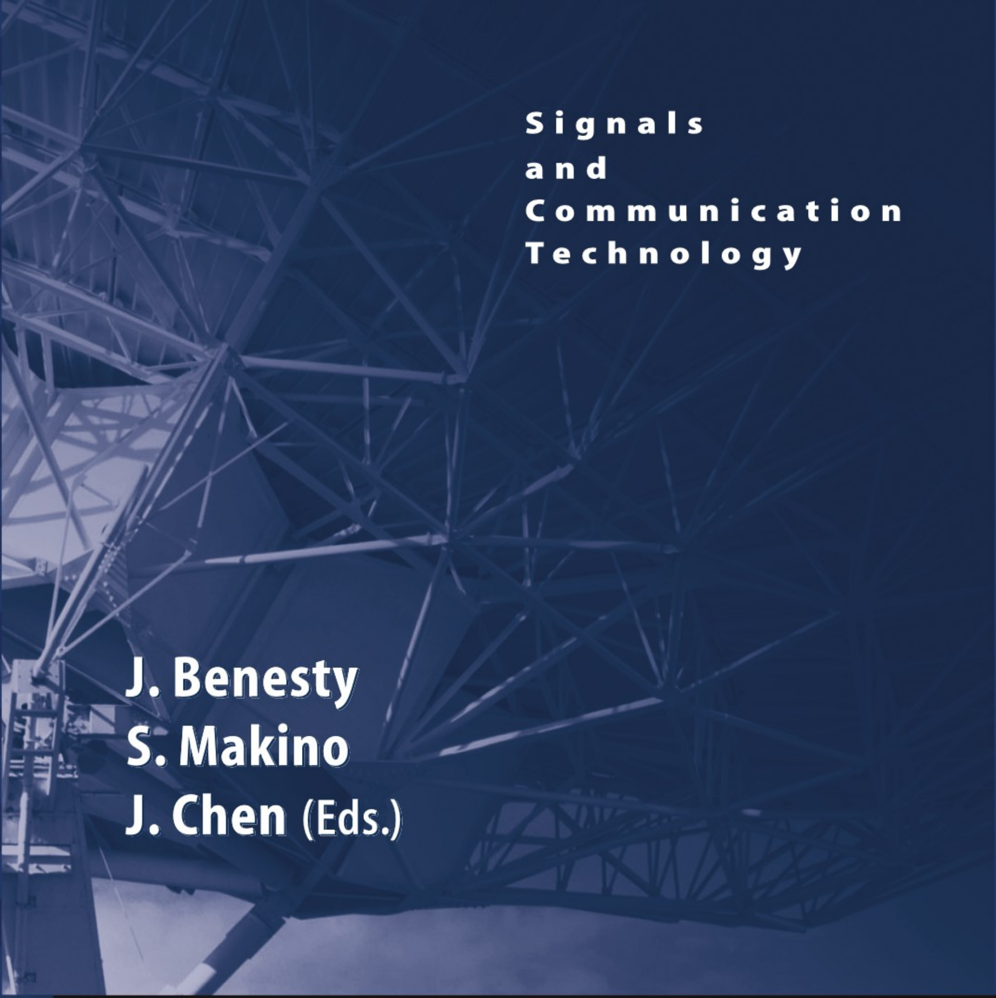


Signals and Communication Technology

J. Benesty
S. Makino
J. Chen (Eds.)

Speech Enhancement

Springer

Springer Series on
SIGNALS AND COMMUNICATION TECHNOLOGY

Signals and Communication Technology

Electronic Noise and Interfering Signals
Principles and Applications
G. Vasilescu
ISBN 3-540-40741-3

DVB
The Family of International Standards
for Digital Video Broadcasting, 2nd ed.
U. Reimers
ISBN 3-540-43545-X

Digital Interactive TV and Metadata
Future Broadcast Multimedia
A. Lugmayr, S. Niiranen, and S. Kalli
ISBN 0-387-20843-7

Adaptive Antenna Arrays
Trends and Applications
S. Chandran (Ed.)
ISBN 3-540-20199-8

Digital Signal Processing
with Field Programmable Gate Arrays
U. Meyer-Baese
ISBN 3-540-21119-5

Neuro-Fuzzy and Fuzzy-Neural Applications
in Telecommunications
P. Stavroulakis (Ed.)
ISBN 3-540-40759-6

SDMA for Multipath Wireless Channels
Limiting Characteristics and Stochastic Models
I.P. Kovalyov
ISBN 3-540-40225-X

Digital Television
A Practical Guide for Engineers
W. Fischer
ISBN 3-540-01155-2

Multimedia Communication Technology
Representation, Transmission
and Identification of Multimedia Signals
J.R. Ohm
ISBN 3-540-01249-4

Information Measures
Information and its Description in Science
and Engineering
C. Arndt
ISBN 3-540-40855-X

Processing of SAR Data
Fundamentals, Signal Processing,
Interferometry
A. Hein
ISBN 3-540-05043-4

Chaos-Based Digital Communication Systems
Operating Principles, Analysis Methods,
and Performance Evaluation
F.C.M. Lau and C.K. Tse
ISBN 3-540-00602-8

Adaptive Signal Processing
Applications to Real-World Problems
J. Benesty and Y. Huang (Eds.)
ISBN 3-540-00051-8

Multimedia Information Retrieval
and Management
Technological Fundamentals
and Applications
D. Feng, W.C. Siu, and H.J. Zhang (Eds.)
ISBN 3-540-00244-8

Structured Cable Systems
A.B. Semenov, S.K. Strizhakov,
and I.R. Suncheley
ISBN 3-540-43000-8

UMTS
The Physical Layer of the Universal Mobile
Telecommunications System
A. Springer and R. Weigel
ISBN 3-540-42162-9

Advanced Theory of Signal Detection
Weak Signal Detection
in Generalized Observations
I. Song, J. Bae, and S.Y. Kim
ISBN 3-540-43064-4

J. Benesty · S. Makino · J. Chen

Speech Enhancement

With 136 Figures and 18 Tables

Springer

Prof. Dr. Jacob Benesty
Universite du Quebec
INRS-EMT
800 de la Gauchetiere Quest
H5A 1K6 Montreal, QC
Canada

Shoji Makino
Communication Science Laboratories, NTT
Hikaridai, Seika-cho 2-4
619-0237 Kyoto
Japan

Jingdong Chen
Bell Labs
Lucent Technologies
600 Mountain Ave.
07974 Murray Hill
USA

ISBN 3-540-24039-X **Springer Berlin Heidelberg New York**

Library of Congress Control Number: 2005921414

Springer is a part of Springer Science+Business Media

springeronline.com

Typesetting and final processing by PTP-Berlin Protago-TEX-Production GmbH, Germany
Cover-Design: design & production GmbH, Heidelberg
Printed on acid-free paper 62/3141/Yu - 5 4 3 2 1 0

Preface

We live in a noisy world! In all applications (telecommunications, hands-free communications, recording, human-machine interfaces, etc) that require at least one microphone, the signal of interest is usually contaminated by background noise and reverberation. As a result, the microphone signal has to be "cleaned" with digital signal processing tools before it is played out, transmitted, or stored.

This book is about speech enhancement. Different well-known and state-of-the-art methods for noise reduction, with one or multiple microphones, are discussed. By speech enhancement, we mean not only noise reduction but also dereverberation and separation of independent signals. These topics are also covered in this book. However, the general emphasis is on noise reduction because of the large number of applications that can benefit from this technology.

The first book on this topic was published in 1983 with the title: *Speech Enhancement*. The editor and publisher were respectively J. S. Lim and Prentice Hall. That book was a collection of journal and conference papers on noise reduction. Since then, we are not aware of any book on the same subject. Obviously, since 1983 research has progressed considerably. While the first methods were all based on spectral subtraction with a single microphone, recently many new concepts have been proposed with one or more microphones. Furthermore, the topic is now much more broadened.

The idea of this edited book came after a short discussion among us that time was quite ripe for such a project, since speech enhancement is not only a fundamental field of research in the applications of digital signal processing but is also of great interest to the industry which is always looking for new solutions that are both effective and practical. It did not take us that much time to decide and launch this project, since we were immediately convinced that the publisher and the contributors would approve it right away. This idea, to finally summarize mature and new concepts in a comprehensive way, was very well received indeed. As a result, we were excited to cover this topic the best way we could with recognized researchers who have made solid contributions to the field of signal enhancement.

One of the main objectives of this book is to provide a strong reference for engineers, researchers, and graduate students who are interested in the problem of signal and speech enhancement. We hope that everyone will find it useful. We think that engineers will find good ideas to implement for new products, researchers will have in their hands a nice tool for further research, and students will be inspired by the ideas presented here in order that one day they will contribute too.

We deeply appreciate the efforts, interest, and enthusiasm of all the contributing authors. Without them, this project would never have been possible. We are very grateful to Dieter Merkle from Springer and his Engineering Editorial Assistant, Petra Jantzen. Working with them is always a pleasure and a wonderful experience since, as usual, everything goes without a bump in the road.

Jacob Benesty,
Shoji Makino,
Jingdong Chen

Contents

List of Contributors

Shoko Araki
NTT Communication Science Laboratories
Kyoto, Japan

Jacob Benesty
Université du Québec, INRS-EMT
Montréal QC, Canada

Guy J. Brown
University of Sheffield, Department of Computer Science
Sheffield, United Kingdom

Benoit Champagne
McGill University
Montreal QC, Canada

Jingdong Chen
Bell Laboratories, Lucent Technologies
Murray Hill NJ, USA

Israel Cohen
Technion – Israel Institute of Technology
Haifa, Israel

Hai Quang Dam
Western Australian Telecommunications Research Institute
Crawley, Australia

Simon Doclo
Katholieke Universiteit Leuven, Dept. of Electrical Engineering
Leuven, Belgium

Sharon Gannot
Bar-Ilan University
Ramat-Gan, Israel

Nedelko Grbić
Blekinge Institute of Technology
Ronneby, Sweden

Yiteng (Arden) Huang
Bell Laboratories, Lucent Technologies
Murray Hill NJ, USA

Firas Jabloun
Toshiba Research
Cambridge, UK

Masanori Kato
Media and Information Research Laboratories, NEC Corporation
Kanagawa, Japan

Keisuke Kinoshita
NTT Communication Science Laboratories
Kyoto, Japan

Thomas Lotter
Siemens Audiological Engineering Group
Erlangen, Germany

Siow Yong Low
Western Australian Telecommunications Research Institute
Crawley, Australia

Shoji Makino
NTT Communication Science Laboratories
Kyoto, Japan

Rainer Martin
Ruhr-Universität Bochum, Institute of Communication Acoustics
Bochum, Germany

Masato Miyoshi
NTT Communication Science Laboratories
Kyoto, Japan

Marc Moonen
Katholieke Universiteit Leuven, Dept. of Electrical Engineering
Leuven, Belgium

Ryo Mukai
NTT Communication Science Laboratories
Kyoto, Japan

Tomohiro Nakatani
NTT Communication Science Laboratories
Kyoto, Japan

Sven Nordholm
Western Australian Telecommunications Research Institute
Crawley, Australia

Hiroshi Sawada
NTT Communication Science Laboratories
Kyoto, Japan

Masahiro Serizawa
Media and Information Research Laboratories, NEC Corporation
Kanagawa, Japan

Ann Spriet
Katholieke Universiteit Leuven, Dept. of Electrical Engineering
Leuven, Belgium

Akihiko Sugiyama
Media and Information Research Laboratories, NEC Corporation
Kanagawa, Japan

DeLiang Wang
The Ohio State University, Department of Computer Science
Columbus OH, USA

Jan Wouters
Katholieke Universiteit Leuven, Laboratory for Exp. ORL
Leuven, Belgium

1 Introduction

Jacob Benesty[1], Shoji Makino[2], and Jingdong Chen[3]

[1] Université du Québec, INRS-EMT
Montréal, QC H5A 1K6, Canada
E-mail: benesty@inrs-emt.uquebec.ca

[2] NTT Communication Science Laboratories
Soraku-gun, Kyoto 619-0237, Japan
E-mail: maki@cslab.kecl.ntt.co.jp

[3] Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07974, USA
E-mail: jingdong@research.bell-labs.com

1.1 Speech Enhancement

What is speech enhancement? Enhancement means the improvement in the value or quality of something. When applied to speech, this simply means the improvement in intelligibility and/or quality of a degraded speech signal by using signal processing tools.

Speech enhancement is a very difficult problem for two reasons. First, the nature and characteristics of the noise signals can change dramatically in time and application to application. It is therefore laborious to find versatile algorithms that really work in different practical environments. Second, the performance measure can also be defined differently for each application. Two perceptual criteria are widely used to measure the performance: quality and intelligibility. While the former is subjective (it reflects individual preferences of listeners), the latter is objective (it gives the percentage of words that could be correctly identified by listeners). It is very hard to satisfy both at the same time. In fact, it can easily be shown that in the single-channel (one microphone) case and when the degradation is due to the uncorrelated additive noise, noise reduction (quality improvement) is possible at the expense of speech distortion (intelligibility reduction). However, we need to be careful when we talk about intelligibility. Indeed, when the speech signal is very noisy, listeners tend to concentrate less after a long period of time and, as a result, there is a reduction in intelligibility. Hence, the intelligibility of the enhanced signal may be perceived higher than that of the noisy signal but only after a long period of listening time, depending on the listener concentration. This phenomenon is known as listener fatigue.

To our knowledge, research on noise reduction techniques started more than 40 years ago at Bell Labs, with pioneering work by Schroeder. He proposed an analog implementation of the spectral magnitude subtraction method. This work is not very well known, probably because it was never published in journals or conferences but only as patents [1], [2]. Boll [3], more than 15 years later, reinvented this method in the digital domain. Since then, many improved and more sophisticated algorithms based on the same approach have been proposed [4], [5], [6], [7]. The spectral subtraction method is by far the most popular and most used in real-world applications. However, this approach introduces some artifacts referred to as musical noise, due to spectral estimation problems. A nice literature review on this particular subject can be found in [8].

Another important approach, based on a signal subspace decomposition, was proposed by Ephraim and Van Trees in 1995 [9]. It consists of decomposing the vector space of the noisy signal into two orthogonal subspaces: the signal-plus-noise subspace and the noise subspace. This is possible because it is well accepted that the clean speech can be modeled with a low-rank model. It is then straightforward to estimate the clean signal. Other algorithms using the same approach were proposed by the same authors. They are based on the observation that signal distortion and residual noise can not be minimized simultaneously. However, we can control a trade-off between the two. The resulting linear estimator is a general Wiener filter with adjustable noise level. It seems that this contribution almost eliminates the musical noise.

The two previous techniques of spectral subtraction and signal subspace are nonparametric. Other important algorithms for speech enhancement belong to the group of parametric methods where the speech signal is modeled as an autoregressive (AR) process embedded in Gaussian noise. Speech enhancement algorithms belonging to this category consist of two steps:

- estimation of the AR coefficients and noise variances, and
- apply the Kalman filter using the estimated parameters to estimate the clean speech.

This important work was started by Paliwal and Basu in 1987 [10].

Most of today's techniques on noise reduction use a single microphone. As we mentioned earlier, it is not possible, in general, to improve both quality and intelligibility at the same time. Usually, quality is improved at the expense of sacrificing intelligibility. Methods using time/spectral information only, have very likely limited performances even though they may give satisfactory results in many applications. One natural way to overcome these limitations or to add more degrees of freedom in searching for a solution is to exploit spatial information by using multiple microphones. In this case, it seems possible to obtain a good amount of noise reduction without distorting the speech signal. Nowadays, researchers are investing efforts in this direction.

Speech communication applications where a noise reduction algorithm is required are numerous; here is a short list:

- hands-free communications,
- voice over IP (VoIP),
- hearing aids,
- local and long distance telecommunications,

- answering machines,
- speech recognition,
- teleconferencing systems,
- car and mobile phones,
- cockpits and noisy manufacturing,
- multiparty conferencing.

To illustrate the importance of noise reduction, we now give the very good example of multiparty conferencing explained by Diethorn in [8]. In multiparty conferencing, the background noise picked up by the microphone of each point of the conference combines additively at the network bridge with the noise signals from all other points. The loudspeaker at each location of the conference therefore reproduces the combined sum of the noise processes from all other locations. Consider a three-point conference in which the room noise at all locations is stationary and independent with power P. Each loudspeaker receives noise from the other two locations, resulting in a total received noise power of $2P$, or 3 dB greater than that of a two-point conference. With N points, each side receives a total noise power that is $10\log_{10}(N-1)$ dB greater than P. For example, in a conference with 11 participating locations, the received noise power at each point is about 10 dB greater than that of the two-party case. Clearly, this problem is extremely serious when the number of conferees is large, and without noise reduction, communication is almost impossible in this context.

To summarize, any speech communication application (and there are many) may require a technique to reduce the level of noise.

1.2 Challenges and Opportunities

In the previous section, we exclusively discussed the noise reduction problem, where the focus is on eliminating or attenuating the additive background noise. However, as the applications are broadened, the definition of speech enhancement is now becoming more general. It should certainly include the reinforcement of the signal from the corruption of competing speech, or even from degradation of the filtered version of the same signal. These are well-known as signal separation and dereverberation problems, which are much more challenging than the classical noise reduction problem.

In a room and in a hands-free context, the signal that is picked up by a microphone from a talker contains not only the direct-path signal, 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 signal, termed as reverberation, which severely deteriorates the source signal. Therefore, dereverberation is required to improve the intelligibility of the speech signal.

Researchers in acoustics have been aware of the negative effect of reverberation in speech communications for more than four decades but, contrary to noise reduction, good and practical solutions are almost inexistent. Dereverberation is a very difficult problem, and we believe it will take a long time before reliable solutions can be derived.

The ability of humans, with normal hearing, to focus on a single talker among a cacophony of conversations and background noise, the so-called *cocktail party effect*, is quite remarkable. One important reason for this ability is the fact that we have two ears. It is well-known that listening in this scenario with only one ear is annoying and it becomes very difficult to concentrate on one particular signal when several of them come from all around simultaneously but it can be done (albeit with much less ability).

In blind source separation with multiple microphones, we try to separate different signals coming, at the same time, from different directions. In a controlled situation, when the number of sources is known and reverberation is not very high, important advances have been made. The cocktail party effect is not solved though. It would be very interesting to be able to separate the signal of interest from the rest, which is not exactly what blind source separation algorithms do.

Many challenges remain in noise reduction as well. In the single-channel case, can we do a better job than what we have now? In the multichannel case, can we derive an optimal solution in the sense that the noise is removed as much as we like without degrading the speech integrity or intelligibility? In the general case, how should we deal with non-stationary noise signals and different acoustic environments?

Needless to say, opportunities are tremendous if great progress is to be in this important topic of research. Noise and reverberation are everywhere around us and they are here to stay. We live in a very noisy world! From hands-free communications to hearing-aids, millions of consumers will benefit from new ideas that go into products. This is already true today. But with new revolutions in ways we telecommunicate with each other and human-machine interfaces, speech enhancement techniques will always be an important part of the whole picture.

Since noise is one of the major problems in speech enhancement in general, more chapters are dedicated to noise reduction than to dereverberation and source separation. In this book, we invited well-known experts to contribute chapters covering the state of the art in the research of this focused field.

1.3 Organization of the Book

This book contains 16 chapters (including this one). We tried to cover the most important topics in the next 15 chapters. Nine chapters (2–10) are dedicated to noise reduction. Dereverberation is covered by two chapters (Chapters 11 and 12). The last four chapters (13–16) are on source separation.

Chapter 2, by Jacob Benesty et al., studies the quantitative performance behavior of the Wiener filter in the context of noise reduction. By defining a speech-distortion index and a noise-reduction factor, it is shown in detail how the Wiener filter achieves noise reduction with detriment to speech integrity. By examining the speech-distortion index, several approaches are also suggested that can better manage the compromise between noise reduction and speech distortion.

Chapter 3, by Rainer Martin, discusses statistical methods for noise reduction in the spectral domain. While the focus is on spectral analysis by means of the discrete Fourier transform, many of these approaches can be also used in conjunction with other analysis techniques such as filterbanks. The MMSE and MAP estimators for the complex spectral coefficients as well as the amplitude of these coefficients are presented. These estimators are analyzed in terms of their input-output characteristics. Emphasis is placed on the use of super-Gaussian density models for the probability density of undisturbed speech coefficients. Furthermore, the estimation of the background noise power is discussed.

In Chapter 4, Thomas Lotter presents maximum *a posteriori* (MAP) spectral amplitude estimators for single- and multi-microphone DFT based speech enhancement systems. The incorporation of a precise super-Gaussian statistical model for the speech spectral amplitude and the generalization of the estimation to multiple-input and output signals provide significant quality improvements compared to common systems.

Chapter 5, by Israel Cohen, describes a new modeling approach for speech signals in the short-time Fourier transform (STFT) domain. It suggests that the expansion coefficients of speech signals have similar characteristics as those of financial time-series, and therefore similar modeling techniques can be exploited. The chapter further demonstrates the application of the new method to speech enhancement, and its advantage compared to using the decision-directed method.

Chapter 6, by Akihiko Sugiyama et al., presents a noise suppression algorithm based on weighted noise estimation. This algorithm continuously updates the estimated noise by weighted noisy speech in accordance with an estimated SNR. Subjective evaluation results show that five-grade mean opinion scores are improved by as much as 0.35, compared with either the MMSE-STSA or the EVRC noise suppression algorithm.

The signal subspace approach (SSA) for speech enhancement is becoming a serious competitor to its already widely used frequency-domain counterparts as it seems to offer a better compromise between signal distortion and the level of the residual noise. Chapter 7, by Firas Jabloun et al., provides a detailed description of the technique in terms of its underlying theory. Various issues associated with the SSA such as the colored noise case and the computational load are addressed. Some of the latest extensions and developments to the SSA are also presented.

In Chapter 8, by Sharon Gannot, the Kalman filter is applied, in the estimate-maximize (EM) framework, to enhance the speech signal received by a single microphone and contaminated by an additive noise signal. The solution iterates between estimating the spectral parameters of the speech and noise signals (at the M-step), and employing the Kalman filter (at the E-step). Generalization to the unscented Kalman filter is also discussed.

Chapter 9, by Simon Doclo et al., discusses a multi-microphone speech enhancement technique that is based on speech distortion weighted multichannel Wiener filtering (SDW-MWF). This speech enhancement technique is more robust against signal model errors than standard adaptive beamforming techniques, since it takes speech distortion due to signal model errors explicitly into account in the design of the adaptive stage. A novel frequency-domain criterion is presented, from which several adaptive frequency-domain algorithms for the SDW-MWF can be derived. The performance of these algorithms is investigated for a small-sized microphone array in a hearing aid application.

Chapter 10, by Sven Nordholm et al., presents two algorithms (using multiple microphones) that show good potential to provide good speech enhancement capability in poor signal-to-noise ratio (SNR) situations. The basic commonality of the adaptive microphone array schemes is that they approximate the Wiener solution according to an estimate using current available data and avoid suppression of the source of interest by employing a quadratic constraint.

Chapter 11, by Tomohiro Nakatani et al., describes a single-channel blind dereverberation method based on the harmonicity of speech signals. A filter that enhances the harmonicity of reverberant speech signals is shown to approximate the inverse filter of the room transfer function; as a result, high quality dereverberation is achieved.

Chapter 12, by Yiteng (Arden) Huang et al., studies the problems of source separation and speech dereverberation in a unified framework based on blind multichannel identification. It is shown that spatial interference and temporal reverberation can be separated and then the extracted interference-free SIMO system is dereverberated using the MINT theorem, leading to the proposal of a novel sequential algorithm. The performance of this algorithm is explored by simulations and the results show some promise.

Chapter 13, by Hiroshi Sawada et al., presents a frequency-domain approach to blind source separation (BSS) of convolutively mixed acoustic signals, where independent component analysis (ICA) is employed in each frequency bin to separate mixed signals. This approach provides good results for separating many sources mixed in a real room environment.

In Chapter 14, Shoko Araki et al. show how to implement BSS in subband. This approach copes with the difficulties of the frequency-domain approach in estimating statistics and the time-domain technique in adapting many parameters. Furthermore, by employing an appropriate separation method for each subband, this method can improve the separation performance.

In Chapter 15, Ryo Mukai et al. present a BSS method for moving sources. Their two-step method employs frequency-domain ICA in the first stage and non-stationary crosstalk cancellation in the second stage. Experimental results using speech signals recorded in a real room show that their method realizes a robust low-delay real-time separation for moving sources.

In Chapter 16, Guy J. Brown et al. review recent developments in the field of computational auditory scene analysis (CASA), which attempts to develop sound separation systems that model human performance. They describe algorithms for the separation of single-channel and binaural acoustic mixtures, and discuss ways of integrating CASA with automatic speech recognition. They also comment on the differences between the CASA and ICA approaches, and suggest ways in which the two might be combined.

1.4 Further Reading

Besides this book, the following is a non-exhaustive list of references for further reading on the subject of speech enhancement in general.

1. H. Drucker, “Speech processing in a high ambient noise environment,” *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 165–168, June 1968.
2. M. M. Sondhi, C. E. Schmidt, and L. R. Rabiner, “Improving the quality of a noisy speech signal,” *Bell Syst. Tech. J.*, vol. 60, pp. 1847–1859, Oct. 1981.
3. J. S. Lim, ed., *Speech Enhancement.* Prentice-Hall, Inc, NJ, 1983.
4. M. Miyoshi and Y. Kaneda, “Inverse filtering of room acoustics,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 145–152, Feb. 1988.
5. J. D. Gibson, B. Koo, and S. D. Gray, “Filtering of colored noise for speech enhancement and coding,” *IEEE Trans. Signal Processing*, vol. 39, pp. 1732–1742, Aug. 1991.
6. S. Furui and M. M. Sondhi, eds., *Advances in Speech Signal Processing.* Marcel Dekker, NY, 1992.
7. Y. Ephraim, “Statistical model based speech enhancement systems,” *Proc. of the IEEE*, vol. 80, pp. 1526–1555, Oct. 1992.
8. O. Cappé, “Elimination of the musical noise phenomenon with the Ephraim and Malah noise suppressor,” *IEEE Trans. Speech Audio Processing*, vol. 2, pp. 345–349, Apr. 1994.
9. S. Gannot, D. Burshtein, and E. Weinstein, “Iterative and sequential Kalman filter-based speech enhancement algorithms,” *IEEE Trans. Speech Audio Processing*, vol. 6, pp. 373–385, July 1998.
10. S. Doclo and M. Moonen, “GSVD-based optimal filtering for single and multimicrophone speech enhancement,” *IEEE Trans. Signal Processing*, vol. 50, pp. 2230–2244, Sept. 2002.
11. A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing.* John Wiley & Sons, 2002.

12. J. Chen, Y. Huang, and J. Benesty, "Filtering techniques for noise reduction and speech enhancement," in *Adaptive Signal Processing–Applications to Real-World Problems*, J. Benesty and Y. Huang, eds., chapter 5, pp. 129–154, Springer-Verlag, Berlin, 2003.
13. R. Martin, "Statistical methods for the enhancement of noisy speech," in *Proc. IWAENC*, 2003, pp. 1–6.
14. Y. Huang and J. Benesty, eds., *Audio Signal Processing for Next-Generation Multimedia Communication Systems.* Kluwer Academic Publishers, Boston, MA, 2004.
15. K. Hermus and P. Wambacq, "Assessment of signal subspace based speech enhancement for noise robust speech recognition," in *Proc. IEEE ICASSP*, 2004, pp. I-945–I-948.

References

1. M. R. Schroeder, U.S. Patent No 3,180,936, filed Dec. 1, 1960, issued Apr. 27, 1965.
2. M. R. Schroeder, U.S. Patent No 3,403,224, filed May 28, 1965, issued Sept. 24, 1968.
3. S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 27, pp. 113–120, Apr. 1979.
4. M. Berouti, R. Schwartz, and J. Makhoul, "Enhancement of speech corrupted by acoustic noise," in *Proc. IEEE ICASSP*, 1979, pp. 208–211.
5. J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proc. of the IEEE*, vol. 67, pp. 1586–1604, Dec. 1979.
6. R. J. McAulay and M. L. Malpass, "Speech enhancement using a soft-decision noise suppression filter," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, pp. 137–145, Apr. 1980.
7. Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 32, pp. 1109–1121, Dec. 1984.
8. 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., chapter 3, pp. 91–115, Kluwer Academic Publishers, Boston, MA, 2004.
9. Y. Ephraim and H. L. Van Trees, "A signal subspace approach for speech enhancement," *IEEE Trans. Speech Audio Processing*, vol. 3, pp. 251–266, July 1995.
10. K. K. Paliwal and A. Basu, "A speech enhancement method based on Kalman filtering," in *Proc. IEEE ICASSP*, 1987, pp. 177–180.

2 Study of the Wiener Filter for Noise Reduction

Jacob Benesty[1], Jingdong Chen[2], Yiteng (Arden) Huang[2], and
Simon Doclo[3]

[1] Université du Québec, INRS-EMT
Montréal, QC H5A 1K6, Canada
E-mail: benesty@inrs-emt.uquebec.ca
[2] Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07974, USA
E-mail: {jingdong, arden}@research.bell-labs.com
[3] Katholieke Universiteit Leuven, Dept. of Electrical Engineering (ESAT-SCD)
Leuven 3001, Belgium
E-mail: doclo@esat.kuleuven.ac.be

Abstract. The problem of noise reduction has attracted a considerable amount of research attention over the past several decades. Numerous techniques were developed, and among them is the optimal Wiener filter, which is the most fundamental approach, and has been delineated in different forms and adopted in diversified applications. It is not a secret that the Wiener filter achieves noise reduction with some integrity loss of the speech signal. However, few efforts have been reported to show the inherent relationship between noise reduction and speech distortion. By defining a speech-distortion index and a noise-reduction factor, this chapter studies the quantitative performance behavior of the Wiener filter in the context of noise reduction. We show that for a single-channel Wiener filter, the amount of noise attenuation is in general proportionate to the amount of speech degradation. In other words, the more the noise is reduced, the more the speech is distorted. This may seem discouraging as we always expect an algorithm to have maximal noise attenuation without much speech distortion. Fortunately, we show that the speech distortion can be better managed by properly manipulating the Wiener filter, or by considering some knowledge of the speech signal. The former leads to a sub-optimal Wiener filter where a parameter is introduced to control the tradeoff between speech distortion and noise reduction, and the latter leads to the well-known parametric-model-based noise reduction technique. We also show that speech distortion can even be avoided if we have multiple realizations of the speech signal.

2.1 Introduction

The existence of noise is inevitable in real-world applications of speech processing. In a voice communication system, for example, a desired speech signal, when propagating through an acoustic channel and picked up by a microphone sensor, is corrupted by unwanted noise, which may result in appreciable or even significant degradation in the quality and intelligibility of the recorded speech. Therefore, it is essential for such systems that we can have some effective noise reduction/speech enhancement techniques to extract the desired speech signal from its corrupted observations.

The noise reduction technique has a broad range of applications, from hearing aids, cellular phones, voice-controlling systems, teleconferencing and multiparty teleconferencing, to automatic speech recognition (ASR) systems. The difference between two systems using and not using such techniques can be significant; therefore, the choice can have a great impact on the functioning of the system.

Research on noise reduction/speech enhancement can be traced back to 40 years ago with 2 patents by Schroeder [1], [2] where an analog implementation of the spectral magnitude subtraction method was described. Since then, it has become an area of active research. Over the past several decades, researchers and engineers have approached this challenging problem by exploiting different facets of the properties of the speech and noise signals [3], [4], [5], [6], [7]. A variety of approaches have been developed, including Wiener filter [8], [9], [10], [11], [12], [13], spectral restoration [3], [11], [14], [15], [16], [17], [18], [19], signal subspace method [20], [21], [22], [23], [24], [25], [26], parametric-model-based approach [27], [28], [29], [30], [31], statistical-model-based method [5], [32], [33], [34], [35], [36], [37], and spatio-temporal filtering [38], [39], [40], [41], [42].

Most of these algorithms were developed independently of each other and their performance on noise reduction were evaluated mostly by assessing the improvement of signal-to-noise ratio (SNR) or subjective speech quality when the methods were formulated. It has been noticed that these algorithms, almost with no exception, achieve noise reduction by some integrity loss of the speech signal. Some algorithms are even formulated explicitly based on the tradeoff between noise reduction and speech distortion, such as the subspace method. However, so far, few efforts have been devoted to analyzing such a tradeoff behavior even though it is a very important issue. In this chapter, we attempt to provide an analysis about the compromise between noise reduction and speech distortion. On the one hand, such a study may offer us some insight into the range of the existing algorithms that can be employed in practical noisy environments. On the other hand, a good understanding may help us to find new algorithms that can work more effectively than the existing ones.

Since there are so many algorithms in the literature, it is extremely difficult if not impossible to find a universal analytical tool that can be applied to any algorithm. In this study, we choose the Wiener filter as the basis since it is the most fundamental approach, and many algorithms are closely connected to this technique. For example, the minimum-mean-square-error (MMSE) estimator presented in [15], which belongs to the category of spectral restoration, converges to the Wiener filter at a high SNR. Also it is widely known that the Kalman filter is tightly related to the Wiener filter.

Starting from the optimal Wiener filtering theory, we introduce two new concepts: the speech-distortion index and the noise-reduction factor. We then show that for a single-channel Wiener filter, the amount of noise attenuation is in general proportionate to the amount of speech degradation. In other words, the more the noise is attenuated, the more the speech is distorted. This observation may seem quite discouraging as we always expect an algorithm to have maximal noise attenuation without much speech distortion. Fortunately, we show that the compromise between noise reduction and speech distortion can be better managed by properly manipulating the Wiener filter, or by considering some knowledge of the speech signal. The former leads to a sub-optimal Wiener filter where, like in the spectral subtraction, a parameter is introduced to control the tradeoff between speech distortion and noise reduction, and the latter leads to the well-known parametric-model-based noise-reduction technique. We also discuss the possibility to avoid speech distortion by using an array of microphones.

2.2 Estimation of the Clean Speech Samples

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

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

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

$$e_x(n) \overset{\triangle}{=} x(n) - \hat{x}(n) = x(n) - \mathbf{h}^T \mathbf{y}(n), \tag{2.2}$$

where superscript ^{T} denotes transpose of a vector or a matrix,

$$\mathbf{h} = \left[\begin{array}{cccc} h_0 & h_1 & \cdots & h_{L-1} \end{array} \right]^T$$

is an FIR filter of length L, and

$$\mathbf{y}(n) = \left[\begin{array}{cccc} y(n) & y(n-1) & \cdots & y(n-L+1) \end{array} \right]^T$$

is a vector containing the L most recent samples of the observation signal $y(n)$.

We now can write the mean-square error (MSE) criterion:

$$J_x(\mathbf{h}) \overset{\triangle}{=} E\left\{ e_x^2(n) \right\}, \tag{2.3}$$

where $E\{\cdot\}$ denotes mathematical expectation. The optimal estimate $\hat{x}_{\mathrm{o}}(n)$ of the clean speech sample $x(n)$ tends to contain less noise than the observation sample $y(n)$, and the optimal filter that forms $\hat{x}_{\mathrm{o}}(n)$ is the Wiener filter which is obtained as follows,

$$\mathbf{h}_{\mathrm{o}} = \arg\min_{\mathbf{h}} J_x(\mathbf{h}). \tag{2.4}$$

Consider the particular filter,

$$\mathbf{u}_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T.$$

This means that the observed signal $y(n)$ will pass this filter unaltered (no noise reduction), thus the corresponding MSE is,

$$\begin{aligned} J_x(\mathbf{u}_1) &= E\left\{\left[x(n) - \mathbf{u}_1^T\mathbf{y}(n)\right]^2\right\} = E\left\{\left[x(n) - y(n)\right]^2\right\} \\ &= E\left\{v^2(n)\right\} = \sigma_v^2. \end{aligned} \tag{2.5}$$

In principle, for the optimal filter $\mathbf{h}_{\mathrm{o}}$, we should have,

$$J_x(\mathbf{h}_{\mathrm{o}}) < J_x(\mathbf{u}_1) = \sigma_v^2. \tag{2.6}$$

In other words, the Wiener filter will be able to reduce the level of noise in the noisy speech signal $y(n)$.

From (2.4), we easily find the Wiener-Hopf equation:

$$\mathbf{R}_y\mathbf{h}_{\mathrm{o}} = \mathbf{p}, \tag{2.7}$$

where

$$\mathbf{R}_y = E\left\{\mathbf{y}(n)\mathbf{y}^T(n)\right\} \tag{2.8}$$

is the correlation matrix of the observed signal $y(n)$ and

$$\mathbf{p} = E\left\{\mathbf{y}(n)x(n)\right\} \tag{2.9}$$

is the cross-correlation vector between the noisy and clean speech signals. However, $x(n)$ is unobservable; as a result, an estimation of $\mathbf{p}$ may seem difficult to obtain. But,

$$\begin{aligned} \mathbf{p} &= E\left\{\mathbf{y}(n)x(n)\right\} = E\left\{\mathbf{y}(n)\left[y(n) - v(n)\right]\right\} \\ &= E\left\{\mathbf{y}(n)y(n)\right\} - E\left\{\left[\mathbf{x}(n) + \mathbf{v}(n)\right]v(n)\right\} \\ &= E\left\{\mathbf{y}(n)y(n)\right\} - E\{\mathbf{v}(n)v(n)\} \\ &= \mathbf{r}_y - \mathbf{r}_v. \end{aligned} \tag{2.10}$$

Now $\mathbf{p}$ depends on the correlation vectors $\mathbf{r}_y$ and $\mathbf{r}_v$. The vector $\mathbf{r}_y$ (which is also the first column of $\mathbf{R}_y$) can be easily estimated during speech and noise periods while $\mathbf{r}_v$ can be estimated during noise-only intervals assuming that the statistics of the noise do not change much with time.

Using (2.10) and the fact that $\mathbf{u}_1 = \mathbf{R}_y^{-1}\mathbf{r}_y$, we obtain the optimal filter:

$$\begin{aligned} \mathbf{h}_{\mathrm{o}} &= \mathbf{u}_1 - \mathbf{R}_y^{-1}\mathbf{r}_v = \left[\mathbf{I} - \mathbf{R}_y^{-1}\mathbf{R}_v\right]\mathbf{u}_1 \\ &= \left[\frac{\mathbf{I}}{\mathrm{SNR}} + \tilde{\mathbf{R}}_v^{-1}\tilde{\mathbf{R}}_x\right]^{-1}\tilde{\mathbf{R}}_v^{-1}\tilde{\mathbf{R}}_x\mathbf{u}_1, \end{aligned} \tag{2.11}$$

where

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

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

$$\begin{aligned} \tilde{\mathbf{R}}_x &\stackrel{\triangle}{=} \frac{\mathbf{R}_x}{\sigma_x^2}, \\ \tilde{\mathbf{R}}_v &\stackrel{\triangle}{=} \frac{\mathbf{R}_v}{\sigma_v^2}. \end{aligned}$$

We have,

$$\lim_{\mathrm{SNR}\to\infty} \mathbf{h}_{\mathrm{o}} = \mathbf{u}_1, \tag{2.13}$$

$$\lim_{\mathrm{SNR}\to 0} \mathbf{h}_{\mathrm{o}} = \mathbf{0}. \tag{2.14}$$

The minimum MSE (MMSE) is,

$$J_x(\mathbf{h}_{\mathrm{o}}) = \sigma_x^2 - \mathbf{p}^T\mathbf{h}_{\mathrm{o}} = \sigma_v^2 - \mathbf{r}_v^T\mathbf{R}_y^{-1}\mathbf{r}_v = \mathbf{r}_v^T\mathbf{h}_{\mathrm{o}}. \tag{2.15}$$

We see clearly from the previous expression that $J_x(\mathbf{h}_{\mathrm{o}}) < J_x(\mathbf{u}_1)$; therefore, noise reduction is possible.

The normalized MMSE is

$$\tilde{J}_x(\mathbf{h}_{\mathrm{o}}) \stackrel{\triangle}{=} \frac{J_x(\mathbf{h}_{\mathrm{o}})}{J_x(\mathbf{u}_1)} = \frac{J_x(\mathbf{h}_{\mathrm{o}})}{\sigma_v^2}, \tag{2.16}$$

and $0 < \tilde{J}_x(\mathbf{h}_{\mathrm{o}}) < 1$.

2.3 Estimation of the Noise Samples

In this section, we will estimate the noise samples from the observations $y(n)$. Define the error signal between the noise sample at time n and its estimate:

$$e_v(n) = v(n) - \hat{v}(n) = v(n) - \mathbf{g}^T\mathbf{y}(n), \tag{2.17}$$

where

$$\mathbf{g} = \left[g_0 \; g_1 \; \cdots \; g_{L-1}\right]^T$$

is an FIR filter of length L. The MSE criterion associated with (2.17) is,

$$J_v(\mathbf{g}) \stackrel{\triangle}{=} E\left\{e_v^2(n)\right\}. \tag{2.18}$$

The estimation of $v(n)$ in the MSE sense will tend to attenuate the clean speech.

The minimization of (2.18) leads to the Wiener-Hopf equation:

$$\begin{aligned}\mathbf{g}_{\mathrm{o}} &= \mathbf{R}_y^{-1}\mathbf{r}_v = \mathbf{R}_y^{-1}\mathbf{R}_v\mathbf{u}_1 \\ &= \left[\mathrm{SNR}\cdot\mathbf{I} + \tilde{\mathbf{R}}_x^{-1}\tilde{\mathbf{R}}_v\right]^{-1}\tilde{\mathbf{R}}_x^{-1}\tilde{\mathbf{R}}_v\mathbf{u}_1.\end{aligned}$$

We have,

$$\lim_{\mathrm{SNR}\to\infty}\mathbf{g}_{\mathrm{o}} = \mathbf{0}, \tag{2.19}$$

$$\lim_{\mathrm{SNR}\to 0}\mathbf{g}_{\mathrm{o}} = \mathbf{u}_1. \tag{2.20}$$

The MSE for the particular filter $\mathbf{u}_1$ (no clean speech reduction) is,

$$J_v(\mathbf{u}_1) = E\left\{x^2(n)\right\} = \sigma_x^2. \tag{2.21}$$

Therefore, the MMSE and the normalized MMSE are respectively,

$$J_v(\mathbf{g}_{\mathrm{o}}) = \sigma_v^2 - \mathbf{r}_v^T\mathbf{R}_y^{-1}\mathbf{r}_v = \sigma_v^2 - \mathbf{r}_v^T\mathbf{g}_{\mathrm{o}}, \tag{2.22}$$

$$\tilde{J}_v(\mathbf{g}_{\mathrm{o}}) \stackrel{\triangle}{=} \frac{J_v(\mathbf{g}_{\mathrm{o}})}{J_v(\mathbf{u}_1)} = \frac{J_v(\mathbf{g}_{\mathrm{o}})}{\sigma_x^2}. \tag{2.23}$$

Since $J_v(\mathbf{g}_{\mathrm{o}}) < J_v(\mathbf{u}_1)$, the Wiener filter will be able to reduce the level of clean speech of the signal $y(n)$. As a result, $0 < \tilde{J}_v(\mathbf{g}_{\mathrm{o}}) < 1$.

In the next section, we will see that while the normalized MMSE, $\tilde{J}_x(\mathbf{h}_{\mathrm{o}})$, of the clean speech estimation plays a key role in noise reduction, the normalized MMSE, $\tilde{J}_v(\mathbf{g}_{\mathrm{o}})$, of the noise process estimation plays a key role in speech distortion.

2.4 Important Relationships Between Noise Reduction and Speech Distortion

Obviously, there are some important relationships between the estimation of the clean speech and noise samples. We immediately see from (2.15) and (2.22) that the two MMSEs are equal,

$$J_x(\mathbf{h}_{\mathrm{o}}) = J_v(\mathbf{g}_{\mathrm{o}}). \tag{2.24}$$

However, the normalized MMSEs are not, in general. Indeed, we have a relation between the two:

$$\begin{aligned}\tilde{J}_v(\mathbf{g}_{\mathrm{o}}) &= \frac{J_v(\mathbf{g}_{\mathrm{o}})}{\sigma_x^2} = \frac{J_x(\mathbf{h}_{\mathrm{o}})}{\sigma_x^2} \\ &= \frac{\sigma_v^2}{\sigma_x^2}\frac{J_x(\mathbf{h}_{\mathrm{o}})}{\sigma_v^2} = \frac{\tilde{J}_x(\mathbf{h}_{\mathrm{o}})}{\mathrm{SNR}}.\end{aligned} \tag{2.25}$$

So the only situation where the two normalized MMSEs are equal is when the SNR is equal to 1. For SNR < 1, $\tilde{J}_x(\mathbf{h}_\mathrm{o}) < \tilde{J}_v(\mathbf{g}_\mathrm{o})$ and for SNR > 1, $\tilde{J}_v(\mathbf{g}_\mathrm{o}) < \tilde{J}_x(\mathbf{h}_\mathrm{o})$. Also, $\tilde{J}_x(\mathbf{h}_\mathrm{o}) <$ SNR and $\tilde{J}_v(\mathbf{g}_\mathrm{o}) < 1/$SNR.

From (2.11) and (2.19), we get a relation between the two optimal filters:

$$\mathbf{h}_\mathrm{o} = \mathbf{u}_1 - \mathbf{g}_\mathrm{o}. \tag{2.26}$$

In fact, minimizing $J_x(\mathbf{h})$ or $J_v(\mathbf{u}_1 - \mathbf{h})$ with respect to $\mathbf{h}$ is equivalent. In the same manner, minimizing $J_v(\mathbf{g})$ or $J_x(\mathbf{u}_1 - \mathbf{g})$ with respect to $\mathbf{g}$ is the same thing. At the optimum, we have,

$$\begin{aligned} e_{x,\mathrm{o}}(n) &= x(n) - \mathbf{h}_\mathrm{o}^T \mathbf{y}(n) = x(n) - [\mathbf{u}_1 - \mathbf{g}_\mathrm{o}]^T [\mathbf{x}(n) + \mathbf{v}(n)] \\ &= -v(n) + \mathbf{g}_\mathrm{o}^T \mathbf{y}(n) = -e_{v,\mathrm{o}}(n). \end{aligned} \tag{2.27}$$

We can easily verify the following:

$$\begin{aligned} J_v(\mathbf{h}_\mathrm{o}) &= J_x(\mathbf{g}_\mathrm{o}) \\ &= \sigma_y^2 - 3J_x(\mathbf{h}_\mathrm{o}), \end{aligned} \tag{2.28}$$

which implies that $J_x(\mathbf{h}_\mathrm{o}) < \sigma_y^2/3$. We already know that $J_x(\mathbf{h}_\mathrm{o}) < \sigma_v^2$ and $J_x(\mathbf{h}_\mathrm{o}) < \sigma_x^2$.

The optimal estimation of the clean speech, in the Wiener sense, is in fact what we call noise reduction:

$$\hat{x}_\mathrm{o}(n) = \mathbf{h}_\mathrm{o}^T \mathbf{y}(n), \tag{2.29}$$

or equivalently, if the noise is estimated first:

$$\hat{v}_\mathrm{o}(n) = \mathbf{g}_\mathrm{o}^T \mathbf{y}(n), \tag{2.30}$$

we can use this estimate to reduce the noise from the observed signal:

$$\hat{x}_\mathrm{o}(n) = y(n) - \hat{v}_\mathrm{o}(n). \tag{2.31}$$

The power of the estimated clean speech signal with the optimal Wiener filter is,

$$\begin{aligned} E\left\{\hat{x}_\mathrm{o}^2(n)\right\} &= \mathbf{h}_\mathrm{o}^T \mathbf{R}_y \mathbf{h}_\mathrm{o} = \sigma_x^2 - J_x(\mathbf{h}_\mathrm{o}) \\ &= \mathbf{h}_\mathrm{o}^T \mathbf{R}_x \mathbf{h}_\mathrm{o} + \mathbf{h}_\mathrm{o}^T \mathbf{R}_v \mathbf{h}_\mathrm{o}, \end{aligned} \tag{2.32}$$

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 (2.32) shows that the price to pay for this is also a reduction of the clean speech [by a quantity equal to $J_x(\mathbf{h}_\mathrm{o}) + \mathbf{h}_\mathrm{o}^T \mathbf{R}_v \mathbf{h}_\mathrm{o}$ and this implies distortion], since $\mathbf{h}_\mathrm{o}^T \mathbf{R}_x \mathbf{h}_\mathrm{o} < \sigma_x^2$. In other words, the power of the attenuated clean speech signal is, obviously, always smaller than the power of the clean speech itself; this means that parts of the clean speech are attenuated in the process and as a result, distortion is unavoidable with this approach.

We now define the speech-distortion index due to the optimal filtering operation as,

$$\begin{aligned} \upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{o}}) &\triangleq \frac{E\left\{\left[x(n) - \mathbf{h}_{\mathrm{o}}^T \mathbf{x}(n)\right]^2\right\}}{\sigma_x^2} \\ &= \frac{\mathbf{g}_{\mathrm{o}}^T \mathbf{R}_x \mathbf{g}_{\mathrm{o}}}{\sigma_x^2} = \frac{1}{\mathrm{SNR}}\left[\tilde{J}_x(\mathbf{h}_{\mathrm{o}}) - \mathbf{h}_{\mathrm{o}}^T \tilde{\mathbf{R}}_v \mathbf{h}_{\mathrm{o}}\right] < \tilde{J}_v(\mathbf{g}_{\mathrm{o}}). \end{aligned} \tag{2.33}$$

Clearly, this index is always between 0 and 1 for the optimal filter. Also,

$$\lim_{\mathrm{SNR}\to 0} \upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{o}}) = 1, \tag{2.34}$$

$$\lim_{\mathrm{SNR}\to \infty} \upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{o}}) = 0. \tag{2.35}$$

So when $\upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{o}})$ is close to 1, the speech signal is highly distorted and when $\upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{o}})$ is near 0, the speech signal is lowly distorted. We deduce that for low SNRs, the Wiener filter can have a disastrous effect on the speech signal.

Similarly, we define the noise-reduction factor due to the Wiener filter as,

$$\begin{aligned} \xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{o}}) &\triangleq \frac{\sigma_v^2}{E\left\{\left[\mathbf{h}_{\mathrm{o}}^T \mathbf{v}(n)\right]^2\right\}} \\ &= \frac{\sigma_v^2}{\mathbf{h}_{\mathrm{o}}^T \mathbf{R}_v \mathbf{h}_{\mathrm{o}}} = \frac{1}{\mathrm{SNR}\left[\tilde{J}_v(\mathbf{g}_{\mathrm{o}}) - \mathbf{g}_{\mathrm{o}}^T \tilde{\mathbf{R}}_x \mathbf{g}_{\mathrm{o}}\right]} > \frac{1}{\tilde{J}_x(\mathbf{h}_{\mathrm{o}})}, \end{aligned} \tag{2.36}$$

and $\xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{o}}) > 1$. The greater is $\xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{o}})$, the more noise reduction we have. Also,

$$\lim_{\mathrm{SNR}\to 0} \xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{o}}) = \infty, \tag{2.37}$$

$$\lim_{\mathrm{SNR}\to \infty} \xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{o}}) = 1. \tag{2.38}$$

Using (2.33) and (2.36), we obtain important relations between the speech-distortion index and the noise-reduction factor:

$$\upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{o}}) = \frac{1}{\mathrm{SNR}}\left[\tilde{J}_x(\mathbf{h}_{\mathrm{o}}) - \frac{1}{\xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{o}})}\right], \tag{2.39}$$

$$\xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{o}}) = \frac{1}{\mathrm{SNR}\left[\tilde{J}_v(\mathbf{g}_{\mathrm{o}}) - \upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{o}})\right]}. \tag{2.40}$$

Therefore, for the optimum filter, when the SNR is very large, there is little speech distortion and little noise reduction (which is not really needed in this situation). On the other hand, when the SNR is very small, speech distortion is large as well as noise reduction. Using the fact that $J_x(\mathbf{h}_\mathrm{o}) < \sigma_y^2/3$, we can easily derive from (2.39) and (2.40) that,

$$\begin{cases} \xi_\mathrm{nr}(\mathbf{h}_\mathrm{o}) > \frac{1}{\mathrm{SNR}} & \text{if} \quad \mathrm{SNR} \leq 1/2 \\ \xi_\mathrm{nr}(\mathbf{h}_\mathrm{o}) \geq \frac{3}{\mathrm{SNR}+1} & \text{if } 1/2 < \mathrm{SNR} \leq 2 \\ \xi_\mathrm{nr}(\mathbf{h}_\mathrm{o}) > 1 & \text{if} \quad \mathrm{SNR} > 2 \end{cases}, \tag{2.41}$$

and

$$\begin{cases} \upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o}) < 1 & \text{if} \quad \mathrm{SNR} \leq 1/2 \\ \upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o}) < \frac{\mathrm{SNR}+1}{3\mathrm{SNR}} & \text{if } 1/2 < \mathrm{SNR} \leq 2 \\ \upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o}) < \frac{1}{\mathrm{SNR}} & \text{if} \quad \mathrm{SNR} > 2 \end{cases}. \tag{2.42}$$

Equations (2.41) and (2.42) give the lower bound for the noise-reduction factor and the upper bound for the speech-distortion index respectively. These bounds can be further refined. But before going further, let us first analyze the *a posteriori* SNR, which is defined, after noise reduction with the Wiener filter, as,

$$\begin{aligned} \mathrm{SNR}_\mathrm{o} &\triangleq \frac{\mathbf{h}_\mathrm{o}^T \mathbf{R}_x \mathbf{h}_\mathrm{o}}{\mathbf{h}_\mathrm{o}^T \mathbf{R}_v \mathbf{h}_\mathrm{o}} \\ &= \mathrm{SNR}\, \frac{\mathbf{h}_\mathrm{o}^T \tilde{\mathbf{R}}_x \mathbf{h}_\mathrm{o}}{\mathbf{h}_\mathrm{o}^T \tilde{\mathbf{R}}_v \mathbf{h}_\mathrm{o}} = -1 + \mathrm{SNR}\, \xi_\mathrm{nr}(\mathbf{h}_\mathrm{o}) \left[1 - \tilde{J}_v(\mathbf{g}_\mathrm{o})\right] \\ &= -1 + \frac{1 - \tilde{J}_v(\mathbf{g}_\mathrm{o})}{\tilde{J}_v(\mathbf{g}_\mathrm{o}) - \upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o})}. \end{aligned} \tag{2.43}$$

It can be easily verified that,

$$\mathrm{SNR}_\mathrm{o} > \frac{\mathrm{SNR}}{\tilde{J}_x(\mathbf{h}_\mathrm{o})} - 2. \tag{2.44}$$

We now give a proposition showing the relationship between the *a priori* SNR and the *a posteriori* SNR.

Proposition: With the Wiener filter, the *a posteriori* SNR and the *a priori* SNR satisfy

$$\mathrm{SNR}_\mathrm{o} = \frac{\mathbf{h}_\mathrm{o}^T \mathbf{R}_x \mathbf{h}_\mathrm{o}}{\mathbf{h}_\mathrm{o}^T \mathbf{R}_v \mathbf{h}_\mathrm{o}} \geq \mathrm{SNR} = \frac{\mathbf{u}_1^T \mathbf{R}_x \mathbf{u}_1}{\mathbf{u}_1^T \mathbf{R}_v \mathbf{u}_1}. \tag{2.45}$$

Proof. From their definitions, we know that all three matrices, $\mathbf{R}_x$, $\mathbf{R}_v$, and $\mathbf{R}_y$ are symmetric, and positive semi-definite. We further assume that $\mathbf{R}_v$ is positive definite so its inverse exists. In addition, based on the independence assumption between the speech signal and noise, we have $\mathbf{R}_y = \mathbf{R}_x + \mathbf{R}_v$. In case that both $\mathbf{R}_x$ and $\mathbf{R}_v$ are diagonal matrices, or $\mathbf{R}_v$ is a scaled version of $\mathbf{R}_x$ (i.e., $\mathbf{R}_x = \mathrm{SNR} \cdot \mathbf{R}_v$), it can be easily seen that $\mathrm{SNR}_\mathrm{o} = \mathrm{SNR}$. Here, we consider more complicated situations where at least one of the $\mathbf{R}_x$ and $\mathbf{R}_v$ matrices is not diagonal. In this case, according to [45], there exists a linear transform that can simultaneously diagonalize $\mathbf{R}_x$, $\mathbf{R}_v$, and $\mathbf{R}_y$. The process is done as follows.

$$\begin{aligned}\mathbf{R}_x &= (\mathbf{B}^T)^{-1}\mathbf{\Lambda}\mathbf{B}^{-1},\\ \mathbf{R}_v &= (\mathbf{B}^T)^{-1}\mathbf{B}^{-1},\\ \mathbf{R}_y &= (\mathbf{B}^T)^{-1}[\mathbf{I}+\mathbf{\Lambda}]\mathbf{B}^{-1},\end{aligned} \tag{2.46}$$

where again $\mathbf{I}$ is the identity matrix,

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & & & \vdots \\ 0 & \cdots & 0 & \lambda_L \end{bmatrix} \tag{2.47}$$

is the eigenvalue matrix of $\mathbf{R}_v^{-1}\mathbf{R}_x$, with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_L \geq 0$, $\mathbf{B}$ is the eigenvector matrix of $\mathbf{R}_v^{-1}\mathbf{R}_x$, and

$$\mathbf{R}_v^{-1}\mathbf{R}_x\mathbf{B} = \mathbf{B}\mathbf{\Lambda}. \tag{2.48}$$

Note that $\mathbf{B}$ is not necessarily orthogonal since $\mathbf{R}_v^{-1}\mathbf{R}_x$ is not necessarily symmetric. Then from the definition of SNR and SNR_o, we immediately have

$$\mathrm{SNR} = \frac{\mathbf{u}_1^T\mathbf{R}_x\mathbf{u}_1}{\mathbf{u}_1^T\mathbf{R}_v\mathbf{u}_1} = \frac{\mathbf{u}_1^T(\mathbf{B}^{-1})^T\mathbf{\Lambda}\mathbf{B}^{-1}\mathbf{u}_1}{\mathbf{u}_1^T(\mathbf{B}^{-1})^T\mathbf{B}^{-1}\mathbf{u}_1}, \tag{2.49}$$

and

$$\begin{aligned}\mathrm{SNR}_\mathrm{o} &= \frac{\mathbf{h}_\mathrm{o}^T\mathbf{R}_x\mathbf{h}_\mathrm{o}}{\mathbf{h}_\mathrm{o}^T\mathbf{R}_v\mathbf{h}_\mathrm{o}} = \frac{\mathbf{u}_1^T\mathbf{R}_x^T\mathbf{R}_y^{-1}\mathbf{R}_x\mathbf{R}_y^{-1}\mathbf{R}_x\mathbf{u}_1}{\mathbf{u}_1^T\mathbf{R}_x^T\mathbf{R}_y^{-1}\mathbf{R}_v\mathbf{R}_y^{-1}\mathbf{R}_x\mathbf{u}_1}\\ &= \frac{\mathbf{u}_1^T(\mathbf{B}^{-1})^T\mathbf{\Lambda}(\mathbf{I}+\mathbf{\Lambda})^{-1}\mathbf{\Lambda}(\mathbf{I}+\mathbf{\Lambda})^{-1}\mathbf{\Lambda}\mathbf{B}^{-1}\mathbf{u}_1}{\mathbf{u}_1^T(\mathbf{B}^{-1})^T\mathbf{\Lambda}(\mathbf{I}+\mathbf{\Lambda})^{-1}(\mathbf{I}+\mathbf{\Lambda})^{-1}\mathbf{\Lambda}\mathbf{B}^{-1}\mathbf{u}_1}\\ &= \frac{\mathbf{u}_1^T(\mathbf{B}^{-1})^T\mathbf{\Sigma_1}\mathbf{B}^{-1}\mathbf{u}_1}{\mathbf{u}_1^T(\mathbf{B}^{-1})^T\mathbf{\Sigma_2}\mathbf{B}^{-1}\mathbf{u}_1},\end{aligned} \tag{2.50}$$

where

$$\begin{aligned}\mathbf{\Sigma_1} &\overset{\triangle}{=} \mathbf{\Lambda}(\mathbf{I}+\mathbf{\Lambda})^{-1}\mathbf{\Lambda}(\mathbf{I}+\mathbf{\Lambda})^{-1}\mathbf{\Lambda}\\ &= \begin{bmatrix} \frac{\lambda_1^3}{(1+\lambda_1)^2} & 0 & \cdots & 0 \\ 0 & \frac{\lambda_2^3}{(1+\lambda_2)^2} & \cdots & 0 \\ \vdots & & & \vdots \\ 0 & \cdots & 0 & \frac{\lambda_L^3}{(1+\lambda_L)^2} \end{bmatrix}\end{aligned}$$

and

$$\begin{aligned}\mathbf{\Sigma_2} &\overset{\triangle}{=} \mathbf{\Lambda}(\mathbf{I}+\mathbf{\Lambda})^{-1}(\mathbf{I}+\mathbf{\Lambda})^{-1}\mathbf{\Lambda} \\ &= \begin{bmatrix} \frac{\lambda_1^2}{(1+\lambda_1)^2} & 0 & \cdots & 0 \\ 0 & \frac{\lambda_2^2}{(1+\lambda_2)^2} & \cdots & 0 \\ \vdots & & & \vdots \\ 0 & \cdots & 0 & \frac{\lambda_L^2}{(1+\lambda_L)^2} \end{bmatrix}\end{aligned}$$

are two diagonal matrices. If for the ease of expression we denote $\mathbf{B}^{-1}$ as $\mathbf{A} = \mathbf{B}^{-1} = [a_{ij}]$, then both SNR and $\mathrm{SNR_o}$ can be rewritten as

$$\begin{aligned}\mathrm{SNR} &= \frac{\sum_{i=1}^{L} \lambda_i a_{i1}^2}{\sum_{i=1}^{L} a_{i1}^2}, \\ \mathrm{SNR_o} &= \frac{\sum_{i=1}^{L} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2}{\sum_{i=1}^{L} \frac{\lambda_i^2}{(1+\lambda_i)^2} a_{i1}^2}. \end{aligned} \tag{2.51}$$

Since $\sum_{i=1}^{L} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2$, $\sum_{i=1}^{L} \frac{\lambda_i^2}{(1+\lambda_i)^2} a_{i1}^2$, $\sum_{i=1}^{L} \lambda_i a_{i1}^2$, and $\sum_{i=1}^{L} a_{i1}^2$ all are nonnegative numbers, as long as we can show that the inequality

$$\sum_{i=1}^{L} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{L} a_{i1}^2 \geq \sum_{i=1}^{L} \frac{\lambda_i^2}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{L} \lambda_i a_{i1}^2 \tag{2.52}$$

holds, then $\mathrm{SNR_o} \geq \mathrm{SNR}$. Now we prove this inequality by way of induction.

- Basic Step: If $L = 2$,

  $$\begin{aligned}&\sum_{i=1}^{2} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{2} a_{i1}^2 \\ &= \frac{\lambda_1^3}{(1+\lambda_1)^2} a_{11}^4 + \frac{\lambda_2^3}{(1+\lambda_2)^2} a_{21}^4 + \left[\frac{\lambda_1^3}{(1+\lambda_1)^2} + \frac{\lambda_2^3}{(1+\lambda_2)^2}\right] a_{11}^2 a_{21}^2.\end{aligned}$$

  Since $\lambda_i \geq 0$, 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

  $$\sum_{i=1}^{2} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{2} a_{i1}^2$$

$$\geq \frac{\lambda_1^3}{(1+\lambda_1)^2} a_{11}^4 + \frac{\lambda_2^3}{(1+\lambda_2)^2} a_{21}^4 + \left[\frac{\lambda_1^2 \lambda_2}{(1+\lambda_1)^2} + \frac{\lambda_1 \lambda_2^2}{(1+\lambda_2)^2}\right] a_{11}^2 a_{21}^2$$
$$= \sum_{i=1}^{2} \frac{\lambda_i^2}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{2} \lambda_i a_{i1}^2,$$

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

- Inductive Step: Assume that the property is true for $L = n$, i.e.,

$$\sum_{i=1}^{n} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{n} a_{i1}^2 \geq \sum_{i=1}^{n} \frac{\lambda_i^2}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{n} \lambda_i a_{i1}^2.$$

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

$$\sum_{i=1}^{n+1} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{n+1} a_{i1}^2$$
$$= \left[\sum_{i=1}^{n} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2 + \frac{\lambda_{n+1}^3}{(1+\lambda_{n+1})^2} a_{n+11}^2\right] \left[\sum_{i=1}^{n} a_{i1}^2 + a_{n+11}^2\right]$$
$$= \left[\sum_{i=1}^{n} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2\right] \left[\sum_{i=1}^{n} a_{i1}^2\right] + \frac{\lambda_{n+1}^3}{(1+\lambda_{n+1})^2} a_{n+11}^4$$
$$+ \sum_{i=1}^{n} \left[\frac{\lambda_i^3}{(1+\lambda_i)^2} + \frac{\lambda_{n+1}^3}{(1+\lambda_{n+1})^2}\right] a_{i1}^2 a_{n+11}^2. \quad (2.53)$$

Using the induction hypothesis, and also the fact that

$$\frac{\lambda_i^3}{(1+\lambda_i)^2} + \frac{\lambda_{n+1}^3}{(1+\lambda_{n+1})^2} \geq \frac{\lambda_i^2 \lambda_{n+1}}{(1+\lambda_i)^2} + \frac{\lambda_i \lambda_{n+1}^2}{(1+\lambda_{n+1})^2},$$

hence

$$\sum_{i=1}^{n+1} \frac{\lambda_i^3}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{n+1} a_{i1}^2$$
$$\geq \sum_{i=1}^{n} \frac{\lambda_i^2}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{n} \lambda_i a_{i1}^2 + \frac{\lambda_{n+1}^3}{(1+\lambda_{n+1})^2} a_{n+11}^4$$
$$+ \sum_{i=1}^{n} \left[\frac{\lambda_i^2 \lambda_{n+1}}{(1+\lambda_i)^2} + \frac{\lambda_i \lambda_{n+1}^2}{(1+\lambda_{n+1})^2}\right] a_{i1}^2 a_{n+11}^2$$
$$= \sum_{i=1}^{n+1} \frac{\lambda_i^2}{(1+\lambda_i)^2} a_{i1}^2 \sum_{i=1}^{n+1} \lambda_i a_{i1}^2, \quad (2.54)$$

where "=" holds when all the λ_i's corresponding to nonzero a_{i1} are equal, where $i = 1, 2, \ldots, n+1$. That completes the proof.

Even though it can improve the SNR, the Wiener filter does not maximize the *a posteriori* SNR. As a matter of fact, (2.43) is well known as the generalized Rayleigh quotient. So the filter that really maximizes the *a posteriori* SNR is the eigenvector corresponding to the maximum eigenvalue of the matrix $\mathbf{R}_v^{-1}\mathbf{R}_x$.

Knowing that $\mathrm{SNR}_\mathrm{o} \geq \mathrm{SNR}$, we can now refine the lower bound for $\xi_\mathrm{nr}(\mathbf{h}_\mathrm{o})$. As a matter of fact, it follows from (2.43) that

$$\mathrm{SNR}_\mathrm{o} = -1 + \frac{1 - \tilde{J}_v(\mathbf{g}_\mathrm{o})}{\tilde{J}_v(\mathbf{g}_\mathrm{o}) - \upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o})} \geq \mathrm{SNR}.$$

Since $\upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o}) < \tilde{J}_v(\mathbf{g}_\mathrm{o})$, and $0 \leq \upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o}) \leq 1$, it can be easily shown that

$$\xi_\mathrm{nr}(\mathbf{h}_\mathrm{o}) \geq \frac{\mathrm{SNR} + 2}{\mathrm{SNR}}. \tag{2.55}$$

This lower bound for $\xi_\mathrm{nr}(\mathbf{h}_\mathrm{o})$ is tighter than the one given in (2.41). Similarly, we can derive that

$$\upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o}) \leq \frac{1}{2\mathrm{SNR} + 1}. \tag{2.56}$$

It can be easily verified that this upper bound for $\upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o})$ is tighter than the one given in (2.42). Figure 2.1 illustrates expressions (2.55) and (2.56).

We now introduce another index for noise reduction:

$$\zeta_\mathrm{nr}(\mathbf{h}_\mathrm{o}) \overset{\triangle}{=} 1 - \tilde{J}_x(\mathbf{h}_\mathrm{o}) < 1. \tag{2.57}$$

The closer is $\zeta_\mathrm{nr}(\mathbf{h}_\mathrm{o})$ to 1, the more noise reduction we get. This index will be helpful to use in the following sections.

2.5 Particular Case: White Gaussian Noise

In this section, we assume that the additive noise is white, so that,

$$\mathbf{r}_v = \sigma_v^2 \mathbf{u}_1. \tag{2.58}$$

From (2.16) and (2.23), we observe that the two normalized MMSEs are

$$\tilde{J}_x(\mathbf{h}_\mathrm{o}) = h_{\mathrm{o},0}, \tag{2.59}$$

$$\tilde{J}_v(\mathbf{g}_\mathrm{o}) = \frac{1 - g_{\mathrm{o},0}}{\mathrm{SNR}} = \frac{h_{\mathrm{o},0}}{\mathrm{SNR}}, \tag{2.60}$$

where $h_{\mathrm{o},0}$ and $g_{\mathrm{o},0}$ are the first components of vectors $\mathbf{h}_\mathrm{o}$ and $\mathbf{g}_\mathrm{o}$, respectively. Clearly, $0 < h_{\mathrm{o},0} < 1$ and $0 < g_{\mathrm{o},0} < 1$. Hence, the normalized MMSE $\tilde{J}_x(\mathbf{h}_\mathrm{o})$ is completely governed by the first element of the Wiener filter $\mathbf{h}_\mathrm{o}$.

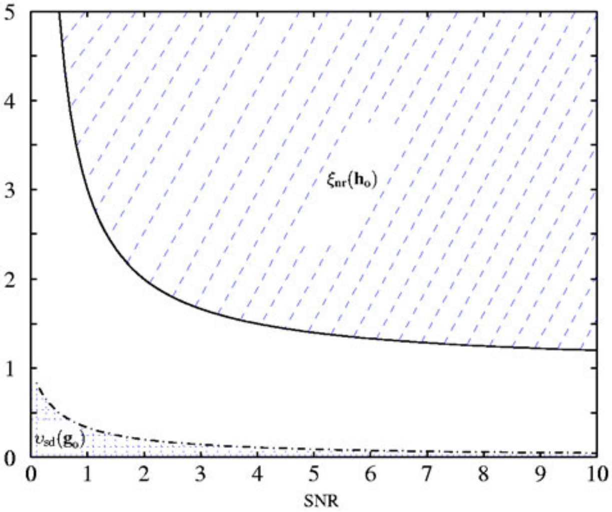


Fig. 2.1. Illustration of the areas where $\xi_{\mathrm{nr}}(\mathbf{h}_\mathrm{o})$ and $\upsilon_{\mathrm{sd}}(\mathbf{g}_\mathrm{o})$ take their values as functions of the SNR. $\xi_{\mathrm{nr}}(\mathbf{h}_\mathrm{o})$ can take any value above the solid line while $\upsilon_{\mathrm{sd}}(\mathbf{g}_\mathrm{o})$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

Now, the speech-distortion index and the noise-reduction factor for the optimal filter can be simplified:

$$\begin{aligned}\upsilon_{\mathrm{sd}}(\mathbf{g}_\mathrm{o}) &= \frac{1}{\mathrm{SNR}}\left[h_{\mathrm{o},0} - \mathbf{h}_\mathrm{o}^T\mathbf{h}_\mathrm{o}\right] \qquad (2.61)\\ &= \frac{\mathbf{g}_\mathrm{o}^T\mathbf{h}_\mathrm{o}}{\mathrm{SNR}} = \frac{1}{\mathrm{SNR}}\left[g_{\mathrm{o},0} - \mathbf{g}_\mathrm{o}^T\mathbf{g}_\mathrm{o}\right],\\ \xi_{\mathrm{nr}}(\mathbf{h}_\mathrm{o}) &= \frac{1}{\mathbf{h}_\mathrm{o}^T\mathbf{h}_\mathrm{o}}. \qquad (2.62)\end{aligned}$$

We also deduce from (2.61) that $h_{\mathrm{o},0} > \mathbf{h}_\mathrm{o}^T\mathbf{h}_\mathrm{o}$ and $g_{\mathrm{o},0} > \mathbf{g}_\mathrm{o}^T\mathbf{g}_\mathrm{o}$.

We know from the linear prediction theory that [43],

$$\mathbf{R}_y\begin{bmatrix}1\\ -\mathbf{a}_y\end{bmatrix} = \begin{bmatrix}E_y\\ \mathbf{0}_{(L-1)\times 1}\end{bmatrix}, \qquad (2.63)$$

where $\mathbf{a}_y$ is the forward linear predictor and E_y is the corresponding error energy. Replacing the previous equation in (2.11), we obtain:

$$\mathbf{h}_\mathrm{o} = \mathbf{u}_1 - \sigma_v^2\mathbf{R}_y^{-1}\mathbf{u}_1 = \begin{bmatrix}h_{\mathrm{o},0}\\ \frac{\sigma_v^2}{E_y}\mathbf{a}_y\end{bmatrix}, \qquad (2.64)$$

where

$$h_{\mathrm{o},0} = \tilde{J}_x(\mathbf{h}_\mathrm{o}) = 1 - \frac{\sigma_v^2}{E_y}. \qquad (2.65)$$

Equation (2.64) shows how the Wiener filter is related to the forward predictor of the observed signal $y(n)$. This expression also gives a hint on how to choose the length of the optimal filter $\mathbf{h}_\mathrm{o}$: it should be equal to the length of the predictor $\mathbf{a}_y$ required to have a good prediction of the observed signal $y(n)$. Equation (2.65) contains some very interesting information. Indeed, if the clean speech signal is completely predictable, this means that $E_y \approx \sigma_v^2$ and $\tilde{J}_x(\mathbf{h}_\mathrm{o}) \approx 0$. On the other hand, if $x(n)$ is not predictable, we have $E_y \approx \sigma_y^2$ and $\tilde{J}_x(\mathbf{h}_\mathrm{o}) \approx 1 - \sigma_v^2/\sigma_y^2$. This implies that the Wiener filter is more efficient to reduce the level of noise for predictable signals than for unpredictable ones.

2.6 Better Ways to Manage Noise Reduction and Speech Distortion

For a noise-reduction/speech-enhancement system, we always expect that it can achieve maximal noise reduction without much speech distortion. From the previous section, however, we see that when noise reduction is maximized with the optimal Wiener filter, speech distortion is also maximized. One may ask a legitimate question: are there better ways to control the tradeoff between the conflicting requirements of noise reduction and speech distortion? Examining (2.33), one can see that to control the speech distortion, we have to minimize $E\left\{\left[x(n) - \mathbf{h}_\mathrm{o}^T\mathbf{x}(n)\right]^2\right\}$. This can be achieved by either manipulating $\mathbf{h}_\mathrm{o}$ or exploiting a speech model.

2.6.1 A Suboptimal Filter

Consider the suboptimal filter:

$$\mathbf{h}_\mathrm{s} = \mathbf{u}_1 - \mathbf{g}_\mathrm{s} = \mathbf{u}_1 - \alpha\mathbf{g}_\mathrm{o}, \tag{2.66}$$

where α is a real number. The MSE of the clean speech estimation corresponding to $\mathbf{h}_\mathrm{s}$ is,

$$\begin{aligned} J_x(\mathbf{h}_\mathrm{s}) &= E\left\{\left[x(n) - \mathbf{h}_\mathrm{s}^T\mathbf{y}(n)\right]^2\right\} \\ &= \sigma_v^2 - \alpha(2-\alpha)\mathbf{r}_v^T\mathbf{R}_y^{-1}\mathbf{r}_v, \end{aligned} \tag{2.67}$$

and, obviously, $J_x(\mathbf{h}_\mathrm{s}) \geq J_x(\mathbf{h}_\mathrm{o}),\ \forall\alpha$; we have equality for $\alpha = 1$. In order to have noise reduction, α must be chosen in such a way that $J_x(\mathbf{h}_\mathrm{s}) < J_x(\mathbf{u}_1)$, therefore,

$$0 < \alpha < 2. \tag{2.68}$$

We can check that,

$$\begin{aligned} J_v(\mathbf{g}_\mathrm{s}) &= E\left\{\left[v(n) - \alpha\mathbf{g}_\mathrm{o}^T\mathbf{y}(n)\right]^2\right\} \\ &= J_x(\mathbf{h}_\mathrm{s}). \end{aligned} \tag{2.69}$$

Let

$$\hat{x}_\mathrm{s}(n) = \mathbf{h}_\mathrm{s}^T\mathbf{y}(n) \tag{2.70}$$

denote the estimation of the clean speech at time n with respect to $\mathbf{h}_\mathrm{s}$. The power of $\hat{x}_\mathrm{s}(n)$ is,

$$\begin{aligned} E\left\{\hat{x}_\mathrm{s}^2(n)\right\} &= \mathbf{h}_\mathrm{s}^T\mathbf{R}_y\mathbf{h}_\mathrm{s} \\ &= \left[\mathbf{u}_1 - \alpha\mathbf{R}_y^{-1}\mathbf{r}_v\right]^T\left[\mathbf{r}_y - \alpha\mathbf{r}_v\right] \\ &= \sigma_x^2 + (1-2\alpha)\sigma_v^2 + \alpha^2\mathbf{r}_v^T\mathbf{R}_y^{-1}\mathbf{r}_v \\ &= \mathbf{h}_\mathrm{s}^T\mathbf{R}_x\mathbf{h}_\mathrm{s} + \mathbf{h}_\mathrm{s}^T\mathbf{R}_v\mathbf{h}_\mathrm{s}. \end{aligned} \tag{2.71}$$

The speech-distortion index corresponding to the filter $\mathbf{h}_\mathrm{s}$ is,

$$\begin{aligned} \upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{s}) &= \frac{E\left\{\left[x(n) - \mathbf{h}_\mathrm{s}^T\mathbf{x}(n)\right]^2\right\}}{\sigma_x^2} \\ &= \alpha^2\mathbf{g}_\mathrm{o}^T\tilde{\mathbf{R}}_x\mathbf{g}_\mathrm{o} = \alpha^2\upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o}). \end{aligned} \tag{2.72}$$

The previous expression shows that the ratio of the speech-distortion indices corresponding to the two filters $\mathbf{g}_\mathrm{s}$ and $\mathbf{g}_\mathrm{o}$ depends on α only.

In order to have less distortion with the suboptimal filter $\mathbf{h}_\mathrm{s}$ than with the Wiener filter $\mathbf{h}_\mathrm{o}$, we must find α in such a way that,

$$\upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{s}) < \upsilon_\mathrm{sd}(\mathbf{g}_\mathrm{o}), \tag{2.73}$$

hence, the condition on α should be

$$-1 < \alpha < 1. \tag{2.74}$$

Finally, the suboptimal filter $\mathbf{h}_\mathrm{s}$ can reduce the level of noise of the observed signal $y(n)$ but with less distortion than the Wiener filter $\mathbf{h}_\mathrm{o}$ if α is taken such as,

$$0 < \alpha < 1. \tag{2.75}$$

For the extreme cases $\alpha = 0$ and $\alpha = 1$ we obtain respectively $\mathbf{h}_\mathrm{s} = \mathbf{u}_1$, no noise reduction at all but no additional distortion added, and $\mathbf{h}_\mathrm{s} = \mathbf{h}_\mathrm{o}$, maximum noise reduction with maximum speech distortion.

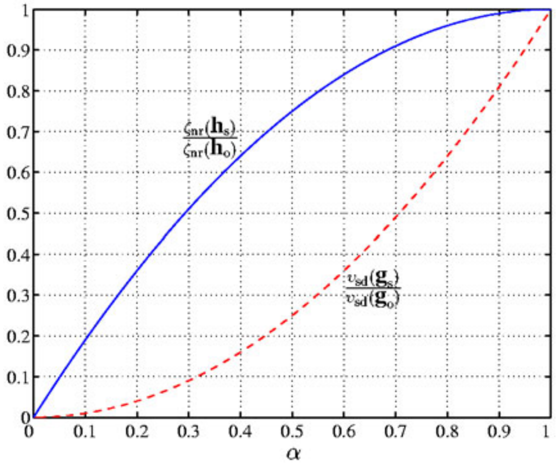


Fig. 2.2. $\upsilon_{\rm sd}(\mathbf{g}_{\rm s})/\upsilon_{\rm sd}(\mathbf{g}_{\rm o})$ (dashed line) and $\zeta_{\rm nr}(\mathbf{h}_{\rm s})/\zeta_{\rm nr}(\mathbf{h}_{\rm o})$ (solid line), both as a function of α.

Since

$$\begin{aligned} J_v(\mathbf{g}_{\rm s}) &= \mathbf{g}_{\rm s}^T \mathbf{R}_x \mathbf{g}_{\rm s} + \mathbf{h}_{\rm s}^T \mathbf{R}_v \mathbf{h}_{\rm s} \\ &= \sigma_x^2 \mathbf{g}_{\rm s}^T \tilde{\mathbf{R}}_x \mathbf{g}_{\rm s} + \sigma_v^2 \mathbf{h}_{\rm s}^T \tilde{\mathbf{R}}_v \mathbf{h}_{\rm s} \\ &= J_x(\mathbf{h}_{\rm s}), \end{aligned} \tag{2.76}$$

it follows immediately that the speech-distortion index and the noise-reduction factor due to $\mathbf{h}_{\rm s}$ are,

$$\upsilon_{\rm sd}(\mathbf{g}_{\rm s}) = \frac{1}{\rm SNR}\left[\tilde{J}_x(\mathbf{h}_{\rm s}) - \frac{1}{\xi_{\rm nr}(\mathbf{h}_{\rm s})}\right], \tag{2.77}$$

$$\xi_{\rm nr}(\mathbf{h}_{\rm s}) = \frac{1}{{\rm SNR}\left[\tilde{J}_v(\mathbf{g}_{\rm s}) - \upsilon_{\rm sd}(\mathbf{g}_{\rm s})\right]}. \tag{2.78}$$

Unlike $\upsilon_{\rm sd}(\mathbf{g}_{\rm s})/\upsilon_{\rm sd}(\mathbf{g}_{\rm o})$ which depends on α only, $\xi_{\rm nr}(\mathbf{h}_{\rm s})/\xi_{\rm nr}(\mathbf{h}_{\rm o})$ does not. However, using (2.67) and (2.15), we find that,

$$\frac{\zeta_{\rm nr}(\mathbf{h}_{\rm s})}{\zeta_{\rm nr}(\mathbf{h}_{\rm o})} = \frac{1 - \tilde{J}_x(\mathbf{h}_{\rm s})}{1 - \tilde{J}_x(\mathbf{h}_{\rm o})} = \alpha(2-\alpha). \tag{2.79}$$

Figure 2.2 plots $\upsilon_{\rm sd}(\mathbf{g}_{\rm s})/\upsilon_{\rm sd}(\mathbf{g}_{\rm o})$ and $\zeta_{\rm nr}(\mathbf{h}_{\rm s})/\zeta_{\rm nr}(\mathbf{h}_{\rm o})$ as functions of α. For example, for $\alpha = 0.7$, we see that the speech-distortion index with the suboptimal filter represents 49% of the speech-distortion index with the Wiener filter while the noise-reduction index is 91%.

2.6.2 Noise Reduction Exploiting the Speech Model

Section 2.5 has shown that the Wiener filter is more efficient to attenuate the level of noise for predictable signals than for unpredictable ones. In fact, it is well known that speech can be represented by an autoregressive (AR) process; thus, speech can be seen as the output of an all-pole linear system where the input is a zero-mean white Gaussian process, $w(n)$, with variance σ_w^2. The clean speech signal is then given by,

$$\begin{aligned} x(n) &= \sum_{l=1}^{L} a_{x,l} x(n-l) + w(n) \\ &= \mathbf{a}_x^T \mathbf{x}(n-1) + w(n), \end{aligned} \tag{2.80}$$

where $a_{x,l}$ are the parameters of the AR process. This model is very often combined with the Kalman filter to enhance a noisy speech signal; see, for example, [27], [28], and [29]. The main challenge in this approach is to get an accurate estimate of the AR parameters from the observations.

We can use this model in the Wiener context with some advantages. For that, in this section, we assume that the additive noise, $v(n)$, of the observed signal, $y(n)$, is white. The cross-correlation vector, $\mathbf{p}$, between the noisy and clean speech signals that appears in the Wiener-Hopf equation is now:

$$\begin{aligned} \mathbf{p} &= E\{\mathbf{y}(n)x(n)\} = E\left\{\mathbf{y}(n)\mathbf{x}^T(n-1)\right\}\mathbf{a}_x + \sigma_w^2 \mathbf{u}_1 \\ &= E\left\{\mathbf{y}(n)\left[\mathbf{y}(n-1) - \mathbf{v}(n-1)\right]^T\right\}\mathbf{a}_x + \sigma_w^2 \mathbf{u}_1 \\ &= (\mathbf{R}_{y,1} - \mathbf{R}_{v,1})\mathbf{a}_x + \sigma_w^2 \mathbf{u}_1, \end{aligned} \tag{2.81}$$

where

$$\mathbf{R}_{y,1} \stackrel{\triangle}{=} E\left\{\mathbf{y}(n)\mathbf{y}^T(n-1)\right\},$$

and

$$\begin{aligned} \mathbf{R}_{v,1} &\stackrel{\triangle}{=} E\left\{\mathbf{v}(n)\mathbf{v}^T(n-1)\right\} \\ &= \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ \sigma_v^2 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \sigma_v^2 & 0 & \cdots & 0 & 0 \\ & \vdots & & & \vdots & \\ 0 & 0 & 0 & \cdots & \sigma_v^2 & 0 \end{bmatrix}. \end{aligned}$$

We deduce the optimal filter:

$$\mathbf{h}_\text{o} = \mathbf{R}_y^{-1}(\mathbf{R}_{y,1} - \mathbf{R}_{v,1})\mathbf{a}_x + \sigma_w^2 \mathbf{R}_y^{-1}\mathbf{u}_1. \tag{2.82}$$

Equation (2.82) shows the relationship between the Wiener filter and the AR parameters of the clean speech signal. When $v(n)$ is a white Gaussian noise signal, (2.82) yields similar results as in (2.64).

2.6.3 Noise Reduction with Multiple Microphones

In more and more applications, multiple microphone signals are available. Therefore, it is interesting to investigate deeply the multichannel case. One of the first papers to do so is a paper written by Doclo and Moonen [42], where the optimal filter is derived as well as a general class of estimators. The authors also show how the generalized singular value decomposition can be used in this spatio-temporal technique. In this section, we take a slightly different approach. We will see, in particular, that we can reduce the level of noise without distorting the speech signal. This result was never observed before.

We suppose that we have a linear array consisting of M microphones whose outputs are denoted as $y_m(n)$, $m = 0, 1, \cdots, M-1$. Without loss of generality, we select microphone 0 as the reference point and to simplify the analysis, we consider the following propagation model:

$$y_m(n) = \beta_m s(n - t - \tau_m) + v_m(n), \; m = 0, 1, \cdots, M-1, \tag{2.83}$$

where β_m is the attenuation factor (with $\beta_0 = 1$), t is the propagation time from the unknown speech source $s(n)$ to microphone 0, $v_m(n)$ is an additive noise signal at the mth microphone, and τ_m is the relative delay between microphones 0 and m, with $\tau_0 = 0$.

In the following, we assume that the relative delays τ_m, $m = 1, \cdots, M-1$, are known or can easily be estimated. So our first step is the design of a simple delay-and-sum beamformer, which spatially aligns the microphone signals to the direction of the speech source. From now on, we will work on the aligned signals:

$$\begin{aligned} z_m(n) &= y_m(n + \tau_m) \\ &= \beta_m s(n - t) + v_m(n + \tau_m), \\ &= x_m(n) + v_m(n + \tau_m), \; m = 0, 1, \cdots, M-1. \end{aligned} \tag{2.84}$$

A straightforward approach for noise reduction is to average the M signals $z_m(n)$,

$$z_\mathrm{a}(n) = \frac{1}{M} \sum_{m=0}^{M-1} z_m(n) = \frac{\beta_\mathrm{a}}{M} s(n - t) + \frac{1}{M} \sum_{m=0}^{M-1} v_m(n + \tau_m), \tag{2.85}$$

where $\beta_\mathrm{a} = \sum_{m=0}^{M-1} \beta_m$. If the noises are added incoherently, the output SNR will, in principle, increase [44]. We can further reduce the noise by passing the signal $z_\mathrm{a}(n)$ through a Wiener filter as was shown in the previous sections. This approach has, however, two drawbacks. The first one is that, since for $m \neq i$, $E\{v_m(n + \tau_m) v_i(n + \tau_i)\} \neq 0$ in general, the output SNR will not improve that much; and the second one, as we know already, is speech distortion that the optimal filter introduces.

Let us now define the error signal, for the mth microphone, between the clean speech sample $x_m(n)$ and its estimate as,

$$\begin{aligned} e_{x_m}(n) &\overset{\triangle}{=} x_m(n) - \mathbf{h}_{:m}^T \mathbf{z}(n) \qquad (2.86) \\ &= x_m(n) - \sum_{i=0}^{M-1} \mathbf{h}_{i:m}^T \mathbf{z}_i(n), \end{aligned}$$

where $\mathbf{h}_{i:m}$ are filters of length L and,

$$\begin{aligned} \mathbf{h}_{:m} &\overset{\triangle}{=} \left[\mathbf{h}_{0:m}^T \ \mathbf{h}_{1:m}^T \ \cdots \ \mathbf{h}_{M-1:m}^T \right]^T, \\ \mathbf{z}(n) &\overset{\triangle}{=} \left[\mathbf{z}_0^T(n) \ \mathbf{z}_1^T(n) \ \cdots \ \mathbf{z}_{M-1}^T(n) \right]^T. \end{aligned}$$

Since $\mathbf{z}_i(n) = \beta_i \mathbf{s}(n-t) + \mathbf{v}_i(n+\tau_i)$, (2.86) becomes:

$$\begin{aligned} e_{x_m}(n) &= \mathbf{s}^T(n-t) \left[\beta_m \mathbf{u}_1 - \sum_{i=0}^{M-1} \beta_i \mathbf{h}_{i:m} \right] - \sum_{i=0}^{M-1} \mathbf{v}_i^T(n+\tau_i) \mathbf{h}_{i:m} \\ &= \mathbf{s}^T(n-t) \left[\beta_m \mathbf{u}_1 - \mathbf{D}\mathbf{h}_{:m} \right] - \mathbf{v}^T(n) \mathbf{h}_{:m} \\ &= e_{s,m}(n) - e_{v,m}(n), \qquad (2.87) \end{aligned}$$

where

$$\begin{aligned} \mathbf{D} &\overset{\triangle}{=} \left[\beta_0 \mathbf{I} \ \beta_1 \mathbf{I} \ \cdots \ \beta_{M-1} \mathbf{I} \right], \\ \mathbf{v}(n) &\overset{\triangle}{=} \left[\mathbf{v}_0^T(n+\tau_0) \ \mathbf{v}_1^T(n+\tau_1) \ \cdots \ \mathbf{v}_{M-1}^T(n+\tau_{M-1}) \right]^T. \end{aligned}$$

Expression (2.87) is the difference between two error signals; $e_{s,m}(n)$ represents signal distortion and $e_{v,m}(n)$ represents the residual noise. The MSE corresponding to the residual noise with the mth microphone as the reference signal is,

$$\begin{aligned} J_{v,m}(\mathbf{h}_{:m}) &= E\left\{ e_{v,m}^2(n) \right\} \\ &= \mathbf{h}_{:m}^T E\left\{ \mathbf{v}(n)\mathbf{v}^T(n) \right\} \mathbf{h}_{:m} \\ &= \mathbf{h}_{:m}^T \mathbf{R}_v \mathbf{h}_{:m}. \qquad (2.88) \end{aligned}$$

Usually, in the single-channel case, the minimization of the MSE corresponding to the residual noise is done while keeping the signal distortion below a threshold [20]. With no distortion, the optimal filter obtained from this optimization is $\mathbf{u}_1$, hence there is not any noise reduction either. The advantage of multiple microphones is that, actually, we can minimize $J_{v,m}(\mathbf{h}_{:m})$ with the constraint that $\beta_m \mathbf{u}_1 = \mathbf{D}\mathbf{h}_{:m}$ (no speech distortion at all). Therefore, our optimization problem is,

$$\min_{\mathbf{h}_{:m}} J_{v,m}(\mathbf{h}_{:m}) \text{ subject to } \beta_m \mathbf{u}_1 = \mathbf{D}\mathbf{h}_{:m}. \qquad (2.89)$$

By using a Lagrange multiplier, we easily find the optimal solution:

$$\mathbf{h}_{\mathrm{o},:m} = \beta_m \mathbf{R}_v^{-1}\mathbf{D}^T \left[\mathbf{D}\mathbf{R}_v^{-1}\mathbf{D}^T\right]^{-1} \mathbf{u}_1, \tag{2.90}$$

where we assumed that the noise signals $v_i(n)$ are not perfectly coherent so that $\mathbf{R}_v$ is not singular.

The MMSE for the mth microphone is,

$$J_{v,m}(\mathbf{h}_{\mathrm{o},:m}) = \beta_m^2 \mathbf{u}_1^T \left[\mathbf{D}\mathbf{R}_v^{-1}\mathbf{D}^T\right]^{-1} \mathbf{u}_1. \tag{2.91}$$

Since we have M microphones, we have M MMSEs as well. The best MMSE from a noise reduction point of view is the smallest one, which is, according to (2.91), the microphone signal with the smallest attenuation factor.

The attenuation factors β_m can be easily determined, if the power of the noise signals are known, by using the formula:

$$\beta_m^2 = \frac{E\{z_m^2(n)\} - E\{v_m^2(n+\tau_m)\}}{E\{z_0^2(n)\} - E\{v_0^2(n)\}}, \; m = 1, 2, \cdots, M-1. \tag{2.92}$$

For the particular case where the noise is spatio-temporally white with a power equal to σ_v^2, the MMSE and the normalized MMSE for the mth microphone are respectively,

$$J_{v,m}(\mathbf{h}_{\mathrm{o},:m}) = \sigma_v^2 \frac{\beta_m^2}{\sum_{i=0}^{M-1} \beta_i^2}, \tag{2.93}$$

$$\tilde{J}_{v,m}(\mathbf{h}_{\mathrm{o},:m}) = \frac{\beta_m^2}{\sum_{i=0}^{M-1} \beta_i^2}. \tag{2.94}$$

We can see that when the number of microphones goes to infinity, the normalized MMSE goes to zero, which means that the noise can be completely removed with no signal distortion at all.

2.7 Simulation Experiments

By defining a noise-reduction factor to quantify the amount of noise being attenuated and a speech-distortion index to valuate the degree to which the speech signal is deformed, we have analytically examined the performance behavior of the Wiener-filter-based noise reduction technique. It is shown that the Wiener filter achieves noise reduction by distorting the speech signal. The more the noise is reduced, the more the speech is distorted. We also proposed several approaches to better manage the tradeoff between noise reduction and speech distortion. To further verify the analysis, and to assess the noise-reduction-and-speech-distortion management schemes, we implemented a time-domain Wiener-filter system. The sampling rate is 8 kHz. The noise signal is estimated in the time-frequency domain using a sequential algorithm presented in [6], [7]. Briefly, this algorithm obtains an estimate of noise using the overlap-add technique on a frame-by-frame basis. The noisy speech signal $y(n)$ is segmented into frames with a frame width of 8 milliseconds and an overlapping factor of 75%. Each frame is then transformed via a DFT into a block of spectral samples. Successive blocks of spectral samples form a two-dimensional time-frequency matrix denoted by $Y_t(j\omega)$, where subscript t is the frame index, denoting the time dimension, and ω is the angular frequency. Then an estimate of the magnitude of the noise spectrum is formulated as

$$\hat{V}_t(\omega) = \begin{cases} \alpha_{\mathrm{a}}\hat{V}_{t-1}(\omega) + (1-\alpha_{\mathrm{a}})|Y_t(j\omega)|, & \text{if } |Y_t(j\omega)| \geq \hat{V}_{t-1}(\omega) \\ \alpha_{\mathrm{d}}\hat{V}_{t-1}(\omega) + (1-\alpha_{\mathrm{d}})|Y_t(j\omega)|, & \text{if } |Y_t(j\omega)| < \hat{V}_{t-1}(\omega) \end{cases}, \quad (2.95)$$

where α_{a} and α_{d} are the the "attack" and "decay" coefficients respectively. Meanwhile, to reduce its temporal fluctuation, the magnitude of the noisy speech spectrum is smoothed according to the following recursion:

$$\bar{Y}_t(\omega) = \begin{cases} \beta_{\mathrm{a}}\bar{Y}_{t-1}(\omega) + (1-\beta_{\mathrm{a}})|Y_t(j\omega)|, & \text{if } |Y_t(j\omega)| \geq \bar{Y}_{t-1}(\omega) \\ \beta_{\mathrm{d}}\bar{Y}_{t-1}(\omega) + (1-\beta_{\mathrm{d}})|Y_t(j\omega)|, & \text{if } |Y_t(j\omega)| < \bar{Y}_{t-1}(\omega) \end{cases}, \quad (2.96)$$

where again β_{a} is the "attack" coefficient and β_{d} the "decay" coefficient. To further reduce the spectral fluctuation, both $\hat{V}_t(\omega)$ and $\bar{Y}_t(\omega)$ are averaged across the neighboring frequency bins around ω. Finally, an estimate of the noise spectrum is obtained by multiplying $\hat{V}_t(\omega)/\bar{Y}_t(\omega)$ with $Y_t(j\omega)$, and the time-domain noise signal is obtained through IDFT and the overlap-add technique. See [6], [7] for more detailed description of this noise-estimation scheme. Figure 2.3 shows a speech signal corrupted by a car noise (SNR = 10 dB), the waveform and the spectrogram of the car noise that is added to the speech, and the waveform and spectrogram of the noise estimate. It can be seen that during the absence of speech, the estimate is a good approximation of the noise signal. It is also noticed from its spectrogram that the noise estimate consists of some minor speech components during the presence of speech. Our listening test, however, shows that the residual speech remained in the noise estimate is almost inaudible. An apparent advantage of this noise-estimation technique is that it does not require an explicit voice activity detector. In addition, our experimental investigation reveals that such a scheme is able to capture the noise characteristics in both the presence and absence of speech, therefore it does not rely on the assumption that the noise characteristics in the presence of speech stay the same as in the absence of speech.

Based on the implemented system, we evaluate the Wiener filter for noise reduction. The first experiment investigates the influence of the filter length on the noise reduction performance. Instead of using the estimated noise, here we assume that the noise signal is known *a priori*. Therefore this experiment demonstrates the upper limit of the performance of the Wiener filter. We consider two cases. In the first one, both the source signal and the background

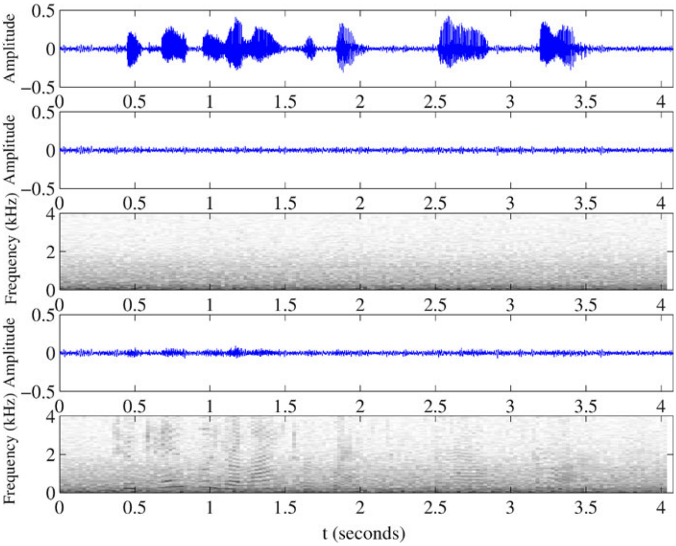


Fig. 2.3. Noise and its estimate. The first trace (from the top) shows the waveform of a speech signal corrupted by a car noise where SNR = 10 dB. The second and third traces plot the waveform and spectrogram of the noise signal. The fourth and fifth traces display the waveform and spectrogram of the noise estimate.

noise are random processes in which the current value of the signal cannot be predicted from its past samples. The source signal is a noise signal recorded from a New York Stock Exchange (NYSE) room. This signal consists of sound from various sources such as speakers, telephone rings, electric fans, etc. The background noise is a computer-generated Gaussian random process. The results for this case is graphically portrayed in Fig. 2.4. It can be seen that both the noise-reduction factor and the speech-distortion index increase linearly with the filter length. Therefore, a longer filter should be applied for more noise reduction. However, the more the noise is attenuated, the more the source signal is deformed, as shown in Fig. 2.4.

In the second case, we test the Wiener filter for noise reduction in the context of speech signal. It is known that a speech signal can be modelled as an AR process, where its current value can be predicted from its past samples. To simplify the situation for the ease of analysis, the source signal used here is an /i:/ sound recorded from a female speaker. Same as in the previous case, the background noise is a computer-generated white Gaussian random process. The results are plotted in Fig. 2.5. Again, the noise-reduction factor, which quantifies the amount of noise being attenuated, increases monotonically with the filter length; but unlike the previous case, the relationship between the noise reduction and the filter length is not linear. Instead, the curve at first grows quickly as the filter length is increased up to 10, and then

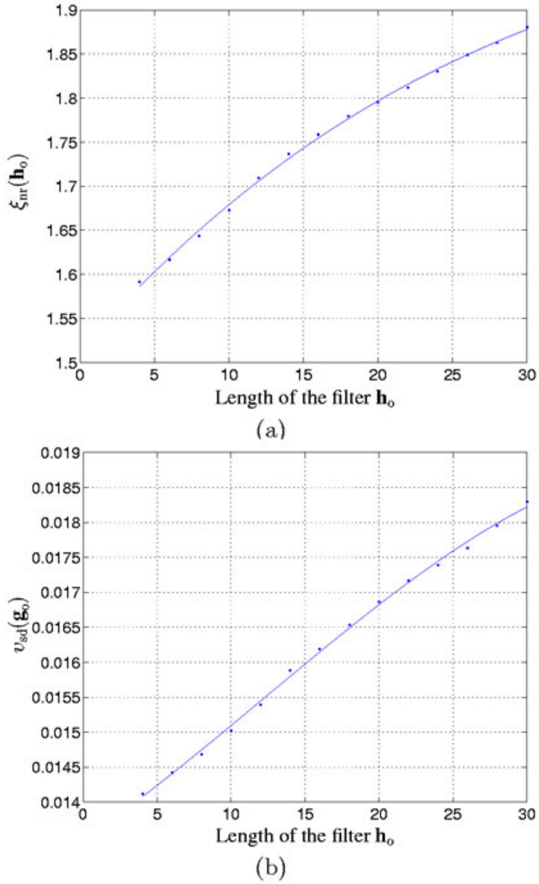
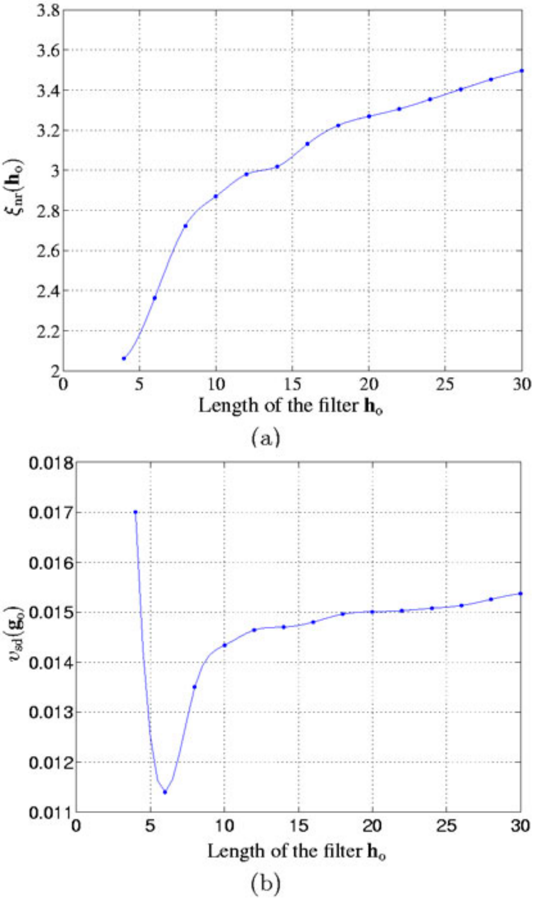


Fig. 2.4. Noise-reduction factor and signal-distortion index, both as a function of the filter length: (a) noise reduction; (b) signal distortion. The source is a signal recorded in a NYSE room; the background noise is a computer-generated white Gaussian random process; and SNR = 10 dB.

continues to grow but with a slower rate. Unlike ξ_{nr}, the speech-distortion index, i.e., υ_{sd}, exhibits a non-monotonic relationship with the filter length. It first decreases to its minimum, and then increases again as the filter length is increased. The reason, as we have explained in Section 2.6.2, is that a speech signal can be modelled as an AR process. Particular to this experiment, the /i:/ sound used here can be well modelled with a 6^{th} order LPC (linear prediction coding) analysis. Therefore, when the filter length is increased to 6, the numerator of (2.33) is minimized, as a result, the speech-distortion index reaches its minimum. Continuing to increase the filter length leads to a higher distortion due to more noise reduction. To further verify this observation, we

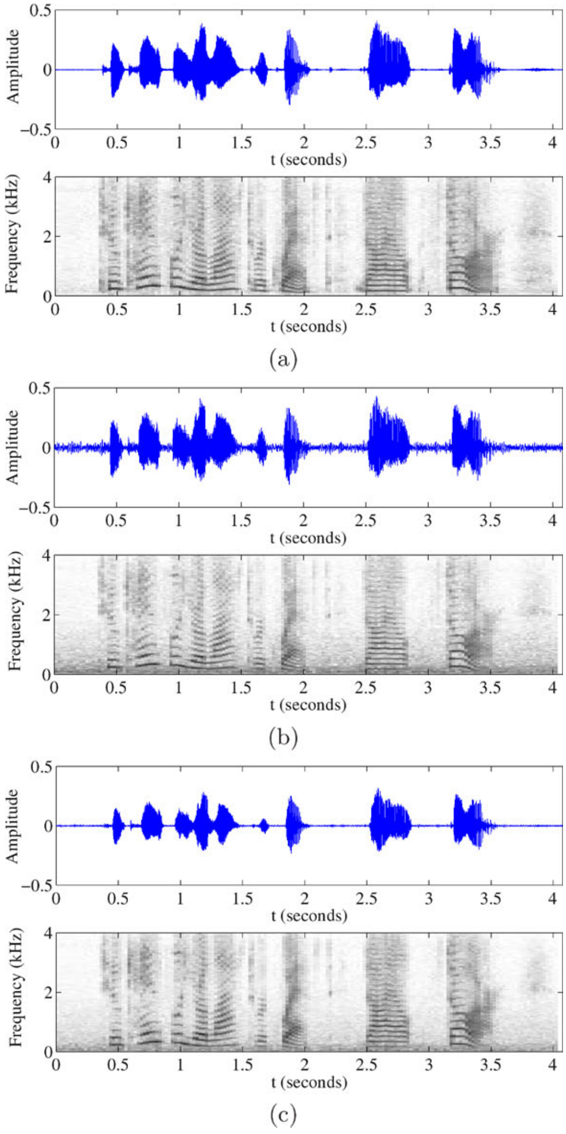


(a)

(b)

Fig. 2.5. Noise-reduction factor and signal-distortion index, both as a function of the filter length: (a) noise reduction; (b) speech distortion. The source signal is an /i:/ sound from a female speaker; the background noise is a computer-generated white Gaussian process; and SNR = 10 dB.

investigated several other vowels, and found that the curve of $\upsilon_{\rm sd}$ vs. filter length follows a similar shape, except that the minimum may appear in a slightly different location. Taking into account the sounds other than vowels in speech that may be less predicable, we find that good performance with the Wiener filter (in terms of the compromise between noise reduction and speech distortion) can be achieved when filter length L is chosen around 20. Figure 2.6 plots the output of our Wiener filter system with $L = 20$, where the speech signal is from a female speaker, the background noise is a car noise signal, and SNR = 10 dB.



(a)

(b)

(c)

Fig. 2.6. Noise reduction in a car noise condition where SNR = 10 dB: (a) clean speech and its spectrogram; (b) noisy speech and its spectrogram; (c) noise reduced speech and its spectrogram.

The second experiment tests the noise reduction performance in different SNR conditions. Here the speech signal is recorded from a female speaker as shown in Fig. 2.6. The computer-generated random Gaussian noise is added to the speech signal to control the SNR. The length of the Wiener filter is set to $L = 20$. The results are presented in Fig.2.7, where besides ξ_{nr} and υ_{sd}, we also plotted the Itakura-Saito (IS) distance, a widely used objective quality measure that performs a comparison of spectral envelopes (AR parameters) between the clean and the processed speech [46]. Studies have shown that the IS measure is highly correlated (0.59) with the subjective quality judgements [47]. A recent report reveals that the difference in mean opinion score (MOS) between two processed speech signals would be less than 1.6 if their IS measure is less than 0.5 for various codecs [48]. Many other reported experiments confirmed that two spectra would be perceptually nearly identical if their IS distance is less than 0.1. All these evidences indicates that the IS distance is a reasonably good objective measure of speech quality.

As SNR decreases, the observation signal becomes more noisy. Therefore the Wiener filter is expected to have more noise reduction for low SNRs. This is verified by Fig. 2.7 (a), where significant noise reduction is obtained for low SNR conditions. However, more noise reduction would correspond to more speech distortion. This is confirmed by Fig. 2.7 (b) and (d) where both the speech-distortion index and the IS distance increase as speech becomes more noisy. Comparing the IS distance before [Fig. 2.7 (c)] and after [Fig. 2.7 (d)] noise reduction, one can see that significant gain in the IS distance has been achieved, indicating that the Wiener filter is able to reduce noise and improve speech quality (but not necessarily speech intelligibility).

The last experiment is to verify the performance behavior of the suboptimal filter derived in Section 2.6.1. The experimental conditions are the same as outlined in the previous experiment. The results are presented in Table 2.1, where for the purpose of comparison, besides the speech-distortion index and the noise-reduction factor, we also show three IS distances (between the clean and filtered speeches denoted as ISD^1, between the clean and noise-reduced speeches marked as ISD^2, and between the clean and noisy speeches denoted as ISD^3, respectively). From the results, one can make the following observations:

- The IS distance between the clean and noisy speech signals increases as SNR drops. The reason for this is apparent. When SNR decreases, the speech signal becomes more noisy. As a result, the difference between the spectral envelope (or AR parameters) of the clean speech and that (or those) of the noisy speech tends to be more significant, which leads to a higher IS distance.
- ISD^2 is much smaller than ISD^3. This significant gain in IS distance indicates that the use of noise reduction technique is able to mitigate noise and improve speech quality.
- A better compromise between noise reduction and speech distortion is accomplished by using the suboptimal filter. For example, when SNR = 20 dB, the speech-distortion index for the suboptimal filter with $\alpha = 0.7$ is 0.0006, which is only 54% of that of the Wiener filter; the corresponding

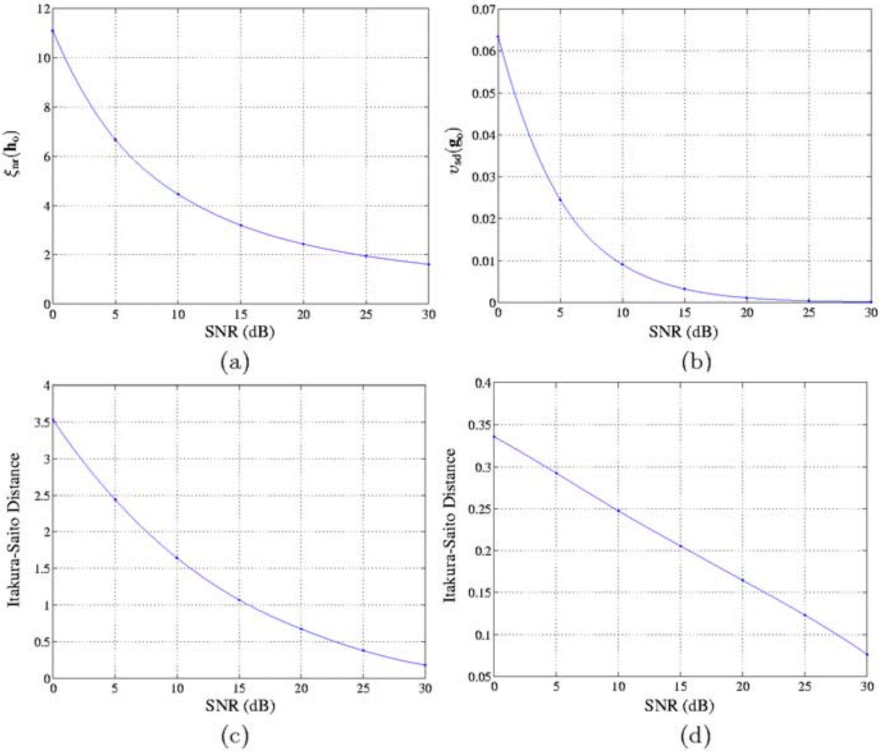


Fig. 2.7. Noise reduction performance as a function of SNR in white Gaussian noise: (a) noise-reduction factor; (b) speech-distortion index; (c) Itakura-Saito distance between the clean and noisy speeches; (d) Itakura-Saito distance between the clean and noise-reduced speeches.

IS distance between the clean and filtered speech is 0.0281, which is only 17% of that of the Wiener filter; but it has achieved a noise reduction of 2.0106, which is 82% of that with the Wiener filter.

- Different from ISD^1, which decreases with α, ISD^2 increases when a smaller α is selected. This is due to the fact that ISD^2 is affected by both speech distortion and the residual noise remained in the noise-reduced speech. As elaborated in Section 2.6.1, as long as α satisfies $0 \leq \alpha \leq 1$, a smaller α would lead to less speech distortion; but a smaller α also means that more residual noise will remain in the noise-reduced speech. While the former may reduce the IS distance, the latter will enlarge the IS distance. As a result, ISD^2 increases when a smaller α is chosen.
- From the analysis shown in Section 2.6.1, we see that both $\frac{\upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{s}})}{\upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{o}})}$ and $\frac{\zeta_{\mathrm{nr}}(\mathbf{h}_{\mathrm{s}})}{\zeta_{\mathrm{nr}}(\mathbf{h}_{\mathrm{o}})}$ are independent of SNR but not $\frac{\xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{s}})}{\xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{o}})}$. From the experimental results, we notice that the ratio between $\upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{s}})$ and $\upsilon_{\mathrm{sd}}(\mathbf{g}_{\mathrm{o}})$ does not

Table 2.1 Noise reduction performance with the suboptimal filter, where ISD^1 is the IS distance between the clean speech [i.e., $x(n)$] and the filtered version of the clean speech [i.e., $\mathbf{h}^T\mathbf{x}(n)$], which purely measures the speech distortion due to the filtering effect; ISD^2 is the IS distance between the clean and noise-reduced speeches; ISD^3 is the IS distance between the clean and noisy speech signals.

SNR		υ_{sd}	ξ_{nr}	ISD^1	ISD^2	ISD^3
20dB	Wiener filter	0.0011	2.4390	0.1691	0.1471	0.6727
	Suboptimal filter ($\alpha = 0.8$)	0.0007	2.1753	0.0423	0.2820	0.6727
	Suboptimal filter ($\alpha = 0.7$)	0.0006	2.0106	0.0281	0.3476	0.6727
15dB	Wiener filter	0.0033	3.1977	0.2133	0.2032	1.0446
	Suboptimal filter ($\alpha = 0.8$)	0.0021	2.7379	0.0488	0.5114	1.0446
	Suboptimal filter ($\alpha = 0.7$)	0.0016	2.4544	0.0352	0.6034	1.0446
10dB	Wiener filter	0.0092	4.4565	0.2622	0.2652	1.5458
	Suboptimal filter ($\alpha = 0.8$)	0.0059	3.5896	0.0582	0.7759	1.5458
	Suboptimal filter ($\alpha = 0.7$)	0.0045	3.0807	0.0441	0.8917	1.5458

vary much when SNR is changed; but $\frac{\xi_{\text{nr}}(\mathbf{h}_{\text{s}})}{\xi_{\text{nr}}(\mathbf{h}_{\text{o}})}$ decreases with SNR. For examples, when $\alpha = 0.7$, the ratio calculated from experiment is 0.82 when SNR = 20 dB, and is 0.77 when SNR = 15 dB. From numerous experiments, we noticed that the speech distortion and noise reduction satisfy $\frac{\xi_{\text{nr}}(\mathbf{h}_{\text{s}})}{\upsilon_{\text{sd}}(\mathbf{g}_{\text{s}})} > \frac{\xi_{\text{nr}}(\mathbf{h}_{\text{o}})}{\upsilon_{\text{sd}}(\mathbf{g}_{\text{o}})}$ if SNR > 5 dB, which indicates that the suboptimal filter can be used to control the tradeoff between noise reduction and speech distortion as long as SNR > 5 dB. The higher is the SNR, the more effective will the suboptimal filter work.

2.8 Conclusions

The problem of speech enhancement has attracted a considerable amount of research attention over the past several decades. Numerous techniques were developed, among them is the optimal Wiener filter, which is the most fundamental approach. It is widely noticed that the Wiener filter achieves noise reduction by deforming the speech signal. However, so far not much has been said on how the Wiener filter really works. This chapter was devoted to analyzing the intrinsic relationship between noise reduction and speech distortion with the Wiener filter. Starting from the speech and noise estimation using the Wiener theory, we introduced a speech-distortion index and a noise-reduction factor. We showed that for the single-channel Wiener filter, the amount of noise attenuation is in general proportionate to the amount of speech degradation, i.e., more noise reduction incurs more speech distortion.

Depending on the nature of the application, some practical noise-reduction systems may require very high-quality speech, but can tolerate a certain amount of noise. While others may want speech as clean as possible even with some degree of speech distortion. Therefore it is necessary that we can have some management schemes to control the contradicting requirements between noise reduction and speech distortion. To do so, we have discussed three approaches. When there is no *a priori* knowledge or no additional information available, a sub-optimal filter with one more free parameter can be used. By setting the free parameter to 0.7, we showed that the sub-optimal filter can achieve 90% of the noise reduction that the Wiener filter can have; but the resulting speech distortion is less than half of that of the Wiener filter. Speech signal can be modeled as an autoregressive (AR) process. If the AR coefficients can be estimated reliably, we showed that these coefficients can be used to construct the Wiener filter for less speech distortion. In scenarios where we can have multiple noisy realizations of the speech signal, then spatio-temporal filtering techniques can be exploited to obtain noise reduction with less or even no speech distortion.

References

1. M. R. Schroeder, U.S. Patent No 3,180,936, filed Dec. 1, 1960, issued Apr. 27, 1965.
2. M. R. Schroeder, U.S. Patent No 3,403,224, filed May 28, 1965, issued Sept. 24, 1968.
3. J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proceedings of the IEEE*, vol. 67, pp. 1586–1604, Dec. 1979.
4. J. S. Lim, *Speech Enhancement*. Englewood Cliffs, NJ: Prentice-Hall, 1983.
5. Y. Ephraim, "Statistical-model-based speech enhancement systems," *Proceedings of the IEEE*, vol. 80, pp. 1526–1554, Oct. 1992.
6. 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., pp. 91–115, Boston, MA: Kluwer, 2004.
7. J. Chen, Y. Huang, and J. Benesty, "Filtering techniques for noise reduction and speech enhancement," in *Adaptive Signal Processing: Applications to Real-World Problems*, J. Benesty and Y. Huang, Eds., pp. 129–154, Berlin, Germany: Springer, 2003.
8. B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1985.

9. S. F. Boll, “Suppression of acoustic noise in speech using spectral subtraction,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 27, pp. 113–120, Apr. 1979.
10. R. J. McAulay and M. L. Malpass, “Speech enhancement using a soft-decision noise suppression filter,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, pp. 137–145, Apr. 1980.
11. P. Vary, “Noise suppression by spectral magnitude estimation–mechanism and theoretical limits,” *Signal Processing*, vol. 8, pp. 387–400, July 1985.
12. R. Martin, “Noise power spectral density estimation based on optimal smoothing and minimum statistics,” *IEEE Trans. Speech Audio Processing*, vol. 9, pp. 504–512, July 2001.
13. 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.
14. D. L. Wang and J. S. Lim, “The unimportance of phase in speech enhancement,” *IEEE Trans. Acoustic. Speech, Signal Processing*, vol. 30, pp. 679–681, Aug. 1982.
15. Y. Ephraim and D. Malah, “Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 32, pp. 1109–1121, Dec. 1984.
16. Y. Ephraim and D. Malah,“Speech enhancement using a minimum mean-square error log-spectral amplitude estimator,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, pp. 443–445, Apr. 1985.
17. N. Virag, “Single channel speech enhancement basd on masking properties of human auditory system,” *IEEE Trans. Speech Audio Processing*, vol. 7, pp. 126–137, Mar. 1999.
18. Y. M. Chang and D. O’Shaughnessy, “Speech enhancement based conceptually on auditory evidence,” *IEEE Trans. Signal Processing*, vol. 39, pp. 1943–1954, Sept. 1991.
19. T. F. Quatieri and R. B. Dunn, “Speech enhancement based on auditory spectral change,” in *Proc. IEEE ICASSP*, vol. 1, 2002, pp. 257–260.
20. Y. Ephraim and H. L. Van Trees, “A signal subspace approach for speech enhancement,” *IEEE Trans. Speech Audio Processing*, vol. 3, pp. 251–266, July 1995.
21. M. Dendrinos, S. Bakamidis, and G. Garayannis, “Speech enhancement from noise: a regenerative approach,” *Speech Commun.*, vol. 10, pp. 45–57, Feb. 1991.
22. P. S. K. Hansen, *Signal Subspace Methods for Speech Enhancement.* Ph.D. dissertation, Techn. Univ. Denmark, Lyngby, Denmark, 1997.
23. H. Lev-Ari and Y. Ephraim, “Extension of the signal subspace speech enhancement approach to colored noise,” *IEEE Trans. Speech Audio Processing*, vol. 10, pp. 104–106, Apr. 2003.
24. A. Rezayee and S. Gazor, “An adpative KLT approach for speech enhancement,” *IEEE Trans. Speech Audio Processing*, vol. 9, pp. 87–95, Feb. 2001.
25. U. Mittal and N. Phamdo, “Signal/noise KLT based approach for enhancing speech degraded by colored noise,” *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 159–167, Mar. 2000.
26. Y. Hu and P. C. Loizou, “A generalized subspace approach for enhancing spech corrupted by colored noise,” *IEEE Trans. Speech Audio Processing*, vol. 11, pp. 334–341, July 2003.
27. K. K. Paliwal and A. Basu, “A speech enhancement method based on Kalman filtering,” in *Proc. IEEE ICASSP*, 1987, pp. 177–180.

28. J. D. Gibson, B. Koo, and S. D. Gray, "Filtering of colored noise for speech enhancement and coding," *IEEE Trans. Signal Processing*, vol. 39, pp. 1732–1742, Aug. 1991.
29. M. Gabrea, E. Grivel, and M. Najim, "A single microphone Kalman filter-based noise canceller," *IEEE Trans. Signal Processing Lett.*, vol. 6, pp. 55–57, Mar. 1999.
30. B. Lee, K. Y. Lee, and S. Ann, "An EM-based approach for parameter enhancement with an application to speech signals," *Signal Processing*, vol. 46, pp. 1–14, Sept. 1995.
31. S. Gannot, D. Burshtein, and E. Weinstein, "Iterative and sequential Kalman filter-based speech enhancement algorithms," *IEEE Trans. Speech Audio Processing*, vol. 6, pp. 373–385, July 1998.
32. Y. Ephraim, D. Malah, and B.-H. Juang, "On the application of hidden Markov models for enhancing noisy speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1846–1856, Dec. 1989.
33. Y. Ephraim, "A Bayesian estimation approach for speech enhancement using hidden Markov models," *IEEE Trans. Signal Processing*, vol. 40, pp. 725–735, Apr. 1992.
34. J. Vermaak, C. Andrieu, A. Doucet, and S. J. Godsill, "Particle methods for Bayesian modeling and enhancement of speech signals," *IEEE Trans. Speech Audio Processing*, vol. 10, pp. 173–185, Mar. 2002.
35. H. Sameti, H. Sheikhzadeh, L. Deng, and R. L. Brennan, "HMM-based strategies for enhancement of speech signals embedded in nonstationary noise," *IEEE Trans. Speech Audio Processing*, vol. 6, pp. 445–455, Sept. 1998.
36. D. Burshtein and S. Gannot, "Speech enhancement using a mixture-maximum model," *IEEE Trans. Speech Audio Processing*, vol. 10, pp. 341–351, Sept. 2002.
37. J. Vermaak and M. Niranjan, "Markov chain Monte Carlo methods for speech enhancement," in *Proc. IEEE ICASSP*, vol. 2, 1998, pp. 1013–1016.
38. S. Gannot, D. Burshtein, and E. Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Trans. Signal Processing*, vol. 49, pp. 1614–1626, Aug. 2001.
39. S. E. Nordholm, I. Claesson, and N. Grbic, "Performance limits in subband beamforming," *IEEE Trans. Speech Audio Processing*, vol. 11, pp. 193-203, May 2003.
40. F. Asano, S. Hayamizu, T. Yamada, and S. Nakamura, "Speech enhancement based on the subspace method," *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 497–507, Sept. 2000.
41. F. Jabloun and B. Champagne, "A multi-microphone signal subspace approach for speech enhancement," in *Proc. IEEE ICASSP*, pp. 205–208, 2001.
42. S. Doclo and M. Moonen, "GSVD-based optimal filtering for single and multimicrophone speech enhancement," *IEEE Trans. Signal Processing*, vol. 50, pp. 2230–2244, Sept. 2002.
43. S. Haykin, *Adaptive Filter Theory.* Fourth Edition, Upper Saddle River, NJ: Prentice Hall, 2002.
44. P. M. Clarkson, *Optimal and Adaptive Signal Processing.* Boca Raton, FL: CRC, 1993.
45. K. Fukunaga, *Introduction to Statistial Pattern Recognition.* San Diego, CA: Academic, 1990.
46. L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition.* Englewood Cliffs, NJ: Prentice Hall, 1993.

47. S. Quakenbush, T. Barnwell, and M. Clements, *Objective Measures of Speech Quality*. Englewook Cliffs, NJ: Prentice Hall, 1988.
48. G. Chen, S. N. Koh, and I. Y. Soon, “Enhanced Itakura measure incorporating masking properties of human auditory system,” *Signal Processing*, vol. 83, pp. 1445–1456, July 2003.

3 Statistical Methods for the Enhancement of Noisy Speech

Rainer Martin

Ruhr-Universität Bochum, Institute of Communication Acoustics
Bochum 44780, Germany
E-mail: rainer.martin@rub.de

Abstract. Speech signals are frequently disturbed by statistically independent additive noise signals. When the power fluctuation of the noise signal is significantly slower than that of the speech signal, a single-microphone approach may be successfully used to reduce the level of the disturbing noise. This chapter outlines algorithms for noise reduction which are based on short term spectral representations of speech and on optimal estimation techniques. We present some of the more prominent estimation methods for complex spectral coefficients, for the amplitude and phase of spectral coefficients, and for related parameters such as the *a priori* signal-to-noise ratio. We interpret these algorithms in terms of their input-output characteristics. Some recent developments such as the use of super-Gaussian speech models and the properties of the resulting estimators are highlighted. Furthermore, we discuss the estimation of the background noise power and the application of these techniques in conjunction with a low bit rate speech coder.

3.1 Introduction

Speech communication devices are often used in environments with high levels of ambient noise such as cars and public places. The noise picked up by the microphones of the device can significantly impair the quality of the transmitted speech signal – especially when the speech source is far from the microphones. When the intelligibility of the transmitted speech is also impaired, the device cannot be used in the desired way. It is therefore sensible to include a noise reduction processor in such devices.

Algorithms for noise reduction have been the subject of intensive research over the last two decades [1–7]. The wide-spread use of mobile communication devices and the introduction of digital hearing aids have contributed to the significant interest in this field. While early approaches focused only on speech quality, it is now generally acknowledged that the perceived quality of the residual noise is also of great importance, e.g., random narrowband fluctuations in the processed noise, also known as *musical tones*, are not accepted by the human listener.

Over the last two decades researchers have found ways to improve the performance of noise reduction algorithms such that musical tones can be avoided and the algorithms are more robust with respect to the great variability of environmental conditions. In this context, statistical models and methods play a prominent role [4,8].

In this chapter, we will outline some well known results as well as some of the recent developments for single-microphone noise reduction algorithms. We will focus on systems which use a short term spectral representation of the speech and noise signals. The noisy signal may be analyzed, for example, by means of a short time discrete Fourier transform (DFT). Most of the results, however, also apply to other non-parametric spectral analysis methods such as filterbanks, subspace algorithms, or wavelet transforms, see e.g., [9,10].

3.2 Spectral Analysis

The advantages of moving into the spectral domain are at least threefold. In the spectral domain we achieve:

- a good separation of speech and noise — especially for voiced speech; thus optimal and/or heuristic approaches can be easily implemented,
- a decorrelation of spectral components; thus frequency bins can be treated independently to some extent and statistical models are simplified, and
- a possibility of integration of psychoacoustic models [11,12].

In most of the relevant applications the noise signal is additive and statistically independent from the source signal. In particular, the noisy speech signal $y(k)$ is generally modeled as the sum of an undisturbed speech signal $s(k)$ and a noise signal $n(k)$. The task of noise reduction is then to recover $s(k)$ "in the best possible way" when only the noisy signal $y(k)$ is given. The estimate of the undisturbed speech signal is denoted by $\widehat{s}(k)$.

Figure 3.1 depicts a typical implementation of a single-channel noise reduction system where the noisy signal is processed in a succession of short signal segments and the spectral coefficients are computed by means of a DFT. The DFT of a segment of M samples of $y(\ell)$, $\ell = k - M + 1, \ldots, k$, is denoted by

$$\boldsymbol{Y}(k) = (Y_0(k), \ldots, Y_\mu(k), \ldots, Y_{M-1}(k))^T, \tag{3.1}$$

with

$$\begin{aligned} Y_\mu(k) &= R_\mu(k) \exp\left(j\theta_\mu(k)\right) \\ &= \sum_{\ell=0}^{M-1} w(\ell) y(k - M + 1 + \ell) \exp\left(-\frac{j2\pi\mu\ell}{M}\right), \end{aligned} \tag{3.2}$$

where a tapered analysis window $w(\ell)$ of length M is applied to the time domain segment before the DFT is computed. k denotes the time index at which the segment of M signal samples is extracted. $\mu = 0, \ldots, M-1$ is the index of the DFT bin which is related to the normalized center frequency Ω_μ

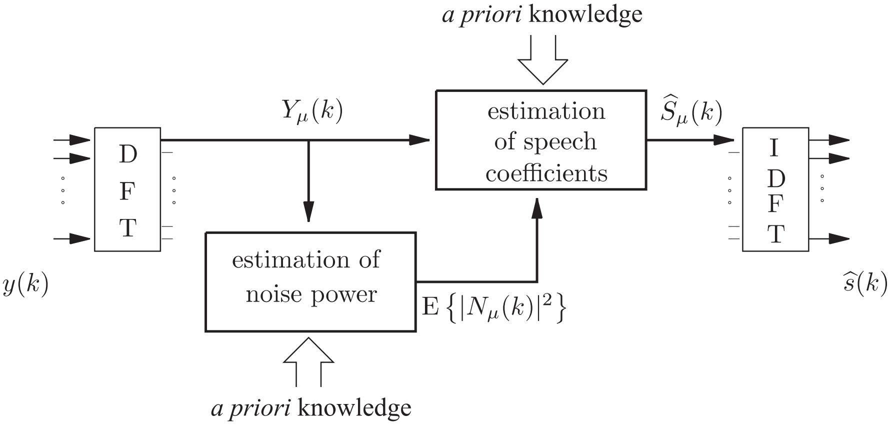


Fig. 3.1. DFT based speech enhancement. k and μ denote the time and the frequency bin index, respectively.

of that bin by $\Omega_\mu = 2\pi\mu/M = 2\pi f_\mu/f_S$ where f_μ and f_S denote the absolute center frequency and the sampling frequency, respectively. An enhanced DFT coefficient is denoted by $\widehat{S}_\mu(k)$. Vectors of the undisturbed speech signal and the enhanced speech signal are defined in the same way. The enhanced signal segments are computed by means of an inverse DFT and a continuous signal is produced by the overlap-add method. For the overlap-add operation the use of a tapered synthesis window is generally beneficial [13,14].

After the short time spectral components are computed by means of a DFT, there are two major tasks which must be addressed:

- estimation of the spectral components $S_\mu(k)$ of the undisturbed speech signal given the noisy spectral components $Y_\mu(k)$,
- estimation of the noise power $\sigma_n^2 = \mathrm{E}\left\{|N_\mu(k)|^2\right\}$ in each frequency bin μ.

Both of these tasks require the application of *a priori* knowledge and will be discussed below.

3.3 The Wiener Filter and its Implementation

Numerous approaches are available for the estimation of the complex coefficients $S_\mu(k) = A_\mu(k)\exp(\alpha_\mu(k))$ of the undisturbed speech signal or functions thereof. Among these are methods based on linear processing models and minimum mean square error (MMSE) estimation such as the Wiener filter. MMSE estimation is suitable for speech processing purposes as large estimation errors are given more weight in the optimization than small estimation errors. The latter might be masked in the human auditory system and might therefore be inaudible. Under the assumption that all signals are wide-sense stationary, the Wiener filter minimizes

$$\mathrm{E}\left\{(\widehat{s}(k)-s(k))^2\right\}, \tag{3.3}$$

where $\mathrm{E}\{\cdot\}$ denotes the statistical expectation operator and

$$\widehat{s}(k)=\sum_{\ell=-\infty}^{\infty} h(\ell)y(k-\ell) \tag{3.4}$$

is the convolution of an impulse response $h(\ell)$ with the noisy signal $y(k)$. For statistically independent and additive speech and noise signals, the frequency response of the Wiener filter is given by

$$G(\Omega)=\mathrm{DTFT}\{h(\ell)\}=\frac{P_{ss}(\Omega)}{P_{ss}(\Omega)+P_{nn}(\Omega)}, \tag{3.5}$$

where $P_{xx}(\Omega)$ denotes the power spectral density of the signal in the subscript and $\mathrm{DTFT}\{\cdot\}$ is the discrete time Fourier transform. Thus, in the case of stationary signals, the spectrum of the enhanced output signal is computed as

$$\widehat{S}(\Omega)=\frac{P_{ss}(\Omega)}{P_{ss}(\Omega)+P_{nn}(\Omega)}Y(\Omega)=\frac{P_{ss}(\Omega)}{P_{yy}(\Omega)}Y(\Omega)=G(\Omega)Y(\Omega). \tag{3.6}$$

In this context, $G(\Omega)$ is frequently called the *spectral gain* function. For the Wiener filter this function depends on the noisy input $y(k)$ or its Fourier transform $Y(\Omega)$ and on the undisturbed speech signal only via statistical expectations. However, an exact numerical implementation of the Wiener filter is not completely straightforward as this filter has an infinite impulse response and a continuous frequency response.

For a numerical implementation in conjunction with the above spectral analysis-synthesis system, the gain function is evaluated at the center frequencies of the spectral bins. Furthermore, as speech and noise signals are not stationary, short-term approximations to the power spectra must be used. However, for the segment-by-segment processing approach outlined above, we prefer an alternative derivation. In analogy to the Wiener filter in (3.6), the output of the filter for the signal segment at time k, $\widehat{\boldsymbol{S}}(k)=(\widehat{S}_0(k),\ldots,\widehat{S}_\mu(k),\ldots,\widehat{S}_{M-1}(k))^T$, is computed by an elementwise multiplication

$$\widehat{\boldsymbol{S}}(k)=\boldsymbol{G}(k)\otimes\boldsymbol{Y}(k) \tag{3.7}$$

of the DFT vector $\boldsymbol{Y}(k)$ and a gain vector

$$\boldsymbol{G}(k)=(G_0(k),G_1(k),\ldots,G_{M-1}(k))^T. \tag{3.8}$$

For independent additive speech and noise signals the minimization of $\mathrm{E}\left\{\left(\widehat{S}_\mu(k) - S_\mu(k)\right)^2\right\}$ with respect to $G_\mu(k)$ leads to

$$G_\mu(k) = \frac{\mathrm{E}\left\{|S_\mu(k)|^2\right\}}{\mathrm{E}\left\{|S_\mu(k)|^2\right\} + \mathrm{E}\left\{|N_\mu(k)|^2\right\}} = \frac{\eta_\mu(k)}{1+\eta_\mu(k)}, \tag{3.9}$$

where the right hand side of (3.9) makes use of the *a priori* SNR

$$\eta_\mu(k) = \frac{\mathrm{E}\left\{|S_\mu(k)|^2\right\}}{\mathrm{E}\left\{|N_\mu(k)|^2\right\}}. \tag{3.10}$$

$\mathrm{E}\left\{|S_\mu(k)|^2\right\} = \sigma^2_{s,\mu}(k)$ and $\mathrm{E}\left\{|N_\mu(k)|^2\right\} = \sigma^2_{n,\mu}(k)$ are the power of the undisturbed speech signal and the noise signal in frequency bin μ, respectively.

In a linear systems framework, the multiplication of the two DFT vectors and the subsequent inverse DFT of the result corresponds to a cyclic convolution in the time domain. Therefore, to implement this Wiener-like filter as a segmentwise linear system the signal and the gain vectors must be zero-padded to the appropriate length.

It is, however, instructive to consider the above estimation task in the framework of *non-linear* estimation, i.e., to derive the best estimator in the MMSE sense for the *short term* spectral coefficients of the undisturbed speech signal given the short term coefficients of the noisy signal. Contrary to the Wiener-like filter (3.9) which relies on second order statistics only, the non-linear solution generally requires knowledge of the probability density functions (pdf) of the speech and noise spectral coefficients. Under the assumption that all frequency bins are mutually independent, the MMSE solution can be stated as the conditional expectation

$$\begin{aligned}\widehat{S}_\mu(k) &= \mathrm{E}\left\{S_\mu(k) \mid Y_\mu(k)\right\} \\ &= \int\!\!\int S_\mu(k) p_{S|Y}(S_\mu(k) \mid Y_\mu(k)) dS_\mu(k) \\ &= \frac{1}{p(Y_\mu(k))} \int\!\!\int S_\mu(k) p_{Y|S}(Y_\mu(k) \mid S_\mu(k)) p(S_\mu(k)) dS_\mu(k),\end{aligned} \tag{3.11}$$

where $p_{S|Y}(S_\mu(k) \mid Y_\mu(k))$ is the pdf of an undisturbed speech coefficient given the coefficient of the noisy signal and $p(S_\mu(k))$ is the density of the undisturbed speech coefficients. Note that $S_\mu(k)$ is a complex quantity and therefore a double integration over the real and imaginary parts or over the magnitude and phase is required.

For additive noise which is statistically independent of the speech signal we have $p_{Y|S}(Y_\mu(k) \mid S_\mu(k)) = p_N(Y_\mu(k) - S_\mu(k))$. Therefore, the application of Bayes theorem in (3.11) leads to a nice decomposition of the density $p_{S|Y}(S_\mu(k) \mid Y_\mu(k))$ in terms of the probability density functions of the noise and the density of the undisturbed speech spectral coefficients. To model the probability density function of the real and the imaginary part of these coefficients, $S_\mu^{<R>}$ and $S_\mu^{<I>}$ respectively, the Gaussian density

$$\begin{aligned} p(S_\mu^{<R>}) &= \frac{1}{\sqrt{\pi}\sigma_s} \exp\left(-\frac{(S_\mu^{<R>})^2}{\sigma_s^2}\right), \\ p(S_\mu^{<I>}) &= \frac{1}{\sqrt{\pi}\sigma_s} \exp\left(-\frac{(S_\mu^{<I>})^2}{\sigma_s^2}\right), \end{aligned} \tag{3.12}$$

is frequently used. These probability densities depend on the speech power σ_s^2 which is, in general, time-variant. When the noise coefficients are also Gaussian distributed it is straightforward to show that for statistically independent and additive speech and noise coefficients, (3.7) with (3.9) is the solution to the estimation problem. For Gaussian signals the non-linear optimal estimator yields a linear function of the observations. However, this does not necessarily hold for practical implementations of these filters.

To illustrate the non-linearity of practical implementations we consider the estimation of the *a priori* SNR $\eta_\mu(k)$ which is required for the computation of the Wiener-like filter in (3.9). $\eta_\mu(k)$ is frequently estimated using the *decision-directed* approach [4]. This scheme assumes that an estimate $\widehat{A}_\mu(k-r)$ for the undisturbed speech amplitudes $A_\mu(k-r) = |S_\mu(k-r)|$ from a previous signal segment at time $k-r$ is available and sufficiently close to the undisturbed speech amplitudes of the current segment. The decision-directed approach then feeds back the estimate of the previous segment and combines it with an instantaneous estimate of the SNR,

$$\gamma_\mu(k) - 1 = \frac{|Y_\mu(k)|^2}{\mathrm{E}\left\{|N_\mu(k)|^2\right\}} - 1 = \frac{R_\mu^2(k)}{\sigma_{n,\mu}^2(k)} - 1, \tag{3.13}$$

such that the estimated SNR $\widehat{\eta}_\mu(k)$ is obtained as

$$\widehat{\eta}_\mu(k) = \alpha_\eta \frac{|S_\mu\widehat{(k-r)}|^2}{\mathrm{E}\left\{|N_\mu(k)|^2\right\}} + (1-\alpha_\eta)\max\left(0, \gamma_\mu(k) - 1\right), \tag{3.14}$$

where the latter contribution is forced to be non-negative and α_η is a smoothing parameter. The term

$$\gamma_\mu(k) = \frac{|Y_\mu(k)|^2}{\mathrm{E}\left\{|N_\mu(k)|^2\right\}} = \frac{R_\mu^2(k)}{\sigma_{n,\mu}^2(k)} \tag{3.15}$$

is the *a posteriori* SNR. For low SNR conditions, this estimator is clearly biased. The bias can be reduced if the maximum operation is applied to the sum of the two contributions:

$$\widehat{\eta}_\mu(k) = \max\left(0, \alpha_\eta \frac{|S_\mu\widehat{(k-r)}|^2}{\mathrm{E}\left\{|N_\mu(k)|^2\right\}} + (1-\alpha_\eta)(\gamma_\mu(k) - 1)\right). \tag{3.16}$$

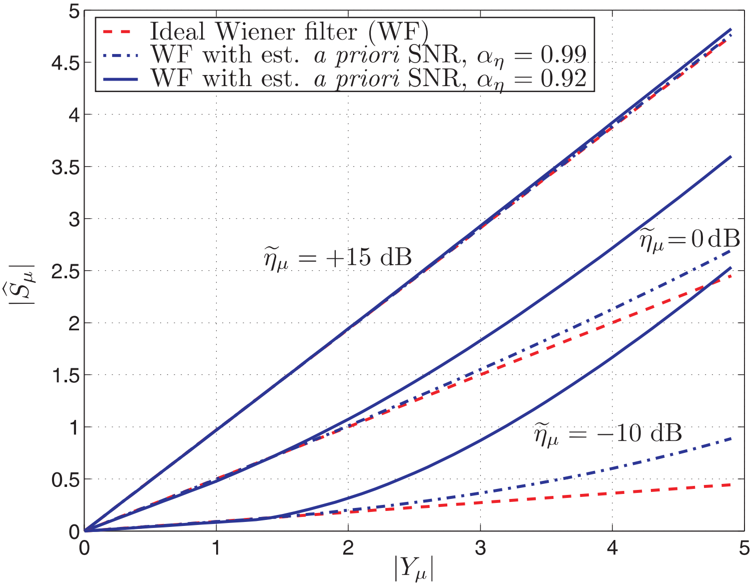


Fig. 3.2. Estimator characteristics for the ideal Wiener-like filter (dashed), the Wiener-like filter with $\alpha_\eta = 0.99$ (dash-dotted) and with $\alpha_\eta = 0.92$ (solid) for three different *a priori* SNR $\widetilde{\eta}_\mu(k-r)$. The decision-directed SNR estimator (3.14) was used and $\sigma_y^2 = \sigma_s^2 + \sigma_n^2 = 2$.

Using (3.14) in (3.9) we find that the spectral components $Y_\mu(k)$ of the current signal segment now have a direct influence on the gain function. Therefore, the combination of the Wiener-like filter and the decision-directed SNR estimation leads to a non-linear system. This non-linear dependency on the observation is clearly visible in Fig. 3.2 which plots the magnitude of the estimated spectral coefficient as a function of the magnitude of the noisy coefficient for $\sigma_y^2 = \sigma_s^2 + \sigma_n^2 = 2$. Three different values for the *a priori* SNR $\widetilde{\eta}_\mu(k-r) = \widehat{A}_\mu^2(k-r)/\mathrm{E}\left\{|N_\mu(k)|^2\right\}$ related to the previous frame are selected. Compared to the ideal Wiener-like filter which is also shown, the non-linear behaviour is visible, especially for low *a priori* SNR conditions. For comparison purposes, the same graphs are shown for the less biased *a priori* SNR estimation (3.16) in Fig. 3.3. For low *a priori* SNR values and small input coefficients, more attenuation is achieved than with the decision-directed approach in (3.14).

For the ideal Wiener-like filter the slope of the filter characteristic does not depend on the noisy input coefficient. On the other hand, the practical implementation using the decision-directed approach provides a larger gain than the Wiener filter when the observed coefficient is larger than its standard deviation. In this case, it is likely that speech is contained in the current segment of the input signal and thus speech distortions are reduced. When the

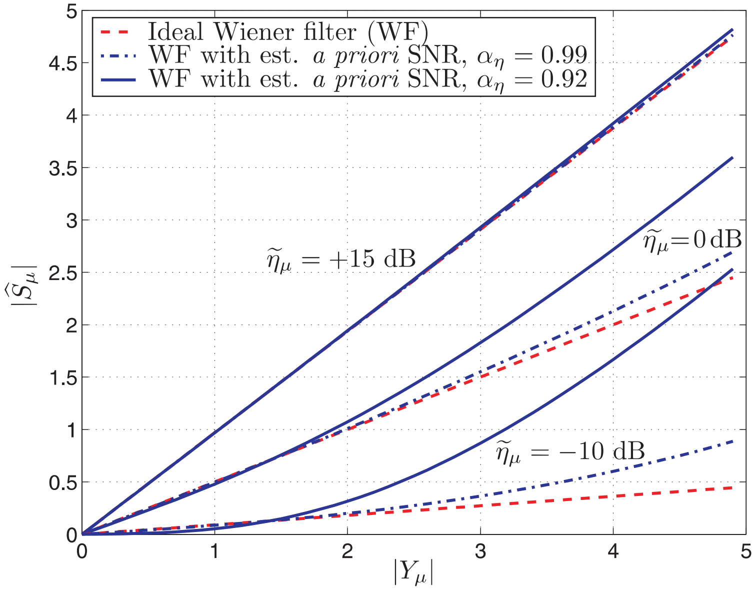


Fig. 3.3. Estimator characteristics for the ideal Wiener-like filter (dashed), the Wiener-like filter with $\alpha_\eta = 0.99$ (dash-dotted) and with $\alpha_\eta = 0.92$ (solid) for three different *a priori* SNR $\widetilde{\eta}_\mu(k-r)$. The decision-directed SNR estimator (3.16) was used and $\sigma_y^2 = \sigma_s^2 + \sigma_n^2 = 2$.

noisy coefficient is relatively small the input coefficient contains mostly noise. In this case it is important to avoid large fluctuations of the output coefficients as these translate into musical tones. With the noise reduction scheme discussed here this can be achieved by choosing the smoothing parameter α_η close to unity and thus smoothing the estimated *a priori* SNR. However, a large amount of smoothing will reduce the non-linearity of the estimation scheme for large amplitudes and thus lead to less transparent speech reproduction. The combination of the Wiener-like filter and the decision-directed estimator therefore requires a balance between these conflicting objectives [8,15]. Nevertheless, the decision-directed estimation procedure is advantageously combined with many noise reduction algorithms where the *a priori* SNR plays a role [15]. Furthermore, there are other ways to exploit the idea of recursive estimation, e.g., [16,17] which in general lead to less musical noise than the standard methods. As an accurate *a priori* SNR estimate is a key factor in the performance of these algorithms, improved *a priori* SNR estimators have also been developed [18,19].

To conclude this discussion we note that noise reduction schemes are frequently non-linear. In general, it is therefore not appropriate to cast spectral estimation procedures into the form of a multiplication of the noisy spectral coefficients with a spectral gain function as in (3.7). Moreover, there are immediate consequences for the synthesis of the enhanced signal. In the framework of non-linear estimation in the spectral domain we strive for the optimal estimate of the spectral coefficients of a short signal segment. The enhanced segments will then be synthesized using an inverse spectral transform and concatenated to produce a continuous signal. By virtue of this approach zero-padding is not necessarily required. For example, the (non-realizable) gain vector $G_\mu(k) = S_\mu(k)/Y_\mu(k)$ will result in a perfect reconstruction of the spectral coefficients and hence of the undisturbed speech signal without zero-padding and any cyclic effects. On the other hand, an MMSE-optimal estimate in the spectral domain does not deliver MMSE-optimal time domain segments. Also, simplifying assumptions such as the independence of adjacent frequency bins lead to estimation errors. Thus, there are no strict guidelines for the implementation of the spectral analysis-synthesis system. To suppress estimation errors in the synthesized signal it is, however, advisable to use a tapered analysis and a tapered synthesis window [14].

3.4 Estimation of Spectral Amplitudes

In the context of single-microphone speech enhancement, the short term spectral amplitudes are much more important than the short term spectral phases [20]. It is therefore sensible to estimate the spectral amplitudes $A_\mu(k)$ of the undisturbed speech signal jointly with the phase $\alpha_\mu(k)$ or directly by using the marginal distribution of the spectral amplitudes. We briefly present *minimum mean square error* (MMSE) and *maximum a posteriori* (MAP) solutions to this problem. These estimators require explicit knowledge of the probability density functions of the spectral coefficients of speech and noise.

3.4.1 MMSE Estimation

For Gaussian speech and noise coefficients the MMSE *short term spectral amplitude* estimator (MMSE-STSA) was derived by Ephraim and Malah [4],

$$\widehat{A}_{\mathrm{STSA},\mu} = \mathrm{E}\left\{A_\mu \mid Y_\mu\right\} = \sigma_n \sqrt{\frac{\eta_\mu}{1+\eta_\mu}}\, \mathit{\Gamma}(1.5)\;\; F_1(-0.5; 1, -v_\mu), \tag{3.17}$$

where we have now dropped the time index k for improved readability. $F_1(\cdot;\cdot,\cdot)$ is a confluent hypergeometric function [21] and v_μ is defined as

$$v_\mu = \frac{\eta_\mu}{1+\eta_\mu}\gamma_\mu. \tag{3.18}$$

The confluent hypergeometric function can be expanded in terms of Bessel functions and may be tabulated for efficient numerical implementations. Besides the MMSE-STSA estimator, the estimate of the logarithm of the spec-

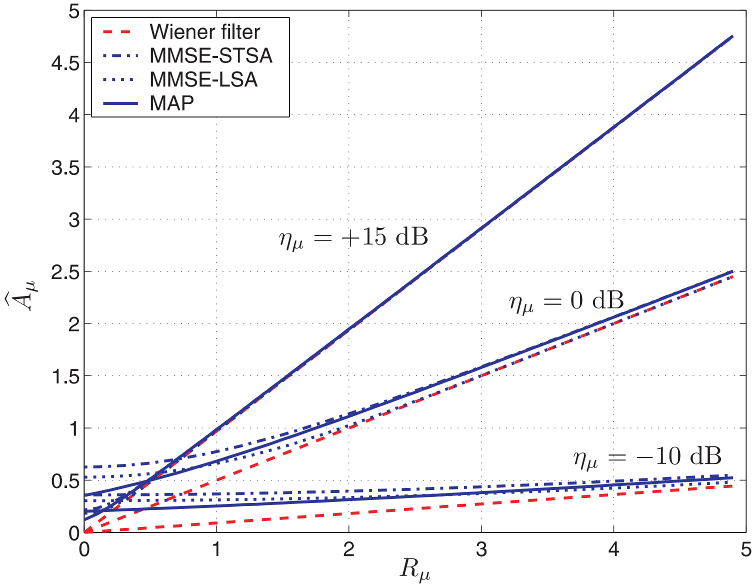


Fig. 3.4. Estimator characteristics for the Wiener filter (dashed), the MMSE-STSA [25] (dash-dotted), the MMSE-LSA [25] (dotted), and the MAP estimator [23] (solid) for three different *a priori* SNR values. $\sigma_y^2 = \sigma_s^2 + \sigma_n^2 = 2$.

tral amplitudes is also widely used. This MMSE *log spectral amplitude* estimator (MMSE-LSA) may be written as

$$\begin{aligned}\widehat{A}_{\mathrm{LSA},\mu} &= \exp\left(\mathrm{E}\left\{\log(A_\mu) \mid Y_\mu\right\}\right) \\ &= \frac{\eta_\mu}{1+\eta_\mu} \exp\left(\frac{1}{2}\int_{v_\mu}^{\infty} \frac{\exp\{-t\}}{t} dt\right) R_\mu,\end{aligned} \tag{3.19}$$

where R_μ denotes the amplitude of the noisy spectral coefficients. For large *a posteriori* SNR values both estimators approach the Wiener filter. For small, noisy amplitudes the estimators deliver an almost constant output value which depends to a greater extent on the *a priori* SNR than on the instantaneous input amplitude. This behaviour contributes significantly to the perceived quality of the residual noise since for small input values the fluctuations of the noisy amplitudes result in much smaller fluctuations in the enhanced output. For a single frequency bin and for $\sigma_s^2 + \sigma_n^2 = 2$ the resulting input-output characteristics are shown in Fig. 3.4 for the *a priori* SNR estimation (3.14) with $\alpha_\eta = 1$ and in Fig. 3.5 for $\alpha_\eta = 0.92$. To compute the enhanced complex spectral coefficient, the estimated spectral amplitude is combined with the short term phase of the noisy input. The observed phase represents the optimal phase estimate in the MMSE sense [4].

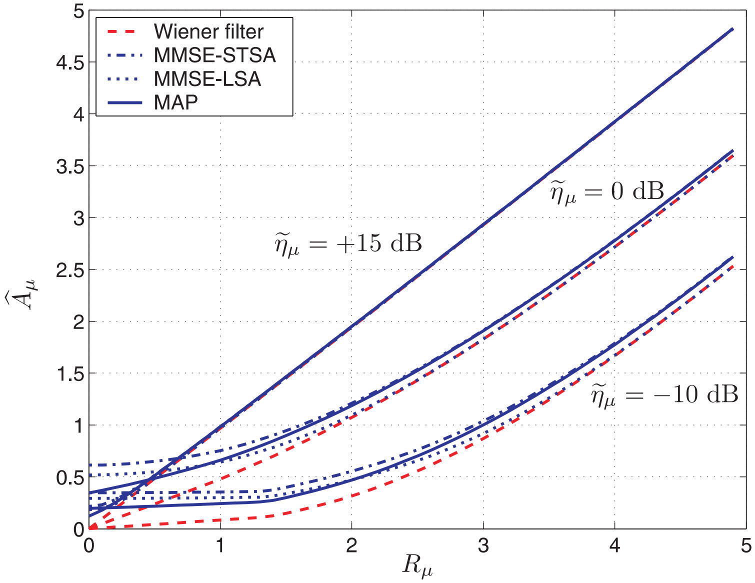


Fig. 3.5. Estimator characteristics for the Wiener filter (dashed), the MMSE-STSA [25] (dash-dotted), the MMSE-LSA [25] (dotted), and the MAP estimator [23] for three different *a priori* SNR and $\sigma_y^2 = \sigma_s^2 + \sigma_n^2 = 2$. The decision-directed SNR estimator (3.14) was used with $\alpha = 0.92$.

3.4.2 Maximum Likelihood and MAP Estimation

The maximum likelihood (ML) and the maximum *a posteriori* (MAP) estimation techniques avoid hard-to-compute integrals and lead to relatively simple solutions. In the case of complex Gaussian distributed spectral coefficients, the ML and the MAP estimators yield the well known Wiener-like solution. An ML estimate for deterministic spectral amplitudes in Gaussian noise was derived in [22],

$$\widehat{A}_{\mathrm{ML},\mu} = \left(0.5R_\mu + 0.5\sqrt{R_\mu^2 - \sigma_{n,\mu}^2}\right). \tag{3.20}$$

This estimator provides only a modest amount of noise reduction and is therefore not often used. Joint MAP estimation of the spectral amplitude and the spectral phase was proposed by Wolfe and Godsill [23]. Also in this case, the optimal estimate of the phase of the undisturbed spectral coefficients is the phase of the noisy input. The estimate of the amplitude is given by

$$\widehat{A}_{\mathrm{JMAP},\mu} = \frac{\eta_\mu + \sqrt{\eta_\mu^2 + 2\,(1+\eta_\mu)\,\frac{\eta_\mu}{\gamma_\mu}}}{2(1+\eta_\mu)} R_\mu. \tag{3.21}$$

Estimation of the spectral amplitude using the marginal density is also feasible but for closed form analytic solutions approximations to the Rician density are required. Using such approximations, the MAP estimation of the spectral amplitudes leads to a solution which, like (3.21), is close in performance to the MMSE methods [23],

$$\widehat{A}_{\mathrm{MAP},\mu} = \frac{\eta_\mu + \sqrt{\eta_\mu^2 + (1+\eta_\mu)\,\frac{\eta_\mu}{\gamma_\mu}}}{2(1+\eta_\mu)} R_\mu . \tag{3.22}$$

The attenuation characteristics of this latter estimator in conjunction with (3.14) is shown in Fig. 3.4 for $\alpha = 1$ and in Fig. 3.5 for $\alpha = 0.92$. A MAP amplitude estimator using super-Gaussian speech models is derived in [24] and discussed in Chapter 4.

To conclude this section, we firstly note that all of these estimators and the underlying statistical models, e.g., (3.12), are conditioned on the signal power. The power of the undisturbed speech signal as well as of the noise signal are random processes by themselves and must be estimated, e.g., using the decision-directed approach. Secondly, all of the above approaches assume that speech is actually present in the frequency bin under consideration. This is, of course, not always the case as there are speech pauses and possibly also a concentration of speech power onto a few dozen harmonics during voiced speech. Frequently, these estimators are used in conjunction with a statistical two-state speech presence/absence model which leads to a soft-decision gain modification procedure. The resulting soft-decision gain functions are dependent on the signal model and are discussed in detail in [22,4,18,30].

3.5 MMSE Estimation Using Super-Gaussian Speech Models

In the time domain, the probability density function of speech samples may be modelled by Laplacian (bilateral exponential) or Gamma, i.e. *super-Gaussian*, densities rather than Gaussian densities [26, page 235]. It has been suggested [27,28] that also in the short time Discrete Fourier domain (frame size $<$ 100 ms), the Laplace and Gamma densities are much better models for the probability density function of the real and imaginary parts of the speech coefficients than the commonly used Gaussian density. In fact, the Gaussian assumption is based on the central limit theorem [29]. However, when the DFT length is shorter than the span of correlation of the signal, the asymptotic arguments do not hold. While for many applications the spectral coefficients of the noise can be modeled by a complex Gaussian random variable, the span of correlation of voiced speech is certainly larger than the typical segment size used in voice communications. Note again that all of these probability functions are conditioned on the signal power which is, in general, time-variant. Therefore, in an experimental verification of the density model great care must be exercised to generate quasi-stationary conditions [30].

Only recently, analytic solutions to the estimation problem under super-Gaussian model assumptions have been found [28,31,32,24]. In this section, we will present an example based on a Laplacian speech pdf and a Gaussian noise model [32]. Estimators for complex spectral coefficients based on Gamma densities as well as soft-decision gain functions for various combinations of speech and noise densities are discussed, e.g., in [30].

When the spectral coefficients of the speech and noise signals are mutually independent with respect to frequency bins and time segments, the optimal instantaneous estimate can be written as a conditional expectation

$$\widehat{S}_\mu(k) = \mathrm{E}\left\{S_\mu(k) \mid Y_\mu(k)\right\} = \mathrm{E}\left\{S \mid Y\right\}. \tag{3.23}$$

On the right hand side we now drop time and frequency bin indices to simplify our notation. For statistically independent real and imaginary parts, we may decompose the optimal estimate into an estimate of its real and its imaginary part

$$\mathrm{E}\left\{S \mid Y\right\} = \mathrm{E}\left\{S^{<R>} \mid Y^{<R>}\right\} + j\mathrm{E}\left\{S^{<I>} \mid Y^{<I>}\right\}, \tag{3.24}$$

where $<R>$ and $<I>$ in the superscript indicate the real and the imaginary parts, respectively. When $\diamondsuit$ denotes either the real or the imaginary part, the MMSE estimate of one of these is given by

$$\mathrm{E}\left\{S^\diamondsuit \mid Y^\diamondsuit\right\} = \int_{-\infty}^{\infty} S^\diamondsuit p(S^\diamondsuit \mid Y^\diamondsuit) dS^\diamondsuit. \tag{3.25}$$

With Bayes theorem we obtain

$$\mathrm{E}\left\{S^\diamondsuit \mid Y^\diamondsuit\right\} = \frac{1}{p(Y^\diamondsuit)} \int_{-\infty}^{\infty} S^\diamondsuit p(Y^\diamondsuit \mid S^\diamondsuit) p(S^\diamondsuit) dS^\diamondsuit. \tag{3.26}$$

Good candidates for the pdf of the real and the imaginary parts of DFT coefficients of speech signals are the Laplacian pdf,

$$p(S^\diamondsuit) = \frac{1}{\sigma_s} \exp\left(-\frac{2|S^\diamondsuit|}{\sigma_s}\right), \tag{3.27}$$

and the Gamma pdf,

$$p(S^\diamondsuit) = \frac{\sqrt[4]{3}}{2\sqrt{\pi\sigma_s}\sqrt[4]{2}} |S^\diamondsuit|^{-\frac{1}{2}} \exp\left(-\frac{\sqrt{3}|S^\diamondsuit|}{\sqrt{2}\sigma_s}\right). \tag{3.28}$$

These two densities are better models than the Gaussian pdf, not only for small amplitudes, but also for large amplitudes where a *heavy-tailed* density leads to a better fit for the observed data [30]. The complexity of the analytic solutions depends upon the density models and the optimization criterion. A relatively simple analytical MMSE solution is based on the Gaussian noise and the Laplacian speech models.

To facilitate the development we introduce the shorthand notations

$$L^{\diamond+} = \frac{\sigma_n}{\sigma_s} + \frac{Y^\diamond}{\sigma_n} = \frac{1}{\sqrt{\eta}} + \frac{Y^\diamond}{\sigma_n},$$
$$L^{\diamond-} = \frac{\sigma_n}{\sigma_s} - \frac{Y^\diamond}{\sigma_n} = \frac{1}{\sqrt{\eta}} - \frac{Y^\diamond}{\sigma_n}, \tag{3.29}$$

where $\eta = \sigma_s^2/\sigma_n^2$ denotes the *a priori* SNR as before.

For the Laplacian speech pdf we obtain the optimal MMSE estimator of either the real part or the imaginary part [21, Theorem 3.462,1] as:

$$\begin{aligned}
&\mathrm{E}\{S^\diamond \mid Y^\diamond\} = \\
&\frac{1}{\sqrt{\pi}\sigma_n\sigma_s p(Y^\diamond)} \int_{-\infty}^{\infty} S^\diamond \exp\left(-\frac{(Y^\diamond - S^\diamond)^2}{\sigma_n^2}\right) \exp\left(-\frac{2|S^\diamond|}{\sigma_s}\right) dS^\diamond \\
&= \frac{\sigma_n \exp(\sigma_n^2/\sigma_s^2)}{2\sigma_s p(Y^\diamond)} \left\{ L^{\diamond+} \exp\left(2\frac{Y^\diamond}{\sigma_s}\right) \mathrm{erfc}(L^{\diamond+}) \right. \\
&\qquad \left. - L^{\diamond-} \exp\left(-2\frac{Y^\diamond}{\sigma_s}\right) \mathrm{erfc}(L^{\diamond-}) \right\},
\end{aligned} \tag{3.30}$$

with [21, Theorem 3.322,2]

$$\begin{aligned}
&p\ (Y^\diamond) = \\
&\frac{1}{\sqrt{\pi}\sigma_n\sigma_s} \int_{-\infty}^{\infty} \exp\left(-\frac{(Y^\diamond - S^\diamond)^2}{\sigma_n^2}\right) \exp\left(-\frac{2|S^\diamond|}{\sigma_s}\right) dS^\diamond \\
&= \frac{\exp\left(\sigma_n^2/\sigma_s^2\right)}{2\sigma_s} \left\{ \exp\left(2\frac{Y^\diamond}{\sigma_s}\right) \mathrm{erfc}(L^{\diamond+}) + \exp\left(-2\frac{Y^\diamond}{\sigma_s}\right) \mathrm{erfc}(L^{\diamond-}) \right\},
\end{aligned} \tag{3.31}$$

where erfc(z) denotes the complementary error function [21, Theorem 8.250]. The optimal estimator for the undisturbed complex speech coefficient is therefore given by $E\{S \mid Y\} = E\{S^{<R>} \mid Y^{<R>}\} + jE\{S^{<I>} \mid Y^{<I>}\}$ with

$$\begin{aligned}
&\mathrm{E}\{S^\diamond \mid Y^\diamond\} \\
&= \frac{\sigma_n \left[L^{\diamond+} \exp(2Y^\diamond/\sigma_s)\mathrm{erfc}(L^{\diamond+}) - L^{\diamond-} \exp(-2Y^\diamond/\sigma_s)\mathrm{erfc}(L^{\diamond-})\right]}{\exp(2Y^\diamond/\sigma_s)\mathrm{erfc}(L^{\diamond+}) + \exp(-2Y^\diamond/\sigma_s)\mathrm{erfc}(L^{\diamond-})}.
\end{aligned} \tag{3.32}$$

We note that both $\mathrm{E}\left\{S^{<R>} \mid Y^{<R>}\right\}$ and $\mathrm{E}\left\{S^{<I>} \mid Y^{<I>}\right\}$ are odd symmetric functions of $Y^{<R>}$ and $Y^{<I>}$, respectively. Figure 3.6 plots the input-output characteristics of this estimator and the Wiener-like filter for $0 \le Y^\diamond \le 5$, $\sigma_y^2 = \sigma_s^2 + \sigma_n^2 = 2$, and three different *a priori* SNR values. Again, the decision-directed SNR estimator is used with two different values of α_η. For high *a priori* SNR values the estimate is almost identical to the estimate delivered by the Wiener filter. Clearly, for a fixed *a priori* SNR, the Wiener filter is a linear estimator, characterized by its constant slope. The estimator based on super-Gaussian densities leads to an increased attenuation of the input when the instantaneous input value is smaller than its standard deviation and a significantly larger output value when the input is larger than

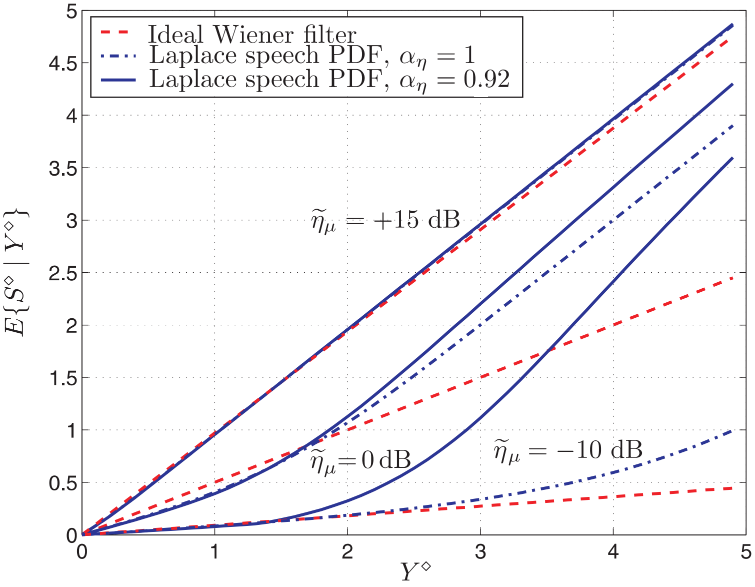


Fig. 3.6. Estimator characteristics $E\{S^\diamond \mid Y^\diamond\}$ for the ideal Wiener filter (dashed) and for the Laplacian speech pdf and the Gaussian noise pdf for $\alpha_\eta = 1$ (dash-dotted) and for $\alpha_\eta = 0.92$ (solid) and for three *a priori* SNR values $\widetilde{\eta}_\mu = 15, 0, -10$ dB. The decision-directed SNR estimator (3.14) was used and $\sigma_y^2 = \sigma_s^2 + \sigma_n^2 = 2$.

the standard deviation. Due to the heavy-tailed speech density, it is highly likely that speech is present in this latter case. Both of these characteristics contribute to the improved SNR of the output coefficients with respect to the linear estimator.

Figure 3.6 also plots the characteristics using the decision-directed SNR estimation technique (3.14) with $\alpha_\eta = 0.92$. The *a priori* SNR $\widetilde{\eta}_\mu(k-r)$ of the preceding signal segment is fixed. The SNR estimate of the present segment is then a function of the instantaneous, magnitude-squared input value which leads to an additional non-linear effect. Compared to the estimators based on Gaussian densities we find that in conjunction with the decision-directed estimator more smoothing can be applied to the SNR estimate without sacrificing the transparency of the enhanced speech components. Furthermore, we note that the proposed estimators may be applied to the magnitude of the spectral coefficients as well if we assume a fixed (hypothetical) phase angle. These procedures are outlined in [30].

3.6 Background Noise Power Estimation

The second estimation task which arises in the processing model of Fig. 3.1 is the estimation of the background noise power in the spectral bins. Most of the proposals in the literature are based on either

- voice activity detection and recursive averaging [22,33],
- soft-decision methods [34,35],
- bias compensated tracking of spectral minima ("minimum statistics") [36,37],

or a combination of these, as, e.g., developed by Cohen [38]. In general, these methods rely on the assumptions that

- speech and noise are statistically independent,
- speech is not always present, and
- noise is more stationary than speech.

For single-microphone systems it is in general difficult to track non-stationary noise mostly because a sudden increase in noise power in one or several frequency bins cannot easily distinguished from a speech onset. Only after a few hundred milliseconds can speech and noise components be reliably discriminated. Therefore, it is difficult to identify and to suppress short noise bursts or competing speakers. Current developments strive to improve the performance of noise estimation under non-stationary conditions [37,38].

In what follows, we briefly outline the minimum statistics approach. The power of this approach relies on the intrinsically non-linear minimum extraction and the subsequent bias compensation. It has been shown that this method can contribute significantly to the intelligibility and the listening ease of the enhanced signal especially in conjunction with a low bit rate speech coder.

3.6.1 Minimum Statistics Noise Power Estimation

Since speech and noise are additive and statistically independent we have

$$\mathrm{E}\left\{|Y_\mu(k)|^2\right\} = \mathrm{E}\left\{|S_\mu(k)|^2\right\} + \mathrm{E}\left\{|N_\mu(k)|^2\right\}. \tag{3.33}$$

Recursive smoothing of the magnitude-squared spectral coefficients leads to

$$P_\mu(k) = \beta_\mu(k)\, P_\mu(k-r) + (1-\beta_\mu(k))\, |Y_\mu(k)|^2, \tag{3.34}$$

where $\beta_\mu(k)$ is a time and frequency dependent smoothing parameter. We now search for the minimum from D samples of the smoothed power $P_\mu(k-\lambda r)$, $\lambda = 0, 1, \ldots, D-1$. Then, we might use this minimum as a first coarse estimate of the noise floor since

$$\begin{aligned} &\min\left(P_\mu(k), \ldots, P_\mu(k-(D-1)r)\right) \\ &\approx \min\left(P_{N,\mu}(k), \ldots, P_{N,\mu}(k-(D-1)r)\right). \end{aligned} \tag{3.35}$$

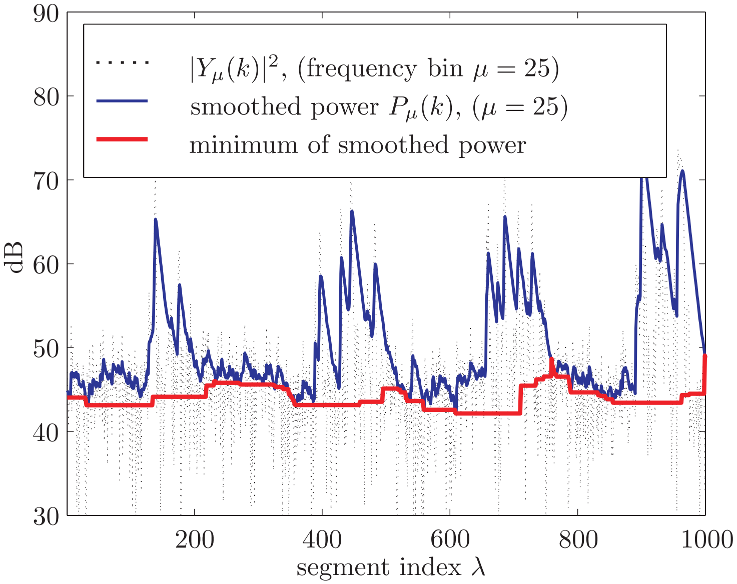


Fig. 3.7. Magnitude-squared DFT coefficient (dotted), smoothed power, and noise floor for a noisy speech signal (6 dB SNR).

$P_{N,\mu}(k)$ denotes a noise power estimate which is smoothed just like $P_\mu(k)$ in (3.34).

An example is shown in Figure 3.7 for a single frequency bin. Obviously, this estimate is biased towards lower values. However, the bias can be computed and compensated. It turns out that the bias depends on the variance of the smoothed power $P_\mu(k)$ which, in turn, is a function of the smoothing parameter $\beta_\mu(k)$ and of the variance of the signal under consideration. For recursively smoothed power estimates and a unity noise power, Fig. 3.8 plots the factor by which the minimum is smaller than the mean as a function of D and $Q_{eq} = 2\mathrm{E}\left\{|N_\mu(k)|^2\right\}^2 / \mathrm{var}\{P_\mu(k)\}$. Q_{eq} is the inverse normalized variance of the smoothed power. When much smoothing is applied $\mathrm{var}\{P_\mu(k)\}$ is relatively small and therefore Q_{eq} is large. Then, the minimum of subsequent values of $P_\mu(k)$ is close to the mean of these values. On the other hand, no smoothing ($Q_{eq} = 2$) requires a large bias compensation.

While earlier versions of the Minimum Statistics algorithm used a fixed smoothing parameter β and hence a fixed bias compensation we note that the full potential is only developed when a time and frequency dependent smoothing method is used. This in turn requires a time and frequency dependent bias compensation [37]. The result when using the adaptive smoothing and bias compensation is shown in Figure 3.9 for the same signal as in Figure 3.7.

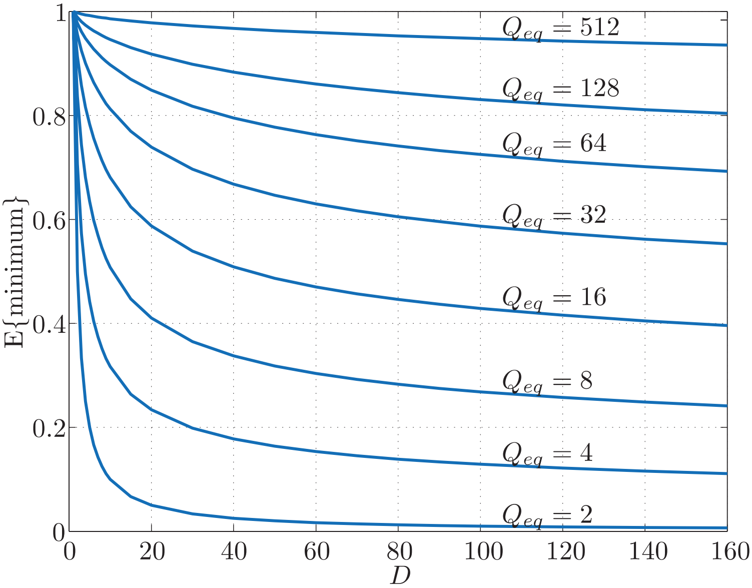


Fig. 3.8. Mean of the minimum of D correlated short term noise power estimates for $\sigma_n^2 = 1$.

3.7 The MELPe Speech Coder

As an application of the above techniques, we consider a speech enhancement algorithm which was developed for a low bit rate speech coder. Low bit rate speech coders are especially susceptible to environmental noise as they use a parametric model to code the input signal. One such example is the *mixed excitation linear prediction* (MELP) coder which operates at bit rates of 1.2 and 2.4 kbps [39]. It is used for secure governmental communications and is expected to succeed the well-known FS 1015 (LPC-10e) and FS 1016 (CELP) speech coding standards. This coder also includes an optional noise reduction preprocessor. The combined system of the preprocessor and the MELP coder is termed *MELPe* [39].

The noise reduction preprocessor [40] of the MELPe coder is based on

- the MMSE log spectral amplitude estimator [25];
- multiplicative soft-decision gain modification [35];
- adaptive gain limiting [14];
- estimation of the *a priori* SNR [35];
- *minimum statistics* noise power estimation [37].

This noise reduction preprocessor turns out to be very robust in a variety of noise environments and SNR conditions. Table 3.1 summarizes the results of a *diagnostic acceptability measure* (DAM) test for undisturbed and noisy

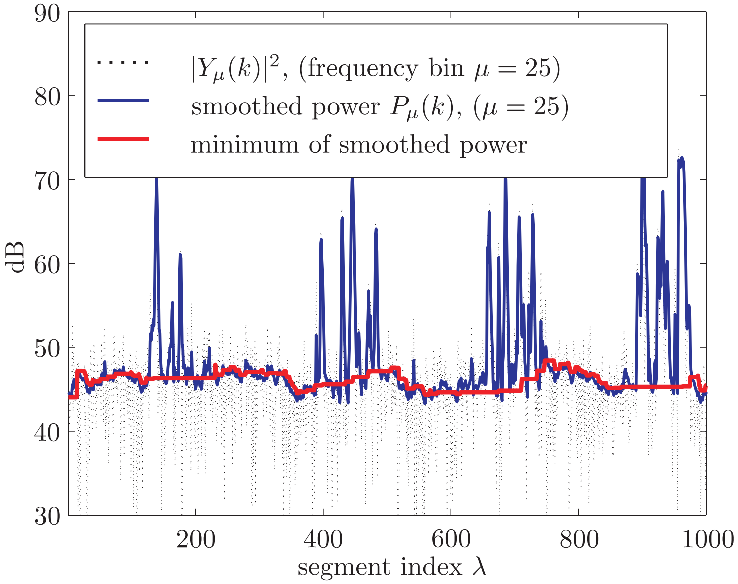


Fig. 3.9. Magnitude-squared DFT coefficient (dotted), smoothed power, and bias corrected noise floor for the same noisy speech signal as in Figure 3.7.

Table 3.1 DAM scores and standard error without noise and with vehicular noise (average SNR ≈ 6 dB).

condition	coder	DAM	standard error
no noise	MELPe	68.6	0.90
noisy	unprocessed	45.0	1.2
noisy	MELP	38.9	1.1
noisy	MELPe	50.3	0.80

conditions. As stated before, the MELP coder is highly sensitive to environmental noise. The noise reduction preprocessor helps to reduce these effects. Table 3.2 shows results of a *diagnostic rhyme test* (DRT) intelligibility evaluation for the same conditions as in the DAM test. We note, that the noisy but unprocessed signal has the highest intelligibility of the noisy conditions in Table 3.2. In conjunction with the MELP coder, the enhancement preprocessor leads to a significant improvement in terms of intelligibility. Thus, for a low bit rate speech coder, single-channel noise reduction systems can improve the quality as well as the intelligibility of the coded speech.

Table 3.2 DRT scores and standard error without noise and with vehicular noise (average SNR $\approx$ 6 dB).

condition	coder	DRT	standard error
no noise	MELPe	93.9	0.53
noisy	unprocessed	91.1	0.37
noisy	MELP	67.3	0.8
noisy	MELPe	72.5	0.58

3.8 Conclusions

Noise reduction technology is still an area of active research. While in the past decade most of these activities were triggered by new developments in mobile communications we now find increasing interest in automatic speech recognition and digital hearing aids applications.

Much of the research in this field is directed towards a better understanding and a better exploitation of the statistical properties of speech signals. As a result, several papers have been published which improve the estimation of critical (yet unknown) quantities such as the *a priori* SNR or the background noise power. Other approaches use optimal time domain estimators like Kalman filters which provide for an easy integration of autoregressive models. The question, however, of how the parameters of such models can be estimated in a robust fashion will require further research.

Further improvements are possible if we can employ more than one microphone and thus sample the sound field at more than one spatial location. There are a number of different ways to exploit multiple microphone signals. The most common are

- to use the spatial directivity of the microphone array [41,42],
- to adapt a single-channel *post-filter* based on the statistics of the microphone signals [43–46],

and combinations thereof. Some of these approaches are discussed, e.g., in [42]. Also, MAP and MMSE estimation of spectral amplitudes has been developed for the multi-microphone case, e.g., [47,48].

Despite these developments and many more which are not discussed here, there are still open questions which need to be addressed in the future:

- What are meaningful optimization criteria for speech enhancement and how can they be mathematically formulated?
- Which method of signal analysis is the most suitable?
- How can we improve the perceived quality of the enhanced signal without compromising intelligibility and vice versa?
- How can we combine signal theoretic and perceptual approaches?

- What kind of processing approach will be optimal for signals perceived by normal or hearing impaired persons, or, for signals processed by speech coders or speech recognition systems, and how are these approaches interrelated?
- What processing takes place in the higher stages of the auditory system and how can we model it?

Given all these questions it is clear that there will not be a single answer. We must, however, pay more attention to how humans process auditory information.

Acknowledgment

The author would like to thank Nilesh Madhu and Dirk Mauler for proofreading this manuscript and for valuable comments.

References

1. S. Boll, “Suppression of acoustic noise in speech using spectral subtraction,” *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 27, pp. 113–120, 1979.
2. M. Berouti, R. Schwartz, and J. Makhoul, “Enhancement of speech corrupted by acoustic noise,” in *Proc. IEEE ICASSP*, 1979, pp. 208–211.
3. J. Lim, ed., *Speech Enhancement.* Prentice-Hall, 1983.
4. Y. Ephraim and D. Malah, “Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator,” *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 32, pp. 1109–1121, Dec. 1984.
5. D. Van Compernolle, “DSP techniques for speech enhancement,” in *Proc. Speech Processing in Adverse Conditions*, 1992, pp. 21–30.
6. R. Martin, “Statistical methods for the enhancement of noisy speech,” in *Proc. IWAENC*, 2003, pp. 1–6.
7. Y. Ephraim and I. Cohen, “Recent advancements in speech enhancement,” book chapter, CRC Press, 2004.
8. O. Cappé, “Elimination of the musical noise phenomenon with the Ephraim and Malah noise suppressor,” *IEEE Trans. Speech and Audio Processing*, vol. 2, pp. 345–349, Apr. 1994.
9. Y. Ephraim and H. Van Trees, “A signal subspace approach for speech enhancement,” *IEEE Trans. Speech and Audio Processing*, vol. 3, no. 4, pp. 251–266, 1995.
10. T. Gülzow and A. Engelsberg, “Comparison of a discrete wavelet transformation and a nonuniform polyphase filterbank applied to spectral subtraction speech enhancement,” *Signal Processing, Elsevier*, vol. 64, no. 1, pp. 5–19, 1998.
11. S. Gustafsson, P. Jax, and P. Vary, “A novel psychoacoustically motivated audio enhancement algorithm preserving background noise characteristics,” in *Proc. IEEE ICASSP*, 1998, pp. 397–400.

12. S. Gustafsson, R. Martin, P. Jax, and P. Vary, “A psychoacoustic approach to combined acoustic echo cancellation and noise reduction,” *IEEE Trans. Speech and Audio Processing*, vol. 10, no. 5, pp. 245–256, 2002.
13. D. Griffin and J. Lim, “Signal estimation from modified short-time Fourier transform,” *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 32, pp. 236–243, Apr. 1984.
14. R. Martin and R. Cox, “New speech enhancement techniques for Low bit rate speech coding,” in *Proc. IEEE Workshop on Speech Coding*, 1999, pp. 165–167.
15. P. Scalart and J. Vieira Filho, “Speech enhancement based on a priori signal to noise estimation,” in *Proc. IEEE ICASSP*, 1996, pp. 629–632.
16. K. Linhard and T. Haulick, “Noise subtraction with parametric recursive gain curves,” in *Proc. EUROSPEECH*, vol. 6, 1999, pp. 2611–2614.
17. C. Beaugeant and P. Scalart, “Speech enhancement using a minimum least-squares amplitude estimator,” in *Proc. IWAENC*, 2001, pp. 191–194.
18. I. Cohen and B. Berdugo, “Speech enhancement for non-stationary noise environments,” *Signal Processing, Elsevier*, vol. 81, pp. 2403–2418, 2001.
19. I. Cohen, “Speech enhancement using a noncausal *a priori* SNR estimator,” *IEEE Signal Processing Letters*, vol. 11, pp. 725–728, 2004.
20. D. Wang and J. Lim, “The unimportance of phase in speech enhancement,” *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 30, no. 4, pp. 679–681, 1982.
21. I. Gradshteyn and I. Ryzhik, *Table of Integrals, Series, and Products.* Academic Press, 5th ed., 1994.
22. R. McAulay and M. Malpass, “Speech enhancement using a soft-decision noise suppression filter,” *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 28, pp. 137–145, Dec. 1980.
23. P. Wolfe and S. Godsill, “Simple alternatives to the Ephraim and Malah suppression rule for speech enhancement,” in *IEEE Workshop on Statistical Signal Processing*, 2001, pp. 496–499.
24. T. Lotter and P. Vary, “Noise reduction by maximum a posteriori spectral amplitude estimation with supergaussian speech modeling,” in *Proc. IWAENC*, 2003, pp. 83–86.
25. Y. Ephraim and D. Malah, “Speech enhancement using a minimum mean-square error log-spectral amplitude estimator,” *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 33, pp. 443–445, Apr. 1985.
26. D. O’Shaughnessy, *Speech Communications.* IEEE Press, 2 ed., 2000.
27. J. Porter and S. Boll, “Optimal estimators for spectral restoration of noisy speech,” in *Proc. IEEE ICASSP*, 1984, pp. 18A.2.1–18A.2.4.
28. R. Martin, “Speech enhancement using MMSE short time spectral estimation with Gamma distributed speech priors,” in *Proc. IEEE ICASSP*, vol. I, 2002, pp. 253–256.
29. D. Brillinger, *Time Series: Data Analysis and Theory.* Holden-Day, 1981.
30. R. Martin, “Speech enhancement based on minimum mean square error estimation and supergaussian priors,” *IEEE Trans. Speech and Audio Processing*,to appear, 2005.
31. C. Breithaupt and R. Martin, “MMSE estimation of magnitude-squared DFT coefficients with supergaussian priors,” in *Proc. IEEE ICASSP*, vol. I, 2003, pp. 848–851.
32. R. Martin and C. Breithaupt, “Speech enhancement in the DFT domain using Laplacian speech priors,” in *Proc. IWAENC*, 2003, pp. 87–90.

33. D. Van Compernolle, “Noise adaptation in a hidden Markov model speech recognition system,” *Computer Speech and Language*, vol. 3, pp. 151–167, 1989.
34. J. Sohn and W. Sung, “A Voice activity detector employing soft decision based noise spectrum adaptation,” in *Proc. IEEE ICASSP*, vol. 1, 1998, pp. 365–368.
35. D. Malah, R. Cox, and A. Accardi, “Tracking speech-presence uncertainty to improve speech enhancement in non-stationary noise environments,” in *Proc. IEEE ICASSP*, 1999, pp. 789–792.
36. R. Martin, “Spectral subtraction based on minimum statistics,” in *Proc. EUSIPCO*, 1994, pp. 1182–1185.
37. R. Martin, “Noise power spectral density estimation based on optimal smoothing and minimum statistics,” *IEEE Trans. Speech and Audio Processing*, vol. 9, pp. 504–512, July 2001.
38. I. Cohen, “Noise estimation in adverse environments: improved minima controlled recursive averaging,” *IEEE Trans. Speech and Audio Processing*, vol. 11, pp. 466–475, Sept. 2003.
39. T. Wang, K. Koishida, V. Cuperman, A. Gersho, and J. Collura, “A 1200/2400 BPS coding suite based on MELP,” in *IEEE Workshop on Speech Coding*, 2002, pp. 90–92.
40. R. Martin, D. Malah, R. Cox, and A. Accardi, “A noise reduction preprocessor for Mobile voice communication,” *EURASIP Journal on Applied Signal Processing*, vol. 2004, pp. 1046–1058, Aug. 2004.
41. G. Elko, “Microphone array systems for hands-free telecommunication,” in *Proc. IWAENC*, 1995, pp. 31–38.
42. M. Brandstein and D. B. Ward, eds., *Microphone Arrays.* Springer-Verlag, Berlin, 2001.
43. R. Zelinski, “A Microphone array with adaptive post-filtering for noise reduction in reverberant rooms,” in *Proc. IEEE ICASSP*, 1988, pp. 2578–2581.
44. C. Marro, Y. Mahieux, and K. Simmer, “Analysis of noise reduction and dereverberation techniques based on microphone arrays with postfiltering,” *IEEE Trans. Speech and Audio Processing*, vol. 6, no. 3, pp. 240–259, 1998.
45. J. Bitzer, K. Simmer, and K.-D. Kammeyer, “Multi-microphone noise reduction by post-filter and superdirective beamformer,” in *Proc. IWAENC*, 1999, pp. 100–103.
46. R. Martin, “Small microphone arrays with postfilters for noise and acoustic echo reduction,” in *Microphone Arrays* (M. Brandstein and D. B. Ward, eds.), Springer-Verlag, Berlin, 2001.
47. R. Balan and J. Rosca, “Microphone array speech enhancement by Bayesian estimation of spectral amplitude and phase,” in *Proc. IEEE Sensor Array and Multichannel Signal Processing Workshop*, 2002.
48. T. Lotter, C. Benien, and P. Vary, “Multichannel speech enhancement using Bayesian spectral amplitude estimation,” in *Proc. IEEE ICASSP*, 2003.

4 Single- and Multi-Microphone Spectral Amplitude Estimation Using a Super-Gaussian Speech Model

Thomas Lotter

Siemens Audiological Engineering Group
Erlangen, Germany
E-mail: thomas.tl.lotter@siemens.com

This work was carried out while being with the Institute of Communication Systems and Data Processing (IND) at the RWTH Aachen University, Germany

Abstract. In this contribution, MAP spectral amplitude estimators for speech enhancement are presented. For single-microphone applications, efficient MAP estimators with a super-Gaussian speech model, that can be adapted with high accuracy towards the real distribution in a given system, are introduced. For multi-microphone applications, joint MAP estimators that also exploit spatial properties of speech and noise are derived. Both the integration of the more accurate speech model as well as the multi-microphone joint spectral amplitude estimation improve the performance of a common DFT domain speech enhancement system.

4.1 Introduction

Many single-channel speech enhancement systems rely on frequency domain weighting [1], commonly consisting of a noise power spectral density estimator and a speech spectral or spectral amplitude estimator. The speech estimator applies a statistical estimation rule based on a statistical model of the discrete Fourier transform (DFT) coefficients. The well known Wiener filter estimates the complex speech DFT coefficients with minimum mean square error (MMSE), whereas the Ephraim-Malah algorithm [2] is an MMSE estimator for the speech DFT amplitude. The second estimator is considered advantageous from a perceptual point of view, since the spectral phase is rather unimportant to the listener. Both the Wiener and the Ephraim-Malah estimators assume zero mean Gaussian distributions of real- and imaginary parts for Fourier coefficients of speech and noise. Whereas the Gaussian model is usually a good approximation for the noise DFT coefficients, the real- and imaginary part of the speech coefficients are better modelled with super-Gaussian densities [3]. More accurate statistical models can often not be incorporated into MMSE estimators due to resulting too complicated integrals in the derivation process.

In this contribution, the probability density function of the speech spectral amplitude is approximated by a function with two parameters. The parameters of the underlying PDF can be fitted to the real distribution of the speech

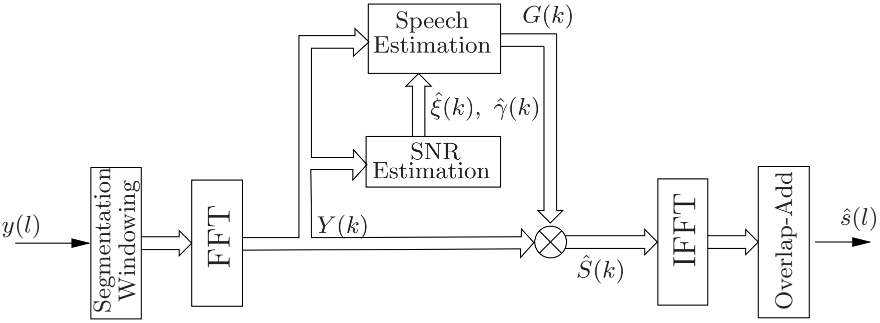


Fig. 4.1. Overview of the single-channel speech enhancement system (l: time index, k: frequency index).

spectral amplitude for a given noise reduction system. Using this statistical model, computationally efficient speech estimators can be found by applying the maximum a posteriori (MAP) estimation rule. The resulting estimators, which are super-Gaussian extensions of the MAP estimators derived by Wolfe and Godsill [4], outperform the commonly applied Ephraim-Malah estimators due to the more accurate statistical model.

To further improve the performance of the noise reduction system, especially in difficult speech-like noise environments, the statistical speech estimation can be extended to multiple input-output signals. A joint statistical model is applied to jointly estimate the speech spectral amplitudes based on the joint observation of all noisy amplitudes. Both MAP estimators with Gaussian and super-Gaussian models of speech can be derived.

The remainder of this Chapter is organized as follows. In Section 4.2 the single-microphone MAP spectral amplitude estimation based speech enhancement system is described. Section 4.3 contains the extension of the MAP estimators for multimicrophone systems and in Section 4.4 experimental performance results are given.

4.2 Single-Channel Statistical Filter

Figure 4.1 shows an overview of the single-channel speech enhancement system examined in this contribution. The noisy time signal $y(l)$ sampled at regular time intervals $l \cdot T$ is composed of clean speech $s(l)$ and additive noise $n(l)$,

$$y(l) = s(l) + n(l). \tag{4.1}$$

After segmentation and windowing with a function $h(l)$, e.g., Hann window, the DFT coefficient of frame λ and frequency bin k is calculated with:

$$Y(\lambda, k) = \sum_{l=0}^{L-1} y(\lambda Q + l) h(l) e^{-j2\pi lk/L}. \tag{4.2}$$

L denotes the DFT frame size. For the noise reduction system applied in this work, $L = 256$ is used at a sampling frequency of 20kHz. For the computation of the next DFT, the window is shifted by Q samples. To decrease the disturbing effects of cyclic convolution, we apply half overlapping Hann windows with 16 zeros at the beginning and end.

The noisy DFT coefficient Y consists of speech part S and noise N

$$Y(\lambda, k) = S(\lambda, k) + N(\lambda, k), \tag{4.3}$$

with $S = S_{\mathrm{Re}} + jS_{\mathrm{Im}}$ and $N = N_{\mathrm{Re}} + jN_{\mathrm{Im}}$, where $S_{\mathrm{Re}} = \mathrm{Re}\{S\}$ and $S_{\mathrm{Im}} = \mathrm{Im}\{S\}$. In polar coordinates, the noisy DFT coefficient of amplitude R and phase ϑ is written as

$$R(\lambda, k) e^{j\vartheta(\lambda,k)} = A(\lambda, k) e^{j\alpha(\lambda,k)} + B(\lambda, k) e^{j\beta(\lambda,k)}. \tag{4.4}$$

The speech DFT amplitude is termed as A, the noise DFT amplitude as B and the respective phases as α, β.

The SNR estimation block calculates a priori SNR ξ and a posteriori SNR γ for each DFT bin k. The SNR calculation requires an estimate of the noise power spectral density $\sigma_N^2(\lambda, k)$. We apply *minimum statistics*, which tracks minima of the smoothed periodogram over a time period that greatly exceeds the speech short time stationarity [5].

Based on the noise estimates $\hat{\sigma}_N^2$ and the observed Fourier amplitudes R, the a priori and the posteriori SNRs are estimated by

$$\hat{\xi}(\lambda, k) = \frac{\hat{\sigma}_S^2(\lambda, k)}{\hat{\sigma}_N^2(\lambda, k)} \quad ; \quad \hat{\gamma}(\lambda, k) = \frac{R^2(\lambda, k)}{\hat{\sigma}_N^2(\lambda, k)}. \tag{4.5}$$

Here, $\hat{\sigma}_S^2$ denotes the instantaneous power spectral density of the speech. Whereas the a posteriori SNRs γ can directly be computed, the a priori SNRs ξ have to be estimated. This is performed using a recursive approach proposed by Ephraim and Malah [2]:

$$\hat{\xi}(\lambda, k) = \alpha_{\mathrm{snr}} \frac{\hat{A}^2(\lambda - 1, k)}{\hat{\sigma}_N^2(\lambda, k)} + (1 - \alpha_{\mathrm{snr}}) F[\gamma(\hat{\lambda}, k) - 1], \tag{4.6}$$

$$\text{with } F[x] = \begin{cases} x \; ; \; x > 0 \\ 0 \; ; \;\; \text{else} \end{cases} .$$

An extended version is presented in [6]. Since the a priori SNR has a high impact on the amount of noise reduction, it is useful to lower limit the a priori SNR according to

$$\tilde{\xi}(\lambda, k) = \begin{cases} \hat{\xi}(\lambda, k); \, \hat{\xi}(\lambda, k) > \xi_{\text{thr}} \\ \xi_{\text{thr}}; \qquad \text{else} \end{cases} . \tag{4.7}$$

The task of the speech estimation block is the calculation of spectral weights G for the noisy spectral components Y, such that the estimated speech DFT coefficient $\hat{S}$ is calculated by

$$\hat{S}(\lambda, k) = G(\hat{\xi}(\lambda, k), \hat{\gamma}(\lambda, k)) \cdot Y(\lambda, k). \tag{4.8}$$

After IFFT and overlap-add, the enhanced time signal $\hat{s}(l)$ is obtained.

4.2.1 Statistical Model

We introduce the statistical model for the speech and noise spectral amplitudes. For the sake of brevity, the frame index λ and frequency index k are omitted, however the following considerations hold independently for every frequency bin k and frame λ.

Motivated by the central limit theorem, real and imaginary part of both speech and noise DFT coefficients are very often modelled as zero mean independent Gaussian [2], [7] with equal variance. The central limit theorem states, that the distribution of the DFT coefficients will converge towards a Gaussian PDF regardless of the PDF of the time samples $y(l)$, if successive samples are statistically independent. This also holds if the correlation in $y(l)$ is short compared to the analysis frame size [8].

For many relevant acoustic noises this assumptions holds. Moreover, multiple noise sources or reverberation often reduce the noise correlation in between the analysis frame size, so that the Gaussian assumption is fulfilled. The variance of the noise DFT coefficient σ_N^2 is assumed to split equally into real and imaginary part. Thus, the probability density function of real- and imaginary part of noise Fourier coefficients can be modelled as:

$$p(N_{\text{Re}}) = \frac{1}{\sqrt{\pi}\sigma_N} \exp\left\{-\frac{N_{\text{Re}}^2}{\sigma_N^2}\right\}. \tag{4.9}$$

Based on (4.9) and the assumption of statistically independent real and imaginary parts, the PDF of the noisy spectrum Y conditioned on the speech amplitude A and phase α can be written as joint Gaussian:

$$p(Y|A, \alpha) = \frac{1}{\pi\sigma_N^2} \exp\left(-\frac{|Y - A\, e^{j\alpha}|^2}{\sigma_N^2}\right). \tag{4.10}$$

A Rice PDF is obtained for the density of the noisy amplitude given the speech amplitude A after polar integration of (4.10) [7]:

$$p(R|A) = \frac{2R}{\sigma_N^2} \exp\left\{-\frac{R^2 + A^2}{\sigma_N^2}\right\} I_0\left(\frac{2AR}{\sigma_N^2}\right), \tag{4.11}$$

where I_0 denotes the modified Bessel function of the first kind and zeroth order.

Considering speech, the span of correlation with typical frame sizes from 10ms to 30ms cannot be neglected. The smaller the frame size, the less Gaussian will the distribution of the speech real- and imaginary part of the Fourier coefficients be. It is well known, that the PDFs of speech samples in the time domain is much better modelled by a Laplace or Gamma density [9]. In the frequency domain, similar distributions can be observed. Martin [3], [10] has abandoned the Gaussian speech model. Instead, the Laplace probability density function,

$$p(S_{\mathrm{Re}}) = \frac{1}{\sigma_S} \exp\left\{ -\frac{2|S_{\mathrm{Re}}|}{\sigma_S} \right\}, \tag{4.12}$$

and Gamma PDFs for statistical independent real and imaginary parts have been proposed,

$$p(S_{\mathrm{Re}}) = \frac{\sqrt[4]{3}|S_{\mathrm{Re}}|^{-\frac{1}{2}}}{2\sqrt[4]{2}\sqrt{\pi\sigma_S}} \exp\left\{ -\frac{\sqrt{3}|S_{\mathrm{Re}}|}{\sqrt{2}\sigma_S} \right\}. \tag{4.13}$$

The same equations hold for the imaginary parts.

Modelling the Spectral Amplitude of Speech. In the following, a simple statistical model for the speech spectral amplitude will be presented [11], which is significantly closer to the real distribution than the commonly applied Gaussian model.

The spectral amplitudes are of special importance, because the phase of the Fourier coefficients can be considered unimportant from a perceptual point of view [12], [13]. Considering noise, the Gaussian assumptions holds due to comparably low correlation in the analysis frame. Assuming statistical independence of real and imaginary parts, the PDF of the noise amplitude B can easily be found as Rayleigh distributed by polar integration,

$$p(B) = \int_0^{2\pi} B \cdot p(N_{\mathrm{Re}}, N_{\mathrm{Im}}) d\beta = \frac{2B}{\sigma_N^2} \exp\left\{ -\frac{B^2}{\sigma_N^2} \right\}. \tag{4.14}$$

For the calculation of an appropriate PDF for A, the Gauss, Laplace, and Gamma PDFs for real and imaginary parts are taken into account. The real and imaginary part of the Fourier coefficients can be considered statistically independent with high accuracy. Then, $p(A)$ can in general be calculated by

$$p(A) = \int_0^{2\pi} A \cdot p(A\cos\alpha) \cdot p(A\sin\alpha) d\alpha, \tag{4.15}$$

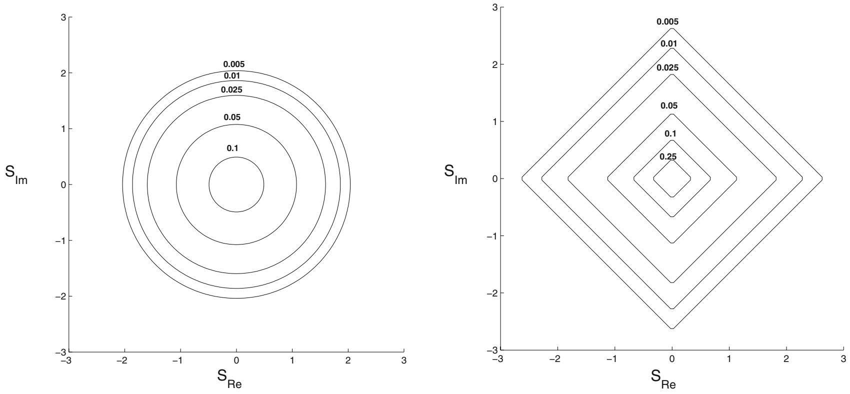


Fig. 4.2. Contour lines of complex Gaussian model with independent cartesian coordinates and of complex Laplace model with independent cartesian coordinates ($\sigma_S^2 = 1$).

with Gaussian PDFs or PDFs according to (4.12) or (4.13) for $p(S_{\mathrm{Re}} = A\cos\alpha),\ p(S_{\mathrm{Im}} = A\sin\alpha)$.

Figure 4.2 shows contour lines of a complex Gaussian or Laplace PDF with independent cartesian components. Compared to the Gaussian PDF, the Laplace PDF has a higher peak, a lower amplitude, and decreases slower towards higher amplitudes visible by the greater distances of the contour lines compared to the complex Gaussian PDF. While the complex Gaussian PDF is rotational invariant, the Laplace amplitude depends on the phase.

Considering Gaussian components, the rotational invariance greatly facilitates the polar integration. Similar to (4.14) the amplitude is Rayleigh distributed:

$$p(A) = \frac{2A}{\sigma_S^2} \exp\left\{-\frac{A^2}{\sigma_S^2}\right\}. \tag{4.16}$$

The PDF of the amplitude of a complex Laplace or Gamma random variable with independent cartesian components varies with the angle α. This makes an analytic calculation of the distribution $A = \sqrt{S_{\mathrm{Re}}^2 + S_{\mathrm{Im}}^2}$ for (4.12) or (4.13) difficult, if not impossible.

Instead of an analytic solution to (4.15), we are looking for a function that approximates the real PDF of the spectral amplitudes with high accuracy regardless of the underlying joint distribution of real and imaginary parts of the Fourier coefficients. However, as indication about how the function should look like, the amplitude of a complex Laplace or Gamma PDF with independent components is taken into account.

Compared to the Rayleigh distributed amplitude of a complex Gaussian, low values are more likely, but the PDF decreases more slowly towards high values.

The fast decay of the Rayleigh PDF results from the second order term of A in the argument of the exponential function in (4.16) similar to the decay of the Gauss function. Similarly, the measured PDFs of the complex Laplace and Gamma amplitude can be assumed to decay like (4.12) and (4.13) with a linear argument in the exponential function.

Apparently, the slope of the Gamma amplitude PDF differs from that of the Laplace amplitude PDF. Hence, a parameter μ is introduced, which enables to approximate both. After normalizing A by the standard deviation σ_S, we thus assume

$$p(A) \sim \exp\left\{-\mu \frac{A}{\sigma_S}\right\}. \tag{4.17}$$

At low values of A, the PDF of the Laplace and Gamma amplitude is much higher than the Rayleigh PDF. Considering the Rayleigh PDF according to (4.16), the behaviour at low values is mainly due to the linear term of A, whereas the exponential term plays a minor role at small values.

Both the PDF of the Laplace amplitude and the PDF of the Gamma amplitude can be approximated by abandoning a linear term in A. Instead, A is taken to the power of a parameter ν after normalization to the standard deviation of speech, i.e., $p(A) \sim \left(\frac{A}{\sigma_S}\right)^{\nu}$ in order to be able to approximate a large variety of PDFs. The smaller the parameter ν, the larger the proposed PDF at low values. The term hardly influences the behaviour of the function at high value due to the dominance of the exponential decay

$$p(A) \sim \frac{A^{\nu}}{\sigma_S^{\nu}} \exp\left\{-\mu \frac{A}{\sigma_S}\right\}. \tag{4.18}$$

After taking $\int_0^{\infty} p(A) dA = 1$ into account, the approximating function with parameters ν, μ is finally obtained using ([14], eq. 3.381.4):

$$p(A) = \frac{\mu^{\nu+1}}{\Gamma(\nu+1)} \frac{A^{\nu}}{\sigma_S^{\nu+1}} \exp\left\{-\mu \frac{A}{\sigma_S}\right\}. \tag{4.19}$$

Here, Γ denotes the Gamma function.

Figure 4.3 shows the approximation of measured histograms of the amplitude of 1.000.000 complex Laplace or Gamma random values with independent cartesian components with $\sigma_S^2 = 1$ by (4.19) using different sets of parameters ν, μ. Apparently, (4.19) allows a very accurate approximation for both Laplace and Gamma components. To approximate the Laplace amplitude, we applied the parameter set ($\nu = 1$, $\mu = 2.5$). To approximate the Gamma amplitude we used ($\nu = 0.01$, $\mu = 1.5$). PDFs in between both or closer to the Rayleigh PDF can be approximated with different sets of parameters ν, μ.

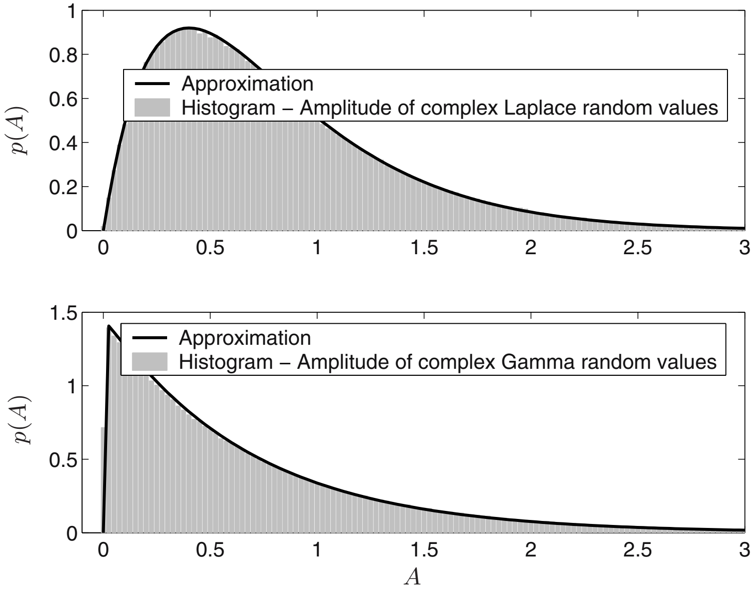


Fig. 4.3. Approximation of amplitudes of complex random values with Laplace and Gamma components using (4.19). (upper plot) Laplace components: ($\nu = 1$, $\mu = 2.5$). (lower plot) Gamma components: ($\nu = 0.01$, $\mu = 1.5$).

Matching with Experimental Data. The real PDF of the speech amplitude will not be exactly like the Laplace amplitude or Gamma amplitude approximation but somewhere in between. Also, it will depend on parameters of the noise reduction system such as the analysis frame size. At a larger frame size, the correlation decreases relatively to the analysis frame size and thus the distribution will be less super-Gaussian. The task is therefore to find a set of parameters (ν,μ) which outperforms the above sets for Laplace or Gamma amplitude approximation for a given system.

To measure the PDF of the speech complex DFT coefficients S or speech DFT amplitudes A, a histogram is built using 1 hour speech from different speakers. Ideally, DFT bins, which solely contain speech of equal variance, should be taken into account.

In practice, the speech variance in a frequency bin is strongly time-variant and can only be estimated in a time frame and frequency bin with a certain estimation error. Thus, we apply (4.6), which is commonly considered as the best performing method to estimate the speech variance in form of the a priori SNR. Hereby, the histogram measurement process also incorporates the same method of estimating the time-varying speech variance as the noise reduction system. Data is collected for the histogram at time instances, when the frequency bin is dominated by speech. For that purpose, a high and

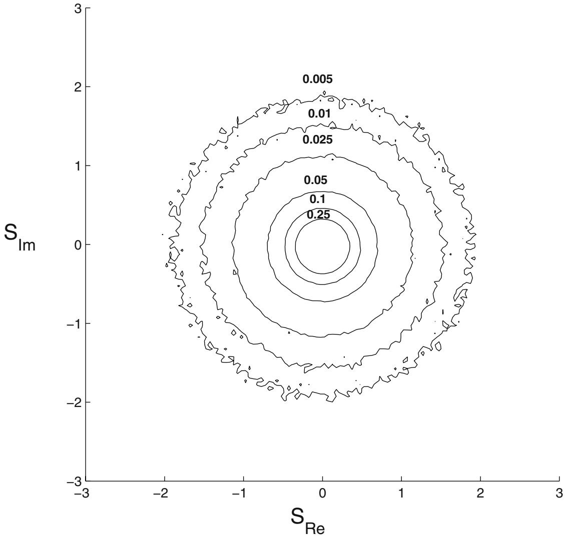


Fig. 4.4. Contour lines of measured speech DFT speech coefficients.

narrow a priori SNR interval is predefined, e.g. 19-21dB. The width of the interval is a tradeoff between the amount of data obtained and the demand to pick samples of same variance.

The left part of Fig. 4.4 shows the contour lines of the measured speech DFT coefficients after normalizing to $\sigma_S^2 = 1$ and averaging over frequency bins afterwards.

Compared to the Gaussian contour lines in Fig. 4.2, a slower decrease towards high amplitudes and faster increase towards low amplitudes is visible. Also, the observed data hardly shows any dependency on the phase like for the Laplace contour lines in Fig. 4.2 like shown for the complex Laplace PDF in the right part of Fig. 4.4. Figure 4.5 plots the histogram of the speech amplitude, which is obtained by integration over the phase of the two-dimensional histogram along with the analytic Rayleigh PDF and the approximation according to (4.19) with the parameter set for Laplace and Gamma amplitude approximation respectively. Apparently, (4.19) provides a much better fit for the speech amplitude than the Rayleigh PDF for both Laplace and Gamma amplitude approximation. For low arguments, the Rayleigh PDF rises too slowly, while for large arguments, the density function decays too fast. The real PDF of the speech amplitude lies between the Laplace and Gamma amplitude approximation.

To find a set (ν, μ), that approximates the real PDF best, a distance measure between the analytic function and the histogram with N bins is numerically minimized. The Kullback divergence [15] can be considered optimal from an information theoretical point of view. Given two random variables

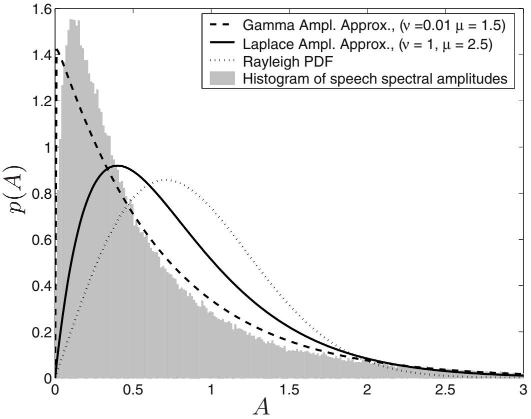


Fig. 4.5. Histogram of speech DFT amplitudes A ($\sigma_S^2 = 1$) fitted with Rayleigh PDF and Laplace/Gamma amplitude approximation (4.19).

of probability density $p_1(x)$ and $p_2(x)$, then I(2:1) describes the mean information per observation of process 2 for discrimination in favor of process 2 and I(1:2) for discrimination in favor of process 1:

$$I(1:2) = \int p_1(x) \log \frac{p_1(x)}{p_2(x)} dx \quad ; \quad I(2:1) = \int p_2(x) \log \frac{p_2(x)}{p_1(x)} dx. \quad (4.20)$$

The sum $J(1:2) = I(1:2) + I(2:1)$ is a measure of divergence between the two processes. To differentiate between the analytical $p_A(n)$ and the histogram PDF $p_h(n)$ with N bins, the divergence can be calculated by

$$J(A:h) = \sum_{n=1}^{N} (p_h(n) - p_A(n)) \log \left(\frac{p_h(n)}{p_A(n)} \right). \quad (4.21)$$

Figure 4.6 shows the best $p(A)$ according to (4.19) determined by minimizing the Kullback divergence. The analytical PDF now fits even better to the observed data than the Laplace of Gamma amplitude approximation. To illustrate the improvement provided by the new model, Table 4.1 shows the Kullback divergences between measured data and model functions. The divergences have been normalized to that of the Rayleigh PDF, i.e., the Gaussian model. When using the Laplace or Gamma amplitude approximation, the Kullback divergence is significantly lower than for the Gaussian model. By determining an optimal parameter set, the divergence further decreases.

Spectral Amplitude of Noise. Compared to speech, the span of noise correlation in an analysis frame is much lower. Thus, the PDF of the real

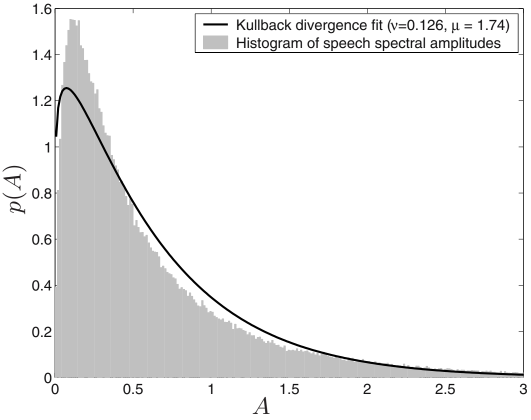


Fig. 4.6. Histogram of speech DFT amplitudes and fitted approximation by (4.19) according to Kullback divergence ($\sigma_S^2 = 1$).

Table 4.1 Normalized Kullback divergence between measured speech PDF and different model function.

$p(A)$	ν, μ	$J(A:h)/J(A:h)_{Rayleigh}$
Rayleigh (4.16)	–	1
Laplace Amplitude Approximation (4.19):	1, 2.5	0.35
Gamma Amplitude Approximation (4.19):	0.01, 1.5	0.05
Kullback Fit (4.19):	0.126, 1.74	0.045

and imaginary parts of the noise spectral coefficients will according to the central limit theorem be closer to a Gaussian function. Martin [3], [10] has proposed spectral estimators with Laplace or Gaussian noise model (and Laplace and Gamma models for the speech coefficients). A Laplace model for noise is motivated by the observation that environmental noises are also super-Gaussian distributed to a certain degree. Figure 4.7 plots histograms of DFT amplitudes measured for three different noise classes. For building of the histograms, the frequency and time dependent noise variances σ_N^2 were estimated using the same system as applied in the noise reduction algorithm, i.e. minimum statistics [5]. Spectral amplitudes with corresponding estimated noise variances inside a narrow predefined interval were then collected for the histogram database. To plot the histogram together with the Rayleigh function (4.16) and the super-Gaussian model function (4.19) in Fig. 4.7, the collected database was normalized to $\sigma_N^2 = 1$.

Unsurprisingly, for the white noise, which was uniformly distributed in the time domain, a Rayleigh function perfectly models the PDF of the noise spectral amplitude. For fan noise, the PDF slightly changes towards the Laplace

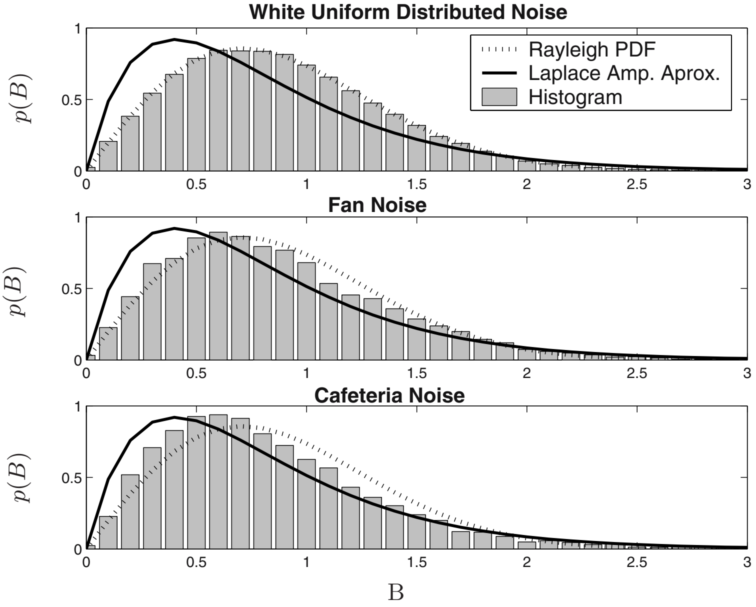


Fig. 4.7. Histogram of noise DFT amplitudes B for white noise, fan noise and cafeteria noise ($\sigma_N^2 = 1$) fitted with Rayleigh-PDF and Laplace amplitude approximation.

amplitude approximation, while the effect is more visible for the cafeteria noise, which contains speech components from many speakers. Due to the low deviation from the Rayleigh PDF, the Gaussian assumption for the noise will be assumed in the following.

4.2.2 Speech Estimators

The task of the speech estimator lies in calculating an estimate for the speech spectral amplitude $\hat{A} = G \cdot R$ given the observed noisy coefficient Y or the noisy amplitude R and the variances of speech σ_S^2 and noise σ_N^2. With probability one, the estimate will not be identical to the real value, therefore a cost function $C(A, \hat{A})$ is introduced [16], which assigns a value to each combination of undisturbed and estimated speech spectral amplitude. The Bayesian estimators aim at minimizing the expectation of the cost according to

$$E\{C(A, \hat{A})\} = \int_{-\infty}^{\infty} \int_{0}^{\infty} C(A, \hat{A}) p(A, Y) dA dY. \tag{4.22}$$

For $C(A, \hat{A}) = (A - \hat{A})^2$, the Ephraim-Malah or conditional expectation estimator [2] is obtained:

$$G \;=\; \frac{\sqrt{v}}{\gamma} \cdot \Gamma(1.5)\, F_1(-0.5\,,\, 1, -v), \quad \text{with } v = \gamma \frac{\xi}{1+\xi}, \tag{4.23}$$

where the confluent hypergeometric series F_1 can be calculated with

$$F_1(-0.5\,,\, 1, -v) \;=\; e^{-v/2} \left[(1+v)\, I_0 \left(\frac{v}{2} \right) + v\, I_1 \left(\frac{v}{2} \right) \right]. \tag{4.24}$$

I_0, I_1 denote the modified Bessel function of zero-th and first order.

The cost function $C(A, \hat{A}) = \log A - \log \hat{A}$ leads to the logarithmic Ephraim-Malah estimator [17]. By choosing a uniform cost function according to

$$C = \begin{cases} 0\,; & |S - \hat{S}| < \epsilon \\ 1\,; & \text{else} \end{cases}, \tag{4.25}$$

MAP estimators can be obtained, which are in general computationally more efficient.

Wolfe and Godsill [4] introduced alternatives to the Ephraim-Malah spectral amplitude estimator based on the maximum a posteriori estimation rule. The spectral weights obtained by the MAP estimators are similar to those of the Ephraim and Malah estimator, thus a quality improvement cannot be expected. However, straightforward implementations without the use of computational expensive Bessel or exponential function are possible.

In the following, we introduce two speech spectral amplitude estimators, which keep the computational simplicity of the Wolfe and Godsill estimators but also achieve a quality gain by applying the super-Gaussian speech model according to (4.19) and a Gaussian model for noise.

First, a MAP estimator for the speech spectral amplitude is derived. Secondly, a joint MAP estimator for the amplitude and phase is introduced. Both estimators are extensions of the MAP estimators proposed in [4].

MAP Spectral Amplitude Estimator. A computationally efficient MAP solution following

$$\hat{A} = \arg\max_{A} p(A|R) = \arg\max_{A} \frac{p(R|A)p(A)}{p(R)} \tag{4.26}$$

similar to [4], where Gaussian distributed S_{Re}, S_{Im} are assumed, can be found. Now, the super-Gaussian function (4.19) is used to model the PDF of the speech spectral amplitude $p(A)$. The Gaussian assumption of noise allows to apply (4.11) for $p(R|A)$. We need to maximize only $p(R|A) \cdot p(A)$, since $p(R)$ is independent of A. A closed form solution can be found if the modified Bessel function I_0 is considered asymptotically, with

$$I_0(x) \approx \frac{1}{\sqrt{2\pi x}}\; e^{x}. \tag{4.27}$$

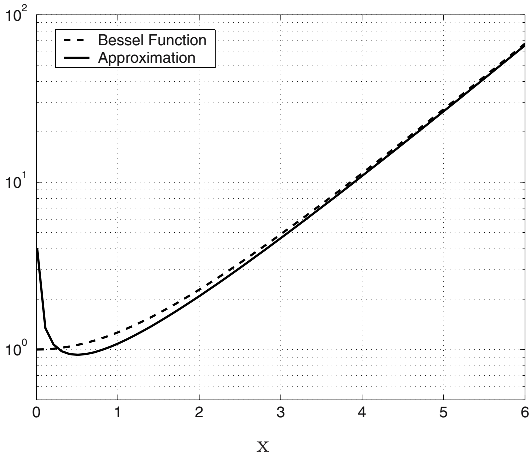


Fig. 4.8. Modified Bessel function of zero-th order I_0 and approximation (4.27).

Figure 4.8 shows that the approximation is reasonable for large arguments and becomes erroneous for low arguments.

After insertion of (4.27) in (4.11), we get for $p(R|A)p(A)$:

$$p(R|A)p(A) \sim A^{\nu-\frac{1}{2}} \exp\left\{-\frac{A^2}{\sigma_N^2} - A(\frac{\mu}{\sigma_S} - \frac{2R}{\sigma_N^2})\right\}. \tag{4.28}$$

Note, that the approximation of the Bessel function has introduced a negative exponent for $\nu > 0.5$.

Instead of differentiating $p(R|A)p(A)$, the maximization can be performed better after applying the natural logarithm, because the product of the polynomial and exponential converts into a sum:

$$\frac{d \log[p(R|A)p(A)]}{dA} = \left(\nu - \frac{1}{2}\right)\frac{1}{A} - \frac{2A}{\sigma_N^2} - \frac{\mu}{\sigma_S} + \frac{2R}{\sigma_N^2} \overset{!}{=} 0. \tag{4.29}$$

After multiplication with A, one reasonable solution $\hat{A} = GR$ to the quadratic equation is found, because the second solution delivers spectral amplitudes $A < 0$ at least for $\nu > 0.5$. The second derivative at $\hat{A}$ is negative, thus a local maximum is guaranteed:

$$G = u + \sqrt{u^2 + \frac{\nu - \frac{1}{2}}{2\gamma}} \quad \text{with} \quad u = \frac{1}{2} - \frac{\mu}{4\sqrt{\gamma\xi}}. \tag{4.30}$$

Whereas the MAP spectral amplitude estimator is very useful for an estimation with an underlying Laplace model of the DFT coefficients, it cannot be applied using a Gamma model or the optimal parameter set. This is due

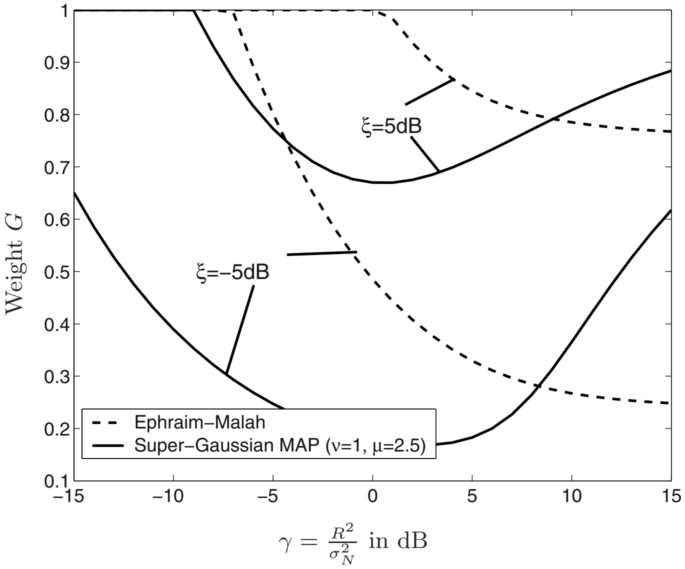


Fig. 4.9. Weights of the super-Gaussian MAP estimator with Laplace amplitude approximation ($\nu = 1, \mu = 2.5$) compared to the Ephraim-Malah weighting rule depending on the a posteriori SNR γ for two a priori SNRs $\xi = -5dB$ and $\xi = 5dB$.

to the inaccuracy introduced by the approximation of the Bessel function (4.27). For $\nu < 0.5$, the approximated a posteriori density $p(A|R)$ has a pole at $A = 0$, which will misplace the maximum found by (4.30).

Figure 4.9 shows the dependency of the weights on the a posteriori SNR γ for two a priori SNRs ξ for the parameter set (ν, μ), that approximates the amplitude of a complex Laplace PDF. Most of the time, the weights of the super-Gaussian estimator are smaller than those of the Ephraim-Malah algorithm due to the larger value of $p(A)$ at low amplitudes compared to the Rayleigh PDF. At high a posteriori SNRs, the Ephraim-Malah weights converge towards the Wiener weights, i.e., $\xi/(1+\xi)$. The weights of the super-Gaussian MAP estimator, however, increase due to the slower decay of the model function towards larger values. Higher observed spectral amplitudes R will result in a higher spectral output compared to the Wiener filter or Ephraim-Malah estimator. This effect is due to the underlying more accurate statistical model of the spectral amplitude of speech, in which high amplitudes are considered more likely than in the Rayleigh model. Consequently, high observed noisy amplitude will be judged to contain more speech components by the super-Gaussian MAP estimator than by the Ephraim-Malah rule

Joint MAP Amplitude and Phase Estimator. To overcome the inability of the proposed MAP estimator with approximation of the Bessel function to cope with an underlying Gamma model or the model that minimizes the Kullback divergence towards the measured data, we introduce a joint MAP estimator of the amplitude and phase. Instead of maximizing the a posteriori probability $p(A|R)$, we now jointly maximize the probability of amplitude and phase conditioned on the observed complex coefficient, i.e., $p(A, \alpha|Y)$,

$$\hat{A} = \arg\max_{A} p(A, \alpha|Y) = \arg\max_{A} \frac{p(Y|A,\alpha)p(A,\alpha)}{p(Y)}, \tag{4.31}$$

$$\text{and} \quad \hat{\alpha} = \arg\max_{\alpha} p(A, \alpha|Y) = \arg\max_{\alpha} \frac{p(Y|A,\alpha)p(A,\alpha)}{p(Y)}. \tag{4.32}$$

If the problem is formulated this way, the Bessel function and its erroneous approximation are avoided. $p(Y|A, \alpha)$ is given by (4.10) using the Gaussian assumption of noise. Up to now we have only dealt with the probability of the speech amplitude, i.e., $p(A)$, while the joint PDF of the amplitude and phase $p(A, \alpha)$ is now asked for. For a rotational invariant PDF it is obtained:

$$p(A, \alpha) = \frac{1}{2\pi} p(A). \tag{4.33}$$

Equations (4.31) and (4.32) can be solved similar to the MAP estimator. Again, the natural logarithm greatly facilitates the optimization process. The partial derivatives of $\log(p(Y|A, \alpha)p(A, \alpha))$ with respect to the phase α and amplitude A need to be zero. Differentiating with respect to α yields

$$\frac{\delta}{\delta\alpha} \log(p(Y|A,\alpha)p(A,\alpha)) = \tag{4.34}$$
$$-\frac{(Y^* - Ae^{-j\alpha})(-jAe^{j\alpha}) + (Y - Ae^{j\alpha})(jAe^{-j\alpha})}{\sigma_N^2}.$$

Setting to zero and substituting $Y = Re^{j\vartheta}$ yields

$$\hat{\alpha} = \vartheta. \tag{4.35}$$

The candidate for the joint MAP phase estimate is simply the noisy phase. Differentiating w.r.t. the speech amplitude gives

$$\frac{\delta}{\delta A} \log(p(Y|A,\alpha)p(A,\alpha)) = \frac{(Y^* - Ae^{-j\alpha})e^{j\alpha} + (Y - Ae^{j\alpha})e^{-j\alpha}}{\sigma_N^2} + \frac{\nu}{A} - \frac{\mu}{\sigma_S}.$$

Setting to zero and replacing $\alpha = \vartheta$ the following quadratic equation is obtained:

$$A^2 + A\left(\frac{\mu\sigma_N^2}{2\sigma_S} - R\right) - \frac{\nu}{2}\sigma_N^2 \overset{!}{=} 0. \tag{4.36}$$

Solving the equation leads to an estimation rule similar to that of the super-Gaussian MAP estimator:

$$G = u + \sqrt{u^2 + \frac{\nu}{2\gamma}} \quad \text{with} \quad u = \frac{1}{2} - \frac{\mu}{4\sqrt{\gamma\xi}}. \tag{4.37}$$

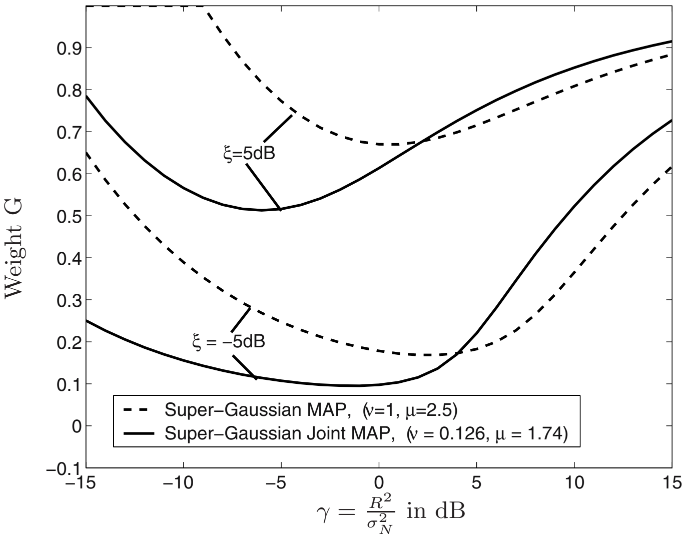


Fig. 4.10. Weights of the joint MAP estimator as a function of the a posteriori SNR γ with different parameter sets, i.e., Laplace and Gamma amplitude approximation as well as Kullback divergence matching, compared to the MAP estimator with Laplace approximation model for $\xi = -5dB$.

Again, checking the second derivatives guarantees, that the extremum found by (4.37) is a local maximum.

Figure 4.10 plots the weights of the joint MAP estimator with optimal parameters for the given speech enhancement system in dependence of the a posteriori SNR for two different a priori SNRs. For comparison, the weights of the MAP estimator with Laplace amplitude approximation are also plotted. The joint MAP weighting rule with optimal parameters delivers lower values at low observed SNRs, while rising faster towards higher a posteriori SNRs compared to the MAP estimation with Laplace model. This behaviour is directly due to the different underlying statistical models of the speech amplitude by using different parameters (ν, μ) in (4.19). Low observed a posteriori SNRs compared to the ratio of variances in form of the a priori SNR will highlight the effect of the statistical model at low values of A, while the behaviour at high a posteriori SNRs will be influenced by the values of the PDF towards high speech spectral amplitudes.

4.3 Multichannel Statistical Filter

In this section, multi-microphone extensions for the MAP spectral amplitude estimators are described [18].

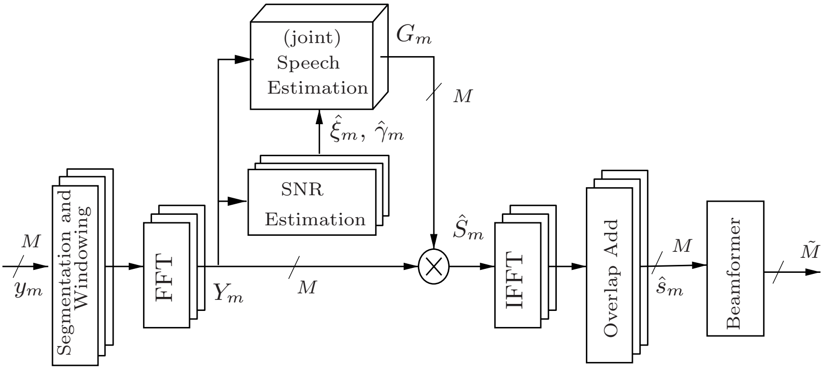


Fig. 4.11. Multichannel statistical filter.

Figure 4.11 depicts the extended noise reduction system. Now, M noisy time signals y_m, microphone index $m \in \{1 \ldots M\}$, are transformed into the frequency domain using DFT. For the sake of brevity, the frequency index k has been omitted. For each channel, the noise power spectral density and the a priori SNR is separately estimated using minimum statistics. The speech estimator however, now calculates real weights G_m using all noisy DFT coefficients Y_m and all estimated a priori $\hat{\xi}_m$ and a posteriori SNRs $\hat{\gamma}_m$. M enhanced signals are resynthesized using IDFTs and overlap add. By applying a beamformer, the outputs can be combined afterwards to $\tilde{M} \leq M$ signals.

The joint speech estimator can similarly to the single channel application be based on the a posteriori PDF $p(A|R)$. However, now the density for each channel index m can be conditioned on the observation of all noisy amplitudes, i.e. $p(A_m|R_1 \ldots R_M)$. Using Bayes rule, the a posteriori PDF can be expressed as

$$p(A_m|R_1 \ldots R_M) = \frac{p(R_1 \ldots R_M|A_m)p(A_m)}{p(R_1 \ldots R_M)}. \tag{4.38}$$

Thus, a joint statistical estimator requires a model for the joint density $p(R_1 \ldots R_M)$ and an a priori model for the speech amplitude $p(A_m)$ similar to Section 4.2.

4.3.1 Joint Statistical Model

To find a simple statistical model for the joint transition density $p(R_1 \ldots R_M|A_m)$, the typical noise reduction scenario of Fig. 4.12, e.g., inside a crowded cafeteria or restaurant or inside a car, is considered. A desired signal s arrives at a microphone array from angle θ_S. Multiple noise sources arrive from various angles. The noisy time signals y_m sampled at time index

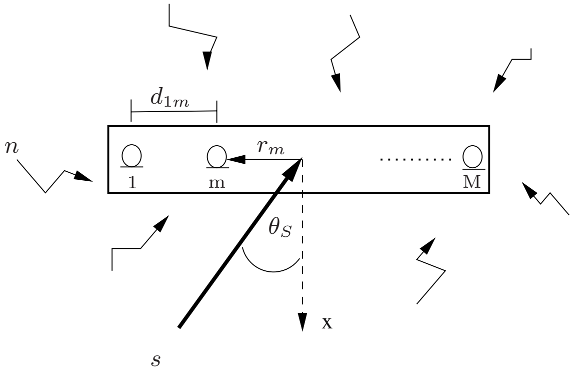


Fig. 4.12. Speech and noise arriving at microphone array.

l can then be expressed by the convolution of the clean signal with a channel dependent impulse response h_m plus channel dependent noise,

$$y_m(l) = h_m(l) * s(l) + n_m(l). \tag{4.39}$$

In the frequency domain, the DFT coefficient Y_m of microphone m consists of speech and noise components

$$Y_m = R_m e^{\vartheta_m} = c_m A e^{j\alpha_m} + N_m. \tag{4.40}$$

For the sake of brevity, the frequency index k is omitted here. Due to different microphone amplifications or near field effects, the speech spectral amplitude differs between the microphones by a factor c_m. Angles of arrival other than $\theta_S = 0°$ cause phase differences in the speech coefficients, i.e., the angle α depends on the channel m. The diffuse noise field in a scenario like depicted in Fig. 4.12 can be characterized by its coherence function. The magnitude squared coherence (MSC) between two omnidirectional microphones m and n of a diffuse noise field is given by, e.g., [19]

$$\mathrm{MSC}_{mn}(f) = \frac{|\varPhi_{mn}(f)|^2}{\varPhi_{mm}(f)\varPhi_{nn}(f)} = \mathrm{si}^2\left(\frac{2\pi f d_{mn}}{c}\right). \tag{4.41}$$

Figure 4.13 plots the theoretical coherence of an ideal diffuse noise field and the measured coherence of the noise field inside a crowded cafeteria with a microphone distance of $d_{mn} = 12$cm. $\mathrm{MSC}_{mn}(f)$ attains its first zero at frequency $f_0 = c/2d_{mn}$, above f_0 the MSC becomes very low and thus the noise components of the noisy spectra can be considered uncorrelated with

$$E\{N_m N_n^*\} = \begin{cases} \sigma_{N_m}^2 & ; m = n \\ 0 & ; m \neq n \end{cases}. \tag{4.42}$$

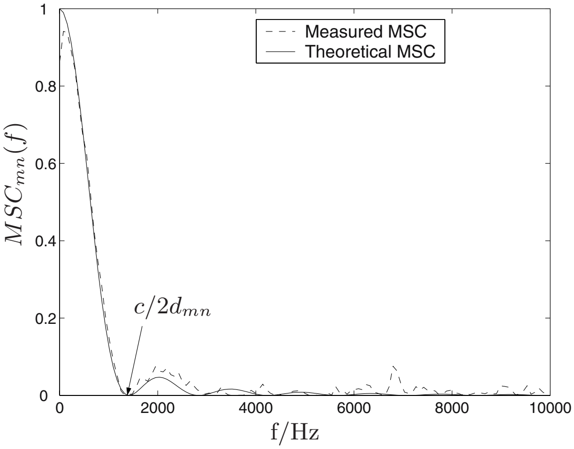


Fig. 4.13. Theoretical MSC of a diffuse noise field and measured MSC inside a crowded cafeteria ($d_{mn} = 12$cm).

Hence (4.11) can be extended to

$$p(R_1, \ldots, R_M|A_m) = \prod_{i=1}^{M} p(R_i|A_m) \tag{4.43}$$

for each $m \in \{1 \ldots M\}$. The time delay of the speech signals between the microphones is assumed to be small compared to the short time stationarity of speech and thus the speech spectral amplitudes A_m to be highly correlated. However, due to effects mentioned above, the speech amplitudes are allowed to deviate by a constant channel dependent factor c_i, i.e., $A_i = c_i \cdot A$ and $\sigma_{S_i}^2 = c_i^2 \sigma_S^2$. Thus is can be written: $p(R_i|A_i = \frac{c_i}{c_m} A_m) = p(R_i|A_m)$. Using the Gaussian model for noise, the joint PDF of all noisy amplitudes R_i given the speech amplitude of channel m can then using (4.43) and (4.11) be written as

$$p(R_1, \ldots, R_M|A_m) = \tag{4.44}$$
$$\exp\left\{-\sum_{i=1}^{M} \frac{R_i^2 + (\frac{c_i}{c_m})^2 A_m^2}{\sigma_{N_i}^2}\right\} \cdot \prod_{i=1}^{M} \left[\frac{2R_i}{\sigma_{N_i}^2} I_0\left(\frac{2\frac{c_i}{c_m} A_m R_i}{\sigma_{N_i}^2}\right)\right],$$

where the c_i are fixed parameters of the joint PDF.

4.3.2 Multichannel MAP Spectral Amplitude Estimation

First, a multichannel MAP estimator with Gaussian speech and noise model, which extends the Wolfe-Godsill [4] estimator to multiple microphones is presented. Now, the speech spectral amplitude $\hat{A}_m$ is searched for, that maximizes the PDF of A_m conditioned on the joint observation of all $R_1 \ldots R_M$, i.e.

$$\begin{aligned}\hat{A}_m &= \arg\max_{A_m} p(A_m|R_1,\ldots,R_M) \\ &= \arg\max_{A_m} \frac{p(R_1,\ldots,R_M|A_m)p(A_m)}{p(R_1,\ldots,R_M)}. \end{aligned} \tag{4.45}$$

Only $C = p(R_1,\ldots,R_M|A_m) \cdot p(A_m)$ needs to be maximized, since $p(R_1,\ldots,R_M)$ is independent of A_m. Using (4.43) and the Gaussian model for speech, i.e. (4.19), it is obtained for $\tilde{C} = \log C$,

$$\begin{aligned}\tilde{C} &= \log\left(\frac{A_m}{\pi\sigma^2_{S_n}}\right) - \frac{A_n^2}{\sigma^2_{S_m}} \\ &\quad + \sum_{i=1}^{M}\left[\log\frac{2R_i}{\sigma^2_{N_i}} - \frac{R_i^2 + (\frac{c_i}{c_m})^2 A_m^2}{\sigma^2_{N_i}} + \log\left(I_0\left(2\frac{\frac{c_i}{c_m}A_m R_i}{\sigma^2_{N_i}}\right)\right)\right]. \end{aligned} \tag{4.46}$$

After approximating the Bessel function with (4.27), differentiation of $\log C$ and multiplication with the amplitude A_m leads to

$$A_m^2\left(-\frac{1}{\sigma^2_{S_m}} - \sum_{i=1}^{M}\frac{(\frac{c_i}{c_m})^2}{\sigma^2_{N_i}}\right) + A_m\sum_{i=1}^{M}\frac{\frac{c_i}{c_m}R_i}{\sigma^2_{N_i}} + \frac{2-M}{4} \overset{!}{=} 0. \tag{4.47}$$

The resulting weight factor for microphone m is then given as

$$G_m = \frac{\frac{1}{2}\sqrt{\frac{\xi_m}{\gamma_m}}}{1+\sum\limits_{i=1}^{M}\xi_i}\mathrm{Re}\left\{\sum_{i=1}^{M}\sqrt{\gamma_i\xi_i} + \sqrt{\left(\sum_{i=1}^{M}\sqrt{\gamma_i\xi_i}\right)^2 + (2-M)(1+\sum_{i=1}^{M}\xi_i)}\right\} \tag{4.48}$$

If $M = 1$, (4.48) simplifies to the single-channel MAP estimator with Gaussian model as given in [4].

Super-Gaussian Speech Model. Similar to the single microphone MAP estimator with super-Gaussian speech model as presented in Section 4.2, the a priori model of $p(A_m)$ for the multi-microphone speech amplitude estimator can be improved by the use of the parametric super-Gaussian statistical model. Now, the starting point is the joint MAP approach

$$\hat{A}_m = \arg\max_{A_m} \frac{p(R_1,\ldots,R_M|A_m)p(A_m)}{p(R_1,\ldots,R_M)}, \tag{4.49}$$

with the model function

$$p(A_m) = \frac{\mu^{\nu+1}}{\Gamma(\nu+1)}\frac{A_m^{\nu}}{\sigma^{\nu+1}_{S_m}}\exp\left\{-\mu\frac{A_m}{\sigma_{S_m}}\right\}, \tag{4.50}$$

for the PDF of the speech spectral amplitude of microphone m.

Again, differentiation of $\tilde{C} = \log(R_1, \ldots, R_M | A_m) \cdot p(A_m)$ with approximation of the the Bessel function gives a quadratic equation

$$\frac{d\tilde{C}}{dA_m} = \frac{\nu}{A_m} - \frac{\mu}{\sigma_{S_m}} + \sum_{i=0}^{M} \left(-\frac{2(\frac{c_i}{c_m})^2 A_m}{\sigma_{N_i}^2} + 2\frac{\frac{c_i}{c_m} R_i}{\sigma_{N_i}^2} - \frac{1}{2A_m} \right), \tag{4.51}$$

which leads to the weight rule of the multichannel MAP estimator with underlying Gaussian model:

$$G_m = \frac{\sqrt{\frac{\xi_m}{\gamma_m}}}{2\sum_{i=1}^{M} \xi_i} \left[\sum_{i=1}^{M} \sqrt{\xi_i \gamma_i} - \frac{\mu}{2} + \sqrt{\left(\sum_{i=1}^{M} \sqrt{\xi_i \gamma_i} - \frac{\mu}{2} \right)^2 + (2\nu - M) \sum_{i=1}^{M} \xi_i} \right]. \tag{4.52}$$

If $M = 1$, (4.52) simplifies to the super-Gaussian spectral amplitude estimator given by (4.30).

4.4 Experimental Results

In this section, the performance of the single and multi-microphone MAP estimators is instrumentally evaluated in comparison with the Ephraim-Malah algorithm.

The noise reduction filter were applied speech with additive noise at different SNRs. To judge the performance of a noise reduction algorithm, the system depicted in Fig. 4.14 was applied. The desired signal s and the interfering undesired signal n are superposed with a given SNR. The noisy signal $y(l)$ is processed with the noise reduction algorithm. Afterwards, the desired and the interfering signal are separately processed with the resulting filter coefficients. Hence, the system enables separate tracking of speech quality and noise reduction amount by comparing outputs to inputs of the fixed filters. Using the master-slave system depicted in Fig. 4.14, the speech quality is tracked in the upper branch by the segmental signal to noise ratio, i.e.,

$$\text{seg. Speech-SNR/dB} = \frac{1}{P} \sum_{p=1}^{P} \left(10 \cdot \log_{10} \left(\frac{\sum_{i=1}^{I} s^2(i + pI)}{\sum_{i=1}^{I} (s(i + pI) - \tilde{s}(i + pI))^2} \right) \right), \tag{4.53}$$

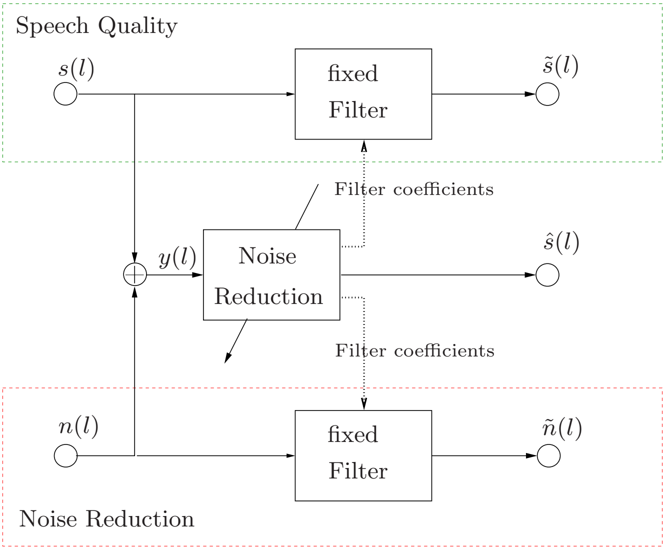


Fig. 4.14. Instrumental performance evaluation of the noise reduction system.

where I denotes the length of the segment, e.g. the frame shift of the DFT analysis window, and P the number of segments, such that $P \cdot I$ is the overall length.

On the other hand, the noise reduction amount is measured in the lower branch of Fig. 4.14 by the segmental noise power attenuation:

$$\text{seg. Noise Reduction/dB} = \frac{1}{P}\sum_{p=1}^{P}\left(10 \cdot \log_{10}\left(\frac{\sum\limits_{i=1}^{I} n^2(i+pI)}{\sum\limits_{i=1}^{I} (\tilde{n}^2(i+pI)}\right)\right). \quad (4.54)$$

To highlight the noise reduction during speech, we only take segments p with global speech activity into account. The global activity is detected in advance by applying a VAD on the clean speech signal.

Single-Microphone MAP Estimators. In the following, the performance of the single microphone system according to Fig. 4.1 when using the Ephraim-Malah estimator, the super-Gaussian MAP estimator with Laplace amplitude approximation, and the super-Gaussian joint MAP estimator with optimal parameters, is compared for speech with three different noises.

The parameters (ν, μ) determine the underlying statistical model of the speech amplitude. For the super-Gaussian MAP estimator, we favor ($\nu = 1$,

$\mu = 2.5$), which approximate the amplitude of a complex RV with independent Laplace components. If the parameters are adjusted for Gamma distributed components or in order to minimize the Kullback divergence, the enhanced signal is greatly disturbed. This is due to the approximation of the Bessel function, which generates an uncompensated pole at $A = 0$ for $\nu < 0.5$. In general, the proposed super-Gaussian MAP estimator cannot be applied for $\nu < 0.5$.

The super-Gaussian joint MAP estimator, however, can be applied to every reasonable set of parameters (ν, μ). Here, we favor the parameters, that were determined by minimizing the Kullback divergence towards the measured data, i.e., ($\nu = 0.126$, $\mu = 1.74$).

The amount of noise reduction using (4.30) with ($\nu = 1$, $\mu = 2.5$) or (4.37) with ($\nu = 0.126$, $\mu = 1.74$) is significantly higher than for the Ephraim-Malah algorithm. Consequently, a lower speech quality will be reached. Comparing speech quality and noise reduction of the super-Gaussian estimators to the Ephraim-Malah estimator would thus be of limited value. For comparability the weights of the super-Gaussian estimators are scaled by a constant factor greater than one so that approximately the same speech quality is reached for all estimators according to (4.53). The amount of noise reduction achieved then allows a comparison between the estimators. In all versions, we include the soft weight given by Ephraim and Malah [2] with tracking of speech absence probabilities [20].

The results for white noise and the three different estimators, i.e., Ephraim-Malah, MAP with ($\nu = 1, \mu = 2.5$), and joint MAP with ($\nu = 0.126, \mu = 1.74$) are shown in Fig. 4.15. The super-Gaussian MAP estimator achieves a significantly higher noise attenuation than the Ephraim-Malah estimator. By applying the super-Gaussian joint MAP estimator with parameters optimally adjusted to the measured data, the noise reduction amount can be increased further without decreasing the speech quality.

Figures 4.16 and 4.17 plot the performance of the estimators for speech with fan noise and cafeteria noise respectively.

The noise reduction amount is lower than for white noise, because the non-stationary cafeteria and fan noise are harder to track by the noise estimation algorithm. The proposed super-Gaussian estimators still outperform the Ephraim-Malah algorithm although the performance gain is lower than for the white noise. Again, the joint MAP estimator with optimal parameters performs best.

Multi-Microphone MAP Estimators. The performance of the multi-microphone estimators is considered with focus to enhancement in a critical cafeteria situation using a microphone array with few elements. The cafeteria noise is generally difficult to suppress due to its instationarity and speech-shaped spectrum.

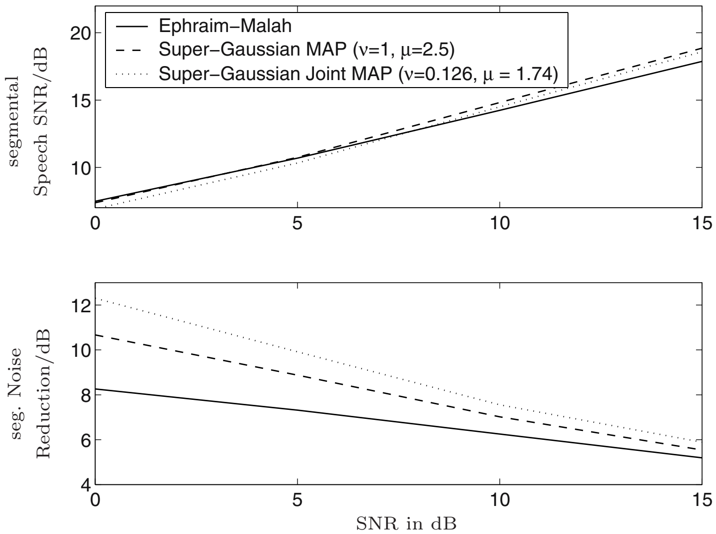


Fig. 4.15. Speech quality and noise reduction amount of statistical filter with Ephraim-Malah estimator (solid) with super-Gaussian MAP estimator (dashed) and super-Gaussian joint MAP estimator (dotted) for speech corrupted with white noise.

The estimators were embedded in the DFT based noise reduction system in Fig. 4.11 at a sampling frequency of $f_s = 20$kHz using half overlapping Hann windowed frames of size 512. Both noise power spectral density $\sigma^2_{N_i}$ and variance of speech $\sigma^2_{S_i}$ were estimated separately for each channel. To instrumentally evaluate the performance, (4.53) and (4.54) were averaged over the microphones. Figure 4.18 plots speech quality and noise reduction amount of the super-Gaussian M-dimensional MAP (Md-MAP) according to (4.52) and the Md-MAP with Gaussian model according to (4.48) compared to the Ephraim-Malah rule for speech mixed with cafeteria noise recorded with a linear microphone array of $M = 4$ elements and an interelement spacing of $d = 12$cm.

By applying the joint estimation over all microphones, the speech enhancement performance can significantly be increased in the difficult cafeteria. It is also important to note, that this performance gain is achieved independently of the direction of arrival of the desired source relative to the microphone array [18]. This is due to the conditioning on the noisy spectral amplitudes in the a posteriori PDF $p(A|R_1 \ldots R_M)$. The DOA information is mainly included in the spectral phases, not in the amplitudes.

Similar to the single-microphone speech enhancement system, including the super-Gaussian model for the speech amplitude also improves the quality of the multi-microphone system.

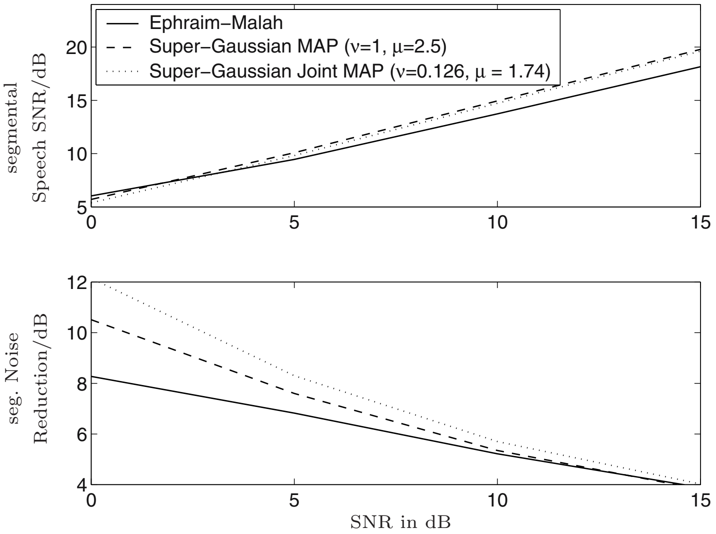


Fig. 4.16. Speech quality and noise reduction amount of statistical filter with Ephraim-Malah estimator (solid) with super-Gaussian MAP estimator (dashed) and super-Gaussian joint MAP estimator (dotted) for speech corrupted with fan noise.

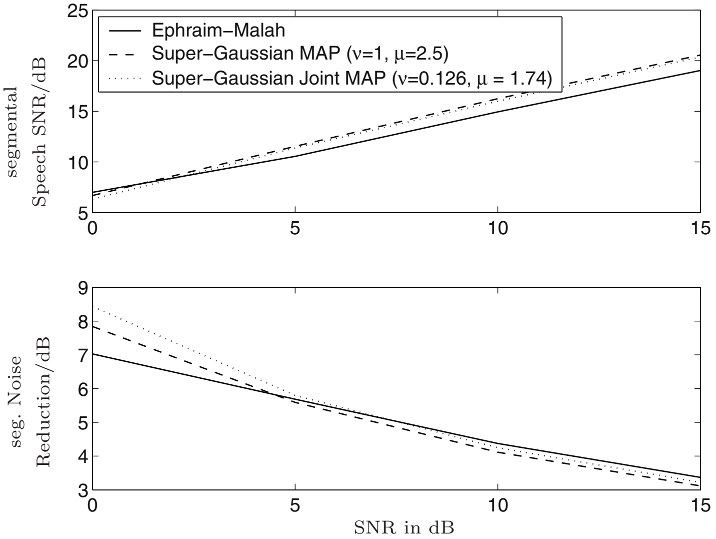


Fig. 4.17. Speech quality and noise reduction amount of statistical filter with Ephraim-Malah estimator (solid) with super-Gaussian MAP estimator (dashed) and super-Gaussian joint MAP estimator (dotted) for speech corrupted with cafeteria noise.

4.5 Conclusions

We have presented MAP spectral amplitude estimators for single- and multi-microphone speech enhancement applications.

For single-microphone noise reduction, efficient MAP estimators with a super-Gaussian speech model have been introduced. The underlying speech model can be tuned in advance to precisely match the distribution in a given system. The application of the MAP estimators improves the quality of the enhanced signal compared to the use of spectral amplitude estimators with common Gaussian speech models while decreasing the computational load.

For noise reduction with multiple microphones, joint MAP spectral amplitude estimators with Gaussian or super-Gaussian speech model have been derived. The joint estimation increases the quality of the enhanced signal in difficult cafeteria situations without introducing a dependency on the position of the desired source relative to the microphone.

References

1. S. F. Boll, “Suppression of Acoustic Noise in Speech Using Spectral Subtraction,” *IEEE Trans. Acoust., Speech and Signal Processing*, vol. 27, pp. 113–120, 1979.
2. Y. Ephraim and D. Malah, “Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator,” *IEEE Trans. Acoust., Speech and Signal Processing*, vol. 32, pp. 1109–1121, 1984.
3. R. Martin, “Speech enhancement using MMSE short time spectral estimation with gamma distributed priors,” in *Proc. IEEE ICASSP*, 2002, pp. 253–256.
4. P. J. Wolfe and S. J. Godsill, “Eficient Alternatives to the Ephraim-Malah Suppression Rule for Audio Signal Enhancement,” *EURASIP Journal on Applied Signal Processing, Special Issue: Digital Audio for Multimedia Communications*, vol. 2003, no. 11, pp. 1043–1051, 2003.
5. R. Martin, “Noise power spectral density estimation based on optimal smoothing and minimum statistics,” *IEEE Trans. Speech and Audio Processing*, vol. 9, pp. 504–512, 2001.
6. I. Cohen, “On the decision-directed estimation approach of Ephraim and Malah,” in *Proc. IEEE ICASSP*, 2004.
7. R. J. McAulay and M. L. Malpass, “Speech enhancement using a soft-decision noise suppression filter,” *IEEE Trans. Acoust., Speech and Signal Processing*, pp. 137–145, Apr. 1980.
8. D. R. Brillinger, *Time Series, Data Analysis and Theory*. McGraw-Hill, 1981.
9. H. Brehm and W. Stammler, “Description and Generation of Spherically Invariant Speech-Model Signals,” *Elsevier Signal Processing*, vol. 12, pp. 119–141, 1987.
10. R. Martin and C. Breithaupt, “Speech Enhancement in the DFT Domain using Laplacian Speech Priors,” in *Proc. IWAENC*, 2003, pp. 87–90.

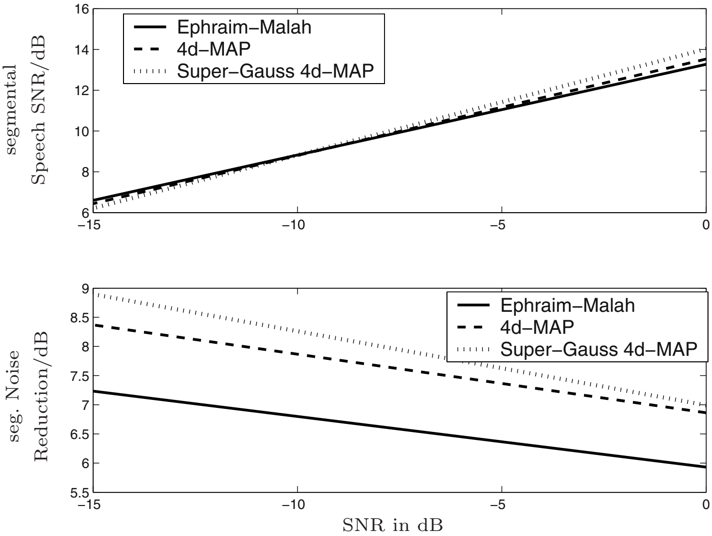


Fig. 4.18. Speech quality and noise reduction of 1d-MMSE estimator (reference) Md-MAP and super-Gaussian Md-MAP for $M = 4$ for noisy signals containing identical speech and cafeteria noise.

11. T. Lotter, *Single and Multichannel Speech Enhancement for Hearing Aids.* Ph.D. Dissertation, Aachener Beiträge zu Digitalen Nachrichtensystemen (ed. P. Vary), RWTH Aachen University, 2004.
12. P. Vary, “Noise suppression by spectral magnitude estimation – Mechanisms and theoretical limits,” *Signal Processing*, vol. 8, pp. 387–400, 1985.
13. D. L. Wang and J. S. Jim, “The unimportance of phase in speech enhancement,” *IEEE Trans. Acoust., Speech and Signal Processing*, vol. ASSP-30, pp. 679–681, 1982.
14. I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products.* Academic Press, Inc., 1994.
15. S. Kullback, *Information Theory and Statistics.* Dover Publication, 1968.
16. J. L. Melsa and D. L. Cohn, *Decision and Estimation Theory.* McGraw-Hill, 1978.
17. Y. Ephraim and D. Malah, “Speech enhancement using a minimum mean-square error log-spectral amplitude estimator,” *IEEE Trans. Acoust., Speech and Signal Processing*, vol. 33, pp. 443–445, 1985.
18. T. Lotter, C. Benien, and P. Vary, “Multichannel direction-independent speech enhancement using spectral amplitude Estimation,” *EURASIP Journal on Applied Signal Processing, Special Issue: Signal Processing for Acoustic Communication Systems*, vol. 2003, no. 11, pp. 1147–1157, 2003.
19. G. W. Elko, “Spatial coherence functions for differential microphone in isotropic noise fields,” in *Microphone Arrays*, edited by M. Brandstein and D. B. Ward, Springer-Verlag, pp. 61–86, 2001.

20. D. Malah, R. V. Cox, and A. J. Accardi, "Tracking speech presence uncertainty to improve speech enhancement in non-stationary noise environments," in *Proc. IEEE ICASSP*, 1999.

5 From Volatility Modeling of Financial Time-Series to Stochastic Modeling and Enhancement of Speech Signals

Israel Cohen

Technion – Israel Institute of Technology
Haifa 32000, Israel
E-mail: icohen@ee.technion.ac.il

Abstract. Modeling speech signals in the short-time Fourier transform (STFT) domain is a fundamental problem in designing speech enhancement systems. This chapter introduces a novel modeling approach, which is based on *generalized autoregressive conditional heteroscedasticity* (GARCH). GARCH is widely-used for volatility modeling of financial time-series such as exchange rates and stock returns. GARCH models take into account the heavy tailed distribution and volatility clustering characteristics of financial time-series. Spectral analysis shows that speech signals in the STFT domain are also characterized by heavy tailed distributions and volatility clustering. We demonstrate the application of GARCH modeling to speech enhancement, and show its advantage compared to using the conventional decision-directed method.

5.1 Introduction

Speech modeling in the short-time Fourier transform (STFT) domain underlies the design of many speech enhancement systems [1]. Ephraim and Malah [2] proposed to model the individual STFT expansion coefficients of the speech signal as zero-mean statistically independent Gaussian random variables. This model is motivated by the central limit theorem, as each expansion coefficient is a weighted sum of random variables resulting from the random sequence of speech samples. It facilitates a mathematically tractable design of useful speech enhancement algorithms in the STFT domain, *e.g.* [2–8]. However, the Gaussian approximation can be very inaccurate in the tail regions of the probability density function [9–12]. Therefore, Martin [10] proposed to model the real and imaginary parts of the expansion coefficients as either Gamma or Laplacian random variables. He showed that a minimum mean-squared error (MMSE) estimator under a Gamma model yields higher improvement in the segmental signal-to-noise ratio (SNR) than an MMSE estimator under a Gaussian model. Furthermore, MMSE estimators under Laplacian speech modeling have similar properties to those estimators derived under Gamma modeling, but are easier to compute and implement [13].

Lotter et al. [7] proposed a parametric probability density function (pdf) for the magnitude of the expansion coefficients, which approximates, with a proper choice of the parameters, the Gamma and Laplacian densities. They derived a maximum a-posteriori (MAP) estimator for the speech spectral amplitude, and showed that under Laplacian speech modeling the MAP estimator demonstrates improved noise reduction compared with the short-term spectral amplitude (STSA) estimator of Ephraim-Malah.

The variances of the speech spectral coefficients are generally referred to as the model parameters, which have to be estimated from the noisy observed signal. Ephraim and Malah [2], [14] proposed three different methods for the variance estimation:

1. **Maximum-likelihood estimation.**
   This method relies on the assumption that the variances are slowly time-varying parameters. It results in musical residual noise, which is annoying and disturbing to the perception of the enhanced signal.
2. **Decision-directed estimation.**
   This method is particularly useful when combined with the MMSE spectral, or log-spectral, magnitude estimators [2], [3], [15]. It results in perceptually colorless residual noise, but is heuristically motivated and its theoretical performance is unknown due to its highly nonlinear nature.
3. **Maximum a-posteriori estimation.**
   This method relies on a first-order Markov model for generating a sequence of speech spectral variances. It involves a set of nonlinear equations, which are solved recursively by using the Viterbi algorithm. The computational complexity of the MAP estimator is relatively high, while it does not provide a significant improvement in the enhanced speech quality over the decision-directed estimator [14].

Consequently, the decision-directed approach has become the most acceptable estimation method for the variances of the speech spectral coefficients. However, the parameters of the decision-directed estimator have to be determined by simulations and subjective listening tests for each particular setup of time-frequency transformation and speech enhancement algorithm. Furthermore, since the decision-directed approach is not supported by a statistical model, the parameters are not adapted to the speech components, but are set to specific values in advance. Ephraim and Malah recognized the limits of their variance estimation methods, and concluded that better speech enhancement performance may be obtained if the variance estimation could be improved [2], [14].

Statistical models based on hidden Markov processes (HMPs) try to circumvent the assumption of specific distributions [16], [17]. The probability distributions of the speech and noise processes are estimated from long training sequences of the speech and noise samples, and then used jointly with a given fidelity criterion to derive an estimator for the speech signal. Unfortunately, the HMP-based speech enhancement relies on the type of training data [18]. It works best with the trained type of noise, but often worse with

Table 5.1 Speech enhancement algorithms.

Algorithm #	Variance Estimation	Fidelity Criterion
1	GARCH	MMSE
2	Decision-Directed	MMSE
3	GARCH	MMSE-LSA
4	Decision-Directed	MMSE-LSA

other type of noise. Furthermore, improved performance generally entails more complex models and higher computational requirements.

This chapter introduces a novel modeling approach for speech signals in the STFT domain [19]. The approach is based on generalized autoregressive conditional heteroscedasticity (GARCH) modeling, which is widely-used for modeling the volatility of financial time-series such as exchange rates and stock returns [20], [21]. We consider four different speech enhancement algorithms, as summarized in Table 5.1. The spectral variance is estimated based on either the proposed GARCH model or the decision-directed method of Ephraim and Malah [2], while the fidelity criterion is either MMSE of the STFT coefficients or MMSE of the log-spectral amplitude (LSA). We show that estimating the variance by using the GARCH modeling method yields lower log-spectral distortion (LSD) and higher perceptual evaluation of speech quality (PESQ) scores (ITU-T P.862) than by using the decision-directed method. Speech spectrograms and informal listening tests confirm that the quality of the enhanced speech obtained by using the GARCH modeling method is better than that obtainable by using the decision-directed method.

The chapter is organized as follows. In Section 5.2, we formulate the problems and objectives. In Section 5.3, we investigate the time-frequency correlation of spectral coefficients. In Section 5.4, we introduce the GARCH model in the STFT domain. In Section 5.5, we address the model estimation problem. Finally, in Section 5.6, we evaluate the performances of MMSE spectral and MMSE-LSA estimators and compare the GARCH modeling method to the decision-directed method.

5.2 Problem Formulation

Let x and d denote speech and uncorrelated additive noise signals, and let $y = x + d$ represent the observed signal. Applying the STFT to the observed signal, we have in the time-frequency domain

$$Y_{tk} = X_{tk} + D_{tk}, \tag{5.1}$$

where t is the time frame index ($t = 0, 1, \ldots$) and k is the frequency-bin index ($k = 0, 1, \ldots, K - 1$). The spectral enhancement problem is generally formulated as deriving an estimator $\hat{X}_{tk}$ for the speech spectral coefficients, such that the expected value of a certain distortion measure is minimized. Let $d\left(X_{tk}, \hat{X}_{tk}\right)$ denote a distortion measure between X_{tk} and its estimate $\hat{X}_{tk}$, and let ψ_t represents the information set that can be employed for the estimation at frame t (*e.g.*, the noisy data observed through time t). Let H_0^{tk} and H_1^{tk} denote, respectively, hypotheses of signal absence and presence in the noisy spectral coefficient Y_{tk}, let $\hat{p}_{tk} = P\left(H_1^{tk} \mid \psi_t\right)$ denote an estimate for the signal presence probability, and let $\hat{\lambda}_{tk} = E\left\{|X_{tk}|^2 \mid H_1^{tk}, \psi_t\right\}$ denote an estimate for the variance of a speech spectral coefficient X_{tk} under H_1^{tk}. Then, we consider an estimator for X_{tk} which minimizes the expected distortion given $\hat{p}_{tk}$, $\hat{\lambda}_{tk}$ and the noisy spectral coefficient Y_{tk}:

$$\min_{\hat{X}_{tk}} E\left\{d\left(X_{tk}, \hat{X}_{tk}\right) \,\middle|\, \hat{p}_{tk}\,, \hat{\lambda}_{tk}\,, Y_{tk}\right\} . \tag{5.2}$$

In particular, restricting ourselves to a squared error distortion measure of the form

$$d\left(X_{tk}, \hat{X}_{tk}\right) = \left|g(\hat{X}_{tk}) - \tilde{g}(X_{tk})\right|^2 , \tag{5.3}$$

where $g(X)$ and $\tilde{g}(X)$ are specific functions of X (*e.g.*, X, $|X|$, $\log|X|$, $e^{j\angle X}$), the estimator $\hat{X}_{tk}$ is calculated from

$$\begin{aligned} g(\hat{X}_{tk}) &= E\left\{\tilde{g}(X_{tk}) \,\middle|\, \hat{p}_{tk}\,, \hat{\lambda}_{tk}\,, Y_{tk}\right\} \qquad (5.4)\\ &= \hat{p}_{tk}\, E\left\{\tilde{g}(X_{tk}) \,\middle|\, H_1^{tk}\,, \hat{\lambda}_{tk}\,, Y_{tk}\right\} + (1-\hat{p}_{tk})\, E\left\{\tilde{g}(X_{tk}) \,\middle|\, H_0^{tk}\,, Y_{tk}\right\} . \end{aligned}$$

The design of a particular estimator for X_{tk} requires the following specifications:

- Functions $g(X)$ and $\tilde{g}(X)$, which determine the fidelity criterion of the estimator.
- A conditional pdf $p\left(X_{tk} \mid \lambda_{tk}\,, H_1^{tk}\right)$ for X_{tk} under H_1^{tk} given its variance λ_{tk}, which determines the statistical model.
- Estimators $\hat{\lambda}_{tk}$ and $\widehat{\sigma_{tk}^2}$ for the speech and noise spectral variances, respectively.
- An estimator $\hat{q}_{tk} = P\left(H_1^{tk} \mid \psi_t'\right)$ for the *a priori* signal presence probability, where $\psi_t' = \psi_t \setminus Y_{tk}$ represents the information set known prior to having the measurement Y_{tk}.

In this chapter, we consider MMSE-spectral and MMSE-LSA fidelity criteria under a Gaussian model, while the speech spectral variance is estimated based on either GARCH modeling or the decision-directed method. Given an estimate $\hat{q}_{tk}$ for the *a priori* signal presence probability, the (*a posteriori*) signal presence probability can be obtained from Bayes' rule:

$$\hat{p}_{tk} = \frac{\hat{q}_{tk}\, P\left(Y_{tk} \mid H_1^{tk}, \psi_t'\right)}{\hat{q}_{tk}\, P\left(Y_{tk} \mid H_1^{tk}, \psi_t'\right) + (1-\hat{q}_{tk}) P\left(Y_{tk} \mid H_0^{tk}, \psi_t'\right)} . \tag{5.5}$$

However, to simplify the comparisons between the speech enhancement algorithms, we focus on implementations that assume speech presence (*i.e.*, $\hat{p}_{tk} = 1$) whenever $20 \log_{10} |X_{tk}| > \epsilon$, where $\epsilon = \max_{tk} \{20 \log_{10} |X_{tk}|\} - 50$ confines the dynamic range of the log-spectrum to 50 dB. In the other time-frequency bins, $\hat{p}_{tk}$ is set to zero. Furthermore, we assume knowledge of the noise variance $\sigma_{tk}^2 \triangleq E\left\{|D_{tk}|^2\right\}$, which in practice can be estimated by using the *minima controlled recursive averaging* approach [22]. Our objectives are as follows:

- Analyze the time-frequency correlation of speech and noise signals in the STFT domain.
- Formulate a statistical model for speech signals in the STFT domain, which takes into consideration the time-frequency correlation and heavy-tailed distribution of the expansion coefficients.
- Evaluate the performances of MMSE-spectral and MMSE-LSA estimators under a Gaussian model, while estimating the speech spectral variance by using the proposed modeling or the decision-directed method.

5.3 Spectral Analysis

To see graphically the relation between successive spectral components of a speech signal, in comparison with a noise signal, we have investigated the sample autocorrelation coefficient sequences (ACSs) of the STFT coefficients along time-trajectories (the frequency-bin index k is held fixed) [23]. We consider a speech signal that is constructed from six different utterances without intervening pauses, and present scatter plots for successive spectral magnitudes [23]. The utterances, half from male speakers and half from female speakers, are taken from the TIMIT database [24]. The speech signal is sampled at 16 kHz, and transformed into the STFT domain using Hamming analysis windows of $N = 512$ samples (32 ms) length, and $M = 256$ samples framing step (50% overlap between successive frames).

Figure 5.1 shows an example of scatter plots for successive spectral magnitudes of white Gaussian noise (WGN) and speech signals. It implies that 50% overlap between successive frames does not yield a significant correlation between the spectral magnitudes of the WGN signal. However, successive spectral magnitudes of the speech signal are highly correlated. Figure 5.2 shows the ACSs of the speech spectral components along time-trajectories, for various frequency-bins and framing steps. The 95 percent confidence limits (*e.g.*, [25]) are depicted as horizontal dotted lines. In order to prevent an upward bias of the autocovariance estimates due to irrelevant (non-speech) spectral components, the ACSs are computed from spectral components whose magnitudes are within 30 dB of the maximal magnitude. Specifically, the sample

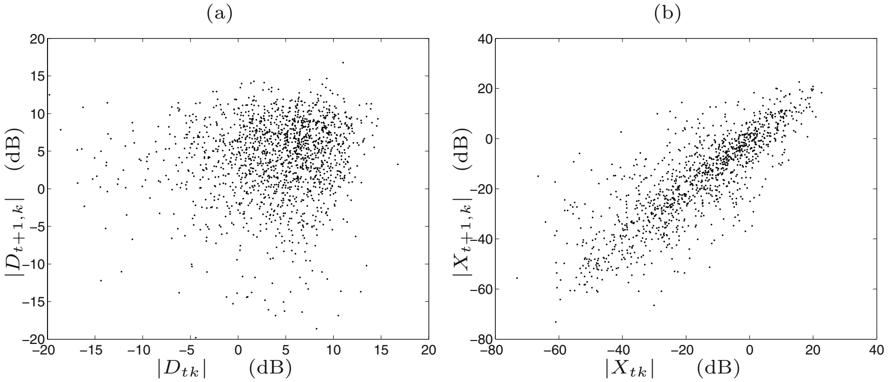


Fig. 5.1. Scatter plots for successive spectral magnitudes of (a) a white Gaussian noise signal, and (b) a speech signal at center frequency 500 Hz ($k = 17$). The overlap between successive frames is 50%.

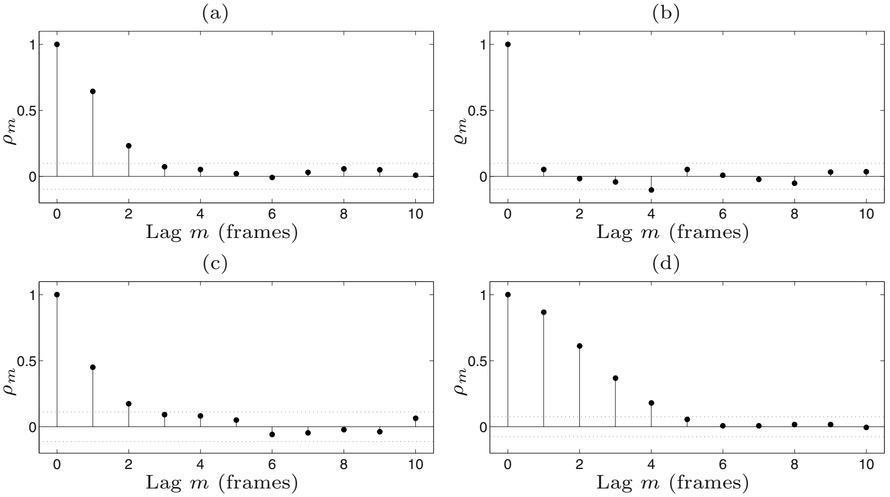


Fig. 5.2. Sample autocorrelation coefficient sequences (ACSs) of clean speech STFT coefficients along time-trajectories, for various frequency-bins and framing steps. The dotted lines represents 95 percent confidence limits. (a) ACS of the spectral magnitude at frequency-bin $k = 17$ (center frequency 500 Hz), framing step $M = N/2$ (50% overlap between frames). (b) ACS of the spectral phase, $k = 17$, $M = N/2$. (c) ACS of the spectral magnitude, $k = 65$ (center frequency 2 kHz), $M = N/2$. (d) ACS of the spectral magnitude, $k = 17$, $M = N/4$ (75% overlap between frames).

autocorrelation coefficients of the spectral magnitudes are calculated by [23]

$$\rho_m = \frac{\sum_{t\in\mathcal{T}} \left[A_{tk} - \bar{A}_k\right] \left[A_{t+m,k} - \bar{A}_k\right]}{\sum_{t\in\mathcal{T}} \left[A_{tk} - \bar{A}_k\right]^2}, \tag{5.6}$$

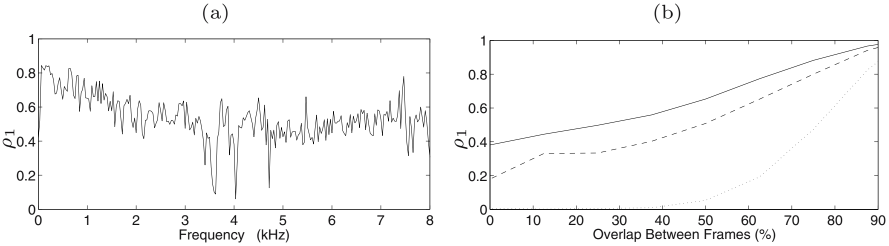


Fig. 5.3. Variation of the correlation coefficient between successive spectral magnitudes. (a) Typical variation of ρ_1 on frequency for a speech signal and 50% overlap between frames. (b) Typical variation of ρ_1 on overlap between frames for a speech signal at center frequencies 1 kHz (solid line) and 2 kHz (dashed line), and for a realization of white Gaussian noise (dotted line).

where $A_{tk} \triangleq |X_{tk}|$ denotes the magnitude of X_{tk}, m is the lag in frames, $\bar{A}_k$ is the sample mean given by

$$\bar{A}_k = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} A_{tk} \, ,$$

and $\mathcal{T}$ represents the set of relevant spectral components

$$\mathcal{T} = \left\{ t \;\middle|\; A_{tk} \geq 10^{-30/20} \max_t \{A_{tk}\} \right\} .$$

The corresponding sample autocorrelation coefficients of the spectral phases are obtained by

$$\varrho_m = \frac{\sum_{t \in \mathcal{T}} \varphi_{tk} \, \varphi_{t+m,k}}{\sum_{t \in \mathcal{T}} \varphi_{tk}^2} \, , \tag{5.7}$$

where φ_{tk} denotes the phase of X_{tk}. Figure 5.3 shows the variation of the correlation between successive spectral magnitudes on frequency and on overlap between successive frames. Figures 5.2 and 5.3 demonstrate that for speech signals, successive spectral magnitudes are highly correlated, while the correlation is generally larger at lower frequencies, and it increases as the overlap between successive frames increases. As a comparison, the variation of ρ_1 on the overlap between frames is shown also for a realization of WGN (see Figure 5.3(b), dotted line). It implies that for a sufficiently large framing step ($M \geq N/2$, *i.e.*, overlap between frames $\leq$ 50%), successive spectral components of the *noise* signal can be assumed uncorrelated.

In view of the above discussion, we may conclude that speech signals in the STFT domain are characterized by volatility clustering. When observing a time series of successive expansion coefficients in a fixed frequency bin, successive magnitudes of the expansion coefficients are highly correlated, whereas successive phases are nearly uncorrelated. Hence, the expansion coefficients are clustered in the sense that large magnitudes tend to follow large magnitudes and small magnitudes tend to follow small magnitudes, while the phase is unpredictable. Therefore, modeling the time-trajectories of the expansion coefficients as GARCH processes offers a reasonable model on which to base the variance estimation, while taking into consideration the heavy-tailed distribution [19].

5.4 Statistical Model for Speech Signals

The variances of the speech coefficients are hidden from direct observation, in the sense that even under perfect conditions of zero noise ($D_{tk} = 0$ for all tk), the values of $\{\lambda_{tk}\}$ are not directly observable. Therefore, our approach is to assume that $\{\lambda_{tk}\}$ themselves are random variables, and to introduce *conditional* variances which are estimated from the available information (*e.g.*, the clean spectral coefficients through frame $t-1$, or the noisy spectral coefficients through frame t) [19]. Let $\mathcal{X}_0^\tau = \{X_{tk} \,|\, t = 0, \dots, \tau,\ k = 0, \dots, K-1\}$ represent the set of clean speech spectral coefficients up to frame τ, and let $\lambda_{tk|\tau} \triangleq E\left\{|X_{tk}|^2 \,|\, H_1^{tk}, \mathcal{X}_0^\tau\right\}$ denote the *conditional* variance of X_{tk} under H_1^{tk} given the clean spectral coefficients up to frame τ. Then, our statistical model in the STFT domain relies on the following set of assumptions [19]:

1. Given $\{\lambda_{tk}\}$ and the state of speech presence in each time-frequency bin (H_1^{tk} or H_0^{tk}), the speech spectral coefficients $\{X_{tk}\}$ are generated by

$$X_{tk} = \sqrt{\lambda_{tk}}\, V_{tk}, \tag{5.8}$$

where $\left\{V_{tk} \,|\, H_0^{tk}\right\}$ are identically zero, and $\left\{V_{tk} \,|\, H_1^{tk}\right\}$ are statistically independent complex random variables with zero mean, unit variance, and independent and identically distributed (iid) real and imaginary parts:

$$\begin{aligned} &H_1^{tk} : E\{V_{tk}\} = 0\,,\ E\left\{|V_{tk}|^2\right\} = 1\,, \\ &H_0^{tk} : V_{tk} = 0\,. \end{aligned} \tag{5.9}$$

2. Let $V_{Rtk} = \Re\{V_{tk}\}$ and $V_{Itk} = \Im\{V_{tk}\}$ denote, respectively, the real and imaginary parts of V_{tk}. Let $p\left(V_{\rho tk} \,|\, H_1^{tk}\right)$ denote the pdf of $V_{\rho tk}$ ($\rho \in \{R, I\}$) under H_1^{tk}. Then,

$$p\left(V_{\rho tk} \mid H_1^{tk}\right) = \frac{1}{\sqrt{\pi}} \exp\left(-V_{\rho tk}^2\right)\,. \tag{5.10}$$

3. The conditional variance $\lambda_{tk|t-1}$, referred to as the *one-frame-ahead conditional variance*, is a random process which evolves as a GARCH$(1,1)$ process:

$$\lambda_{tk|t-1} = \lambda_{\min} + \mu\, |X_{t-1,k}|^2 + \delta\left(\lambda_{t-1,k|t-2} - \lambda_{\min}\right), \tag{5.11}$$

where

$$\lambda_{\min} > 0\,, \quad \mu \geq 0\,, \quad \delta \geq 0\,, \quad \mu + \delta < 1, \tag{5.12}$$

are the standard constraints imposed on the parameters of the GARCH model [21]. The parameters μ and δ are, respectively, the moving average and autoregressive parameters of the GARCH(1,1) model, and $\lambda_{\min}$ is a lower bound on the variance of X_{tk} under H_1^{tk}.

4. The noise spectral coefficients $\{D_{tk}\}$ are zero-mean statistically independent Gaussian random variables. The real and imaginary parts of D_{tk} are iid random variables $\sim \mathcal{N}\left(0, \frac{\sigma_{tk}^2}{2}\right)$.

The first assumption implies that the speech spectral coefficients $\left\{X_{tk} \mid H_1^{tk}\right\}$ are conditionally zero-mean statistically independent random variables given their variances $\{\lambda_{tk}\}$. The real and imaginary parts of X_t under H_1^t are conditionally iid random variables given λ_{tk}, satisfying

$$p\left(X_{\rho tk} \mid \lambda_{tk}\,,\, H_1^{tk}\right) = \frac{1}{\sqrt{\lambda_{tk}}} p\left(V_{\rho tk} = \frac{X_{\rho tk}}{\sqrt{\lambda_{tk}}} \,\middle|\, H_1^{tk}\right)\,, \quad \rho \in \{R, I\}\ . \tag{5.13}$$

5.5 Model Estimation

The maximum-likelihood (ML) estimation approach is commonly used for estimating the parameters of a GARCH model [20]. In this section, we present the ML function of the model parameters, by using the spectral coefficients of the clean speech signal on some interval $t \in [0, T]$ [19]. For simplicity, we assume that the parameters are constant during the above interval and are independent of the frequency-bin index k. As noted in [19], the speech signal can be divided in practice into short time segments and split in frequency into narrow subbands, such that the parameters can be assumed to be constant in each time-frequency region. Furthermore, we generally do not have a direct access to the clean spectral coefficients. However, the expectation-maximization (EM) algorithm [26], [27] can be utilized for solving this problem by iteratively estimating the clean spectral coefficients and the model parameters from the noisy measurements.

Let $\mathcal{X}_0^T$ denote the set of clean speech spectral coefficients employed for the model estimation, let $\mathcal{H}_1$ denote the set of time-frequency bins where the signal is present, and let $\boldsymbol{\phi} = \left[\mu\ \delta\ \lambda_{\min}\right]$ denote the vector of unknown parameters. Then, the conditional variance $\lambda_{tk|t-1}$ can recursively be calculated from past spectral coefficients $\mathcal{X}_0^{t-1}$ by using (5.11) and the parameter vector $\boldsymbol{\phi}$. Hence, the logarithm of the conditional density of X_{tk} given the clean spectral coefficients up to frame $t-1$ can be expressed as [19]

$$\log p\left(X_{tk} \;\middle|\; \mathcal{X}_0^{t-1}\,; \boldsymbol{\phi}\right) = -\frac{|X_{tk}|^2}{\lambda_{tk|t-1}} - \log \lambda_{tk|t-1} - \log \pi\,, \tag{5.14}$$

where $tk \in \mathcal{H}_1$. For sufficiently large sample size, the spectral coefficients of the first frame make a negligible contribution to the total likelihood. Therefore, the values of $\lambda_{0,k|-1}$ in the first frame can be initialized to their minimal value $\lambda_{\min}$, and the log-likelihood can be maximized when conditioned on the first frame. The log-likelihood conditional on the spectral coefficients of the first frame is given by

$$\mathcal{L}(\boldsymbol{\phi}) = \sum_{tk \in \mathcal{H}_1 \cap t \in [1,T]} \log p\left(X_{tk} \;\middle|\; H_1^{tk}, \mathcal{X}_0^{t-1}; \boldsymbol{\phi}\right) . \tag{5.15}$$

Substituting (5.14) into (5.15) and imposing the constraints in (5.12) on the estimated parameters, the maximum-likelihood estimates of the model parameters can be obtained by solving the following constrained nonlinear minimization problem [19]:

$$\underset{\hat{\lambda}_{\min},\, \hat{\mu},\, \hat{\delta}}{\text{minimize}} \sum_{tk \in \mathcal{H}_1 \cap t \in [1,T]} \left[\frac{|X_{tk}|^2}{\lambda_{tk|t-1}} + \log \lambda_{tk|t-1} \right] , \tag{5.16}$$

subject to

$$\hat{\lambda}_{\min} > 0 \, , \; \hat{\mu} \geq 0 \, , \; \hat{\delta} \geq 0 \, , \; \hat{\mu} + \hat{\delta} < 1 \, . \tag{5.17}$$

For given numerical values of the parameters, the sequences of conditional variances $\left\{\lambda_{tk|t-1}\right\}$ can be calculated from (5.11) and used to evaluate the series in (5.16). The result can then be minimized numerically by using the Berndt, Hall, Hall, and Hausman [28] algorithm as in Bollerslev [29].

5.6 Experimental Results

In this section, we demonstrate the performances of MMSE spectral and LSA estimators (see Appendices), while the speech variance is estimated by using either the GARCH modeling or the decision-directed method. The evaluation includes two objective quality measures, and informal listening tests. The first quality measure is log-spectral distortion, in dB, which is defined by

$$\text{LSD} = \left[\frac{1}{|\mathcal{H}_1|} \sum_{tk \in \mathcal{H}_1} \left(20 \, \log_{10} |X_{tk}| - 20 \, \log_{10} |\hat{X}_{tk}| \right)^2 \right]^{\frac{1}{2}} , \tag{5.18}$$

where $\mathcal{H}_1 = \{tk \mid 20 \, \log_{10} |X_{tk}| > \epsilon\}$ denotes the set of time-frequency bins which contain the speech signal, $|\mathcal{H}_1|$ denotes its cardinality, and $\epsilon = \max_{tk} \{20 \, \log_{10} |X_{tk}|\} - 50$ confines the dynamic range of the log-spectrum to 50 dB. The second quality measure is the PESQ score (ITU-T P.862).

The speech signals used in our evaluation are taken from the TIMIT database [24]. They include 20 different utterances from 20 different speakers, half male and half female. The speech signals are sampled at 16 kHz and degraded by white Gaussian noise with SNRs in the range $[0, 20]$ dB. The noisy signals are transformed into the STFT domain using half overlapping Hamming analysis windows of 32 milliseconds length. The GARCH model (*i.e.*, the parameters μ, δ and $\lambda_{\min}$) is estimated independently for each speaker from the clean signal of that speaker, as described in Section 5.5. The variance of an expansion coefficient is recursively estimated by iterating propagation and update steps following the rational of Kalman filtering as described in [30]. Four different speech enhancement algorithms are then applied to each noisy speech signal, as summarized in Table 5.1. The speech variance is estimated by using either the GARCH modeling method or the decision-directed method, and the fidelity criterion is either MMSE of the spectral coefficients or MMSE of the log-spectral amplitude. The decision-directed estimate of the speech variance is given by [2], [15]

$$\hat{\lambda}_{tk}^{\mathrm{DD}} = \max\left\{\alpha\,|\hat{X}_{t-1,k}|^2 + (1-\alpha)\left(|Y_{tk}|^2 - \sigma_{tk}^2\right)\,,\, \xi_{\min}\,\sigma_{tk}^2\right\}\,, \tag{5.19}$$

where α $(0 \leq \alpha \leq 1)$ is a weighting factor that controls the trade-off between noise reduction and transient distortion introduced into the signal, and $\xi_{\min}$ is a lower bound on the *a priori* SNR. These parameters are set to the values $\xi_{\min} = -15$ dB and $\alpha = 0.98$ as specified in [2], [3], [15]. The noise spectral variance σ_{tk}^2 is estimated by averaging over time the spectral power values of the noise signal itself. Speech presence is determined (*i.e.*, $\hat{p}_{tk} = 1$) whenever $20\,\log_{10}|X_{tk}| > \epsilon$. The attenuation factor $G_{\min}$ during speech absence is -20 dB.

Table 5.2 shows the results of the LSD obtained by using the different algorithms for various SNR levels. The results of the PESQ scores are presented in Table 5.3. The results show that the MMSE-LSA estimator yields lower LSD and higher PESQ scores than the MMSE spectral estimator, whether the variance is estimated by using the GARCH modeling method or the decision-directed method. Furthermore, speech variance estimation based on GARCH modeling yields lower LSD and higher PESQ scores than those obtained by using the decision-directed method, whether the fidelity criterion is MMSE of the spectral coefficients or MMSE-LSA.

A subjective study of speech spectrograms and informal listening tests confirm that the quality of the enhanced speech obtained by using the GARCH modeling method is significantly better than that obtainable by using the decision-directed method. Figure 5.4 demonstrates the spectrograms and waveforms of a clean signal, noisy signal (SNR = 5 dB) and enhanced speech signals obtained by using the GARCH modeling and the decision-directed methods. It shows that weak speech components and unvoiced sounds are significantly more emphasized in the signal enhanced by using the GARCH modeling method than in the signal enhanced by using the decision-directed method.

Table 5.2 Log-spectral distortion obtained by using different variance estimation methods (GARCH modeling method vs. decision-directed method) and fidelity criteria (MMSE vs. MMSE-LSA).

Input SNR	GARCH modeling method		Decision-Directed method	
[dB]	MMSE	MMSE-LSA	MMSE	MMSE-LSA
0	7.77	**4.85**	18.89	11.35
5	5.78	**4.04**	17.29	11.03
10	4.14	**3.27**	13.87	9.13
15	2.50	**2.25**	9.19	6.05
20	1.30	**1.28**	4.88	3.13

Table 5.3 PESQ scores obtained by using different variance estimation methods (GARCH modeling method vs. decision-directed method) and fidelity criteria (MMSE vs. MMSE-LSA).

Input SNR	GARCH modeling method		Decision-Directed method	
[dB]	MMSE	MMSE-LSA	MMSE	MMSE-LSA
0	2.52	**2.55**	1.91	2.21
5	2.97	**2.98**	2.30	2.61
10	3.37	**3.38**	2.70	2.99
15	3.67	**3.69**	3.09	3.31
20	3.88	**3.89**	3.53	3.64

5.7 Conclusions

In this chapter, we described a GARCH modeling approach for speech signals in the STFT domain. We assumed that the conditional variances of the STFT expansion coefficients are random variables, and that the *one-frame-ahead* conditional variance evolves as a GARCH$(1,1)$ process. We compared the performances of MMSE spectral and MMSE-LSA estimators, while the variance estimation is based on either the GARCH modeling approach or the decision-directed method of Ephraim and Malah. We showed that the MMSE-LSA estimator yields lower log-spectral distortion and higher PESQ scores than the MMSE spectral estimator, whether the variance is estimated by using the GARCH modeling method or the decision-directed method. However, speech variance estimation based on GARCH modeling yields lower LSD and higher PESQ scores than those obtained by using the decision-directed method, whether the fidelity criterion is MMSE of the spectral coefficients or MMSE-LSA.

Appendix A: MMSE Spectral Estimation

An MMSE estimator for X_{tk} is obtained by using the functions

$$g(\hat{X}_{tk}) = \hat{X}_{tk}\,, \quad \tilde{g}(X_{tk}) = \begin{cases} X_{tk}\,, & \text{under } H_1^{tk} \\ G_{\min} Y_{tk}\,, & \text{under } H_0^{tk} \end{cases}, \tag{5.20}$$

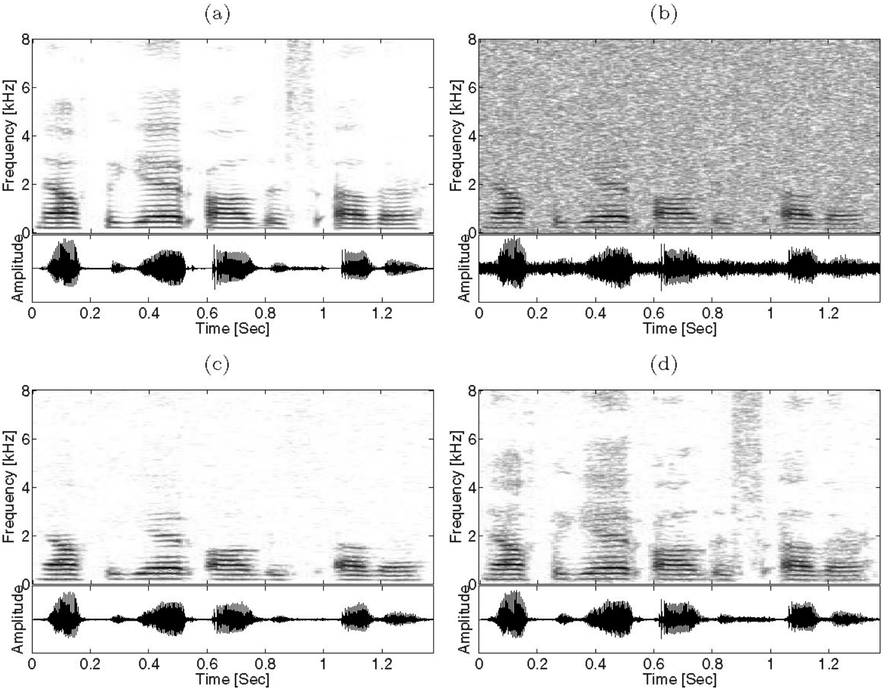


Fig. 5.4. Speech spectrograms and waveforms. (a) Original clean speech signal: "Now forget all this other." (b) Noisy signal (SNR = 5 dB, LSD = 13.75 dB, PESQ= 1.76). (c) Speech reconstructed by using the decision-directed method and MMSE-LSA estimator (LSD = 9.00 dB, PESQ = 2.57). (d) Speech reconstructed by using the GARCH modeling method and MMSE-LSA estimator (LSD = 3.59 dB, PESQ = 2.88).

where $G_{\min} \ll 1$ represents a constant attenuation factor. Substituting (5.20) into (5.5), we have

$$\begin{aligned}\hat{X}_{tk} = \hat{p}_{tk} \left[G_{\mathrm{MSE}} \left(\hat{\xi}_{tk|t}, \gamma_{Rtk} \right) Y_{Rtk} + j\, G_{\mathrm{MSE}} \left(\hat{\xi}_{tk|t}, \gamma_{Itk} \right) Y_{Itk} \right] \\ + (1 - \hat{p}_{tk})\, G_{\min}\, Y_{tk}\,, \end{aligned} \tag{5.21}$$

where $\hat{\xi}_{tk|t} = \hat{\lambda}_{tk|t} / \sigma^2_{tk}$ is an estimate for the *a priori* SNR, and $G_{\mathrm{MSE}}(\xi, \gamma_\rho)$ ($\rho \in \{R, I\}$) represents the MMSE spectral gain function under H_1. The specific expression for $G_{\mathrm{MSE}}(\xi, \gamma_\rho)$ depends on the particular statistical model. For a Gaussian model, the gain function is the Wiener filter given by [31]

$$G_{\mathrm{MSE}}(\xi) = \frac{\xi}{1 + \xi}\,. \tag{5.22}$$

When the signal is surely absent (*i.e.*, when $\hat{p}_{tk} = 0$), the resulting estimator $\hat{X}_{tk}$ reduces to a constant attenuation of Y_{tk} (*i.e.*, $\hat{X}_{tk} = G_{\min} Y_{tk}$). This retains the noise naturalness, and is closely related to the "spectral floor" proposed by Berouti *et al.* [32].

Appendix B: MMSE Log-Spectral Amplitude Estimation

In speech enhancement applications, estimators which minimize the mean-squared error of the log-spectral amplitude have been found advantageous to MMSE spectral estimators [2], [3], [9]. An MMSE-LSA estimator is obtained by substituting into (5.5) the functions

$$g(\hat{X}_{tk}) = \log|\hat{X}_{tk}|\,, \quad \tilde{g}(X_{tk}) = \begin{cases} \log|X_{tk}|\,, & \text{under } H_1^{tk} \\ \log\left(G_{\min}|Y_{tk}|\right)\,, & \text{under } H_0^{tk} \end{cases}. \tag{5.23}$$

Combing the resulting amplitude estimate with the phase of the noisy spectral coefficient Y_{tk} yields

$$\hat{X}_{tk} = \left[G_{\mathrm{LSA}}(\hat{\xi}_{tk|t}, \gamma_{tk})\right]^{\hat{p}_{tk}} G_{\min}^{1-\hat{p}_{tk}}\, Y_{tk}, \tag{5.24}$$

where $\gamma_{tk} = \gamma_{Rtk} + \gamma_{Itk}$ denotes *a posteriori* SNR,

$$G_{\mathrm{LSA}}\,(\xi,\,\gamma) \triangleq \frac{\xi}{1+\xi}\exp\left(\frac{1}{2}\int_{\vartheta}^{\infty}\frac{e^{-x}}{x}dx\right) \tag{5.25}$$

represents the LSA gain function under H_1^{tk} which was derived by Ephraim and Malah [3], and ϑ is defined by $\vartheta \triangleq \frac{\xi\,\gamma}{1+\xi}$. Similar to the MMSE spectral estimator, the MMSE-LSA estimator reduces to a constant attenuation of Y_{tk} when the signal is surely absent (*i.e.*, $\hat{p}_{tk} = 0$ implies $\hat{X}_{tk} = G_{\min}\,Y_{tk}$). However, for a fixed value of the *a priori* SNR, the LSA gain is a monotonically decreasing function of γ. This behavior of $G_{\mathrm{LSA}}\,(\xi,\,\gamma)$ is related to the useful mechanism that counters the musical noise phenomenon [15]. Local bursts of the *a posteriori* SNR, during noise-only frames, are "pulled down" to the average noise level, thus avoiding local buildup of noise whenever it exceeds its average characteristics. As a result, the MMSE-LSA estimator generally produces lower levels of residual musical noise, when compared with the MMSE spectral estimator.

Acronyms

ACS	autocorrelation coefficient sequence
EM	expectation-maximization
GARCH	generalized autoregressive conditional heteroscedasticity
HMP	hidden Markov process
iid	independent and identically distributed
LSA	log-spectral amplitude
LSD	log-spectral distortion
MAP	maximum a posteriori
ML	maximum-likelihood
MMSE	minimum mean-squared error
MSE	mean-squared error
PESQ	perceptual evaluation of speech quality
pdf	probability density function
SNR	signal-to-noise ratio
STFT	short-time Fourier transform
STSA	short-term spectral amplitude
WGN	white Gaussian noise

References

1. Y. Ephraim and I. Cohen, "Recent advancements in speech enhancement," in *The Electrical Engineering Handbook*, 3rd ed. CRC Press, to be published. [Online]. Available: http:// ece.gmu.edu/∼yephraim/ ephraim.html
2. Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-32, pp. 1109–1121, Dec. 1984.
3. ——, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-33, pp. 443–445, Apr. 1985.
4. A. J. Accardi and R. V. Cox, "A modular approach to speech enhancement with an application to speech coding," in *Proc. IEEE ICASSP*, 1999, pp. 201–204.
5. J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detector," *IEEE Signal Processing Letters*, vol. 6, pp. 1–3, Jan. 1999.
6. I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments," *Signal Processing*, vol. 81, pp. 2403–2418, Nov. 2001.
7. T. Lotter, C. Benien, and P. Vary, "Multichannel speech enhancement using bayesian spectral amplitude estimation," in *Proc. IEEE ICASSP*, 2003, pp. I_832–I_835.
8. P. J. Wolfe and S. J. Godsill, "Efficient alternatives to the Ephraim and Malah suppression rule for audio signal enhancement," *special issue of EURASIP JASP on Digital Audio for Multimedia Communications*, vol. 2003, pp. 1043–1051, Sept. 2003.
9. J. Porter and S. Boll, "Optimal estimators for spectral restoration of noisy speech," in *Proc. IEEE ICASSP*, 1984, pp. 18A.2.1–18A.2.4.

10. R. Martin, "Speech enhancement using MMSE short time spectral estimation with gamma distributed speech priors," in *Proc. IEEE ICASSP*, 2002, pp. I-253–I-256.
11. S. Gazor and W. Zhang, "Speech probability distribution," *IEEE Signal Processing Letters*, vol. 10, pp. 204–207, July 2003.
12. ——, "A soft voice activity detector based on a laplacian-gaussian model," *IEEE Trans. Speech and Audio Processing*, vol. 11, pp. 498–505, Sept. 2003.
13. R. Martin and C. Breithaupt, "Speech enhancement in the DFT domain using Laplacian speech priors," in *Proc. IWAENC*, 2003, pp. 87–90.
14. Y. Ephraim and D. Malah, "Signal to noise ratio estimation for enhancing speech using the Viterbi algorithm," Technion - Israel Institute of Technology, Haifa, Israel, Technical Report, EE PUB 489, Mar. 1984.
15. O. Cappé, "Elimination of the musical noise phenomenon with the Ephraim and Malah noise suppressor," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 2, pp. 345–349, Apr. 1994.
16. B. H. Juang and L. R. Rabiner, "Mixture autoregressive hidden Markov models for speech signals," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-33, pp. 1404–1413, Dec. 1985.
17. Y. Ephraim and N. Merhav, "Hidden Markov processes," *IEEE Trans. Information Theory*, vol. 48, pp. 1518–1568, June 2002.
18. H. Sameti, H. Sheikhzadeh, L. Deng, and R. L. Brennan, "HMM-based strategies for enhancement of speech signals embedded in nonstationary noise," *IEEE Trans. Speech and Audio Processing*, vol. 6, pp. 445–455, Sept. 1998.
19. I. Cohen, "Modeling speech signals in the time-frequency domain using GARCH," *Signal Processing*, vol. 84, pp. 2453–2459, Dec. 2004.
20. R. F. Engle, Ed., *ARCH Selected Readings.* New York: Oxford University Press Inc., 1995.
21. T. Bollerslev, R. Y. ChouKenneth, and F. Kroner, "ARCH modeling in finance: A review of the theory and empirical evidence," *Journal of Econometrics*, vol. 52, pp. 5–59, Apr.–May 1992.
22. I. Cohen, "Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging," *IEEE Trans. Speech and Audio Processing*, vol. 11, pp. 466–475, Sept. 2003.
23. ——, "Relaxed statistical model for speech enhancement and a priori SNR estimation," *to appear in IEEE Trans. Speech and Audio Processing.*
24. J. S. Garofolo, "Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database," National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, Tech. Rep., (prototype as of Dec. 1988).
25. A. Stuart and J. K. Ord, *Kendall's Advanced Theory of Statistics.* 6th ed. London, UK: Edward Arnold, vol. 1, 1994.
26. A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of Royal Statistical Society (B)*, vol. 39, pp. 1–38, 1977.
27. G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions.* New York: Wiley, 1997.
28. E. K. Berndt, B. H. Hall, R. E. Hall, and J. A. Hausman, "Estimation and inference in nonlinear structural models," *Annals of Economic and Social Measurement*, vol. 4, pp. 653–665, 1974.

29. T. Bollerslev, “Generalized autoregressive conditional heteroskedasticity,” *Journal of Econometrics*, vol. 31, pp. 307–327, Apr. 1986.
30. I. Cohen, “Speech spectral modeling and enhancement based on autoregressive conditional heteroscedasticity model,” Technion - Israel Institute of Technology, Haifa, Israel, Technical Report, EE PUB 1425, Apr. 2004.
31. J. S. Lim and A. V. Oppenheim, “Enhancement and bandwidth compression of noisy speech,” *Proceedings of the IEEE*, vol. 67, pp. 1586–1604, Dec. 1979.
32. M. Berouti, R. Schwartz, and J. Makhoul, “Enhancement of speech corrupted by acoustic noise,” in *Proc. IEEE ICASSP*, 1979, pp. 208–211.

6 Single-Microphone Noise Suppression for 3G Handsets Based on Weighted Noise Estimation

Akihiko Sugiyama, Masanori Kato, and Masahiro Serizawa

Media and Information Research Laboratories, NEC Corporation
Kawasaki, Kanagawa 211-8666, Japan
E-mail: aks@ak.jp.nec.com, m-kato@df.jp.nec.com, serizawa@ah.jp.nec.com

Abstract. A noise suppression algorithm with high speech quality based on weighted noise estimation is presented. This algorithm continuously updates the estimated noise by weighted noisy speech in accordance with an estimated SNR. With a better noise estimate, a more correct SNR is obtained, resulting in the enhanced speech with low distortion. Subjective evaluation results show that five-grade mean opinion scores of this algorithm with a speech codec is improved by as much as 0.35, compared with either the MMSE-STSA or the EVRC noise suppression algorithm. A noise suppressor based on a later version of this noise suppression algorithm satisfies all the 3GPP minimum performance requirements. It is employed in the world's first 3G handset equipped with a 3GPP-endorsed noise suppressor.

6.1 Introduction

Applications of speech coding and speech recognition have been widespreading these days. Among these are cellular phones and car navigation systems, to name a few. One of the challenges in those applications is that they are often used in noisy environment such as in a car, on a street, at a station, or in an office. The quality of the speech conveyed to the other party or that of the voice to be recognized is seriously degraded, resulting in uncomfortable communication or a lower recognition rate. This is because speech coding and speech recognition have assumed noise-free environment in their development. A remedy for these problems is a noise canceler or a noise suppressor.

A variety of noise cancellation or noise suppression algorithms can be found in the literature [1]-[15]. Noise suppressors or cancelers can be categorized into two groups; single-microphone noise suppressors [1]-[7] and multiple-microphone noise cancelers [8]-[15]. In relatively high SNR (signal-to-noise ratio) environments, single-microphone solutions are common. Single-microphone noise suppression algorithms include those based on STSA (short time spectral amplitude) analysis [1]-[6]. STSA based algorithms extract the spectral amplitude of the clean speech from that of the noisy speech, *i.e.* the desired speech contaminated by noise. The phase information is kept unaltered.

STSA is most widely used for its computational advantage. Most of the noise suppression algorithms which have been evaluated in combination with a codec (coder and decoder) through subjective assessments can be categorized as an STSA [6,16–18]. Subjective assessment of the enhanced speech quality is a must from a viewpoint of recent international standards. Actually, 3GPP (the third generation partnership project), which standardizes the next generation cellular phone system, has determined the procedure for subjective evaluation of a noise suppression system [20]. Therefore, STSA is one of the most reliable noise suppression algorithms, which has shown its superiority by subjective assessment. Among others, MMSE (minimum mean square error)-STSA proposed by Ephraim and Malah [4] is a promising STSA based algorithm. It minimizes the mean squared error of the estimated short time spectral amplitude. It is reported that MMSE-STSA can provide good noise suppression without unpleasant residual noise called "musical noise" [21,22].

Studies of MMSE-STSA have been focused on the spectral gain, which determines the degree of noise suppression. Though noise estimation is directly related to the SNR used for spectral gain calculation, it has been almost untouched. Actually, the noise estimation in the original MMSE-STSA is carried out during nonspeech periods. The estimated noise may be incorrect during voiced periods, especially for nonstationary noise, leading to degraded speech quality. Specifically, overestimation of the noise may lead to fatal distortion in the enhanced speech. This is because overestimation of the noise is equivalent to underestimation of the SNR, which causes oversuppression of the noise.

Another STSA-based popular algorithm is the one employed for EVRC (enhanced variable rate codec) [16] which is the North American CDMA digital cellular phone standard [23]. This is the most successful algorithm which is commercially available. Its quality has been proven to be good through commercial products. Nevertheless, the quality may not be sufficiently good for a wide range of SNRs which were not given much attention to when it was standardized.

This chapter presents a noise suppression algorithm with good speech quality for a wide range of SNRs. This algorithm continuously estimates the noise with a noisy speech weighted by an estimated SNR. This makes more accurate SNR estimate available for gain calculation, resulting in good speech quality and sufficient noise suppression simultaneously. The spectral gain is modified so that the improved noise estimation can be utilized more effectively. In the next section, the original MMSE-STSA is reviewed with its drawback. Section 6.3 is devoted to a new algorithm based on the new noise estimate and a modified spectral gain. Finally, in Section 6.4, listening test results are presented and analyzed to show the superiority of the new noise suppression algorithm.

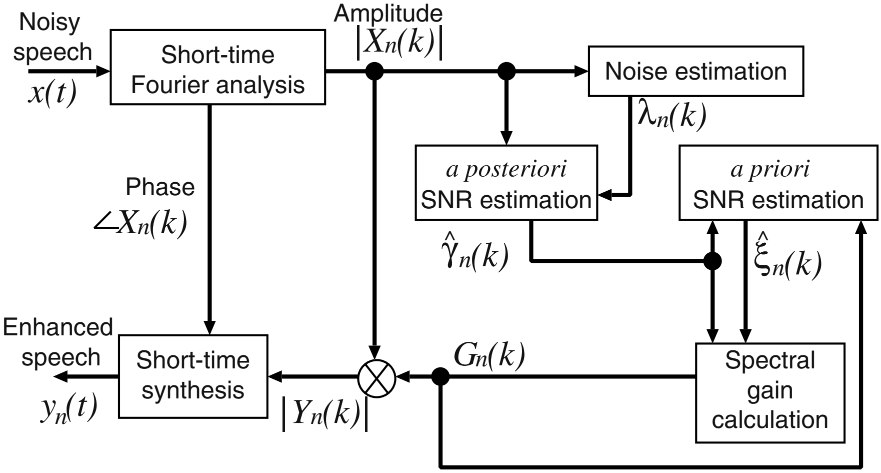


Fig. 6.1. Structure of the conventional noise suppression (MMSE-STSA).

6.2 Conventional Noise Suppression Algorithm

6.2.1 MMSE-STSA

Figure 6.1 shows the structure of the original MMSE-STSA [5]. It mainly consists of six functions; short-time Fourier analysis, noise estimation, *a posteriori* and *a priori* SNR estimation, spectral gain calculation and short-time synthesis. Short-time Fourier analysis computes a discrete Fourier transform of the noisy speech to obtain its spectral amplitude and phase. The amplitude of the noisy speech is multiplied by a spectral gain to make the amplitude of the enhanced speech. Short-time synthesis computes the inverse discrete Fourier transform of the enhanced speech amplitude multiplied by the phase of the noisy speech. After overlap-add processing of the inverse transform, an enhanced speech is obtained in the time domain. The spectral gain is calculated with an estimated *a priori* and *a posteriori* SNR. These SNR estimates are obtained based on the estimated noise power spectrum which is calculated from the spectral amplitude of the noisy speech during nonspeech periods.

Assuming that the clean speech $s(t)$ is degraded by an additive noise $d(t)$, the noisy speech $x(t)$ is given by

$$x(t) = s(t) + d(t), \tag{6.1}$$

where t is the time index. The noisy speech $x(t)$ is segmented into frames of M samples. An analysis window $h(t)$ of a size $2M$ with a 50 % overlap is applied to the segmented noisy speech $x_n(t)$ in frame n to obtain a windowed noisy speech $\tilde{x}_n(t)$ with $2M$ samples as follows:

$$\tilde{x}_n(t) = \begin{cases} h(t)x_{n-1}(t), & 1 \leq t \leq M \\ h(t)x_n(t-M), & M < t \leq 2M \end{cases} . \tag{6.2}$$

Let $X_n(k)$ denote a discrete Fourier transform of $\tilde{x}_n(t)$, where n and k refer to the analysis frame and the frequency bin index. Noise suppression is applied only to the spectral amplitude $|X_n(k)|$ of the noisy speech in each frequency bin. The phase $\angle X_n(k)$ of the noisy speech is kept unaltered to be used for the enhanced speech since the phase is not important for speech intelligibility and quality [2].

Based on $|X_n(k)|$ and estimated noise power spectrum $\lambda_n(k)$, the *a priori* SNR $\xi_n(k)$ and the *a posteriori* SNR $\gamma_n(k)$ are calculated by

$$\gamma_n(k) = \frac{|X_n(k)|^2}{E\{|D_n(k)|^2\}}, \tag{6.3}$$

$$\xi_n(k) = \frac{E\{|S_n(k)|^2\}}{E\{|D_n(k)|^2\}}, \tag{6.4}$$

where $S_n(k)$ and $D_n(k)$ are the discrete Fourier transform of $s(t)$ and $d(t)$, respectively. $E\{\cdot\}$ stands for the expectation operator. Because $E\{|D_n(k)|^2\}$ is not available, its estimate $\lambda_n(k)$ is substituted to approximate $\gamma_n(k)$ by $\hat{\gamma}_n(k)$ as

$$\hat{\gamma}_n(k) = \frac{|X_n(k)|^2}{\lambda_n(k)}. \tag{6.5}$$

$\lambda_n(k)$ is obtained[1] by averaging the power spectrum of the noisy speech in the first nonspeech section[2]. With this $\hat{\gamma}_n(k)$ and $\hat{\gamma}_{n-1}(k)$ in the previous frame, the *a priori* SNR $\xi_n(k)$ is approximated by $\hat{\xi}_n(k)$ calculated in a decision-directed approach by

$$\hat{\xi}_n(k) = \alpha\hat{\gamma}_{n-1}(k)G_{n-1}^2(k) + (1-\alpha)P[\hat{\gamma}_n(k) - 1]. \tag{6.6}$$

$P[x]$ is a rectifying function and α is a forgetting factor satisfying $0 < \alpha < 1$.

The spectral gain $G_n(k)$ is calculated with the estimated *a priori* and *a posteriori* SNRs by

$$G_n(k) = \left[(1 + v_n(k))I_0\left(\frac{v_n(k)}{2}\right) + v_n(k)I_1\left(\frac{v_n(k)}{2}\right)\right] \\ \cdot \frac{\varLambda_n(k)}{1 + \varLambda_n(k)} \frac{\sqrt{\pi v_n(k)}}{2\hat{\gamma}_n(k)} \exp\left(-\frac{v_n(k)}{2}\right), \tag{6.7}$$

[1] The exact noise estimation method is not disclosed in [4].

[2] It is generally assumed that the beginning of the noisy speech contains no speech components [33].

where $I_0(z)$ and $I_1(z)$ denote the modified Bessel functions of zero-th and first order [24], respectively. $v_n(k)$ and $\Lambda_n(k)$ are defined by

$$v_n(k) = \frac{\eta_n(k)}{1+\eta_n(k)}\hat{\gamma}_n(k), \tag{6.8}$$

$$\Lambda_n(k) = \frac{1-q}{q} \cdot \frac{\exp(v_n(k))}{1+\eta_n(k)}, \tag{6.9}$$

where

$$\eta_n(k) = \frac{\hat{\xi}_n(k)}{1-q}. \tag{6.10}$$

q is the probability of speech absence which is defined by the presence of nonspeech components and insignificant speech in the noisy speech. It is of interest to see that (6.7) is reduced to

$$G_n(k) = \frac{\Lambda_n(k)}{1+\Lambda_n(k)} \cdot \frac{\eta_n(k)}{1+\eta_n(k)} \tag{6.11}$$

for $\xi_n(k) \gg 1$ [4].

The enhanced speech spectrum $Y_n(k)$ is constructed with $\angle X_n(k)$ and $|Y_n(k)|$ as

$$\begin{aligned} Y_n(k) &= |Y_n(k)| \cdot \exp\{j\angle Y_n(k)\} \\ &= G_n(k)|X_n(k)| \cdot \exp\{j\angle X_n(k)\}, \end{aligned} \tag{6.12}$$

where $j = \sqrt{-1}$. After the inverse discrete Fourier transform of $Y_n(k)$, which is denoted by $\tilde{y}_n(t)$, is calculated, the enhanced speech $y_n(t)$ is obtained by performing the overlap-add processing as follows.

$$y_n(t) = \tilde{y}_{n-1}(t+M) + \tilde{y}_n(t), \;\; 1 \leq t \leq M. \tag{6.13}$$

6.2.2 Problem in Noise Estimation

The original MMSE-STSA estimates the noise power spectrum based on the noisy speech only in the first nonspeech period where the pure noise is available. This means that, for a nonstationary noise, a change in noise characteristics cannot be tracked and the enhanced speech quality becomes poor.

As a continuous noise estimation which has the tracking capability, a noise estimation method based on minimum statistics [25]-[29] is widely used. This is simple compared to [6] which requires VAD (voice activity detection). This fact imposes additional constraints on computations, not to mention the tuning difficulty of the VAD in low SNRs and computations for psychoacoustic analysis [30] used in noise suppression of [6].

The minimum statistics uses the minimum value of the smoothed noisy speech power within a finite time-window length L_{MS} as the estimated noise. The estimated noise $\lambda_n(k)$ in the n-th frame for k-th bin is obtained by

$$\lambda_n(k) = C_{MS} \cdot \min_{l=0,1,\ldots,L_{MS}} \overline{|X_{n-l}(k)|}^2, \tag{6.14}$$

where min is a minimum-value operator. C_{MS} is a constant to compensate for the bias of the minimum estimate. $\overline{|X_n(k)|}^2$ denotes the smoothed noisy speech power in the n-th frame given by

$$\overline{|X_n(k)|}^2 = \beta\overline{|X_{n-1}(k)|}^2 + (1-\beta)|X_n(k)|^2, \tag{6.15}$$

with a constant β $(0 < \beta < 1)$.

Because of the statistical nature, a larger L_{MS} provides more accurate noise estimation for a stationary noise. However, the tracking capability for a nonstationary noise is degraded. A short window, on the other hand, may introduce overestimation[3] which results in poor speech quality for high SNRs, although it achieves better tracking capability. As a result, there is a trade-off in the selection of L_{MS}. Therefore, it is not easy to select an appropriate window length for good tracking capability without overestimation.

Overestimation may cause serious distortion in the enhanced speech. This is because overestimation of the noise is equivalent to underestimation of the *a posteriori* SNR based on the definition in (6.5). A low *a posteriori* SNR naturally leads to a small spectral gain as is seen from (6.5) through (6.10). This fact agrees with a common intuition that a smaller spectral gain is preferable for a lower SNR to achieve stronger suppression of the noise. It should be noted that distortion by overestimation is more audible in speech periods where the signal power should be larger and the SNR is generally high. Equation (6.7) suggests this fact. A lower *a posteriori* SNR would result in a lower *a priori* SNR, which leads to a smaller spectral gain than that for the correct *a posteriori* SNR. Actually, from Fig. 4 of [4], it is seen that a 5 dB overestimation in the instantaneous SNR, thus, the *a posteriori* SNR, at a 15 dB SNR would cause almost half as much the spectral gain calculated by (6.7) as that obtained by correct estimation. It is straightforward to explain, for a similar reason, that underestimation would cause insufficient suppression of the noise, causing degraded quality of the enhanced speech. Correct noise estimation is essential to good quality of the enhanced speech.

6.3 New Noise Suppression Algorithm

To achieve good tracking capability without overestimation for various non-stationary noise sources, the new noise suppression algorithm employs a noise

[3] A short window is defined by a small value of L_{MS} and the probability that the minimum value is encountered in the short window is naturally small.

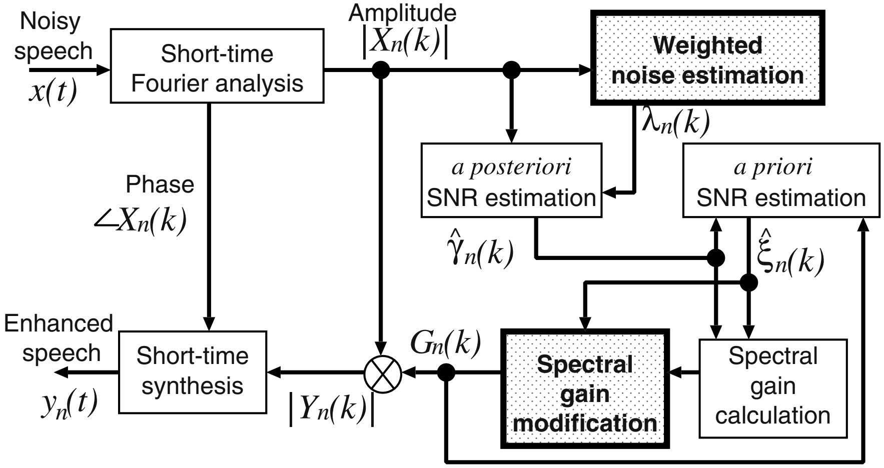


Fig. 6.2. Structure of the new noise suppression.

estimation with a weighting factor based on the estimated SNR. This estimation is not based on a stochastic processing and thus, is free from an appropriate choice of the window size. The weighting factor makes continuous noise estimation possible without overestimation even in speech periods. As a result, the weighted noise estimation tracks the change of the noise characteristics in both speech and nonspeech periods. To obtain a suitable spectral gain for the new noise estimation, the MMSE-STSA-original spectral gain is modified in accordance with the SNR. Figure 6.2 shows the structure of the new noise suppression.

6.3.1 Weighted Noise Estimation

The weighted noise estimation mainly consists of three steps; SNR estimation, weighting factor calculation, and averaging. The noisy speech is weighted by a weighting factor calculated based on the estimated SNR. The estimated noise is obtained as an average of the weighted noisy speech.

In the first step, the estimated SNR $\tilde{\gamma}_n(k)$ is obtained from the power spectrum $|X_n(k)|^2$ of the noisy speech in the n-th frame and the estimated noise $\lambda_{n-1}(k)$ in the previous frame as follows:

$$\tilde{\gamma}_n(k) = \frac{|X_n(k)|^2}{\lambda_{n-1}(k)}. \tag{6.16}$$

In the second step, the weighting factor $W_n(k)$ is calculated in accordance with the estimated SNR $\tilde{\gamma}_n(k)$. A nonlinear function in Fig. 6.3 is used to calculate $W_n(k)$, where $\tilde{\gamma}_1, \tilde{\gamma}_2$, and θ_Z are constants. This nonlinear function

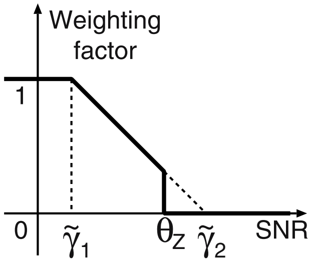


Fig. 6.3. Nonlinear function for weighting factor.

is designed such that the weighting factor is almost inversely proportional to the estimated SNR. Overestimation for high SNRs does not happen by appropriately suppressing the contribution of the noisy speech to the estimate.

The weighted noisy speech $z_n(k)$ and its average $\lambda_n(k)$, which is used as the estimated noise, are given by

$$z_n(k) = W_n(k)|X_n(k)|^2, \tag{6.17}$$

$$\lambda_n(k) = \frac{trace\{\boldsymbol{Z}_n(k)\}}{\Psi(\boldsymbol{Z}_n(k))}, \tag{6.18}$$

where

$$\boldsymbol{Z}_n(k) = \begin{cases} [z_n(k), \tilde{\boldsymbol{Z}}_{n-1}(k)], \; n \le T_{init} \\ [z_n(k), \tilde{\boldsymbol{Z}}_{n-1}(k)], \; \tilde{\gamma}_n(k) < \theta_Z, \\ \boldsymbol{Z}_{n-1}(k), \qquad\quad \text{otherwise} \end{cases} \tag{6.19}$$

$$\tilde{\boldsymbol{Z}}_0(k) = \mathbf{0}_{1\times(L_Z-1)}, \tag{6.20}$$

$$\tilde{\boldsymbol{Z}}_n(k) = \boldsymbol{Z}_n(k) \, [\, \mathbf{I}_{L_Z-1} \; \mathbf{0}^T_{1\times(L_Z-1)}]^T. \tag{6.21}$$

Equation (6.17) represents that the power spectrum of the noisy speech is weighted by a factor $W_n(k)$ based on an estimated SNR as in Fig. 6.3. $\Psi(\boldsymbol{Z}_n(k))$ in (6.18) is the number of non-zero elements in $\boldsymbol{Z}_n(k)$ and $trace\{\cdot\}$ is an operator to take the sum of its diagonal elements. Because $\boldsymbol{Z}(k)$ is a row vector, $trace\{\boldsymbol{Z}(k)\}$ is simply a sum of its elements except zero-valued ones. Therefore, (6.18) defines an averaging operation for non-zero values of $z_n(k)$. L_Z and $\mathbf{I}_{L_Z-1}$ are the number of samples for the average and the identity matrix of size $L_Z - 1$, respectively. Equation (6.19) means that $\boldsymbol{Z}_n(k)$ is updated when the estimated SNR is lower than a threshold θ_Z, or the frame index is smaller than or equal to T_{init}. An inappropriate weighted noisy speech sample by an unreliable SNR estimate is eliminated by θ_Z to obtain a better value of $\lambda_n(k)$. Equations (6.20) and (6.21) are introduced to express the initial status as it is clear by carefully following from (6.17) through (6.21). $W_n(k) = 1$ for $0 < n \le T_{init}$ under the assumption that the speech does not start in the first T_{init} frames. Noise estimation is performed independently for each bin, enabling more precise results depending on the SNR at each bin.

6.3.2 Spectral Gain Modification

The spectral gain is modified in two ways; conditional scaling and limitation. Conditional scaling further suppresses the residual noise for high SNRs resulting in clearer enhanced speech. The minimum value of the spectral gain is limited with G_{floor} so that excessive suppression, which causes speech distortion, can be avoided [19].

The original spectral gain $G_n(k)$ obtained by the MMSE-STSA is first multiplied by a scaling factor G_{mod} as follows:

$$G_n(k) = \begin{cases} G_{mod}G_n(k), & \xi_n(k) < \theta_G \\ G_n(k), & \text{otherwise} \end{cases} . \tag{6.22}$$

The scaling is performed only when the *a priori* SNR $\xi_n(k)$ is smaller than a threshold θ_G. $G_{mod} < 1$ makes the value of $G_n(k)$ smaller to further suppress the noise for low SNRs.

Following the limiting operation in (6.22), the spectral gain $G_n(k)$ is modified so that its minimum value is larger than or equal to a flooring parameter G_{floor} as

$$G_n(k) = \begin{cases} G_n(k), & G_n(k) > G_{floor} \\ G_{floor}, & \text{otherwise} \end{cases} . \tag{6.23}$$

Equation (6.23) helps eliminate undesirable oversuppression for a better quality of the enhanced speech.

6.3.3 Computational Requirements

The new noise suppression algorithm is computationally simple. Equation (6.16) needs one division (DIV) since the power spectrum has been calculated already in another part. Equation (6.17) introduces one multiplication (MPY) per bin. Expression (6.18) represents a moving average which can be performed by two additions (ADD) per sample and a division. Expression (6.19) consists of two comparisons (CMP) at the most followed by a pointer modification and data storage (STR) in a memory. Storage of L_z zeros, at the most, is required by (6.21). For spectral gain modification, one comparison and one multiplication in (6.22) and one comparison in (6.23) are needed. All together, operations summarized in Table 6.1 are additionally required for the new noise suppression compared to MMSE-STSA. Because these numbers are for a single bin, the total number should be obtained by multiplying these numbers by M.

Table 6.1 Additionally required computations per bin.

Operations	MPY	ADD	DIV	CMP	STR
Noise Estimation	1	2	2	2	$L_z + 1$
Spectral Gain Modification	1	0	0	2	0

6.4 Evaluation

The new noise suppression algorithm was compared with the conventional noise suppression algorithms in terms of noise estimation accuracy and subjective quality of the enhanced speech. Noise estimation accuracy shows the pure contribution of the newly introduced weighted noise estimation. Subjective quality of the enhanced speech reflects the overall quality improvement as a complete noise suppression system. Because a noise suppressor is often used in combination with a codec, it is of great significance to show the combined subjective quality. Following the global standard [20], the overall quality of the noise suppressors were evaluated through subjective assessments[4].

As one of the conventional noise suppression algorithms, MMSE-STSA was evaluated to show the quality enhancement mainly by the new weighted noise estimation. Instead of its original noise estimation, MMSE-STSA was combined with the Minimum Statistics for fair comparison[5].

As the other conventional algorithm, the most successful commercial algorithm, namely, the noise suppression for EVRC (NS-EVRC)[6], which is employed by *cdmaOne* [31], was used. Comparison with these algorithms helps understand the position of the new noise suppressor in this chapter among state-of-the-art algorithms.

Both speech and noise had been sampled at 8 kHz before they were digitally mixed to generate the noisy speech. Four kinds of background noise sources, namely, the babble, the street, the office, and the vehicle noise were used. These noise sources cover all required kinds by 3GPP in the evaluation of codecs and noise suppressors [20,33]. Hamming window was used as the analysis window $h(t)$. Other parameters are shown in Table 6.2.

[4] This is a natural and realistic evaluation because nobody looks at the SNR when the noise suppressor is in an actual use. The user cares for the subjective quality, not the SNR.

[5] The Minimum Statistics is employed in the algorithm proposed by Siemens, which was tested in the 3GPP noise suppression evaluation [32]. This algorithm was ranked close to the best in its ACR (absolute category rating) tests. Therefore, MMSE-STSA combined with the minimum statistics is a good candidate for comparison. None of the evaluated algorithms are disclosed in details, which makes direct comparison impossible.

[6] NS-EVRC is the basis for the algorithm proposed by Motorola, which was tested in the 3GPP noise suppression evaluation [32]. This algorithm received the highest scores in its CCR (comparison category rating) tests.

Table 6.2 Parameters for new noise suppression.

Parameter	Value
M	128
α	0.98
q	0.20
L_Z	20
$\tilde{\gamma}_1$	0 dB
$\tilde{\gamma}_2$	10 dB
θ_Z	7 dB
T_{init}	4 frames
θ_G	10 dB
G_{mod}	-1.0 dB
G_{floor}	-6.8 dB

6.4.1 Objective Evaluation for Noise Estimation

Noise estimation accuracy was evaluated frame by frame based on the normalized estimation error ε_n given by

$$\varepsilon_n = 10\log_{10}\left(\frac{\sum_{k=0}^{M}\left||D_n(k)|^2 - \lambda_n(k)\right|}{\sum_{k=0}^{M}|D_n(k)|^2}\right). \tag{6.24}$$

For the minimum statistics, $L_{MS} = 50$ (*i.e.* 0.80 sec), $\beta = 0.90$ and $C_{MS} = 1.5$ were used as specified in [25]. The initial averaging method, which is the original noise estimation used for the MMSE-STSA, estimates the noise power spectrum in the initial $T_{IA} = 20$ frames. This estimate $\lambda_n(k)$ is defined by

$$\lambda_n(k) = \begin{cases} |X_n(k)|^2, & n \leq T_{IA} \\ \frac{1}{T_{IA}}\sum_{n=1}^{T_{IA}}|X_n(k)|^2, & \text{otherwise} \end{cases}. \tag{6.25}$$

Figures 6.5–6.8 show the normalized estimation error for the evaluated noise estimation methods under a 5 dB SNR condition with the corresponding clean speech in Fig. 6.4. The normalized estimation error of the initial averaging method (Init. Ave.), the minimum statistics method (Min. Stat.), and the new method (New) are shown by a dashed, a thin solid, and a thick solid line, respectively.

In Figs. 6.5–6.8, the new noise estimation achieves the best accuracy among the three. It is shown that the estimation error of the minimum statistics generally becomes greater in the latter part of a speech period. This is a consequence of a small value of L_{MS}. In the latter part of a speech period, overestimation occured for the reason described in Section 6.2.2. The new noise estimation is clearly more accurate than the initial averaging in

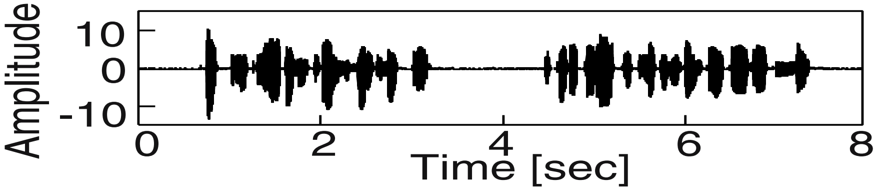


Fig. 6.4. Original speech.

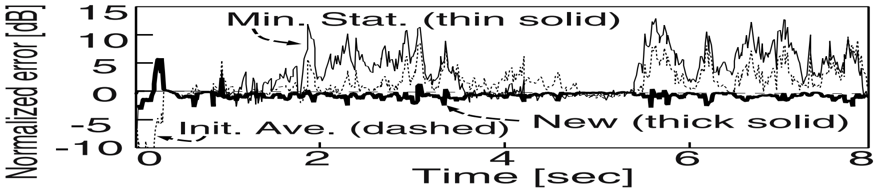


Fig. 6.5. Normalized error in noise estimation (babble noise, SNR=5dB).

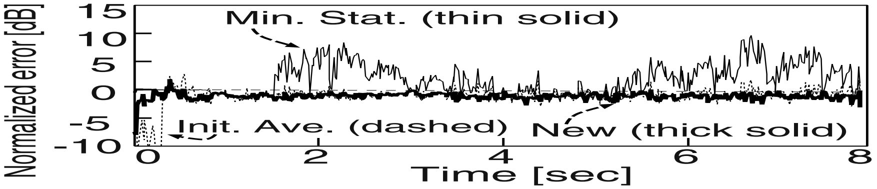


Fig. 6.6. Normalized error in noise estimation (street noise, SNR=5dB).

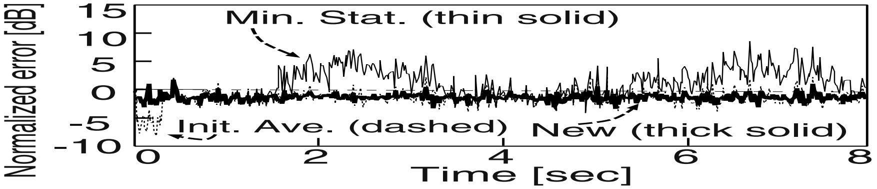


Fig. 6.7. Normalized error in noise estimation (office noise, SNR=5dB).

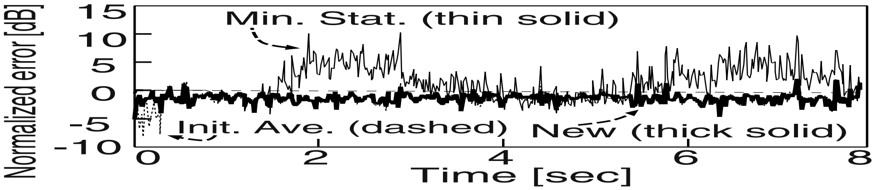


Fig. 6.8. Normalized error in noise estimation (vehicle noise, SNR=5dB).

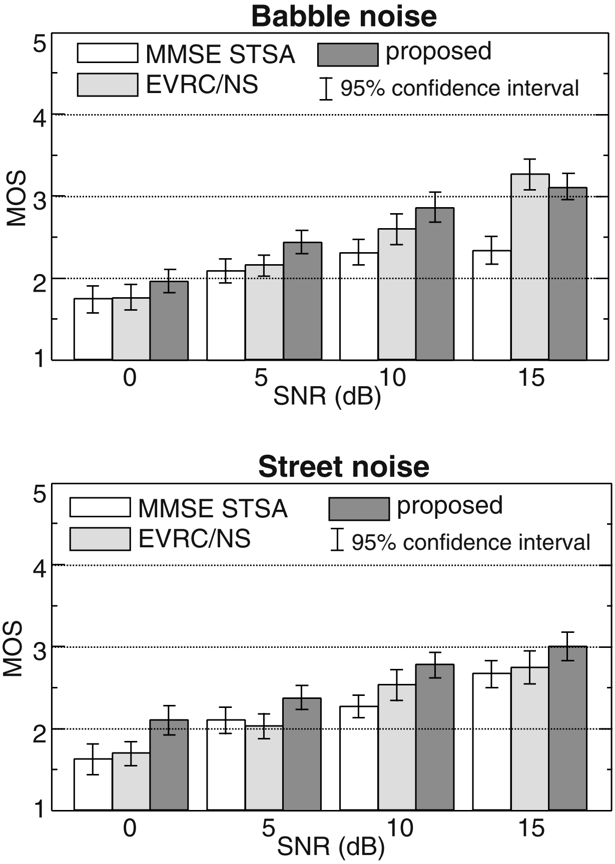


Fig. 6.9. Results of listening test without codec (babble and street Noise).

Fig. 6.5. while the overall difference is insignificant in Figs. 6.6–6.8. However, instantaneous values still exhibit significant differences.

Although it was reported that the minimum statistics was superior to the conventional methods [25], the results in this chapter are different. It is perhaps caused by inappropriate parameter settings even though the employed values were all from [25]. This may be considered as another proof for the trade-off in the minimum statistics. Because its parameter optimization is not in the scope of this chapter, further optimization is not provided.

6.4.2 Subjective Evaluation

Two listening tests with and without a speech codec were carried out by using a five-grade mean opinion score (MOS) based on the absolute category rating (ACR) [34]. Twelve listeners evaluated the enhanced speech obtained by noise

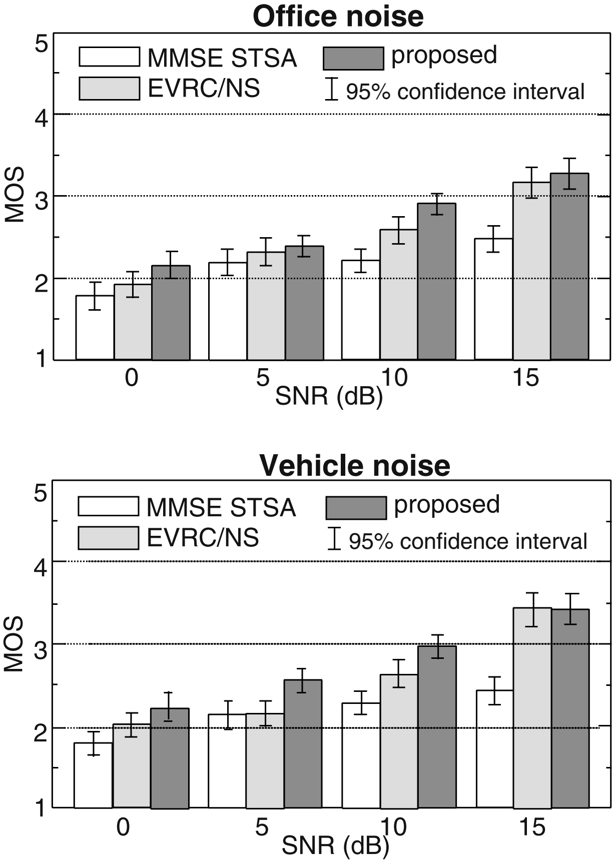


Fig. 6.10. Results of listening test without codec (office and vehicle noise).

suppressions under the test. Each listener scored between one and five, with five being the best. In the test with a codec, the noise-suppressed speech was encoded and decoded by the EVRC codec. Six clean speech signals by four different speakers (2 males and 2 females) were used as the clean speech. A noise was added to the clean speech with different SNRs (0, 5, 10 and 15 dB) to produce noisy speech. Noisy speech degraded under 0, 6, 12, 18, 24 and 30 dB SNR conditions were also used as anchors.

Figures 6.9 and 6.10 show the listening test results without a speech codec for babble, street, office, and vehicle noise. The corresponding results with a speech codec are depicted in Figs. 6.11 and 6.12.

The scores of the anchors in both tests, which are not shown in Figs. 6.9 through 6.12, distributed in the range between 1.6 and 4.2 in proportion to SNR. The mean values of the scores are represented with the height of

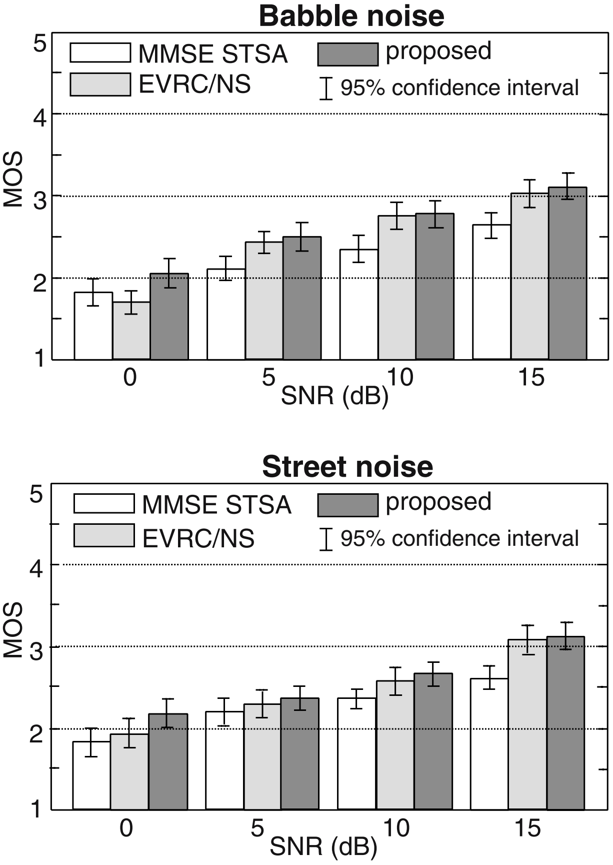


Fig. 6.11. Result of listening test with codec (babble and street noise).

a vertical bar. The vertical line centered around a mean value is the 95% confidence interval.

Scores of the new method are higher than those of the original MMSE-STSA and the EVRC noise suppression in most conditions. Differences between the new algorithm and the other two are smaller with a codec than without a codec. This is because speech distortion introduced by noise suppression and residual noise are masked by the speech encoding/decoding process. The differences between the scores of the new algorithm and those of the original MMSE-STSA combined with the minimum statistics noise estimation become larger for high SNR conditions such as 10 and 15 dB. Large differences for high SNR conditions are mainly caused by overestimation of the minimum statistics method. The differences are statistically significant

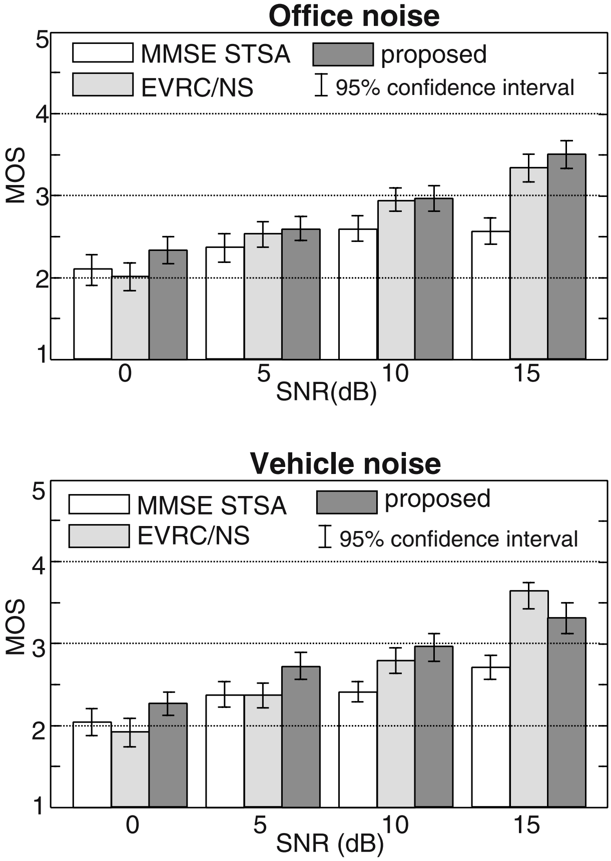


Fig. 6.12. Result of listening test with codec (office and vehicle noise).

for SNRs above 10 dB and for 44 % of the SNRs below 5 dB. The maximum difference is 0.93 and 0.90 with and without a codec, respectively.

When the new algorithm is compared to the EVRC noise suppression, the difference becomes larger for low SNRs such as 0 and 5 dB. The differences are statistically significant for 44 % of the low SNRs. The maximum difference is 0.35 and 0.40 with and without a codec, respectively. It is shown in Figs. 6.11 and 6.12 that the new algorithm achieves better scores than the EVRC noise suppression, even though the latter is optimized for the EVRC codec. This fact is another evidence for the superiority of the new noise suppression algorithm.

The noise suppression algorithm presented in this chapter was further modified with pseudo noise injection and synthesis windowing for better enhanced-speech quality [35]. The performance evaluation results [36] of this latest version, which shows that it satisfies all the 3GPP requirements specified in a standard [20], have officially been endorsed by 3GPP [37].

6.5 Conclusions

A noise suppression algorithm based on weighted noise estimation and MMSE-STSA has been presented. The new algorithm continuously updates the noise estimate by weighted noisy speech in accordance with an estimated SNR. The spectral gain is modified with the SNR so that it better fits the new noise estimate for higher speech quality. With the improved noise estimate, distortion in the enhanced speech is reduced due to a more correct SNR, resulting in smaller oversuppression.

In the subjective evaluation with a five-grade mean opinion score (MOS), the new noise suppression obtained better scores than the original MMSE-STSA and the EVRC noise suppression for most conditions. The maximum improvement in the score reached 0.35 and 0.40 with and without a speech codec, respectively.

Under 80 % of all tested conditions, the new algorithm outperforms either or both of the conventional algorithms with a statistically significant MOS difference. A later version of this noise suppressor is equipped with in millions of 3G handsets as the one and only commercially available 3GPP-endorsed noise suppressor.

References

1. S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 113–120, Apr. 1979.
2. J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proc. of the IEEE*, vol. 67, pp. 1586–1604, Dec. 1979.
3. R. J. McAulay and M. L. Malpass, "Speech enhancement using a soft- decision noise suppression filter," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 137–145, Apr. 1980.
4. Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 1109–1121, Dec. 1984.
5. Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 443–445, Apr. 1985.
6. N. Virag, "Single channel speech enhancement based on masking properties of the human auditory system," *IEEE Trans. Speech and Audio Processing*, vol. 7, pp. 126–137, Mar. 1999.
7. Z. Goh, K.-C. Tan, and B. T. G. Tan, "Kalman-filtering speech enhancement method based on a voiced- unvoiced speech model," *IEEE Trans. Speech and Audio Processing*, vol. 7, pp. 510–524, Sept. 1999.

8. B. Widrow, J. R. Grover, Jr., J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. r. Zeidler, E. Dong, Jr., and R. C. Goodlin, "Adaptive noise cancelling: principles and applications," *Proc. of the IEEE*, vol. 63, pp. 1692–1716, Dec. 1975.
9. S. F. Boll and D. C. Publisher, "Suppression of acoustic noise in speech using two microphone adaptive noise cancellation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 752–753, Dec. 1980.
10. M. J. Al-Kindi and J. Dunlop, "A low distortion adaptive noise cancellation structure for real time applications," in *Proc. IEEE ICASSP*, 1987, pp. 2153–2156.
11. J. Dunlop and M. J. Al-Kindi, "Application of adaptive noise cancelling to diver voice communication," in *Proc. IEEE ICASSP*, 1987, pp. 1708–1711.
12. G. Mirchandani, R. L. Zinser, and J. B. Evans, "A New Adaptive Noise Cancellation Scheme in the Presence of Crosstalk," *IEEE Trans. Circuits and Systems*, vol. 39, pp. 681–694, Oct. 1992.
13. V. Parsa, P. A. Parker, and R. N. Scott, "Performance analysis of a crosstalk resistant adaptive noise canceller," *IEEE Trans. Circuits and Systems*, vol. 43, pp. 473–482, July 1996.
14. S. Ikeda and A. Sugiyama, "An adaptive noise canceller with low signal distortion for speech codecs," *IEEE Trans. Signal Processing*, vol. 47, pp.665–674, Mar. 1999.
15. S. Ikeda and A. Sugiyama, "An adaptive noise canceller with low signal-distortion in the presense of crosstalk," *IEICE Trans. Fund.*, vol. E82-A, pp. 1517–1525, Aug. 1999.
16. T. V. Ramabadran, J. P. Ashley, and M. J. McLaughlin, "Background noise suppression for speech enhancement and coding," in *IEEE Workshop on Speech Coding and Tel.*, 1997, pp. 43–44.
17. D. Malah, R. V. Cox, and A. J. Accardi, "Tracking speech-presence uncertainty to improve speech enhancement in non-stationary noise environments," in *Proc. IEEE ICASSP*, 1999, pp. 789–792.
18. N. S. Kim and J. H. Chang, "Spectral enhancement based on global soft decision," *IEEE Signal Processing Letters*, vol. 7, pp. 108–110, May 2000.
19. M. Berouti, R. Schwartz, and J. Makhoul, "Enhancement of speech corrupted by acoustic noise," in *Proc. IEEE ICASSP*, 1979, pp. 208–211.
20. 3GPP TS 26.077, *Minimum Performance Requirements for Noise Suppresser Application to the AMR Speech Encoder*. Mar. 2001.
21. O. Cappe, "Elimination of the musical noise phenomenon with Ephraim and Malah noise suppressor," *IEEE Trans. Speech Audio Processing*, vol. 2, pp. 345–349, Apr. 1994.
22. P. Scalart and J. V. Filho, "Speech enhancement based on a priori signal to noise estimation," in *Proc. IEEE ICASSP*, 1996, pp. 629–632.
23. TIA/EIA/IS-127, *Enhanced Variable Rate Codec*. 1996.
24. M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. John Wiley & Sons, 1964.
25. R. Martin, "Spectral subtraction based on minimum statistics," in *Proc. EUSIPCO*, 1994, pp. 1182–1185.
26. R. Martin, "An MMSE soft-decision estimation for combined noise and residual echo reduction," in *Proc. IWAENC*, 1999, pp. 84–87.

27. V. Stahl, A. Fischer, and R. Bippus, “Quantile based noise estimation for spectral subtraction and Wiener filtering,” in *Proc. IEEE ICASSP*, 2000, pp. 1875–1878.
28. S. Gustafsson, P. Jax, and P. Vary, “A novel psychoacoustically motivated audio enhancement algorithm preserving background noise characteristics,” in *Proc. IEEE ICASSP*, 1998, pp. 397–400.
29. T. Gulzow, “Spectral-subtraction-based speech enhancement using a new estimation technique for non-stationary noise,” in *Proc. IWAENC*, 1999, pp. 76–79.
30. J. D. Johnston, “Transform coding of audio signal using perceptual noise criteria,” *IEEE J. of Selec. Areas Commun.*, vol. 6, pp. 314–323, Feb. 1988.
31. B. Sawa, M. Nashioka, K. Nakamura, and M. Enoki, “Cellular telephones and PHS terminals,” *Toshiba Review*, vol. 54, pp. 38-43, Apr. 1999 (in Japanese).
32. 3GPP TS 26.978, *Results of the AMR Noise Suppression Selection Phase.* Mar. 2001.
33. 3GPP TS 26.975, *Performance Characterization of the AMR Speech Codec.* Apr. 2001.
34. ITU-T COM 12, *Methods for Subjective Determination of Transmission Quality.* Recommendation P.800.
35. A. Sugiyama, T. P. Hua, M. Kato, and M. Serizawa, “Noise suppression with synthesis windowing and pseudo noise injection,” in *Proc. IEEE ICASSP*, 2002, pp. 545–548.
36. M. Kato, A. Sugiyama, and M. Serizawa, “A family of 3GPP noise suppressors for the AMR codec and the evaluation results,” in *Proc. IEEE ICASSP*, 2003, pp. 916–919.
37. “TSG SA WG4 status report at TSG SA#17,” TSGS#16-020431, Sept. 2002.
38. M. Kato, M. Serizawa, N. Toki, U. Mori, Y. Morishita, and K. Hayashi, “Noise suppression with high speech quality based on weighted noise estimation for 3G handsets,” *NEC Res. & Develop.*, vol. 44, pp. 65–73, Oct. 2003.

7 Signal Subspace Techniques for Speech Enhancement

Firas Jabloun[1] and Benoit Champagne[2]

[1] Toshiba Research
Cambridge, UK
E-mail: firas.jabloun@crl.toshiba.co.uk

[2] McGill University
Montreal, Canada
E-mail: champagne@ece.mcgill.ca

Abstract. In this chapter, we present the signal subspace approach (SSA) for speech enhancement. The SSA is becoming a serious competitor to its already widely used frequency-domain counterparts since it seems to offer a better compromise between signal distortion and the level of the residual noise. We provide a detailed description of the technique in terms of its underlying theory as well as the implementation issues. We also discuss the methods, proposed in the literature, to deal with the colored noise case and to cope with the complexity concerns usually associated with the SSA. In addition to that, we provide a filterbank interpretation to the SSA which allows it to be viewed from a frequency-domain perspective which is a more intuitive domain as far as speech signals are concerned. Finally, we present some of the latest variations and extensions to the SSA found in the literature which also serve as suggestions to further research in this area.

7.1 Introduction

In modern speech communication applications it is no longer assumed that the developed systems operate in favorable environments. Indeed, the currently existing systems are constantly being used under very adverse conditions where the users still expect to receive satisfactory services. For this reason, speech enhancement research has intensified and innovative strategies to face the new challenges are being sought.

Usually, frequency-domain techniques are the most preferred approaches due to their relative simplicity and ease of implementation. Moreover, speech is traditionally well analyzed and understood in the frequency domain which makes processing in this domain more appealing and easier to design. Nonetheless, these methods, namely spectral subtraction [2] and its variants, are nowhere close to offering fully satisfactory solutions to their inherent problems: the musical noise artifact and the inevitable trade-off between signal distortion and the level of the residual noise. Therefore, different research horizons need to be investigated such as processing in a different transform domain. One such potential domain is the eigendomain in which the signal subspace approach operates.

The signal subspace approach (SSA) has demonstrated itself as a powerful signal processing tool in various applications including array processing via the popular MUSIC algorithm [32] and variations thereof [27]. In speech enhancement, the SSA has been originally introduced by Dendrinos *et al.* [6] who propose to use the singular value decomposition (SVD) of a data matrix to remove the noise subspace and then reconstruct the desired speech signal from the remaining signal subspace. This approach gained more popularity when Ephraim and Van Trees proposed a new technique based on the eigenvalue decomposition (EVD) of the covariance matrix of the input speech vector [8]. This method consists in estimating a transform, namely the Karhunen-Loeve transform (KLT), which will project the input speech vectors into a subspace called the signal subspace hence readily eliminating the components in the orthogonal noise only subspace. The signal coefficients in the signal subspace (or the eigendomain) have the property of being uncorrelated which allows to process them individually using a diagonal gain matrix in order to suppress any remaining noise components. The entries of this matrix are estimated according to a particular optimization criterion leading to several alternatives for the gain function. Finally, the enhanced signal is reconstructed in the time domain using an inverse KLT. The SSA was found to outperform frequency-domain methods particularly by offering a more satisfactory compromise between signal distortion and residual noise level leading to a less annoying musical noise [8].

The SSA has not yet received as much attention as its frequency-domain counterparts and its use in practice is still scarce. The reason for that remains its relatively high computational load (due to the costly EVD step) and because it operates in a less familiar domain for speech signals. Recently, however, with the emergence of new SSA based techniques coupled with the substantial developments in the available computational power, the SSA has become a serious candidate to compete with the currently employed noise reduction methods. Several new studies have recently been published where the superiority of the SSA has been further confirmed and variations to the original approach has been developed [21,26,31,29,15,24,38]. The SSA was also tested in a speech recognition task and promising results have been reported [16], [14].

In this chapter, we provide a thorough presentation of the SSA from an EVD perspective, i.e. based on the Ephraim and Van Trees approach. In Section 7.2, we introduce the technique by presenting the underlying signal and noise models used. Then, in Section 7.3, we provide the different signal estimators found in the literature. In Section 7.4, we review the techniques developed to handle colored noise situations. Section 7.5 offers a different insight into the SSA by analyzing it from a frequency-domain perspective. Such interpretation can shed more light on the SSA thus potentially leading to even further improvements. The computational cost issue is addressed in Section 7.7 and state of the art techniques to deal with it are presented.

Finally, variations to the signal subspace technique found in the literature are reviewed in Section 7.8 and conclusions are given in Section 7.9.

7.2 Signal and Noise Models

In this section, we present the underlying theory of the SSA namely the signal and noise models employed and their representation in terms of the EVD of the speech covariance matrix.

Assume that the speech signal can be represented by a linear model of the form

$$\mathbf{s} = \mathbf{A}\mathbf{c} = \sum_{i=1}^{Q} \mathbf{a}_i c_i, \tag{7.1}$$

where $\mathbf{s} = [s_1, \ldots, s_P]^T$ is a sequence of random signal samples and $\mathbf{c} = [c_1, \ldots, c_Q]^T$ is, in general, a zero mean random coefficient vector. $\mathbf{A} \in \mathbb{C}^{P \times Q}$ is a model matrix with linearly independent columns, $\mathbf{a}_i$. Therefore $\text{rank}(A) = Q \leq P$, in general. An example of such a model used with speech signals is the *damped complex sinusoid* model whose basis vector is given by [3]

$$\mathbf{a}_i = [1, \rho_i^1 e^{j\omega_i 1}, \ldots, \rho_i^{P-1} e^{j\omega_i (P-1)}]^T. \tag{7.2}$$

In the SSA framework, the precise underlying model is not important. What is important, however, is that $Q < P$ which is a valid assumption for speech signals [8]. Hence the columns of $\mathbf{A}$ do not span the entire Euclidean space but rather a subspace referred to as the *signal subspace*. Here, the column span of matrix $\mathbf{A}$, also called its range, will be denoted by $\mathcal{R}\{\mathbf{A}\}$.

The covariance matrix of the vector $\mathbf{s}$ in (7.1) is given by[1]

$$\mathbf{R}_s = E\{\mathbf{s}\mathbf{s}^T\} = \mathbf{A}\mathbf{R}_c\mathbf{A}^T, \tag{7.3}$$

where $\mathbf{R}_c = E\{\mathbf{c}\mathbf{c}^T\}$ is the covariance matrix of vector $\mathbf{c}$, where we assume that $\mathbf{R}_c > 0$. Accordingly, $\mathbf{R}_s$ is rank deficient with $\text{rank}(\mathbf{R}_s) = Q < P$ and hence it has $P - Q$ zero eigenvalues.

Suppose now that we have available a P-dimensional noisy observation vector $\mathbf{x}$ such that

$$\mathbf{x} = \mathbf{s} + \mathbf{w}, \tag{7.4}$$

where $\mathbf{w}$ is the noise vector. The noise is assumed to be zero mean and uncorrelated with the speech signal. The noise covariance matrix $\mathbf{R}_w$ is assumed to be known and is given by

$$\mathbf{R}_w = E\{\mathbf{w}\mathbf{w}^T\} = \sigma^2\mathbf{I}, \tag{7.5}$$

[1] Unless otherwise mentioned, all signals in this chapter are considered to be real valued.

meaning that the noise is white with variance σ^2. The whiteness assumption is necessary for the time being in order to analyze the signal subspace method. The more practical case of colored noise will need further processing and will be addressed in Section 7.4. With these assumptions, the noisy signal covariance matrix $\mathbf{R}_x$ can be written as

$$\mathbf{R}_x = \mathbf{R}_s + \mathbf{R}_w = \mathbf{R}_s + \sigma^2\mathbf{I}. \tag{7.6}$$

Now consider the eigenvalue decomposition (EVD) of $\mathbf{R}_x$ defined as

$$\mathbf{R}_x = \mathbf{U}\boldsymbol{\Lambda}_x\mathbf{U}^T, \tag{7.7}$$

where the eigenvalue matrix is given by

$$\boldsymbol{\Lambda}_x = \mathrm{diag}(\lambda_{x,1}, \ldots, \lambda_{x,P}), \tag{7.8}$$

and $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_P]$ is the matrix of orthonormal eigenvectors (i.e. $\mathbf{U}^T\mathbf{U} = \mathbf{I}$). Without loss of generality it is assumed throughout this chapter that $\lambda_{x,1} \geq \lambda_{x,2} \geq \ldots \geq \lambda_{x,P}$.

Since the noise is white, the eigenvectors $\mathbf{u}_i$ are also the eigenvectors of $\mathbf{R}_s$ and the eigenvalues $\lambda_{x,i}$ are given by

$$\lambda_{x,i} = \begin{cases} \lambda_{s,i} + \sigma^2 & \text{for } i = 1, \ldots, Q \\ \sigma^2 & \text{for } i = Q+1, \ldots, P \end{cases}, \tag{7.9}$$

where $\lambda_{s,i}$, for $i = 1, \ldots, Q$, are the Q eigenvalues of $\mathbf{R}_s$ which are strictly greater than zero.

Accordingly, $\mathbf{U}$ can be partitioned as $\mathbf{U} = [\mathbf{U}_1 \quad \mathbf{U}_2]$ where $\mathbf{U}_1 = [\mathbf{u}_1, \ldots, \mathbf{u}_Q]$ and $\mathbf{U}_2 = [\mathbf{u}_{Q+1}, \ldots, \mathbf{u}_P]$. Since $\mathbf{U}$ is orthogonal, we have

$$\mathbf{U}_1\mathbf{U}_1^T + \mathbf{U}_2\mathbf{U}_2^T = \mathbf{I}. \tag{7.10}$$

Indeed, $\mathbf{U}_1\mathbf{U}_1^T$ is the orthogonal projector onto the subspace spanned by the columns of $\mathbf{U}_1$ which is the same as $\mathcal{R}\{\mathbf{A}\}$. This subspace is called the *signal subspace*. $\mathbf{U}_2\mathbf{U}_2^T$, on the other hand, is the orthogonal projector onto the complementary orthogonal subspace called the *noise subspace*. It should be noted, however, that the noise actually fills the entire space and is not just confined to the noise subspace.

7.3 Linear Signal Estimation

With the signal and noise modeling assumptions described above, a linear filter $\mathbf{H}$ can be designed to estimate the desired speech vector $\mathbf{s}$ from the noisy observation $\mathbf{x}$ in (7.4). Let $\hat{\mathbf{s}}$ denote the estimate of $\mathbf{s}$ at the filter output,

$$\hat{\mathbf{s}} = \mathbf{H}\mathbf{x} = \mathbf{H}\mathbf{s} + \mathbf{H}\mathbf{w}. \tag{7.11}$$

The linear estimator $\mathbf{H}$ can be calculated in different ways depending on the optimization criteria employed. We next present some of the most popular estimators proposed in the literature.

7.3.1 Least-Squares Estimator

A straightforward and simple solution to the estimation problem is to use the *least-squares* (LS) estimate. It is obtained by minimizing the squared fitting error between the observation vector $\mathbf{x}$ and the linear low order speech model of (7.1),

$$\hat{\mathbf{s}} = \mathbf{A}\mathbf{c}_0, \qquad \mathbf{c}_0 = \arg\min_{\mathbf{c}} ||\mathbf{x} - \mathbf{A}\mathbf{c}||^2. \tag{7.12}$$

Setting the gradient of the above cost function to zero, the LS solution is obtained as

$$\hat{\mathbf{s}} = \mathbf{H}\mathbf{x} = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{x}. \tag{7.13}$$

It can be seen that $\hat{\mathbf{s}}$ is the projection of the observation vector onto the signal subspace spanned by the columns of $\mathbf{A}$ as discussed earlier. Hence $\mathbf{H}$ can alternatively be written in terms of the eigenvalue decomposition of $\mathbf{R}_s$ as follows

$$\mathbf{H} = \mathbf{U}_1\mathbf{U}_1^T. \tag{7.14}$$

This estimator does not result in any signal distortion (provided that the subspace dimension Q was correctly estimated) but has the highest possible residual noise since it allows the noise components in the signal subspace to remain intact. The SNR gain obtained with this estimator is in the order of P/Q [8].

Other LS estimators rely on approximating the speech model matrix $\mathbf{A}$ (e.g. [22], [30]), which is usually a difficult problem. Unlike these methods, (7.14) shows that such a model is not required and the desired signal can be simply estimated using the eigenvalue decomposition of the noisy covariance matrix.

7.3.2 The Linear Minimum Mean Squared Error Estimator

The linear minimum mean squared error (LMMSE) estimator is obtained by minimizing the residual error energy as follows

$$\min_{\mathbf{H}} \; E\{||\mathbf{r}||^2\}, \tag{7.15}$$

where the residual error signal is defined as

$$\mathbf{r} = \hat{\mathbf{s}} - \mathbf{s} = \mathbf{H}\mathbf{x} - \mathbf{s}. \tag{7.16}$$

The solution to this classical problem is given by the Wiener filtering matrix

$$\mathbf{H} = \mathbf{R}_s(\mathbf{R}_s + \sigma^2\mathbf{I})^{-1}. \tag{7.17}$$

Rewriting (7.17) in terms of the EVD of $\mathbf{R}_s$ and recalling that $\lambda_{s,i} = 0$ for $i > Q$, we get

$$\mathbf{H} = \mathbf{U}\boldsymbol{\Lambda}_s(\boldsymbol{\Lambda}_s + \sigma^2\mathbf{I})^{-1}\mathbf{U}^T = \mathbf{U}_1\mathbf{G}\mathbf{U}_1^T, \tag{7.18}$$

where $\mathbf{G}$ is a $Q\times Q$ diagonal gain matrix with entries

$$g_i = \frac{\lambda_{s,i}}{\lambda_{s,i} + \sigma^2} \quad \text{for} \quad i = 1, \ldots, Q. \tag{7.19}$$

The matrix $\mathbf{U}_1^T$ is in fact the Karhunen-Loeve transform[2] (KLT) and its effect on the noisy signal vector $\mathbf{x}$ is to calculate the coefficients of its projection onto the signal subspace. These coefficients have the property of being uncorrelated so that they can be processed independently using a diagonal gain matrix according to (7.18). The enhanced signal vector is finally reconstructed in the time domain using the matrix $\mathbf{U}_1$, the inverse KLT.

The gain function in (7.19) is actually analogous to the frequency-domain Wiener filter [5]. For this reason, this gain function will be referred to as the Wiener gain function.

7.3.3 The Time-Domain Constrained Estimator

Instead of minimizing the total residual error energy, the time-domain constrained estimator (TDC), proposed in [8], is obtained by minimizing the signal distortion subject to forcing the residual noise energy to be below some predefined threshold. This constrained optimization approach aims to offer control over the trade-off between signal distortion and residual noise level. This can be achieved by decomposing the residual error signal as follows

$$\mathbf{r} = \hat{\mathbf{s}} - \mathbf{s} = (\mathbf{H} - \mathbf{I})\mathbf{s} + \mathbf{H}\mathbf{w}. \tag{7.20}$$

Accordingly, define the signal distortion as

$$\mathbf{r}_s \triangleq (\mathbf{H} - \mathbf{I})\mathbf{s}, \tag{7.21}$$

and

$$\mathbf{r}_w \triangleq \mathbf{H}\mathbf{w} \tag{7.22}$$

as residual noise. The filter $\mathbf{H}$ is then obtained as the solution to the following optimization problem

$$\min_{\mathbf{H}} \; E\{||\mathbf{r}_s||^2\} \quad \text{subject to} \quad \frac{1}{P}E\{||\mathbf{r}_w||^2\} \leq \alpha\sigma^2, \tag{7.23}$$

where $0 \leq \alpha \leq 1$.

[2] To be precise, the KLT is the matrix $\mathbf{U}^T$. However since all eigenvectors in $\mathbf{U}_2^T$ will have, according to (7.18), a weight of zero, $\mathbf{U}_1^T$ can indeed be considered to be the KLT.

Using the Kuhn-Tucker necessary conditions for the above constrained minimization problem [28], the optimum filter $\mathbf{H}$ is a feasible stationary point if the gradient of the Lagrangian

$$L(H, \mu) = E\{||\mathbf{r}_s||^2\} + \mu(E\{||\mathbf{r}_w||^2\} - \alpha P\sigma^2) \tag{7.24}$$

is equal to zero and

$$\mu(E\{||\mathbf{r}_w||^2\} - \alpha P\sigma^2) = 0 \quad \text{for } \mu \geq 0. \tag{7.25}$$

The solution is then given by [8]

$$\mathbf{H} = \mathbf{R}_s(\mathbf{R}_s + \mu\sigma^2\mathbf{I})^{-1}, \tag{7.26}$$

where μ is the Lagrange multiplier. The latter can be shown to satisfy the following relationship with α [8]

$$\alpha = \frac{1}{P}\text{tr}\{\mathbf{R}_s^2(\mathbf{R}_s + \mu\sigma^2\mathbf{I})^{-2}\}. \tag{7.27}$$

In terms of the EVD of $\mathbf{R}_s$, the filter $\mathbf{H}$ (7.26) can be written as

$$\mathbf{H} = \mathbf{U}_1\mathbf{G}\mathbf{U}_1^T, \tag{7.28}$$

where $\mathbf{G}$ is a $Q \times Q$ diagonal gain matrix with entries

$$g_i = \frac{\lambda_{s,i}}{\lambda_{s,i} + \mu\sigma^2} \quad \text{for} \quad i = 1, \ldots, Q. \tag{7.29}$$

Note that (7.29) only differs from (7.19) by the Lagrange multiplier μ, and that both are indeed the same when $\mu = 1$. Equation (7.29) can then be interpreted as a Wiener filter with a variable noise level (controlled by μ).

Equation (7.27) can also be simplified and we can find that the Lagrange multiplier satisfies

$$\alpha = \frac{1}{P}\sum_{i=1}^{Q}\left(\frac{\lambda_{s,i}}{\lambda_{s,i} + \mu\sigma^2}\right)^2. \tag{7.30}$$

7.3.4 The Spectral-Domain Constrained Estimator

The second estimator proposed in [8], is the spectral domain constrained approach (SDC), where the enhancement filter $\mathbf{H}$ is the solution to the following optimization problem:

$$\min_{\mathbf{H}} E\{||\mathbf{r}_s||^2\} \quad \text{subject to} \quad \begin{cases} E\{|\mathbf{u}_i^T\mathbf{r}_w|^2\} \leq \alpha_i\sigma^2 & \text{for } 1 \leq i \leq Q \\ E\{|\mathbf{u}_i^T\mathbf{r}_w|^2\} = 0 & \text{for } Q < i \leq P \end{cases}. \tag{7.31}$$

The goal here is to minimize the signal distortion subject to keeping every spectral component of the residual noise, within the signal subspace, below some predefined threshold. Those spectral components in the noise subspace, on the other hand, are set to zero. Again, using the Kuhn-Tucker necessary conditions, the solution to this problem is given by [8]

$$\mathbf{H} = \mathbf{U}_1 \mathbf{G} \mathbf{U}_1^T, \tag{7.32}$$

where the entries of the gain matrix $\mathbf{G} = \mathrm{diag}\{g_1, \ldots, g_Q\}$ are given by

$$g_i = \sqrt{\alpha_i} \quad \text{for} \quad i = 1, \ldots, Q. \tag{7.33}$$

In theory, the gain matrix entries in (7.33) can be independent of the input data. However, exploiting information available from the signal and noise statistics may lead to a better choice for the gain coefficients. To this end, a commonly used quantity is the SNR on the i^{th} spectral component defined as

$$\gamma_i = \lambda_{s,i}/\sigma^2. \tag{7.34}$$

Ideally, one would like to turn off spectral components with very low SNR and keep those components with very high SNR unchanged. This may be achieved by letting $g_i = f(\gamma_i)$, where $f(.)$ is an increasing function satisfying

$$\begin{aligned} f(0^+) &\to 0, \quad \text{and} \\ f(+\infty) &\to 1. \end{aligned} \tag{7.35}$$

A possible choice of f is

$$f(\gamma) = \frac{\gamma}{\gamma + \mu}, \tag{7.36}$$

leading to the TDC solution given in (7.29) (the Wiener gain function with variable noise level). A second choice is the exponential function

$$f(\gamma) = \exp(-\nu/\gamma), \tag{7.37}$$

which gives

$$g_i = \sqrt{\alpha_i} = e^{-\nu\sigma^2/\lambda_{s,i}}, \quad i = 1, \ldots, Q. \tag{7.38}$$

This gain function is found to have more noise suppression capabilities. Besides, for $\nu = 1$, the first order Taylor development of g_i^{-1} [eq. (7.38)] with respect to $\sigma^2/\lambda_{s,i}$ is the inverse of the Wiener gain function in (7.29) with $\mu = 1$, pointing to the equivalence of these approaches at high SNR.

Figure 7.1 shows a plot of these gain functions for comparison. Note that the LS estimator discussed in Section 7.3.1 is also a special case of the SDC with $g_i = 1$ for all $i = 1, \ldots, Q$. The gain functions associated with the four different estimators presented in this section are summarized in Table 7.1.

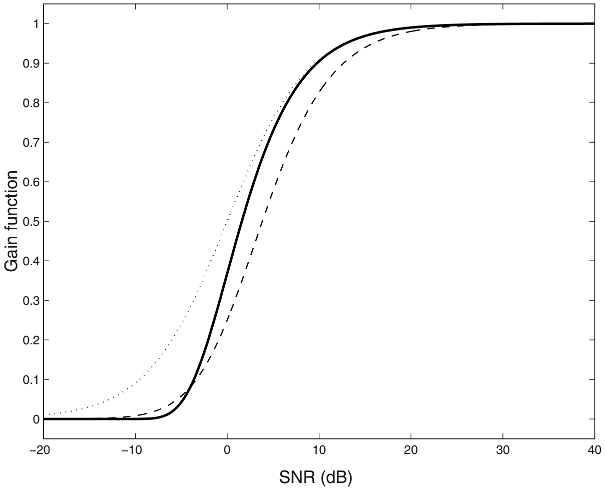


Fig. 7.1. The gain function $f(\gamma)$: exponential (7.37) with $\nu = 1$ (thick), Wiener (7.36) with $\mu = 1$ (dotted) and with $\mu = 2$ (dashed).

Table 7.1 The gain functions corresponding to different linear signal estimators.

Signal Estimator	Gain Function g_i
LS	1
LMMSE	$\frac{\lambda_{s_i}}{\lambda_{s_i}+\sigma^2}$
TDC	$\frac{\lambda_{s_i}}{\lambda_{s_i}+\mu\sigma^2}$
SDC	$\sqrt{\alpha_i}$

7.4 Handling Colored Noise

One problem with the signal subspace approach is that it is based on the white noise assumption. However, almost all common noise types encountered in real life are colored. Therefore, extra techniques should be included with the signal subspace method to handle the colored noise case for it to be useful in practice. Fortunately, several such techniques have been proposed in the literature with satisfying results.

7.4.1 Prewhitening

In [8], prewhitening is proposed as a remedy to the colored noise issue. It consists of multiplying the noisy input vector $\mathbf{x}$ by $\mathbf{R}_w^{-\frac{1}{2}}$, the inverse of the square root of the colored noise covariance matrix $\mathbf{R}_w = E\{\mathbf{w}\mathbf{w}^T\}$. The prewhitened signal is obtained as

$$\breve{\mathbf{x}} = \mathbf{R}_w^{-\frac{1}{2}}\mathbf{x} = \mathbf{R}_w^{-\frac{1}{2}}\mathbf{s} + \mathbf{R}_w^{-\frac{1}{2}}\mathbf{w} = \breve{\mathbf{s}} + \breve{\mathbf{w}}. \tag{7.39}$$

It can be verified that $E\{\breve{\mathbf{w}}\breve{\mathbf{w}}^T\} = \mathbf{I}$. Hence $\breve{\mathbf{w}}$, the prewhitened noise component, is now white with variance equal to one.

The EVD obtained from the signal $\breve{\mathbf{x}}$ can then be used instead of the EVD obtained from $\mathbf{x}$ to calculate a filter $\breve{\mathbf{H}}$ using any of the linear estimators presented earlier. However, since the desired speech signal is also affected, the inverse of the prewhitening matrix, i.e. $\mathbf{R}_w^{\frac{1}{2}}$, is applied as a postfilter to undo the effect of prewhitening. This is called dewhitening. Accordingly, the overall effective enhancing filter becomes

$$\mathbf{H} = \mathbf{R}_w^{\frac{1}{2}}\breve{\mathbf{H}}\mathbf{R}_w^{-\frac{1}{2}}. \tag{7.40}$$

The prewhitening and dewhitening matrices can be obtained using the Cholesky decomposition of the noise covariance matrix or more safely (in case the latter is not invertible or near singular) using its eigenvalue decomposition. Consider the EVD $\mathbf{R}_w = \mathbf{U}_w\boldsymbol{\Lambda}_w\mathbf{U}_w^T = \mathbf{U}_{w,1}\boldsymbol{\Lambda}_{w,1}\mathbf{U}_{w,1}^T$, where $\boldsymbol{\Lambda}_{w,1}$ contains only non-zero eigenvalues and $\mathbf{U}_{w,1}$ has the corresponding eigenvectors as its columns, then[3]

$$\mathbf{R}_w^{\frac{1}{2}} = \mathbf{U}_{w,1}\boldsymbol{\Lambda}_{w,1}^{\frac{1}{2}}\mathbf{U}_{w,1}^T, \tag{7.41}$$

$$\mathbf{R}_w^{-\frac{1}{2}} = \mathbf{U}_{w,1}\boldsymbol{\Lambda}_{w,1}^{-\frac{1}{2}}\mathbf{U}_{w,1}^T. \tag{7.42}$$

We shall refer to this method as the preWhitening based signal subspace method (PWSS).

In [16], prewhitening is accomplished using a filter designed from the coefficients of an autoregressive model of the noise.

7.4.2 The Generalized Eigenvalue Decomposition Method

Prewhitening can alternatively be realized as an integral part of the subspace decomposition using the generalized EVD [15] or the generalized SVD [23]. The idea is to find a matrix that would diagonalize both $\mathbf{R}_s$ and $\mathbf{R}_w$ simultaneously. Such a matrix would satisfy [15],

$$\mathbf{V}^T\mathbf{R}_s\mathbf{V} = \boldsymbol{\Lambda}, \tag{7.43}$$

$$\mathbf{V}^T\mathbf{R}_w\mathbf{V} = \mathbf{I}, \tag{7.44}$$

where $\mathbf{V}$ and $\boldsymbol{\Lambda}$ are the eigenvector matrix and the eigenvalue matrix of $\mathbf{R}_w^{-1}\mathbf{R}_s$, respectively. Hence the optimal filter (7.32) can be modified as follows

$$\mathbf{H} = \mathbf{V}^{-T}\mathbf{G}\mathbf{V}^T. \tag{7.45}$$

[3] Actually, $\mathbf{R}_w^{-\frac{1}{2}}$ is rather the pseudo-inverse of $\mathbf{R}_w^{\frac{1}{2}}$.

It should be noted that $\mathbf{V}^T$ is no longer the KLT corresponding to $\mathbf{R}_s$ and that $\mathbf{V}$ is not orthogonal. The gain matrix $\mathbf{G}$ is chosen as discussed earlier to satisfy the desired optimization criterion. The noise variance, however, should now be set to one, that is $\sigma^2 = 1$.

7.4.3 The Rayleigh Quotient Method

As discussed earlier, the prewhitening technique consists of using $\breve{\mathbf{x}}$, in (7.39), instead of $\mathbf{x}$ for the filter design. Therefore, the filter will shape the noise spectrum according to the spectrum of $\breve{\mathbf{s}}$, the modified speech vector, rather than $\mathbf{s}$. Hence the filter in equation (7.40) is not necessarily optimal in the sense of its noise shaping capabilities [29].

Alternatively, another method to handle colored noise, consists of replacing the constant noise variance in (7.34) by the noise energy in the direction of the i^{th} eigenvector, given by

$$\xi_i = \mathbf{u}_i^T \mathbf{R}_w \mathbf{u}_i, \tag{7.46}$$

which is the Rayleigh quotient associated with $\mathbf{u}_i$ and $\mathbf{R}_w$ for $i = 1, \ldots, Q$. Here $\mathbf{u}_i$ is the i^{th} eigenvector of the clean speech covariance matrix estimate, $\hat{\mathbf{R}}_s$, with corresponding eigenvalue $\lambda_{s,i}$. $\hat{\mathbf{R}}_s$ is estimated from the noisy covariance matrix as follows

$$\hat{\mathbf{R}}_s = \mathbf{R}_x - \mathbf{R}_w. \tag{7.47}$$

$\hat{\mathbf{R}}_s$, so obtained, is no longer guaranteed to be positive definite and may have negative eigenvalues. Hence, the rank Q is chosen as the number of strictly positive eigenvalues $\lambda_{s,i}$.

The gain function is calculated, for example, using the exponential function (7.38), in the following way,

$$g_i = f(\lambda_{s,i}/\xi_i) = e^{-\nu \xi_i / \lambda_{s,i}} \quad \text{for } i = 1, \ldots, Q. \tag{7.48}$$

RQSS was the basis for the methods described in [29] and [31]. In the latter it was used in conjunction with a subspace tracking technique in order to reduce the computational load. In [29], further processing is added to RQSS by classifying the speech frames as *speech dominated* or *noise dominated.* The procedure described above is applied during speech dominated frames. During noise dominated frames on the other hand, the EVD of the noise covariance matrix is used instead of that of the estimated clean speech covariance matrix. In both works [31], [29], RQSS was reported to be superior to the prewhitening technique in the sense that better noise shaping[4] is achieved.

[4] Noise shaping refers to giving the noise spectrum a shape which follows that of the desired speech signal, hence masking it.

7.5 A Filterbank Interpretation

By design, the SSA is always viewed and analyzed from a linear algebra perspective. However since our understanding of speech signals is best in terms of its frequency spectrum, it seems beneficial if we can provide a frequency-domain interpretation of the SSA in order to better understand its behaviour. A filterbank interpretation has been given for example in [10] and [24] yielding a modified SSA based method (with an SVD implementation).

In this section, we use a similar filterbank approach. To this end, we first present a frequency to eigendomain transformation which facilitates the filterbank interpretation.

7.5.1 The Frequency to Eigendomain Transformation

Consider a zero mean stationary signal $x(n)$ with covariance matrix $\mathbf{R}_x = E\{\mathbf{x}_n\mathbf{x}_n^T\}$ where $\mathbf{x}_n = [x(n), \ldots, x(n-P+1)]$. Let λ_i be the i^{th} eigenvalue of $\mathbf{R}_x$ and $\mathbf{u}_i = [u_i(0), \ldots, u_i(P-1)]^T$ its corresponding unit norm eigenvector. It can be shown that λ_i can be written in terms of $\Phi(\omega)$, the power spectral density (PSD) of $x(n)$, in the following way [12]

$$\lambda_i = \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi(\omega)|V_i(\omega)|^2 d\omega \quad \text{for} \quad i = 1, \ldots, Q, \tag{7.49}$$

where

$$V_i(\omega) = \sum_{p=0}^{P-1} u_i(p)e^{-j\omega p} \tag{7.50}$$

is the discrete-time Fourier transform (DTFT) of the entries $u_i(p)$ of the eigenvector $\mathbf{u}_i$. Equation (7.49) will be called the frequency to eigendomain transformation (FET) [18], [21] and will serve as a bridge between the eigendomain and the more intuitive frequency domain.

This relationship was used in [18], [21] to incorporate the human hearing properties in the SSA by allowing to map the masking threshold estimated in the frequency domain to the eigendomain.

7.5.2 The Eigen Filterbank

Consider a filter bank with P analysis filters with frequency responses $V_i(\omega)$ for $i = 1, \ldots, P$ as shown in Fig. 7.2. That is, every filter has a finite impulse response $u_i(p)$ for $p = 0, \ldots, P-1$. Now let $x(n)$, a random process with PSD $\Phi(\omega)$, be the input to this filterbank. Thus, the PSD $\Phi_i(\omega)$ of the output $x_i(n)$ of the i^{th} filter is given by [13]

$$\Phi_i(\omega) = \Phi(\omega)|V_i(\omega)|^2. \tag{7.51}$$

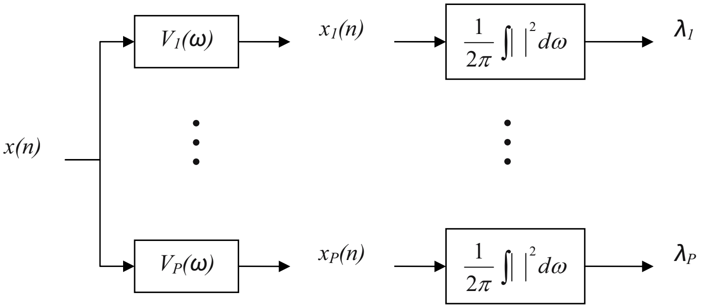


Fig. 7.2. A block diagram of the signal subspace filterbank interpretation.

Hence, Using the FET (7.49), it can be seen that the total energy at the output of the i^{th} filter is actually the i^{th} eigenvalue λ_i,

$$\lambda_i = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Phi_i(\omega) d\omega. \tag{7.52}$$

Therefore, the SSA actually consists of dividing the input signal into several "subbands." In every band, a gain function depending on the average SNR in that particular band is applied and then the whole signal is resynthesized in the time domain.

This filterbank, however, is different from other common ones, such as the DFT filter banks, in that instead of having the passbands of the analysis filters uniformly distributed over the frequency range of interest, the "eigen" analysis filters are signal dependent. In Fig. 7.3, these filters are shown for the case of a vowel (/a/). The figures show the PSD of the input signal (thick line) together with the magnitude squared of the frequency response of the $P = 32$ eigen analysis filters, $|V_i(\omega)|^2$.

It can be seen in Fig. 7.3 that the first four filters correspond to the first formant whereas the next two filters correspond to the second formant. The third formant (also important for intelligibility) can be found in the passbands of the 12^{th}, 13^{th} and 14^{th} filters.

It is widely known that the output of the single-channel frequency domain methods usually suffer from spectral peaks randomly distributed over time and frequency. This processing artifact, commonly referred to as musical noise, mostly occurs due to poor estimation of the speech and noise statistics resulting in "random" fluctuations in the suppression filters both over time and frequency. Therefore, the proposed remedy to this problem usually consists of trying to smooth the filter coefficients. In the SSA, and using the filterbank interpretation, it can be readily noted that this approach accomplishes such smoothing by obtaining the average SNR within every

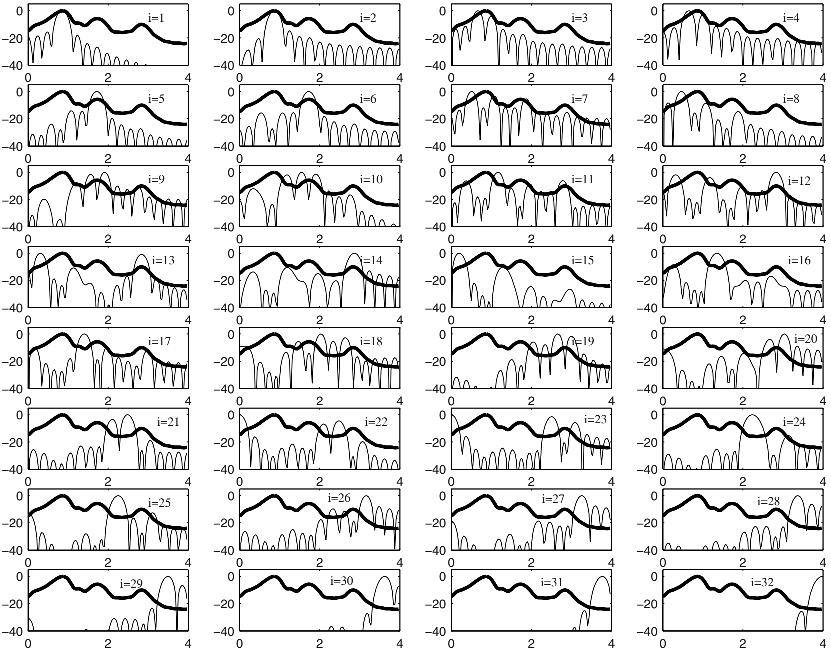


Fig. 7.3. The magnitude squared of the i^{th} eigenfilter $V_i(\omega)$ for the vowel /a/ in the word "cats," for $i = 1, \ldots, P$. The thick line shows the PSD of the speech signal.

passband of the eigen analysis filters. The reduction of the musical noise, commonly reported for SSA based methods, may then be attributed to this phenomenon.

In addition, and since the passbands of the analysis filters are usually located around the speech formants, the residual noise spectrum will eventually be shaped according to the spectrum of the desired speech signal. This shaping entails a masking effect which would further suppress the noise, from a perceptual standpoint, with a relatively lower signal distortion.

This filterbank interpretation has been exploited in [18] to provide further insight into some signal subspace results and observations reported in the literature, particularly on the effect of noise.

7.6 Implementation Issues

In this section, we address the implementation issues of the SSA in more details. We first present the original implementation scheme where the white noise assumption is retained[5]. After that, we discuss the effect of the different involved parameters on the overall performance.

To implement the SSA, length-P speech vectors are input at a rate of $P/2$ samples. To preserve the whiteness of the noise, only a rectangular window is used in the analysis phase. Each of these vectors, $\mathbf{x}_n = [x(n), \ldots, x(n - P + 1)]^T$, is multiplied by an enhancing linear filter $\mathbf{H}$.

Since the speech signal is not stationary over the whole utterance, the filter $\mathbf{H}$ should be updated as a new vector comes in. This is done by estimating the noisy speech covariance matrix $\mathbf{R}_x$ and calculating its EVD. Using (7.9), the eigenvalues of the clean covariance matrix are estimated as follows[6]

$$\lambda_{s,i} = \max\{\lambda_{x,i} - \sigma^2, 0\}, \tag{7.53}$$

where $\lambda_{x,i}$ is the i^{th} eigenvalue of $\mathbf{R}_x$ and σ^2 is the noise variance estimated during non-speech activity periods. The so obtained eigenvectors and eigenvalues are used with one of the signal estimators discussed in Section 7.3 to compute the subspace filter.

Finally, to synthesize the signal, the 50% overlapping enhanced vectors are Hanning windowed and combined using the overlap-add approach [5].

7.6.1 Estimating the Covariance Matrix

The linear signal estimators described in Section 7.3 assume exact knowledge of the second order statistics of the noisy signal and noise processes. In practice however this information needs to be estimated from the available noisy observation vectors, $\mathbf{x}_n = [x(n), \ldots, x(n - P + 1)]^T$.

An estimate $\mathbf{R}_{x,n}$ of the covariance matrix of $\mathbf{x}_n$ can be obtained from the empirical covariance of $2N + 1$ non-overlapping noisy vectors in the neighborhood of $\mathbf{x}_n$. To this end, we assume that conditions of stationarity and ergodicity are satisfied for a data window of length $(2N + 1)P$. For speech, these conditions are considered to be satisfied for a window which is around 30 msec long [5]. The estimate $\mathbf{R}_{x,n}$ can then be obtained as follows

$$\begin{aligned}\mathbf{R}_{x,n} &= \frac{1}{2PN} \sum_{i=-NP+1}^{i=NP} \mathbf{x}_{n+i}\mathbf{x}_{n+i}^T \\ &= \mathbf{X}_n\mathbf{X}_n^T,\end{aligned} \tag{7.54}$$

where $\mathbf{X}_n$ is a $P \times 2PN$ data matrix given by

$$\mathbf{X}_n = \frac{1}{\sqrt{2PN}}[\mathbf{x}_{n-NP+1}, \ldots, \mathbf{x}_{n-1}, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+NP}]. \tag{7.55}$$

[5] Extension to the colored noise case using PWSS or RQSS can be approached in a similar way.

[6] For simplicity of notation, we avoid the use of a hat to denote estimated quantities. Such notation will only be used when it is necessary to avoid ambiguity.

The signal subspace can now be calculated either by EVD of the covariance estimate $\mathbf{R}_{x,n}$ or via the SVD of the data matrix $\mathbf{X}_n$. Since it does not require the explicit computation of the covariance matrix, the SVD needs less computations in addition to being more stable in the case of an ill-conditioned data matrix [9]. However, the SVD does not allow the use of more structured covariance matrices. Namely, it was observed that a Toeplitz covariance matrix would better represent speech signals and would yield a better noise reduction performance [8].

To derive such a Toeplitz covariance matrix, the biased autocorrelation function estimator obtained from $L = 2NP$ observation samples is calculated as follows

$$r_x(p) = \frac{1}{L} \sum_{i=-NP+1}^{NP-p} x(n+i)x(n+i+p) \quad \text{for} \quad p = 0, \ldots, P-1. \tag{7.56}$$

The Toeplitz covariance is then formed as follows

$$\mathbf{R}_x = \begin{bmatrix} r_x(0) & r_x(1) & \cdots & r_x(p-1) \\ r_x(1) & r_x(0) & \cdots & r_x(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r_x(p-1) & r_x(p-2) & \cdots & r_x(0) \end{bmatrix}. \tag{7.57}$$

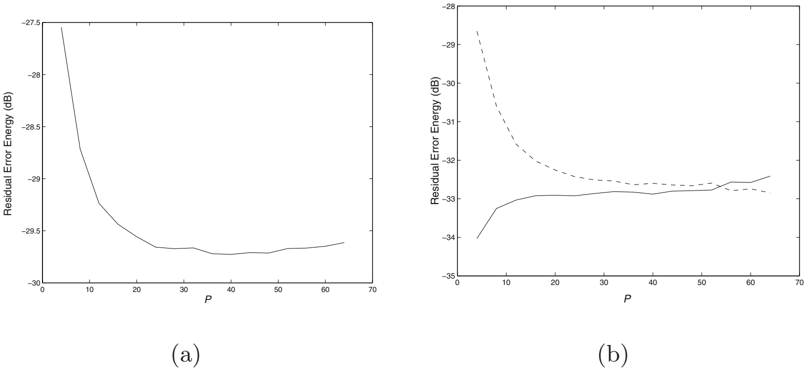
The EVD of this matrix is calculated and is used to compute the signal subspace filter as described earlier.

7.6.2 Parameter Analysis

The advantage of the SSA is that it can offer a better compromise between signal distortion and the level of the residual noise. To achieve that, proper tuning of the system parameters is required. We discuss, next, some of these parameters in order to assist the reader in selecting the appropriate values. The discussion provided will be based on the case of 8 KHz sampling frequency. The conventional SSA approach of [8] has been used and the noisy signal has been obtained by artificially adding a white noise signal to the clean speech.

The Window Length *L*. The choice of the window length $L = 2NP$ is a crucial design decision. To obtain better covariance estimates, L should be as long as possible. However, in the current application, we are limited by the non-stationarity of the speech signal.

Simulation experiments show that for shorter windows (or frames), the covariance estimates are not reliable resulting in a higher level of the musical noise. Longer frames, on the other hand, considerably reduce the level of the residual noise at the cost of more signal distortion (due to the violation of the



(a)

(b)

Fig. 7.4. (a) The residual error signal and (b) the signal distortion energy (dashed) and residual noise energy (continuous), as a function of the model order P.

stationarity assumption). Such distortion will be more evident at unvoiced instances of speech because they are generally shorter in duration and weaker in energy [18]. Satisfactory results can be obtained with a window of length 30-40 msec, i.e. at 8 KHz sampling rate, so we can choose $L = 256$.

The Model Order *P*. Another important parameter is the model order P. Figure 7.4 (a) shows the effect of P on the total residual error energy $E\{\|\mathbf{r}\|^2\}$ while in Fig.7.4 (b) the residual noise energy $E\{\|\mathbf{r}_w\|^2\}$ and the signal distortion energy $E\{\|\mathbf{r}_s\|^2\}$ are shown separately. At low values of P, the SSA exhibits high signal distortion due to the fact that not enough correlation coefficients are available to accurately estimate the signal subspace. This results in the loss of signal components important for intelligibility. The residual noise, however, is low because, for the same reason, many of its components would have been forced to zero [18].

The figure also shows that the higher P is, the lower the residual error energy. The latter converges to a minimum value for $P > 30$ suggesting that no more gain in performance can be achieved by further increasing the value of P. Moreover, higher values of P may even increase the residual error signal energy [as can be seen in Fig. 7.4 (a)] because not enough samples are available for estimating the autocorrelation function at high lags (these results were obtained for a fixed window length $L = 256$).

Besides, increasing P would drastically increase the computational load. This is because the SSA is based on the exact EVD of a $P \times P$ covariance matrix which requires a complexity in $\mathcal{O}(P^3)$.

Generally, P is chosen somewhere between 20 and 40.

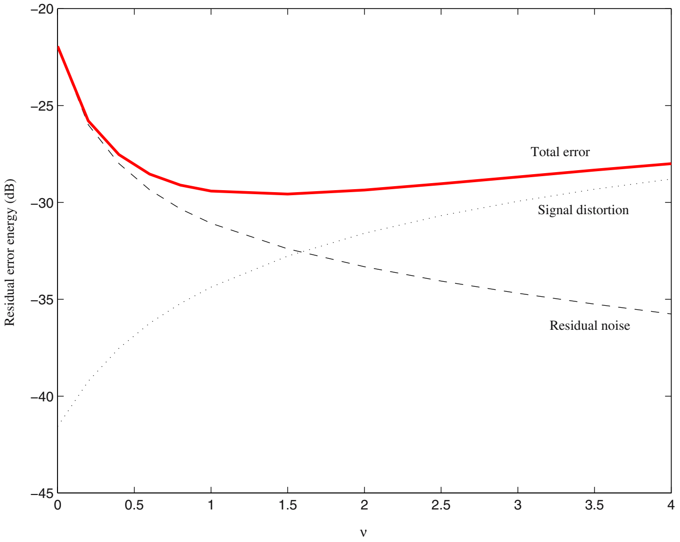


Fig. 7.5. The total residual error signal energy (thick), the signal distortion energy (dotted), and the residual noise energy (dashed), as a function of ν.

The Control Parameter ν. In the exponential gain function, the parameter ν serves as a free parameter that controls the trade off between the residual noise level and the signal distortion, defined in (7.20). Figure 7.5 shows a plot of the signal distortion energy $E\{\|\mathbf{r}_s\|^2\}$, the residual noise energy $E\{\|\mathbf{r}_w\|^2\}$, and the total residual error energy $E\{\|\mathbf{r}\|^2\}$ as a function of the parameter ν in the exponential gain function (7.38).

It can be seen that as ν increases, the signal distortion increases and the residual noise level decreases. Consequently, the minimum values for the total residual error energy is obtained when ν is around 1.5. Listening tests however show that $\nu = 2$ is a better choice from a perceptual perspective. This can be explained by the fact that humans prefer a lower noise level at the expense of a slightly more signal distortion. Note that since the noise and the speech signal are uncorrelated, $E\{\|\mathbf{r}\|^2\} = E\{\|\mathbf{r}_s\|^2\} + E\{\|\mathbf{r}_w\|^2\}$.

7.7 Fast Subspace Estimation Techniques

The disadvantage of the signal subspace approach is the relatively high computational load mainly due to the expensive eigenvalue decomposition. This drawback made the engineers and scientists working on speech enhancement rather reluctant to use the SSA in practice. However, with the impressive development in the DSP technology and the continuous increase in the available processing speed and computational power, it is believed that the SSA can eventually compete with the widely used frequency-domain methods.

The complexity issue has however been addressed in the literature and several approaches have been proposed to tackle this problem. These techniques include fast EVD and subspace tracking methods. Moreover, attempts to approximate the KLT using the discrete-cosine transform (DCT) have also been recently applied to speech enhancement [17,36].

7.7.1 Fast Eigenvalue Decomposition Methods

One solution to the complexity issue is to replace the exact EVD by alternative fast methods which are capable to reduce the complexity from $\mathcal{O}(P^3)$ to $\mathcal{O}(P^2Q)$ operations per sample where Q is the rank of the matrix. For example in [39], the structure of the covariance matrix is exploited and the so-called Lanczos algorithm [9] is used to reduce the required computations by only calculating the Q principal eigenvectors (and their corresponding eigenvalues) which span the signal subspace.

For instance, for a speech signal sampled at 8 KHz, the effective rank of the covariance matrix of a voiced frame (which constitute the majority of a speech sentence) would be around 10 to 15, for a $P = 32$ model. Therefore, substantial computational savings could be achieved by approximating the exact EVD using such fast methods.

7.7.2 Subspace Tracking Methods

Another possible approach is to make use of the so called EVD or subspace tracking algorithms to efficiently update the required EVD information. Rather than trying to calculate the EVD from scratch, these algorithms seek to recursively update an already existing EVD as more data becomes available. In [31], a fast implementation of the SSA for speech enhancement has been developed based on the projection approximation subspace tracking (with deflation) method (PASTd) which reduces the complexity to $\mathcal{O}(PQ)$ per sample using the recursive least-squares (RLS) algorithm [40]. The PASTd algorithm does not guarantee the orthogonality of the eigenvectors, this is why it was reported in [24] that better results can be achieved using the fast orthogonal iteration (FOI) based algorithm [34]. In [11], a rank-revealing ULV decomposition [33] has been also used for speech enhancement. Several other $\mathcal{O}(PQ)$ subspace tracking algorithms have been developed in recent years which guarantee the orthogonality of the eigenvectors (see for example [4] and references therein) but their applicability to the speech enhancement problem has yet to be tested.

Unfortunately, there seem to be some problems associated with applying subspace trackers to speech enhancement. The reason is that these methods are based on estimating the covariance matrix using a sliding exponential window[7]. During our experiments however, we noticed that shifting the win-

[7] Note that a few subspace trackers based on sliding rectangular windows have been proposed (e.g. [1]) but they usually require more computations .

dow at a high rate added reverberation to the enhanced speech signal. This result, which has also been confirmed in [24], suggests that a sliding exponential window may be inadequate for speech enhancement applications.

Subspace trackers can then only be applied if the EVD update scheme is carried out on a sample by sample basis[8], which does not lead to the apparent great computational savings. In fact, an exact EVD has a computational cost of $\mathcal{O}(P^3)$, but since it is only calculated every $P/2$ samples, this complexity is reduced to $\mathcal{O}(P^2)$ per sample[9]. A subspace tracker on the other hand can at best achieve a $\mathcal{O}(PQ)$ per sample complexity.

7.7.3 The Frame Based EVD (FBEVD) Method

In [18], [21] a novel implementation scheme which helps to overcome the computational issue of the SSA has been presented. The so called Frame based EVD (FBEVD) method is a modification of the approach used in [8] and described in Section 7.6. The idea is based on exploiting the stationarity assumption of the speech signal. This assumption is already exploited in [8] to calculate the covariance matrix required for the subspace filter design, but it is applied here in a different manner.

Let the speech signal $x(n)$ be divided into length L overlapping frames $x_i(m)$ with a shift of D samples,

$$x_i(m) = x(iD + m), \quad m = 0, \ldots, L-1. \tag{7.58}$$

This frame is used to obtain the biased autocorrelation function estimate as follows

$$r_x(p, i) = \frac{1}{L} \sum_{m=0}^{L-p} x_i(m) x_i(m+p), \quad p = 0, \ldots, P-1. \tag{7.59}$$

These autocorrelation coefficients are used as described in Section 7.6 to calculate the subspace filter $\mathbf{H}_i$. Note that this filter has now a subscript i to emphasize the fact that it is computed based on the signal samples of the i^{th} frame.

Every frame is divided into smaller P-dimensional overlapping vectors with a 50% overlap. The frame length L is chosen to be a multiple of P so that there will be in total $\frac{2L}{P} - 1$ vectors in one frame. Like all frequency-domain methods, the speech signal within every frame is assumed to be stationary so that these vectors would all have the same covariance matrix and hence the same subspace filter $\mathbf{H}_i$. Therefore we have

$$\hat{\mathbf{s}}_{i,jP/2} = \mathbf{H}_i \mathbf{x}_i(jP/2) \quad \text{for } j = 2, \ldots, \frac{2L}{P}, \tag{7.60}$$

[8] As in the case of the method reported in [31].

[9] Moreover, if a fast eigenvalue decomposition is used, for example [39], the complexity would be $\mathcal{O}(PQ)$ per sample.

where the input vector $\mathbf{x}_i(m) = [x_i(m), x_i(m-1), \ldots, x_i(m-P+1)]^T$ and the filter output $\hat{\mathbf{s}}_{i,m}$ is defined in a similar way. The output vectors are then multiplied by a length-P Hanning window and synthesized using the overlap-add method to obtain one enhanced frame $\hat{s}_i(m) = \hat{s}(iD+m)$. Finally every frame is multiplied by a second length-L Hanning window and the total enhanced speech signal is recovered using the overlap-add synthesis technique. A 50% overlap is also applied to these larger analysis frames, that is $D = L/2$. In this way, every input vector is enhanced using filters designed from two different analysis frames which allows to compensate for any speech non-stationarity. This is analogous to frequency domain methods where frame overlap is applied, with every length-L frame of noisy speech being enhanced using a unique filter.

In the original SSA implementation described earlier the EVD, with complexity $\mathcal{O}(P^3)$, is carried out every $P/2$ samples resulting in a total complexity in the order of $\mathcal{O}(P^2)$ per sample. In the new FBEVD scheme, the EVD is only needed every frame at a rate of $L/2$ samples, where L is the frame length. Thus, if $L = \kappa P$ then the computational cost of the EVD would reduce to $\mathcal{O}(P^2/\kappa)$. That is the computational savings will be proportional to $\kappa = L/P$.

For example for $L = 256$ and $P = 32$, we have $\kappa = 8$. This results in reducing the cost of EVD calculation by a factor of 8. Knowing that the largest computational burden of the SSA arises from the EVD, this factor constitutes a significant saving at almost no performance degradation [18]. Coupling this method with one of the fast EVD techniques discussed earlier would considerably reduce the overall computational load.

7.8 Some Recent Developments

Compared to its frequency-domain counterparts, the signal subspace approach for speech enhancement is relatively young. Research in this area has been mainly focused on resolving the problem of colored noise and on reducing the computation complexity. Recently, however, new interesting ideas have emerged with regard to the application of the SSA method to the general problem of speech enhancement. The increased interest in SSA has been mainly stirred by the availability of more processing power at low cost hence making the main handicap of the SSA, namely its computational complexity, be no more a serious issue. In this section, we present recent developments and extensions to the conventional SSA method which may guide the reader towards more innovative ideas.

7.8.1 Auditory Masking

The masking properties of the human ear is a very interesting phenomenon that has attracted the interest of speech enhancement specialists, who found that it may offer an adequate solution to the inevitable trade-off between signal distortion and residual noise level. Indeed the use of masking in frequency-domain methods has resulted in improved performances with less annoying musical noise [37], [35].

Until recently, however, the use of masking in conjunction with the SSA had not been attempted; one apparent reason for this is that the human perceptual properties are usually understood from a frequency-domain perspective. Consequently, all available masking models were developed in the frequency domain and there was no clear way to represent these human hearing properties in the eigendomain. Lately, a few solutions to this problem started to emerge.

In [26], a perceptual post-filter is appended to the output of the signal subspace filter to further suppress the remaining noise components without distorting the desired speech signal from a perceptual stand-point. The post-filter, however, is designed in the frequency domain.

In [20], [21], on the other hand, the masking properties were used to actually adjust the gain function in the eigendomain by accounting for the masking threshold. The latter is calculated in the frequency domain then mapped to the eigendomain using (7.49).

A similar approach is adopted in [25] where a different gain function is obtained by modifying the optimization criterion used.

7.8.2 Multi-Microphone Systems

Microphone array processors are known to offer an improved performance over their single-channel counterparts by exploiting the added spatial information. Few attempts have been made to generalize the SSA into a multi-microphone design.

In [7], a multi-microphone beamformer was presented which used the SVD of a composite data matrix to simultaneously compute the filter coefficients for every channel. This method was found to yield improved results under directional noise but the performance degraded under reverberant conditions or as the number of noise sources increased.

In [19], [18], an eigendomain postfilter is designed which uses the EVD of a composite covariance matrix coupled with averaging in the eigendomain to calculate the filter coefficients. This method was found to be mainly useful under diffuse noise fields, such as reverberant enclosures, and to be relatively insensitive to the reverberation time.

7.8.3 Subband Processing

Recently in [38], the SSA has been used in a subband design. In this scheme, the input noisy signal is split into subbands via a perceptual filterbank implemented using a wavelet packet transform. Every subband signal is then enhanced using a separate subspace filter. To reduce the computational load, subspace tracking similar to the one adopted in [31] has been used. The enhanced fullband signal is then recovered using a synthesis filterbank.

Applying the SSA in conjunction with a subband design can give extra flexibility to the system by allowing for instance to select different parameter values in every band depending on the noise characteristics. Less sever suppression can be applied in higher bands where car noise for example is known to be absent. In addition, the subband signals have a narrower bandwidth which may allow to select a lower value for the model order P without incurring significant signal distortion as would be the case with the fullband signal. This would result in reducing the cost of the EVD and with a proper design may lead to substantial overall computational savings.

Another advantage of the reduced bandwidth is that within every band the noise can be more accurately assumed to be white. This may be an alternative solution to the colored noise issue usually associated with the SSA.

7.9 Conclusions

In this chapter, we discussed the signal subspace approach (SSA) for speech enhancement. The SSA consists of transforming the input noisy speech into the eigendomain which provides a transform domain different from the traditionally widely used frequency domain. Processing in the eigendomain readily offers the possibility to eliminate the components residing in the noise subspace and to recover the desired speech from its orthogonal signal subspace. To this end, different linear estimators are available in the literature. They mainly vary according to the underlying optimization criterion adopted.

We have described the different techniques developed to deal with the colored noise case and to cope with the complexity issue usually associated with the SSA. We also provided a frequency-domain interpretation in which the eigendomain linear estimators can be viewed as functions of the SNR in the passbands of signal dependent analysis filters which track the formant locations of the input speech. This interpretation can shed more light on the SSA allowing the emergence of new ideas that tackle the problem from a different perspective.

We finally made a review of some of the most promising and interesting extensions and developments to the SSA, namely the use of masking, subband processing, and multi-microphone implementations. These methods can stir the readers motivation to pursue further research in this direction as it is believed that the SSA is a powerful signal processing tool that has yet to be fully exploited in the speech enhancement field.

References

1. R. Badeau, K. Abed-Meraim, G. Richard, and B. David, “Sliding window orthonormal past algorithm,” in *Proc. IEEE ICASSP*, vol. 5, 2003, pp. 261–264.

2. S. F. Boll, “Suppression of acoustic noise in speech using spectral subtraction,” *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 27, pp. 113–120, Apr. 1979.
3. Y. Bresler and A. Macovski, “Exact maximum likelihood parameter estimation of superimposed exponential signals in noise,” *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 34, pp. 1081–1089, Oct. 1986.
4. T. Chonavel, B. Champagne, and C. Riou, “Fast adaptive eigenvalue decomposition: a maximum likelihood approach,” *IEEE Trans. on Signal Processing*, vol. 83, pp. 317–324, Feb. 2003.
5. J. R. Deller, J. G. Proakis, and J. H. L. Hansen, *Discrete-Time Processing of Speech Signals*. Prentice-Hall, Englewood Cliffs, NJ, 1993.
6. M. Dendrinos, S. Bakamidis, and G. Carayannis, “Speech enhancement from noise: a regenerative approach,” *Speech Communications*, vol. 10, pp. 45–57, Feb. 1991.
7. S. Doclo and M. Moonen, “GSVD-based optimal filtering for single and multimicrophone speech enhancement,” *IEEE Trans. on Signal Processing*, vol. 50, pp. 2230–2244, Sept. 2002.
8. Y. Ephraim and H. L. Van Trees, “A signal subspace approach for speech enhancement,” *IEEE Trans. on Speech and Audio Processing*, vol. 3, pp. 251–266, July 1995.
9. G. H. Golub and C. F. Van Loan, *Matrix Computations*. Johns Hopkins University Press, Baltimore, MD, 2nd edition, 1989.
10. P. C. Hansen and S. H. Jensen, “FIR filter representation of reduced-rank noise reduction,” *IEEE Trans. on Signal Processing*, vol. 46, pp. 737–1741, June 1998.
11. P. S. K. Hansen, P. C. Hansen, S. D. Hansen, and J. A. Sorensen, “Noise reduction of speech signals using the rank-revealing ULLV decomposition,” in *Proc. EUSIPCO*, 1996, pp. 182–185.
12. M. H. Hayes, *Statistical Digital Signal Processing and Modeling*. John Wiley & Sons, Inc., New York, 1996.
13. S. Haykin, *Adaptive Filter Theory*. Prentice-Hall, Englewood Cliffs, NJ, 4th edition, 2002.
14. K. Hermus and P. Wambacq, “Assessment of signal subspace based speech enhancement for noise robust speech recognition,” in *Proc. IEEE ICASSP*, vol. 1, 2004, pp. 954–948.
15. Y. Hu and C. Loizou, “A subspace approach for enhancing speech corrupted by colored noise,” in *Proc. IEEE ICASSP*, vol. 1, 2002, pp. 573–576.
16. J. Huang and Y. Zhao, “An energy-constrained signal subspace method for speech enhancement and recognition in colored noise,” *Speech Communications*, vol. 1, pp. 165–181, 1998.
17. J. Huang and Y. Zhao, “A DCT-based fast signal subspace technique for robust speech recognition,” *IEEE Trans. on Speech and Audio Processing*, vol. 8, pp. 747–751, Nov. 2000.
18. F. Jabloun, *Perceptual and Multi-Microphone Signal Subspace Techniques for Speech Enhancement*. Ph.D. thesis, McGill University, Montreal, Canada, 2004.
19. F. Jabloun and B. Champagne, “A multi-microphone signal subspace approach for speech enhancement,” in *Proc. IEEE ICASSP*, vol. 1, 2001, pp. 205–208.
20. F. Jabloun and B. Champagne, “A perceptual signal subspace approach for speech enhancement in colored noise,” in *Proc. IEEE ICASSP*, vol. 1, 2002, pp. 569–572.

21. F. Jabloun and B. Champagne, “Incorporating the human hearing properties in the signal subspace approach for speech enhancement,” *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 11, pp. 700–708, Nov. 2003.
22. J. Jensen and J. H. L. Hansen, “Speech enhancement using a constrained iterative sinusoidal model,” *IEEE Trans. on Speech and Audio Processing*, vol. 9, pp. 731–740, Oct. 2001.
23. 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. on Speech and Audio Processing*, vol. 3, pp. 439–448, Nov. 1995.
24. M. Jeppesen, C. A. Rodbro, and S. H. Jensen, “Recursively updated eigenfilterbank for speech enhancement,” in *Proc. IEEE ICASSP*, vol. 1, 2001, pp. 653–656.
25. J. U. Kim, S. G. Kim, and C. D. Yoo, “The incorporation of masking threshold to subspace speech enhancement,” in *Proc. IEEE ICASSP*, vol. 1, 2003, pp. 76–79.
26. M. Klein and P. Kabal, “Signal subspace speech enhancement with perceptual post-filtering,” in *Proc. IEEE ICASSP*, vol. 1, 2002, pp. 537–540.
27. H. Krim and M. Viberg, “Two decades of array signal proceeising research: the parametric approach,” *IEEE Signal Processing Magazine*, pp. 67–94, July 1996.
28. D. G. Luenberger, *Linear and Nonlinear Programming.* Addison-Wesley, Reading, MA, 1984.
29. U. Mittal and N. Phamdo, “Signal/noise KLT based approach for enhancing speech degraded by colored noise,” *IEEE Trans. on Speech and Audio Processing*, vol. 8, pp. 159–167, Mar. 2000.
30. T. F. Quatieri and R. J. McAulay, “Noise reduction using a soft-decision sinewave vector quantizer,” in *Proc. IEEE ICASSP*, 1990, pp. 821–824.
31. A. Rezayee and S. Gazor, “An adaptive KLT approach for speech enhancement,” *IEEE Trans. on Speech and Audio Processing*, vol. 9, pp. 87–95, Feb. 2001.
32. R. O. Schmidt, “Multiple emitter location and signal parameter estimation,” *IEEE Trans. on Antennas and Propagation*, vol. 34, pp. 276–280, Mar. 1986.
33. G. W. Stewart, “An updating algorithm for subspace tracking,” *IEEE Trans. on Signal Processing*, vol. 40, pp. 1535–1541, June 1992.
34. P. Strobach, “Low-rank adaptive filters,” *IEEE Trans. on Signal Processing*, vol. 44, pp. 2932–2947, Dec. 1996.
35. D. E. Tsoukalas, J. N. Mourjopoulos, and G. Kokkinakis, “Speech enhancement based on audible noise suppression,” *IEEE Trans. on Speech and Audio Processing*, vol. 5, pp. 479–514, Nov. 1997.
36. R. Vetter, “Single channel speech enhancement using MDL-based subspace approach in bark domain,” in *Proc. IEEE ICASSP*, vol. 1, 2001, pp. 641–644.
37. N. Virag, “Single channel speech enhancement based on masking properties of the human auditory system,” *IEEE Trans. on Speech and Audio Processing*, vol. 7, pp. 126–137, Mar. 1999.
38. J. F. Wang, C. H. Yang, and K. H. Chang, “Subsapce tracking for speech enhancement in car noise environments,” in *Proc. IEEE ICASSP*, vol. 2, 2004, pp. 789–792.
39. G. Xu and T. Kailath, “Fast subspace decomposition,” *IEEE Trans. on Signal Processing*, vol. 42, pp. 539–551, Mar. 1994.
40. B. Yang, “Projection approximation subspace tracking,” *IEEE Trans. on Signal Processing*, vol. 43, pp. 95–107, Jan. 1995.

8 Speech Enhancement: Application of the Kalman Filter in the Estimate-Maximize (EM) Framework

Sharon Gannot

Bar-Ilan University
Ramat-Gan 52900, Israel
Email: gannot@macs.biu.ac.il

Abstract. The application of the Kalman filter to the single-microphone speech enhancement task is presented in this chapter. Among numerous published algorithms, an important sub-group employs the estimate–maximize (EM) procedure to iteratively estimate the spectral parameters of the speech and noise signals. We elaborate on a specific member of this sub-group. In the E-step, the Kalman smoother is applied and in the M-step, a non-standard Yule-Walker equation set is solved. An approximated EM algorithm is derived by applying the gradient-descent method to the likelihood function. We obtain a sequential, computationally efficient, algorithm. It is then shown, that the sequential parameter estimation can be replaced by a Kalman filter to obtain a dual speech and parameters Kalman filter. A natural generalization to the dual scheme is an estimation scheme in which both speech and parameters are jointly estimated by applying a nonlinear extension to the Kalman filter, namely the *unscented Kalman filter*. Extensive experimental study, using real speech and noise signals is provided to compare the proposed methods with alternative speech enhancement algorithms. Kalman filter based algorithms are shown to maintain the natural speech quality. However, their noise reduction ability is limited.

8.1 Introduction

Speech quality and intelligibility might significantly deteriorate in the presence of background noise, especially when the speech signal is subject to subsequent processing. In particular, speech coders, and *automatic speech recognition* (ASR) systems, that were designed or trained to act on clean speech signals, might be rendered useless in the presence of background noise. Speech enhancement algorithms have therefore attracted a great deal of interest in the past three decades.

Among these speech enhancement algorithms there are numerous algorithms based on Wiener [1] or Kalman filtering [2]. Employing both Wiener and Kalman filters requires the knowledge of the parameters involved (e.g. noise gains, linear predictive codes (LPC) coefficients). Since these parameters are usually unknown, the problem of joint estimation of signal and parameters arises.

Roughly, there are two families of algorithms applicable to the problem at hand. The first involving an off-line training stage, where representative parameters are extracted from the clean speech utterances and then used in the enhancement stage. The second family, consists of online methods in which the signal and the parameters are estimated jointly from the corrupted signal. A typical procedure, applied to this type of problems, is the *estimate-maximize* (EM) algorithm [3]. In this work we will concentrate on the latter, but for the completeness of the presentation we will briefly give a literature survey of the former.

A pioneering speech enhancement work in the framework of the EM procedure was presented by Lim and Oppenheim [4]. They used an *auto-regressive* (AR) model for the speech signal and assumed that the speech is picked-up by the microphone together with additive white Gaussian noise. The proposed algorithm is iterative in nature. It consists of estimating the speech AR parameters by solving the Yule-Walker equations, using the current estimate of the speech signal, and then applying the (non-causal) Wiener filter to the observed signal to obtain an (hopefully) improved estimate of the desired speech signal. It can be shown that the version of the algorithm which uses the covariance of the speech signal estimate, given at the output of the Wiener filter, is in fact the estimate-maximize (EM) algorithm (up to a scale factor) for the problem at hand. As such, it is guaranteed to converge to the *maximum likelihood* (ML) estimate of the AR parameters, or at least to a local maximum of the likelihood function, and to yield the best linear filtered estimate of the speech signal, computed at the ML parameter estimate.

Hansen and Clements [5], [6] proposed to incorporate auditory domain constrains, in the *line spectral pair* (LSP) domain, for improving the convergence behavior of the Lim and Oppenheim algorithm. They use smoothness constraints, both across time and iterations, to produce a consistent stopping criterion for the iterative procedure. Later, Pellom and Hansen [7] extended this idea by incorporating the constraints according to the signal-to-noise ratio (SNR) in various sub-bands. Masgrau *et al.* [8] proposed to incorporate third-order cumulants in the Yule-Walker equations for improving the immunity of the AR parameters estimate to additive Gaussian noise.

Paliwal and Basu [9] were, perhaps, the first to use the Kalman filter in the context of speech enhancement. Their experimental results reveal its distinct advantage over the Wiener filter. However, the estimated speech parameters are obtained from the clean speech signal and not from the corrupted signal. Gibson *et al.* [10], [11] proposed to extend the use of the Kalman filter by incorporating a colored noise model for improving the enhancement performance for certain class of noise sources. The proposed algorithm iterates between Kalman filtering of the given corrupted speech measurements, and estimation of the speech parameters given the enhanced speech waveform. As the authors suggest using the regular Yule-Walker equations for estimating the speech AR parameters, the resulting algorithm is only an approximated version of the EM algorithm. The estimated parameters prove to improve speech coding systems that rely on AR modelling of the speech signal.

A comprehensive study of the use of the EM algorithm in diverse problems of joint estimation of signals and parameters is given in a series of works by Weinstein *et al.*. In [12], the noise cancellation problem presented by Widrow [13] is solved using the EM algorithm in the frequency domain. The more general two-channel noise cancellation problem is addressed in [14]. Both are comprised of iterations between parameter estimation and Wiener filtering. In [15], a time-domain formulation to the single-microphone speech enhancement problem is presented. The approach consists of representing the signal model using linear dynamic state equation, and applying the EM method. The resulting algorithm is similar in structure to the Lim and Oppenheim [4] algorithm, only that the non-causal Wiener filter is replaced by the Kalman smoothing equations. Sequential speech enhancement algorithms are presented as well. These sequential algorithms are characterized by a forward Kalman filter whose parameters are continuously updated by gradient-decent search on the likelihood function. In [16], [17], sequential approximations to the EM algorithm are elaborated on in the context of two-channels noise cancellation. The related problem of single-microphone *active noise cancellation* (ANC) is presented in [18].

Lee *et al.* [19], [20] extend the sequential single-sensor algorithm of Weinstein *et al.* by replacing the white Gaussian excitation of the speech signal with a Gaussian-mixture term that may account for the presence of an impulse train in the excitation sequence of voiced speech. A recursive gradient-based approach is applied to the parameter estimation. Lee *et al.* examined the SNR improvement of the algorithm when applied to synthetic speech input. Goh *et al.* [21] propose replacing the speech innovation sequence. The proposed excitation sequence is comprised of both Gaussian white noise (modelling the unvoiced part) and impulse train (presenting the voiced part). The latter is modelled as a long-term AR process. The resulting high-dimensional Kalman filter is implemented efficiently by exploiting the sparsity of the involved matrices. The parameter estimation is conducted via EM iterations. When the standard Kalman filter gain is recursively computed, one needs to estimate the speech and noise gains. To avoid this estimation stage, Gabrea *et al.* [22] propose checking the whiteness of the innovation sequence to test whether the asymptotically optimal solution has been reached. Since the estimation of the AR parameters cannot be avoided, Gabrea *et al.* propose to use the modified Yule-Walker procedure. Another extension to Weinstein *et al.* works was proposed by Gannot *et al.* [23]. In this work both iterative-batch and sequential versions of the EM algorithm are treated. The E-step is implemented by applying the Kalman filter. *Higher-order statistics* (HOS) is employed for obtaining a robust initialization to the parameter estimation stage (M-step). Fujimoto and Ariki [24] use Kalman filtering in the frequency domain (without using the AR model). Initialization of their algorithm is obtained by the classical *spectral subtraction* [25] algorithm.

Several nonlinear extensions to the standard Kalman filter exist. Lee *et al.* [26] propose the application of a robust Kalman filter. Similar to other contributions, iterations between signal enhancement and parameter estimation are conducted. The novelty of this paper stems from the use of nonlinear estimation procedures. Both parameters and signals are estimated in a robust manner by introducing a saturation function into the cost function, rather than using the standard squared cost function. Ma *et al.* [27] introduce the application of a post-filter based on masking properties of the human auditory system to further enhance the resulting speech signal. However, not much attention is paid in this work to the AR parameter estimation task itself.

Shen and Deng in [28] present a new and interesting approach to speech enhancement based on H_∞ filtering. This approach differs from the traditional Kalman filtering approach in the definition of the error criterion for the filter design. Rather than minimizing a squared error term, as in the standard Kalman filter, their procedure consists of calculating a filter which minimizes the worst possible amplification of the estimation error in terms of the modelling errors and additive noises. The parameter estimation is conducted in parallel using H_∞ criterion as well. The authors claim that their resulting *minimax* estimation method is highly robust and more appropriate in practical speech enhancement. It should be noted however, that the implementation of the minimax criterion in the parameter estimation stage of the algorithm, seems to be much more complicated than the conventional estimation procedure.

Wan *et al.* [29] assume nonlinear model of the speech production, i.e. that the speech utterance is the output of a *neural network* (NN) with unknown parameters. Their algorithm is comprised of iterations between parameter estimation and signal enhancement. The nonlinearity inherent to the NN is negotiated by the application of the *extended Kalman filter* (EKF). The recently proposed *unscented transform* (UT), suggested by Julier *et al.* [30], is a novel method for calculating first- and second-order statistics of a random variable undergoing a nonlinear transformation, that was used for constructing a Kalman filter for the nonlinear case. The resulting filter, named the *unscented Kalman filter* (UKF) is shown to be superior to the well-established EKF in many application of interest. Wan *et al.* [31] propose to replace the EKF in [29] by the UKF, resulting in an improved performance. Gannot and Moonen [32] use the UKF in the speech enhancement application (as well as speech dereverberation), where the nonlinearity arises from the multiplication of the speech and the parameters. Their proposed method is only applied to artificial AR process. Fong and Godsill [33] use the *particle Kalman filter* for the speech enhancement task. The speech signal gain is given a random walk model, while its *partial correlation coefficients* (PARCOR) are given a constrained random walk model (as their absolute value must be less than 1). Monte Carlo filtering is applied for estimating these parameters, whereas linear Kalman filter is applied in parallel for enhancing the speech signal.

A distinct family of algorithms, using the Kalman filter as well, tackles the parameter estimation task by conducting a training stage. In this approach, a *hidden Markov model* (HMM) for the clean signals is estimated from the training data, and the signal is estimated from the noisy signal by applying Bayesian estimators. This method was first proposed by Ephraim *et al.* [34], [35], where a bank of HMM state-related Wiener filters is used. The Wiener filter is later replaced by the Kalman filter in [36], [37]. The problem of unknown speech gain contours and noise parameters is alleviated by using EM iterations. This method, proposed by Lee and Jung [38], use Kalman filter in the E-step, based on the trained AR parameters and the estimated noise parameters. In the M-step, the noise and gain parameters are recursively estimated. An *interacting multiple model* (IMM) algorithm, in which the Kalman filters in the different states interact with one another, is applied for enhancing speech contaminated by additive white or colored noise by Kim *et al.* [39]. Finally, a nonlinear extension of the HMM concept is proposed by Lee *et al.* [40]. The speech is assumed to be an output of a *neural network* with time-varying parameters controlled by a hidden Markov chain. Both the training stage and the enhancement stage become nonlinear. The nonlinearity is negotiated by the application of the EKF.

In this work, we are elaborating on a representative of the online methods family. We derive the exact EM solution to the case where the speech signal is contaminated by colored noise. An iterative-batch solution is obtained by dividing the noisy signal into short frames where the quasi-stationarity assumption for the speech signal holds. In each frame, a predetermined number of iterations is conducted. In the E-step, the Kalman smoother is applied and the speech signal estimate is obtained together with the respective covariance matrix. In the M-step, non-standard Yule-Walker equations, comprised of both the speech estimate and the covariance matrix, are solved. Special attention is given to the initialization stage, in which HOS are used. We proceed to sequential approximations of the EM method. The obtained algorithm consists of a forward Kalman filter (or fixed-lag smoother), rather than the Kalman smoother, and a gradient-based search of the likelihood function for the optimal parameters. The latter is implemented as a sequential update, where a new estimate is produced at each new sample. We then give the parameters a dynamic and a stochastic model. Due to the multiplication of the parameter vector and the speech state-vector, nonlinearity results in. We tackle this problem by using the UKF. Two version of the solution are proposed, namely a *dual* scheme, in which two Kalman filters are applied in parallel, and a *joint* scheme, which consists of an application of a single nonlinear UKF.

Our discussion is supported by an extensive experimental study using speech and noise signals taken from databases. The outcome consists of objective distortion measures (such as total output SNR, weighted segmental SNR, log-spectral distance and ASR performance evaluation), as well as subjective tests (i.e. assessment of sound spectrograms and informal listening tests). We first show that the sequential algorithm is generally inferior to the iterative-batch algorithm. However, at low SNR levels, the degradation is usually insignificant. The iterative-batch algorithm is then compared to various methods, including the log-spectral amplitude (LSA) estimator [41], the HMM-based filtering algorithms [34], [35], the optimally-modified LSA [42], the mixture-maximum algorithm [43], and the Wiener filter approach of [4]. We end our study by applying the UKF version to a simple and artificial problem, showing the potential of the method.

The organization of the chapter is as follows. In Sect. 8.2, we present the signal model. In Sect. 8.3, we present the iterative-batch algorithm. In Sect. 8.4, we show how higher-order statistics might be incorporated in order to improve the performance of the iterative-batch algorithm. The sequential algorithm is presented in Sect. 8.5. The all-Kalman algorithms are discussed in Sect. 8.6. Experimental results are provided in Sect. 8.7. We draw some conclusions and discuss further directions in Sect. 8.8.

8.2 Signal Model

Consider a speech signal received by a single microphone and contaminated by a colored noise signal. Let the signal measured by the microphone be given by:

$$z(t) = s(t) + v(t), \tag{8.1}$$

where $s(t)$ represents the sampled speech signal, and $v(t)$ represents additive background noise. We assume the standard LPC modelling for the speech signal over an analysis frame, i.e. $s(t)$ is modelled as a stochastic AR process:

$$s(t) = -\sum_{k=1}^{p} \alpha_k s(t-k) + g_s u_s(t), \tag{8.2}$$

where the excitation $u_s(t)$ is a normalized (zero-mean unit variance) white noise, g_s represents the spectral level, and $\alpha_1, \ldots, \alpha_p$ are the AR coefficients. We may incorporate the more detailed voiced speech model suggested in [44] in which the excitation process is composed of a weighted linear combination of an impulse train and a white noise sequence to represent voiced and unvoiced speech, respectively. However, in our experiments, this approach did not yield any significant performance improvements over the standard LPC modelling. Equation (8.2) can be reformulated in a state-space presentation.

Define the state-vector

$$\boldsymbol{s}_p^T(t) = \left[\, s(t-p+1)\ s(t-p+2)\ \ldots\ s(t) \,\right],$$

the speech transition matrix

$$\Phi_s = \begin{bmatrix} 0 & 1 & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \\ -\alpha_p & -\alpha_{p-1} & \cdots & \cdots & -\alpha_2 & -\alpha_1 \end{bmatrix},$$

and the p-dimensional vectors $\boldsymbol{g}_s^T = [\,0\ \ldots\ 0\ g_s\,]$ and $\boldsymbol{h}_s^T = [\,0\ \ldots\ 0\ 1\,]$. Then (8.2) can be rewritten as

$$\begin{aligned} \boldsymbol{s}_p(t) &= \Phi_s(t)\boldsymbol{s}_p(t-1) + \boldsymbol{g}_s(t)u_s(t), \\ z(t) &= \boldsymbol{h}_s^T\boldsymbol{s}_p(t) + v(t). \end{aligned} \tag{8.3}$$

The additive noise $v(t)$ is also assumed to be a realization from a zero-mean, possibly non-white stochastic AR process:

$$v(t) = -\sum_{k=1}^{q} \beta_k v(t-k) + g_v u_v(t), \tag{8.4}$$

where $\beta_1, \ldots, \beta_q$ are the AR parameters of the noise process, and g_v represents its power level. Many of the actual noise sources may be closely approximated as low-order, all-pole (AR) processes. In this case a significant improvement may be achieved by incorporating the noise model into the estimation process as indicated in [11], [23].

Equation (8.4) can be rewritten in a state-space formulation as well. Define the state-vector $\boldsymbol{v}_q^T(t) = \left[\, v(t-q+1)\ v(t-q+2)\ \ldots\ v(t) \,\right]$, the noise transition matrix

$$\Phi_v = \begin{bmatrix} 0 & 1 & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \\ -\beta_q & -\beta_{q-1} & \cdots & \cdots & -\beta_2 & -\beta_1 \end{bmatrix},$$

and the q-dimensional vectors $\boldsymbol{g}_v^T = [\,0\ \ldots\ 0\ g_v\,]$ and $\boldsymbol{h}_v^T = [\,0\ \ldots\ 0\ 1\,]$. Then (8.4) can be rewritten as

$$\begin{aligned} \boldsymbol{v}_q(t) &= \Phi_v(t)\boldsymbol{v}_q(t-1) + \boldsymbol{g}_v(t)u_s(t), \\ z(t) &= s(t) + \boldsymbol{h}_v^T\boldsymbol{v}_q(t). \end{aligned} \tag{8.5}$$

Equations (8.3) and (8.5) can be concatenated into a one, larger, state-space equation:

$$\begin{aligned}\boldsymbol{x}(t) &= \varPhi\boldsymbol{x}(t-1) + G\boldsymbol{u}(t), \\ \boldsymbol{z}(t) &= \boldsymbol{h}^T\boldsymbol{x}(t),\end{aligned} \tag{8.6}$$

where the state-vector $\boldsymbol{x}(t)$ is defined by

$$\boldsymbol{x}^T(t) = \left[\, \boldsymbol{s}_{p-1}^T(t-1)\; s(t)\; \boldsymbol{v}_{q-1}^T(t-1)\; v(t) \,\right],$$

the state-transition matrix $\varPhi$ is given by

$$\varPhi = \begin{bmatrix} \varPhi_s & \mathbf{0} \\ \mathbf{0} & \varPhi_v \end{bmatrix},$$

the driving noise vector is

$$\boldsymbol{u}^T(t) = \left[\, u_s(t)\; u_v(t) \,\right],$$

G is given by

$$G = \begin{bmatrix} \boldsymbol{g}_s & \mathbf{0} \\ \mathbf{0} & \boldsymbol{g}_v \end{bmatrix},$$

and the measurement vector is given by

$$\boldsymbol{h}^T = \left[\, \boldsymbol{h}_s^T\; \boldsymbol{h}_v^T \,\right].$$

Assuming that all signal and noise parameters are known, which implies that $\varPhi$, $\boldsymbol{h}$ and G are known, the optimal minimum mean square error (MMSE) linear state estimate of $\boldsymbol{x}(t)$, which includes the desired speech signal $s(t)$, is obtained using the Kalman smoothing equations. However, since the signal and noise parameters are not known a priori, they must be estimated within the algorithm as well.

8.3 EM - Based Algorithm

Applying the EM method to the problem at hand, and following the considerations in [45], [15] (see also [16] that considers the two channel case), we obtain in Appendix A the following algorithm that iterates between state and parameter estimation.

Let $\boldsymbol{\theta}$ be the vector of unknown parameters in the extended model,

$$\boldsymbol{\theta}^T = \left[\, \boldsymbol{\alpha}^T\; g_s\; \boldsymbol{\beta}^T\; g_v \,\right] \quad \boldsymbol{\alpha}^T = \left[\, \alpha_p\; \alpha_{p-1}\; \ldots\; \alpha_1 \,\right] \quad \boldsymbol{\beta}^T = \left[\, \beta_q\; \beta_{q-1}\; \ldots\; \beta_1 \,\right] \tag{8.7}$$

and let $\widehat{\boldsymbol{\theta}}^{(l)}$ be the estimate of $\boldsymbol{\theta}$ after l iterations of the algorithm. Finally, let

$$\boldsymbol{z}^T = \left[\, z(0)\; z(1) \ldots z(N-1) \,\right] \tag{8.8}$$

be the vector of observed data in the current analysis frame. We use the notation

$$\widehat{(\quad\cdot\quad)} = E_{\widehat{\boldsymbol{\theta}}^{(l)}}\{\cdot \mid \boldsymbol{z}\} \tag{8.9}$$

for designating an estimate of the signal statistics based on the observation vector $\boldsymbol{z}$ and using the current parameter estimate $\widehat{\boldsymbol{\theta}}^{(l)}$. To obtain $\widehat{\boldsymbol{\theta}}^{(l+1)}$ we use the following two-stage iterative procedure.

8.3.1 State Estimation (E-Step)

Define:

$$\boldsymbol{\mu}(t|N) = \widehat{\boldsymbol{x}(t)},$$
$$P(t|N) = \widehat{\boldsymbol{x}(t)\boldsymbol{x}^T(t)} - \widehat{\boldsymbol{x}(t)}\widehat{\boldsymbol{x}(t)}^T,$$

i.e., $\boldsymbol{\mu}(t|N)$ represents the current state estimate based on $t = 0, \ldots, N-1$, and $P(t|N)$ represents the associated error covariance matrix. Then $\boldsymbol{\mu}(t|N)$ and $P(t|N)$ are computed using a forward, Kalman filtering recursion, followed by a backward Kalman smoothing recursion, as follows:

Forward (filtering) recursion:

For $t = 0, 2, \ldots, N-1$:

Propagation Equations

$$\boldsymbol{\mu}(t|t-1) = \hat{\Phi}^{(l)}\boldsymbol{\mu}(t-1|t-1), \tag{8.10}$$

$$P(t|t-1) = \hat{\Phi}^{(l)}P(t-1|t-1)\hat{\Phi}^T + \hat{G}^{(l)}\left(\hat{G}^{(l)}\right)^T. \tag{8.11}$$

Updating Equations

$$\boldsymbol{\mu}(t|t) = \boldsymbol{\mu}(t|t-1) + \boldsymbol{k}(t)\left[z(t) - \boldsymbol{h}^T\boldsymbol{\mu}(t|t-1)\right], \tag{8.12}$$

$$P(t|t) = P(t|t-1) - \boldsymbol{k}(t)\boldsymbol{h}^T P(t|t-1), \tag{8.13}$$

where

$$\boldsymbol{k}(t) = \frac{P(t|t-1)\boldsymbol{h}}{\boldsymbol{h}^T P(t|t-1)\boldsymbol{h}}.$$

Backward (smoothing) recursion:

For $t = N-1, N-2, \dots, 0$:

$$\boldsymbol{\mu}(t-1|N) = \boldsymbol{\mu}(t-1|t-1) + S(t-1)\left(\boldsymbol{\mu}[t-1|N] - \hat{\Phi}^{(l)}\boldsymbol{\mu}(t-1|t-1)\right), \tag{8.14}$$

$$P(t-1|N) = P(t-1|t-1) - S(t-1)\left(P(t|N) - P(t|t-1)\right)S^T(t-1), \tag{8.15}$$

where

$$S(t-1) = P(t-1|t-1)\left(\hat{\Phi}^{(l)}\right)^T P^{-1}(t|t-1).$$

$\hat{\Phi}^{(l)}$ and $\hat{G}^{(l)}$ are the matrices Φ and G respectively, at the current iteration stage.

8.3.2 Parameter Estimation (M-Step)

Parameters are estimated by using equation sets similar to the standard Yule-Walker (YW) solution for estimating the coefficients of an AR process, except that the correlation values are replaced by their a posteriori value:

$$\widehat{\boldsymbol{\alpha}}^{(l+1)} = -\left[\sum_{t=0}^{N-1}\widehat{\boldsymbol{s}_p(t-1)\boldsymbol{s}_p^T(t-1)}\right]^{-1}\sum_{t=0}^{N-1}\widehat{\boldsymbol{s}_p(t-1)s(t)}, \tag{8.16}$$

$$\widehat{g_s}^{(l+1)} = \frac{1}{N}\sum_{t=0}^{N-1}\left[\widehat{s^2(t)} + \left(\widehat{\boldsymbol{\alpha}}^{(l+1)}\right)^T\widehat{\boldsymbol{s}_p(t-1)s(t)}\right], \tag{8.17}$$

$$\widehat{\boldsymbol{\beta}}^{(l+1)} = -\left[\sum_{t=0}^{N-1}\widehat{\boldsymbol{v}_q(t-1)\boldsymbol{v}_q^T(t-1)}\right]^{-1}\sum_{t=0}^{N-1}\widehat{\boldsymbol{v}_q(t-1)v(t)}, \tag{8.18}$$

$$\widehat{g_v}^{(l+1)} = \frac{1}{N}\sum_{t=0}^{N-1}\left[\widehat{v^2(t)} + \left(\widehat{\boldsymbol{\beta}}^{(l+1)}\right)^T\widehat{\boldsymbol{v}_q(t-1)v(t)}\right]. \tag{8.19}$$

We note that $\widehat{\boldsymbol{s}_p(t-1)\boldsymbol{s}_p^T(t-1)}$ is the upper left $p \times p$ sub-matrix of $\widehat{\boldsymbol{x}(t)\boldsymbol{x}^T(t)}$. $\widehat{\boldsymbol{s}_p(t-1)s(t)}$, $\widehat{s^2(t)}$, $\widehat{\boldsymbol{v}_q(t-1)\boldsymbol{v}_q^T(t-1)}$, $\widehat{\boldsymbol{v}_q(t-1)v(t)}$, and $\widehat{v^2(t)}$ may similarly be extracted from $\widehat{\boldsymbol{x}(t)\boldsymbol{x}^T(t)}$.

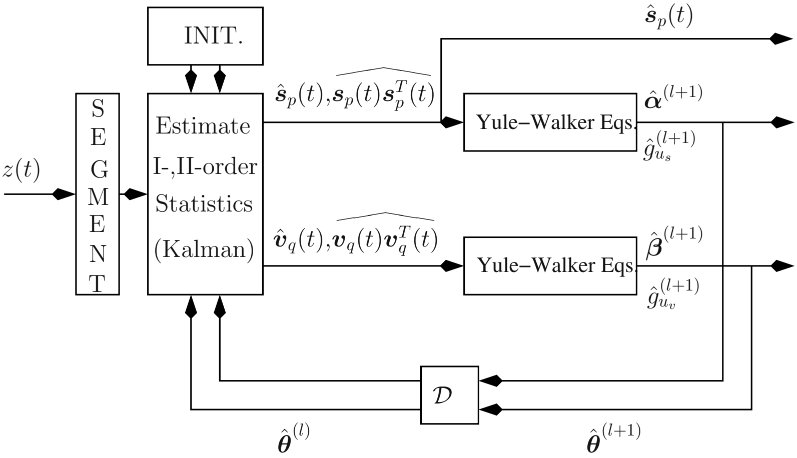

INIT.
$z(t)$
S E G M E N T
Estimate I-,II-order Statistics (Kalman)
$\hat{\boldsymbol{s}}_p(t),\widehat{\boldsymbol{s}_p(t)\boldsymbol{s}_p^T(t)}$
$\hat{\boldsymbol{v}}_q(t),\widehat{\boldsymbol{v}_q(t)\boldsymbol{v}_q^T(t)}$
Yule–Walker Eqs
Yule–Walker Eqs
$\hat{\boldsymbol{s}}_p(t)$
$\hat{\boldsymbol{\alpha}}^{(l+1)}$
$\hat{g}_{u_s}^{(l+1)}$
$\hat{\boldsymbol{\beta}}^{(l+1)}$
$\hat{g}_{u_v}^{(l+1)}$
$\mathcal{D}$
$\hat{\boldsymbol{\theta}}^{(l)}$
$\hat{\boldsymbol{\theta}}^{(l+1)}$

Fig. 8.1. Iterative-batch algorithm based on the EM procedure.

8.3.3 Reduced Complexity

In order to reduce the computations involved, we suggest to replace the full smoothing operation with fixed-lag smoothing (delayed Kalman filter estimate) [9] or even just by filtering. That is, instead of using $\hat{s}(t|N)$, the p-th entry of $\boldsymbol{\mu}(t|N)$, as the enhanced signal estimate, it is proposed to use $\hat{s}(t-p+1|t)$ (fixed lag smoothing) or $\hat{s}(t|t)$ (filtering), that are the first and the p-th entries of $\boldsymbol{\mu}(t|t)$, respectively. Similar observations apply to the enhanced noise estimate, used by the EM algorithm. With these modifications we do not need to apply the smoothing equations (8.14)–(8.15) that are computationally expensive. As indicated in [23], the obtained algorithm still maintains its nice monotonic convergence behavior.

8.3.4 Discussion

The *iterative-batch* algorithm is summarized in Fig. 8.1. Note, that two other blocks are depicted in Fig. 8.1. The first is the initialization block, we will elaborate on in Sect. 8.4. The second block is responsible to segmenting the noisy speech signal.

Since the algorithm is based on the EM method, it is guaranteed to converge monotonically to the ML estimate of all unknown parameters (under Gaussian assumptions), or at least to a local maximum of the likelihood function, where each iteration increases the likelihood of the estimate of the parameters. As a byproduct, it yields the optimal linear state (signal) estimate, computed using the estimated parameters.

Refer to the simplified EM algorithm, proposed by Koo *et al.* [10]. This simplified EM algorithm is obtained by iteratively estimating the speech parameters using the enhanced speech signal (by employing the ordinary YW equation set), and then using these parameters to improve the estimate of the enhanced signal (the noise parameters are estimated, using signal segments at which voice activity is assumed not to be present). We found that unlike the EM algorithm (even when using filtering or fixed-lag smoothing), which is guaranteed to be stable and to monotonically increase the likelihood function, the simplified EM algorithm [10] does not possess such properties. The simplified EM algorithm results in performance degradation, which is very significant at the lower SNR range. Similar behavior was noticed by Lim and Oppenheim [4] in the context of an iterative Wiener filter algorithm for the enhancement of speech in the presence of white Gaussian noise.

The obtained algorithm is an extension of the algorithm presented in [15] for the case in which the additive noise is modelled more generally as a colored AR process. Since the signal and the noise parameter estimates are computed separately within the algorithm, the increase in computational complexity is quite moderate. However, the realizable improvement in the enhancement performance may be quite significant, as indicated in [11], [23].

8.4 Parameter Estimation Using Higher-Order Statistics

To obtain a reliable estimate of the speech signal, it is essential to have a powerful initialization algorithm for the speech and noise parameters. Otherwise, the estimation algorithm might converge to a local minimum of the likelihood function. When the SNR is high, an initial estimate of the speech parameters may be obtained using standard LPC processing, and an initial estimate of the noise parameters may be obtained by employing a *voice activity detector* (VAD), so that the noise statistics are accumulated during silence periods.

Unfortunately, this initialization procedure breaks down at low SNR conditions, below 5 dB in our experiments. However, if the additive noise $v(t)$ is assumed to be Gaussian, then higher-order statistics (HOS) may be incorporated in order to improve the initial estimate of the speech parameters. In that case, the quality of the enhanced speech signal is significantly improved compared to the standard initialization method that was indicated above.

It can be shown [by invoking the basic cumulant properties in [46][Section 2.2.3] and recalling (8.2)] that

$$\begin{aligned}&\mathrm{cum}\left(s(t), s(t-l_1), s(t-l_2), \ldots, s(t-l_M)\right) \qquad (8.20)\\ &= -\sum_{k=1}^{p} \alpha_k \mathrm{cum}\left(s(t-k), s(t-l_1), s(t-l_2), \ldots, s(t-l_M)\right),\end{aligned}$$

where cum$(\cdot, \cdot, \ldots)$ denotes the joint cumulant of the bracketed variables and $M \geq 1$. We note that

$$\mathrm{cum}(z(t)) = E\{z(t)\},$$

$$\mathrm{cum}(z(t), z(t-l_1)) = E\{z(t)z(t-l_1)\},$$

$$\mathrm{cum}(z(t), z(t-l_1), z(t-l_2)) = E\{z(t)z(t-l_1)z(t-l_2)\},$$

$$\begin{aligned}
&\mathrm{cum}(z(t), z(t-l_1), z(t-l_2), z(t-l_3)) \\
&\ = E\{z(t)z(t-l_1)z(t-l_2)z(t-l_3)\} \\
&\ -E\{(z(t)z(t-l_1)\} \cdot E\{z(t-l_2)z(t-l_3)\} \\
&\ -E\{(z(t)z(t-l_2)\} \cdot E\{z(t)z(t-l_3)\} \\
&\ -E\{(z(t)z(t-l_3)\} \cdot E\{z(t-l_1)z(t-l_2)\}.
\end{aligned}$$

A general formula for expressing cumulants in terms of moments can be found in [46].

Now, under the assumption that $v(t)$ is Gaussian, it can be shown [by invoking the same basic cumulant properties in [46] and recalling (8.1) and the statistical independence of $v(t)$ and $s(t)$] that

$$\begin{aligned}
&\mathrm{cum}\left(z(t), z(t-l_1), z(t-l_2), \dots, z(t-l_M)\right) \\
&\ = -\sum_{k=1}^{p} \alpha_k \mathrm{cum}\left(z(t-k), z(t-l_1), z(t-l_2), \dots, z(t-l_M)\right),
\end{aligned}$$

whenever $M \geq 2$. For $M = 1$, we obtain the standard Yule-Walker equations based on second-order statistics. However, in this case the equations do not hold due to the contribution of the additive noise. This explains why the parameter initialization breaks down at low SNR. For $M \geq 2$, we obtain additional Yule-Walker type equations that are insensitive to the presence of additive Gaussian noise. These equations appear to be very useful if the additive noise is "more Gaussian" than the speech signal in the sense that its higher-order cumulants are relatively small in magnitude (this assumption is verified in Sect. 8.7).

In practice the cumulants are approximated by substituting the unavailable ensemble averages with sample averages, thus obtaining a set of linear equations that may be used to compute the AR parameters $\alpha_1, \dots, \alpha_p$ directly from the observed signal $z(t)$. The spectral level g_s of the excitation signal may be computed by applying a whitening filter using the estimated AR parameters.

Since the equations are satisfied for all $M \geq 2$ and any combinations of lags $l_1, l_2, \dots, l_M$, we have an over-determined set of equations that may be used to improve numerical and statistical stability of the resulting parameters estimates.

Experimental results using actual speech signal in several typical noise environments indicated that at low SNR conditions, below 5 dB, using fourth-order cumulants ($M = 3$) one typically obtains a better and more robust initial estimate of the speech parameters as compared with the conventional LPC approach based on second-order statistics. The use of third-order cumulants ($M = 2$), as suggested in [47], was not that effective.

We also tried to incorporate HOS into the iterative algorithm, and not merely as an initialization tool. For that purpose consider (8.20) for $M \geq 1$. By using the cumulants of the enhanced speech signal, $\hat{s}(t)$, in (8.20) we obtain an iterative algorithm that employs HOS to iteratively estimate the speech AR parameters. However, experiments showed that the resulting algorithm produces low quality enhanced speech, with reduced bandwidth formants. Similar observation was noted by Masgrau *et al.* [8], when incorporating third order statistics into the approximated EM, Wiener filter-based algorithm of [4].

Note that the incorporation of HOS implies a nonlinear framework. This issue is elaborated on in the context of the application of the UKF in Sect. 8.6.

8.5 Gradient-Based Sequential Algorithm

The iterative-batch EM algorithm requires the use of an analysis window over which the signal and noise statistics are assumed to be stationary. To avoid this assumption, we now suggest a sequential speech enhancement algorithm which is no longer an EM algorithm. The resulting sequential algorithm is computationally more efficient than the iterative-batch algorithm. Another benefit of the sequential algorithm is that it is delay-less, unlike the iterative-batch algorithm that has an inherent delay of one processing window frame.

Following the considerations in [15] (see also [16], that considers the two-channel case) we obtain in Appendix B the following sequential speech enhancement algorithm. This algorithm consists of a forward Kalman filter, given by (8.10)–(8.13), whose parameters are continuously updated according to:

$$\hat{\boldsymbol{\alpha}}(t+1) = \hat{\boldsymbol{\alpha}}(t) - \frac{\rho_s}{g_s}\left[Q^s_{12}(t) + Q^s_{11}(t)\hat{\boldsymbol{\alpha}}(t)\right], \tag{8.21}$$

$$\hat{g}_s(t+1) = \frac{1-\lambda_s}{1-\lambda_s^t}\left[Q^s_{22}(t) + \hat{\boldsymbol{\alpha}}^T(t)Q^s_{12}(t)\right], \tag{8.22}$$

$$\hat{\boldsymbol{\beta}}(t+1) = \hat{\boldsymbol{\beta}}(t) - \frac{\rho_v}{g_v}\left[Q^v_{12}(t) + Q^v_{11}(t)\hat{\boldsymbol{\beta}}(t)\right], \tag{8.23}$$

$$\hat{g}_v(t+1) = \frac{1-\lambda_v}{1-\lambda_v^t}\left[Q^v_{22}(t) + \hat{\boldsymbol{\beta}}^T(t)Q^v_{12}(t)\right], \tag{8.24}$$

where $Q^s(t)$, $Q^v(t)$ are defined by

$$\begin{aligned} Q^s(t) &= \begin{bmatrix} Q^s_{11}(t) & Q^s_{12}(t) \\ Q^s_{21}(t) & Q^s_{22}(t) \end{bmatrix} \\ &= \sum_{\tau=0}^{t} \lambda_s^{t-\tau}\, \widehat{\boldsymbol{s}_{p+1}(\tau)\boldsymbol{s}^T_{p+1}(\tau)} = \lambda_s Q^s(t-1) + \widehat{\boldsymbol{s}_{p+1}(t)\boldsymbol{s}^T_{p+1}(t)}, \end{aligned}$$

$$Q^v(t) = \begin{bmatrix} Q^v_{11}(t) \; Q^v_{12}(t) \\ Q^v_{21}(t) \; Q^v_{22}(t) \end{bmatrix}$$
$$= \sum_{\tau=0}^{t} \lambda_v^{t-\tau} \, \widehat{\boldsymbol{v}_{q+1}(\tau)\boldsymbol{v}^T_{q+1}(\tau)} = \lambda_v Q^v(t-1) + \widehat{\boldsymbol{v}_{q+1}(t)\boldsymbol{v}^T_{q+1}(t)} \; .$$

$Q^s_{11}(t)$ is a $p \times p$ matrix, $Q^s_{12}(t) = \left(Q^s_{21}(t)\right)^T$ is a $p \times 1$ vector, and $Q^s_{22}(t)$ is a scalar value. Similarly, $Q^v_{11}(t)$ is a $q \times q$ matrix, $Q^v_{12}(t) = \left(Q^v_{21}(t)\right)^T$ is a $q \times 1$ vector, and $Q^v_{22}(t)$ is a scalar value. $0 \le \lambda_s, \lambda_v \le 1$ are exponential weighting factors and ρ_s, ρ_v are the update step sizes.

An improvement in the convergence behavior of the algorithm is obtained by normalizing the step sizes, i.e. using:

$$\hat{\boldsymbol{\alpha}}(t+1) = \hat{\boldsymbol{\alpha}}(t) - \rho_s \frac{Q^s_{12}(t) + Q^s_{11}(t)\hat{\boldsymbol{\alpha}}(t)}{||Q^s_{12}(t) + Q^s_{11}(t)\hat{\boldsymbol{\alpha}}(t)||},$$
$$\hat{\boldsymbol{\beta}}(t+1) = \hat{\boldsymbol{\beta}}(t) - \rho_v \frac{Q^v_{12}(t) + Q^v_{11}(t)\hat{\boldsymbol{\beta}}(t)}{||Q^v_{12}(t) + Q^v_{11}(t)\hat{\boldsymbol{\beta}}(t)||}.$$

We can further simplify the algorithm by setting $\lambda_s = \lambda_v = 0$, i.e. estimating the respective correlation matrices, $Q^s(t)$ and $Q^v(t)$, using only the current state-vectors. This simplification leads to a *least-mean-square* (LMS) type algorithm.

8.6 All-Kalman Speech and Parameter Estimation

Until now, our discussion was limited to the non-Bayesian framework. The parameters of the problem (i.e. $\boldsymbol{\alpha}, \boldsymbol{\beta}, g_s, g_v$) were assumed to be deterministic (although time varying). For this reason, the *maximum likelihood* solution for the parameters could be obtained via the *estimate-maximize* algorithm and a sequential variant thereof could be obtained. However, carefully looking at the gradient-based sequential algorithm [depicted in (8.10)–(8.13) and (8.21)–(8.24)] an interesting interpretation can be derived. Since both signals and parameters are continuously updated, other parameter adaptation mechanisms could be applied, while maintaining the forward-recursive structure of the algorithm. As the non-Bayesian framework was merely an instrument for deriving the sequential algorithm, it is not obligatory for deriving a sequential variant. The Bayesian framework could be used instead. It is therefore proposed to replace the gradient-based parameter search by a Kalman-based adaptation. The obtained algorithm which apply the Kalman filter both to the signal enhancement stage and the parameter estimation stage will be denoted all-Kalman recursion. Two variants of the all-Kalman based algorithm are addressed, denoted *dual* scheme and *joint* scheme.

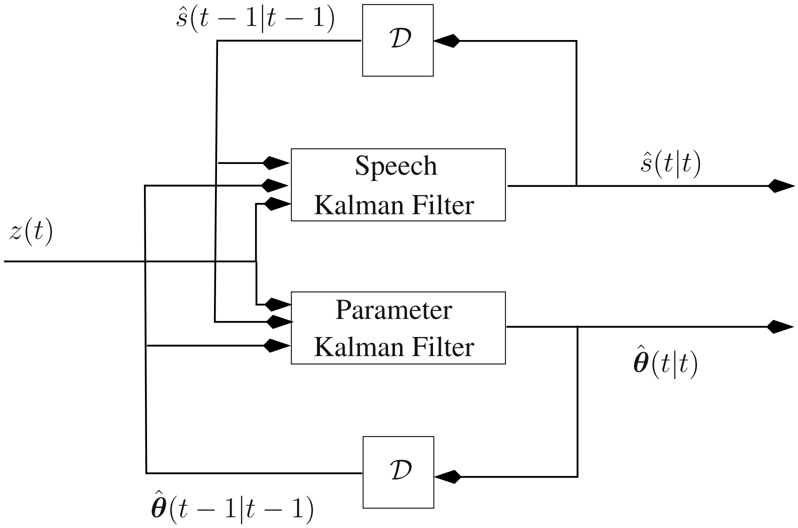


Fig. 8.2. All-Kalman speech and parameter estimation. *Dual* estimation procedure.

8.6.1 Dual Scheme

The first variant of the algorithm in the Bayesian framework follows. In this algorithm, we maintain the structure of the gradient-based algorithm by just replacing the parameter adaptation by a Kalman filter. The resulting method, referred to by Wan *et al.* [31] as *dual* estimation method, is comprised of two steps. In each time instant a speech Kalman filter step is applied based on the current estimate of the parameters. In parallel a parameter Kalman-based estimate step is applied based on the current signal-state estimate. The concept is depicted in Fig. 8.2. At time instant t, $\hat{s}(t|t)$ is calculated by applying the speech Kalman filter to the previous speech estimate $\hat{s}(t|t)$ using the parameter set $\hat{\boldsymbol{\theta}}(t-1|t-1)$. In parallel, another Kalman filter is applied to obtain a new parameter estimate $\hat{\boldsymbol{\theta}}(t|t)$, based on the previous parameter estimate $\hat{\boldsymbol{\theta}}(t-1|t-1)$ and the speech estimate $\hat{s}(t-1|t-1)$.

Signals Model. For the speech signal we assume that the state-space model given in (8.6) holds. For brevity of the exposition we assume that the additive noise is white, i.e. its AR order is set to $q = 0$. Thus, the noise model is simply $v(t) = g_v u_v(t)$.

Parameter Model. In the Bayesian framework the parameters are given a dynamic model. Since we do not have any a priori knowledge on this dynamics we assume a very simple process, namely (almost) Brownian motion. Finding a better model is still an open research issue. Define the parameter vector

$$\boldsymbol{\alpha}^T(t) = [\,\alpha_1(t)\ \alpha_2(t)\ \ldots \alpha_p(t)\,],$$

and the innovation vector

$$\boldsymbol{u}_{\boldsymbol{\alpha}}^T(t) = \left[\,u_{\alpha_1}(t)\ u_{\alpha_2}(t)\ \ldots\ u_{\alpha_p}(t)\,\right],$$

with the respective covariance matrix

$$Q_{\boldsymbol{\alpha}}(t) = E\{\boldsymbol{u}_{\boldsymbol{\alpha}}(t)\boldsymbol{u}_{\boldsymbol{\alpha}}^T(t)\}.$$

The parameter state-space equations are:

$$\begin{aligned} \boldsymbol{\alpha}(t) &= \Phi_{\boldsymbol{\alpha}}\boldsymbol{\alpha}(t-1) + \boldsymbol{u}_{\boldsymbol{\alpha}}(t), \\ z(t) &= \boldsymbol{h}_{\boldsymbol{\alpha}}^T(t)\boldsymbol{\alpha}(t) + \boldsymbol{g}_s(t)u_s(t) + v(t), \end{aligned} \tag{8.25}$$

where

$$\boldsymbol{h}_{\boldsymbol{\alpha}}^T(t) = \left[\,s(t-1)\ s(t-2)\ \ldots\ s(t-p)\,\right]$$

is comprised of speech signal samples and $\Phi_{\boldsymbol{\alpha}} = I_{p\times p}$ or very close to it.

Speech Kalman Filter. Using the speech state-model (8.3) and the current estimate of the parameter set (obtained by the, running in parallel, parameters Kalman filter) we obtain the following speech Kalman filter.

Propagation equations:

$$\hat{\boldsymbol{s}}_p(t|t-1) = \hat{\Phi}_s(t-1)\hat{\boldsymbol{s}}_p(t-1|t-1), \tag{8.26}$$

$$\begin{aligned} P(t|t-1) &= \hat{\Phi}_s(t-1|t-1)P(t-1|t-1)\hat{\Phi}_s^T(t-1|t-1) \\ &\quad + \hat{\boldsymbol{g}}_s(t-1|t-1)\hat{\boldsymbol{g}}_s^T(t-1|t-1). \end{aligned} \tag{8.27}$$

Kalman gain:

$$\boldsymbol{k}(t) = \frac{P(t|t-1)\boldsymbol{h}_s}{\boldsymbol{h}_s^T P(t|t-1)\boldsymbol{h}_s + \hat{g}_v^2(t-1|t-1)}. \tag{8.28}$$

Update equations:

$$\hat{\boldsymbol{s}}_p(t|t) = \hat{\boldsymbol{s}}_p(t|t-1) + \boldsymbol{k}(t)\left[z(t) - \boldsymbol{h}_s^T\hat{\boldsymbol{s}}_p(t|t-1)\right], \tag{8.29}$$

$$P(t|t) = P(t|t-1) - \boldsymbol{k}(t)\left[\boldsymbol{h}_s^T P(t|t-1)\boldsymbol{h}_s + \hat{g}_v^2(t-1|t-1)\right]\boldsymbol{k}^T(t). \tag{8.30}$$

Parameters Kalman Filter. Using the parameters state-model (8.25) and the current estimate of the speech signal (obtained by the, running in parallel, speech Kalman filter) we obtain the following parameters Kalman filter.

Propagation equations:

$$\begin{aligned}\hat{\boldsymbol{\alpha}}(t|t-1) &= \varPhi_{\boldsymbol{\alpha}}\hat{\boldsymbol{\alpha}}(t-1|t-1), && (8.31)\\ P_{\boldsymbol{\alpha}}(t|t-1) &= \varPhi_{\boldsymbol{\alpha}}P_{\boldsymbol{\alpha}}(t-1|t-1)\varPhi_{\boldsymbol{\alpha}}^T + Q_{\boldsymbol{\alpha}}. && (8.32)\end{aligned}$$

Kalman gain:

$$\boldsymbol{k}_{\boldsymbol{\alpha}}(t) = \frac{P_{\boldsymbol{\alpha}}(t|t-1)\hat{\boldsymbol{h}}_{\boldsymbol{\alpha}}(t|t)}{\hat{\boldsymbol{h}}_{\boldsymbol{\alpha}}^T(t|t)P_{\boldsymbol{\alpha}}(t|t-1)\boldsymbol{h}_{\boldsymbol{\alpha}}(t|t) + \hat{g}_s^2(t-1|t-1) + \hat{g}_v^2(t-1|t-1)}. \quad (8.33)$$

Update equations:

$$\begin{aligned}\hat{\boldsymbol{\alpha}}(t|t) &= \hat{\boldsymbol{\alpha}}(t|t-1) + \boldsymbol{k}_{\boldsymbol{\alpha}}(t)\left[z(t) - \hat{\boldsymbol{h}}_{\boldsymbol{\alpha}}^T(t|t)\hat{\boldsymbol{\alpha}}(t|t-1)\right], && (8.34)\\ P_{\boldsymbol{\alpha}}(t|t) &= P_{\boldsymbol{\alpha}}(t|t-1) - \boldsymbol{k}_{\boldsymbol{\alpha}}(t)\left[\hat{\boldsymbol{h}}_{\boldsymbol{\alpha}}^T(t|t)P_{\boldsymbol{\alpha}}(t|t-1)\hat{\boldsymbol{h}}_{\boldsymbol{\alpha}}(t|t)\right. && (8.35)\\ &\qquad \left.+\hat{g}_s^2(t-1|t-1) + \hat{g}_v^2(t-1|t-1)\right]\boldsymbol{k}_{\boldsymbol{\alpha}}^T(t).\end{aligned}$$

$\hat{g}_s^2(t|t)$ may be estimated similarly to $\hat{\boldsymbol{\alpha}}(t|t)$ by applying another Kalman filter, or by recursively calculating the obtained innovation sequence gain. $\hat{g}_v^2(t|t)$ could be estimated by averaging over non-speech samples, using decisions obtained by VAD.

8.6.2 Joint Scheme

Consider the speech and parameters state-space presentations given by (8.3) and (8.25). In our Bayesian framework, both speech and parameters are assumed to be stochastic processes. Note, that (8.3) involve a multiplication of these processes. Define an augmented state-vector comprised of the speech signal and the parameters $\boldsymbol{\eta}^T(t) = \left[\boldsymbol{s}_p^T(t)\,\boldsymbol{\alpha}^T(t)\right]$ and an augmented innovation vector $\boldsymbol{u}^T(t) = \left[\boldsymbol{g}_s(t)u_s(t)\,\boldsymbol{u}_{\boldsymbol{\alpha}}(t)\right]$. The state-space equation becomes nonlinear:

$$\begin{aligned}\boldsymbol{\eta}(t) &= \underbrace{\begin{bmatrix}\varPhi_s & \mathbf{0}\\ \mathbf{0} & \varPhi_{\boldsymbol{\alpha}}\end{bmatrix}\boldsymbol{\eta}(t-1)}_{\text{nonlinearity}} + \boldsymbol{u}(t), && (8.36)\\ z(t) &= \begin{bmatrix}1 & 0 & 0 & \ldots & 0\end{bmatrix}\boldsymbol{\eta}(t) + v(t).\end{aligned}$$

Thus, in generalized notation, the nonlinear transition and measurement equations are given by,

$$\begin{aligned}\boldsymbol{\eta}(t) &= \varPhi\left(\boldsymbol{\eta}(t-1), \boldsymbol{u}(t)\right), && (8.37)\\ \boldsymbol{z}(t) &= \boldsymbol{h}\left(\boldsymbol{\eta}(t-1), \boldsymbol{v}(t)\right).\end{aligned}$$

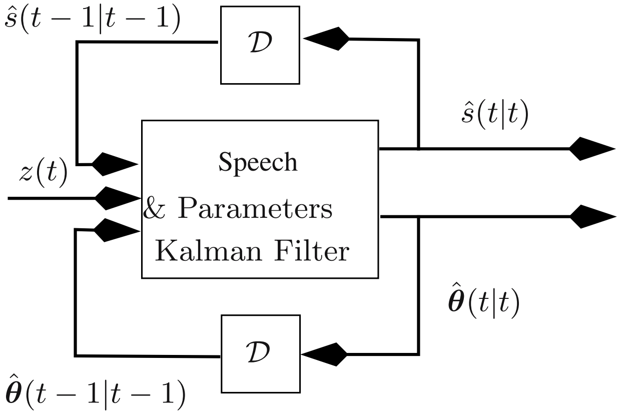
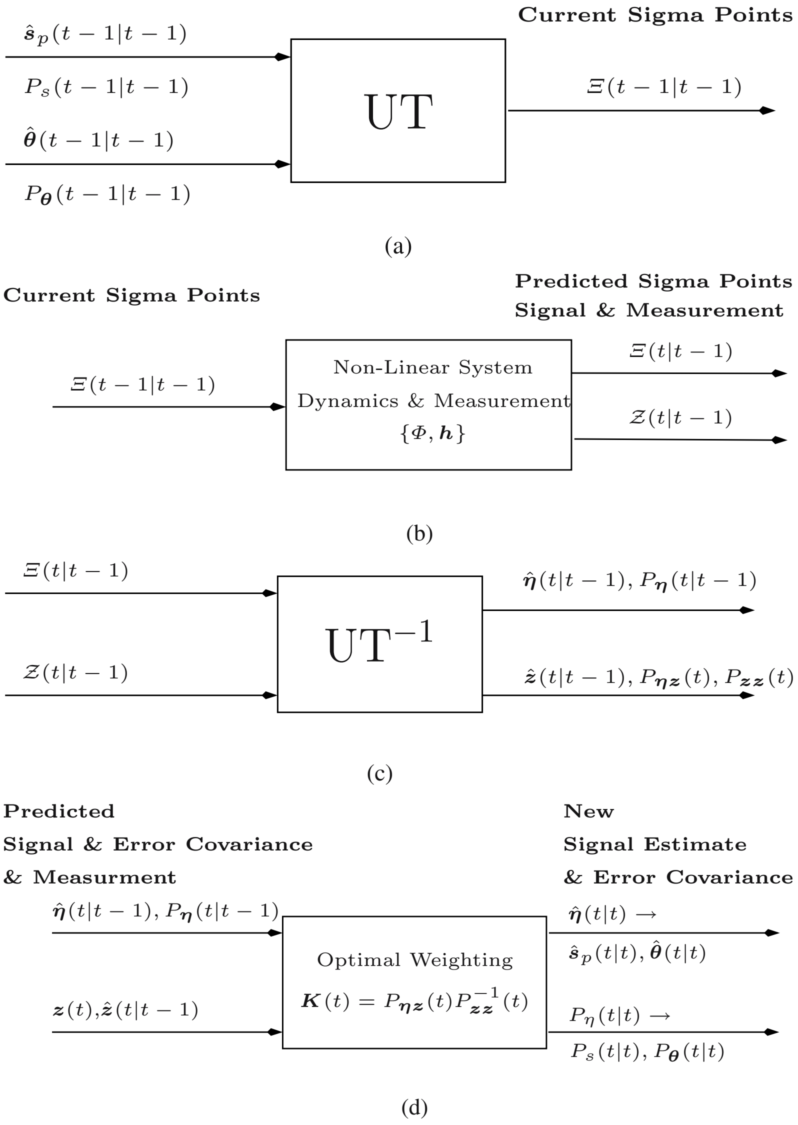


Fig. 8.3. All-Kalman speech and parameter estimation. *Joint* estimation procedure.

If we wish to apply the Kalman filter we will need a nonlinear modification thereof. Following Wan *et al.* [31], we will define the *joint* scheme for both speech and parameters estimation. The system is depicted in Fig. 8.3. The two components of $\hat{\boldsymbol{\eta}}(t|t)$, namely $\hat{s}(t|t)$ and $\hat{\boldsymbol{\theta}}(t|t)$, are jointly obtained by applying an approximated Kalman filter. We will show in the sequel that this filter should tackle with the inherent nonlinearity of the system. In the past, the *extended Kalman filter* (EKF), based on the linearization of the equations, was used. This method might be quite cumbersome, as it involves the calculation of derivatives, but yet it is not accurate enough, as only first-order approximation is applied.

The Unscented Kalman Filter. A better method for linearizing the nonlinear state equations was proposed by Julier and Uhlmann in [30]. This extension to the Kalman filter is making use of the *unscented transform* summarized in Appendix C. Figure 8.4 summarizes the steps involved in unscented Kalman filter (UKF). The method consists of calculating the mean and covariance of the augmented state-vector, given in (8.37), undergoing a known nonlinear transform by using the *unscented transform.* Denote by $\hat{\boldsymbol{\eta}}(t-1|t-1)$ the current state-vector estimate and by $P_{\boldsymbol{\eta}\boldsymbol{\eta}}(t-1|t-1)$ its respective covariance. The method is comprised of four stages. In stage (a), $\hat{\boldsymbol{\eta}}(t-1|t-1)$ is split into σ-points $\Xi(t-1|t-1)$ approximating the probability density function of the vector. By using this method the mean and covariance better propagate through the nonlinearities. However, it must be stressed that no claims of optimality holds. Then, in stage (b), each of the σ-points is undergoing the known nonlinearity yielding the σ-points of the *predicted* state-vector, $\Xi(t|t-1)$. The σ-points of the predicted noisy measurement, $\mathcal{Z}(t|t-1)$, are calculated as well. In step (c), the σ-points are


$\hat{\boldsymbol{s}}_p(t-1|t-1)$
$P_s(t-1|t-1)$
$\hat{\boldsymbol{\theta}}(t-1|t-1)$
$P_{\boldsymbol{\theta}}(t-1|t-1)$
UT
Current Sigma Points
$\Xi(t-1|t-1)$
Current Sigma Points
$\Xi(t-1|t-1)$
Non-Linear System
Dynamics & Measurement
$\{\Phi, \boldsymbol{h}\}$
Predicted Sigma Points
Signal & Measurement
$\Xi(t|t-1)$
$\mathcal{Z}(t|t-1)$
$\Xi(t|t-1)$
$\mathcal{Z}(t|t-1)$
UT^{-1}
$\hat{\boldsymbol{\eta}}(t|t-1), P_{\boldsymbol{\eta}}(t|t-1)$
$\hat{\boldsymbol{z}}(t|t-1), P_{\boldsymbol{\eta z}}(t), P_{\boldsymbol{zz}}(t)$
Predicted
Signal & Error Covariance
& Measurment
$\hat{\boldsymbol{\eta}}(t|t-1), P_{\boldsymbol{\eta}}(t|t-1)$
$\boldsymbol{z}(t),\hat{\boldsymbol{z}}(t|t-1)$
Optimal Weighting
$\boldsymbol{K}(t) = P_{\boldsymbol{\eta z}}(t)P_{\boldsymbol{zz}}^{-1}(t)$
New
Signal Estimate
& Error Covariance
$\hat{\boldsymbol{\eta}}(t|t) \rightarrow$
$\hat{\boldsymbol{s}}_p(t|t), \hat{\boldsymbol{\theta}}(t|t)$
$P_\eta(t|t) \rightarrow$
$P_s(t|t), P_{\boldsymbol{\theta}}(t|t)$

(a)

(b)

(c)

(d)

Fig. 8.4. Unscented Kalman filter: (a) Unscented transform. (b) Propagation equations. (c) Inverse unscented transform. (d) Update equations.

collected together yielding the predicted values $\hat{\boldsymbol{\eta}}(t|t-1)$ and $\hat{\boldsymbol{z}}(t|t-1)$. This concludes the propagation stage of the UKF. In step (d), similar to the conventional filter, the Kalman gain is calculated by $\boldsymbol{K}(t) = P_{\boldsymbol{\eta z}}(t)P_{\boldsymbol{zz}}^{-1}(t)$. Note, that the covariance matrices estimates are obtained by the UT. Finally, the update stage is implemented by properly weighting the predicted values and the current measurement yielding the new estimate $\hat{\boldsymbol{\eta}}(t|t)$.

The complexity of the suggested method is quite low. Suppose that the dimension of $\hat{\boldsymbol{\eta}}(t|t)$ is L. Then, only an increase of the computational load by a factor of $2L+1$ compared with the standard Kalman filter is required. The obtained UKF, although not optimal, is a better and more sophisticated linearization of the nonlinear system.

8.7 Experimental Study

In this section, we provide a performance evaluation of the proposed iterative-batch algorithm, denoted for brevity *Kalman estimate-maximize* (KEM) and its sequential version *Kalman gradient descent* (KGD) algorithm. Both objective and subjective tests are conducted. The performance of these algorithms is compared with the following algorithms:

1. The *log spectral amplitude* (LSA) estimator, suggested by Ephraim and Malah [41].
2. The optimally-modified LSA[1], proposed by Cohen and Berdugo [42] and denoted hereinafter OM-LSA.
3. The HMM-based speech enhancement algorithm suggested by Ephraim *et al.* [34], [35].
4. The Wiener-EM algorithm of Lim and Oppenheim [4], denoted WEM.
5. The Mixture-Maximum algorithm proposed by Burshtein and Gannot [43], denoted MixMax.

Comparison with other algorithms could be found in [23].

The organization of this section is as follows. We start by defining the experiment setup. We proceed by assessing the Gaussian assumption for the noise signal, assumption that forms the basis of the initialization procedure of the KEM version. Some objective tests are then defined and presented, namely the total SNR, the weighted segmental SNR, the noise reduction level and the log spectral distance. The word recognition rate of an *automatic speech recognition* (ASR) system is used as another figure-of-merit for evaluating the enhancement algorithms. In speech processing field, the best assessment is obtained by conducting subjective tests. For this purpose, we provide speech spectrograms and describe informal listening tests. Finally, a preliminary experimental study is conducted for the UKF, showing the potential of the method.

8.7.1 Experimental Setup

In the following experiments, five iterations are required for the KEM algorithm to converge. The AR order used to model the speech signal is $p = 10$.

[1] The author would like to thank Dr. I. Cohen of the Technion–IIT for providing the OM-LSA code.

The frame size is 16 mSec, although small changes in this value did not degrade the performance. The analysis frames are non-overlapping (overlapping frames did not yield improved performance). In all experiments (unless otherwise stated), we used the fixed-lag smoothing version. The HMM algorithm was based on the *minimum mean square error* (MMSE) criterion, since it was reported to be superior over the alternative maximum a-posteriori (MAP) criterion [35]. The WEM algorithm was the RLMAP variant in [4], since it yielded the best results. The OM-LSA gain floor and forgetting factor were adjusted according to the desired output characteristics. In the MixMax algorithm the post-processing level was set according to the amount of affordable distortion. The speech signal was modelled by using 40 Gaussian mixtures.

We used speech signal drawn from TIMIT [48] or TIDIGITS [49], depending on the experiment conducted. All signals were down-sampled to 8 kHz. The speech signal was degraded by additive noise at various SNR levels. As for the noise sources, we used signals drawn from the NOISEX-92 [50] database together with computer-fan noise signal, recorded in our lab, and computer generated white Gaussian source. In the KEM algorithm, an AR process of order $q = 4$ was used for modelling the non-white noise signals. For the white noise, we used $q = 1$ to allow deviation from the exact model due to short data-segments.

8.7.2 Verifying the Gaussian Assumption

We first want to assess the assumption that the noise signals are more Gaussian than the speech signal. Recall, that this assumption led to the HOS-based initialization method, used by the KEM method. In Fig. 8.5, we assess the validity of the Gaussian approximation, by plotting the empirical *cumulative distribution function* (CDF) of four signals, 125 mSec long each. The first is a speech-like noise drawn from the NOISEX-92 database [50], the second is a factory noise from the same database, and the others are voiced and unvoiced segments of a speech signal drawn from TIMIT database [48]. The vertical axis employs a nonlinear division of the interval $[0, 1]$, such that the vertical coordinate of a sample with CDF c, is $y = \Psi(c)$, where Ψ is the CDF of a Gaussian random variable, whose mean and variance are the empirical mean and variance of the given signal segment. Hence, a Gaussian random variable corresponds to a straight line (presented by a dashed line in the figure). As depicted in Fig. 8.5, the noise segments are very close to an ideal Gaussian curve. Unvoiced speech segments possess similar curves, although not as close to Gaussian distribution as the noise signals. Opposed to that, the voiced segment deviates significantly from the Gaussian curve. It should be stressed, however, that the difference between the voiced speech signal and the other waveforms is not as large when shorter segments are used. Therefore, the benefit arising from using HOS-based initialization is expected to be less significant.

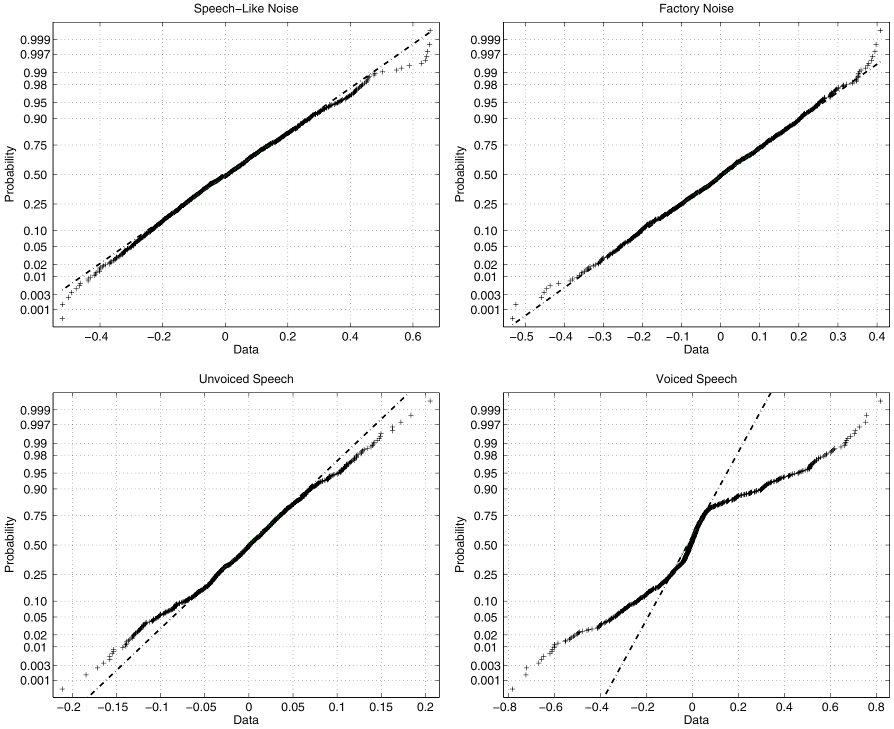


Fig. 8.5. Gaussian curves for speech-like noise (upper-left), factory noise (upper-right), unvoiced speech segment (lower-left) and voiced speech segment (lower-right).

8.7.3 Objective Evaluation

We used two types of objective tests. The first measure relates directly to the waveform properties, while the other is an indirect measure of the enhancement capabilities, namely the increase in detection rate of an ASR system.

Waveform Assessment. Define $y(t)$ to be the signal to be assessed (noisy signal or one of the algorithms' output) and recall that $s(t)$ is the desired speech signal. Four objective quality measures were used to assess the algorithms' performance.

The first is the total output SNR defined by

$$\mathrm{SNR} = \frac{\sum_t s^2(t)}{\sum_t \left(s(t) - y(t)\right)^2}, \tag{8.38}$$

where the time summations are over the entire duration of the signals. Although this distortion measure is not very correlative with speech quality, it is still informative.

The second objective quality measure is the *noise level* (NL) during nonactive speech periods, defined as,

$$\mathrm{NL} = \mathrm{Median}_n \left\{ 10\log_{10}\left(E(n)\right) \;\; n \in \text{Speech Nonactive} \right\}, \tag{8.39}$$

where $E(n) = \sum_{\tau \in T_n} y^2(\tau)$, and T_n are the time instances corresponding to segment number n. Note that the lower the NL figures are, the better the result obtained by the respective algorithm is.

The third figure-of-merit is the *weighted segmental SNR* (W-SNR). This measure applies weights to the segmental SNR within frequency bands. The frequency bands are spaced proportionally to the ear's critical bands, and the weights are constructed according to the perceptual quality of speech. Define, $S(t, B_k)$ and $Y(t, B_k)$ to be the clean speech signal and the signal to be assessed at frequency band B_k, respectively. Now, define $\mathrm{SNR}(n, B_k) = \frac{\sum_{\tau \in T_n} Y^2(\tau, B_k)}{\sum_{\tau \in T_n}(Y(\tau,B_k)-S(\tau,B_k))^2}$ the SNR in segment number n and frequency band B_k. W-SNR is defined as,

$$\begin{aligned} &\text{W-SNR} = \\ &\mathrm{Median}_n \left\{ 10\log_{10}\left(\sum_k W(B_k)\mathrm{SNR}(n, B_k) \right) \;\; n \in \text{Speech Active} \right\}. \end{aligned} \tag{8.40}$$

The frequency bands B_k and their corresponding *importance weights* $W(B_k)$ are according to the ANSI standard [51]. Studies have shown that the W-SNR measure is more closely related to a listener's perceived notion of quality than the classical SNR or segmental SNR [52].

The fourth objective speech quality measure, which is with better correlation with the *mean opinion score* (MOS) than the other mentioned distortion measures, is the *log spectral distance* (LSD) defined by,

$$\begin{aligned} \mathrm{LSD} = \mathrm{Median}_n \Big\{ &\sqrt{\mathrm{Mean}_\omega \left\{ [20\log_{10}|S(n, e^{j\omega})| - 20\log_{10}|Y(n, e^{j\omega})|]^2 \right\}}, \\ &n \in \text{Speech Active} \}. \end{aligned} \tag{8.41}$$

$S(t, e^{j\omega})$ and $Y(t, e^{j\omega})$ are the *short-time Fourier transforms* (STFT) of the input and the assessed signals, respectively. Note, that a lower LSD level corresponds to a better performance.

In the last three figures-of-merit, we used the median value of the individual frames readings. Median averaging eliminates outliers, and is therefore superior to the common definition involving simple averaging.

For the evaluation depicted in Fig. 8.6, we used 50 sentences drawn from TIMIT database (25 uttered by male speakers and 25 – by female speakers) contaminated by speech-like noise drawn from NOISEX-92 database at various SNR levels. The four figures-of-merit, (8.38)–(8.41) were calculated separately for each sentence and averaged over all 50 sentences. It is clearly shown that the best results were obtained by the OM-LSA algorithm in all comparison parameters. It should be noted, however, that in this setup the

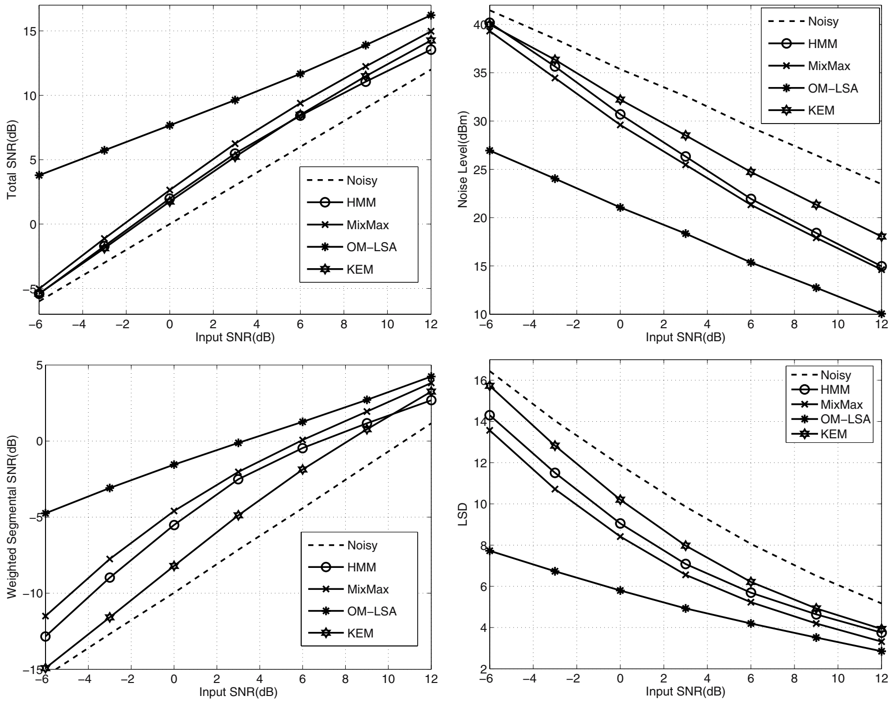


Fig. 8.6. Figures-of-merit for 50 TIMIT sentences contaminated by speech-like noise.

OM-LSA was tuned to a working point in which large noise reduction is traded-off for increased speech distortion. The MixMax algorithm was tuned to low distortion output. We will elaborate on the working point of the algorithm in the subjective test section. The KEM algorithm was found to be inferior to all other tested algorithms, although the difference in performance was not so crucial. Similar trends were observed in factory noise signal (from NOISEX-92) and the white Gaussian noise.

Automatic Speech Recognition Tests. ASR systems are very sensitive to additive noise and speech distortion. Therefore, detection rate of such systems can be used as another method for comparing the performance of several enhancement algorithms.

We used an ASR system[2] using a continuous density HMM-based speech recognition system. The acoustic front-end is comprised of 8 cepstral values and their time derivatives, computed by using standard LPC analysis. The resulting 16-dimensional feature vector is modelled by a mixture of 3

[2] The author would like to thank Dr. J. Goldberger of Bar-Ilan University for providing the speech recognition software.

Table 8.1 Single digits recognition rate.

SNR	Noisy	KEMI	LSAE	HMM
0	71.2	87.1	91.2	72.5
3	78.3	92.5	95.0	87.5
6	84.6	97.1	97.5	93.3
9	91.7	97.5	97.5	95.0
12	97.5	97.9	98.3	96.2
15	97.5	98.3	97.9	97.9

diagonal-covariance Gaussians. Each word in the vocabulary was modelled by one, 5-state, left-to-right HMM. Training was performed using the Baum-Welch algorithm. The decoder was a conventional Viterbi algorithm.

The speech database was the speaker independent, high quality connected digits recorded at TI [49]. This database is divided into training and testing digit strings uttered by 225 adult talkers. The single digits (word) recognition rate of the system, when tested on the clean isolated digits sentences, was 99.1%. The noise signal for this experiment was computer-fan noise, recorded at our lab. This noise is typical to office environments. The designated digits were contaminated by the noise signal at various SNR levels. The SNR was measured in the frequency region of interest, between 200 Hz and 3200 Hz.

The single digits recognition rate of the system when subject to speech signals contaminated by the noise is summarized in Table 8.1. We also show the corresponding recognition rate, when the noisy speech is pre-processed by the KEM algorithm (fixed-lag smoothing version; the filtered version was slightly inferior), by the LSA algorithm [41], and by the HMM-MMSE algorithm [34]. As can be seen, the KEM algorithm improves the performance by about 6dB (i.e., when speech enhancement is not employed, the SNR needs to be increased by 6dB in order to obtain the same recognition rate). The KEM algorithm shows superior performance compared to the HMM-MMSE algorithm, and is comparable to the LSA at input SNR levels higher than 6dB. Below 6dB input SNR, the LSA is superior to the KEM algorithm.

8.7.4 Subjective Evaluation

A useful subjective quality measure is the assessment of speech spectrograms (sonograms). Figure 8.7 shows the spectrograms of some clean speech segment, the corresponding noisy segment, and the outcome of several enhancement algorithms, namely the HMM, MixMax, OM-LSA and KEM (fixed lag smoothing version).

As can be seen, the OM-LSA, the KEM, and especially the MixMax algorithms seem to preserve the speech detailed structure. However, the KEM

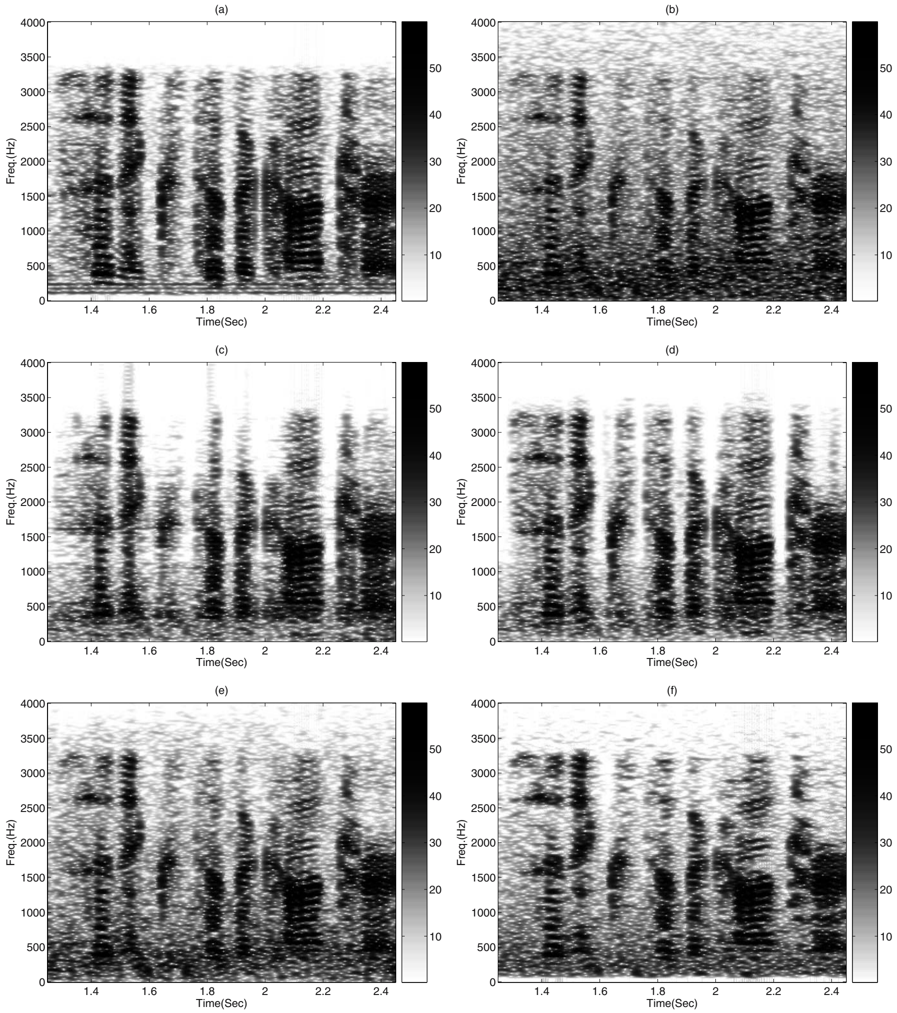


Fig. 8.7. Sonograms of "jokes cartoons and cynics to the contrary mothers in law make good friends" for (a) clean, (b) noisy, (c) HMM, (d) MixMax, (e) KEM, and (f) OM-LSA algorithms. SNR 5dB speech-like noise.

algorithm is inferior to other algorithms in terms of noise reduction. As depicted in Fig. 8.7, the HMM algorithm output has distorted speech sonogram. The HMM algorithm is therefor inferior to other algorithms in respect to speech quality. This observation was also noted in [35]. These results corresponds also to informal listening tests, conducted in our lab. The KEM algorithm preserve the speech natural sound. However its noise reduction is lower than that obtained by the MixMax algorithm and especially by the OM-LSA algorithm. Some speech samples can be found in [53]. As mentioned before, we should emphasize that the noise reduction ability of both OM-LSA and MixMax algorithms can be sacrificed for the sake of more natural speech outcome. Overall, the state-of-the-art OM-LSA algorithm has the best noise suppression ability while keeping the speech distortion level sufficiently low. The MixMax algorithm outcome has the most natural sound characteristics. The KEM performance has limited performance compared to these algorithms but it is superior to the HMM-MMSE method. These phenomena are demonstrated in [53] as well.

8.7.5 Comparison Between EM-Based Algorithms

We also test the variants of the EM-based algorithm, namely the Wiener filter based WEM algorithm, the Kalman filter based KEM algorithm and its sequential version, the KGD.

According to our informal listening rests, at SNR levels below 5dB the speech quality of the KGD algorithm is only slightly inferior to that of the KEM. Above 5dB the KGD algorithm tends to be unstable, with a time-varying signal level. We attribute this phenomenon to the fact that at high SNR levels the estimated noise model parameters might be very inaccurate (since the noise is masked by the signal). A possible solution is to replace the sequential update equation of the noise parameters by an estimator that, based on a voice activity detector, considers only signal segments where speech activity is not detected.

A comparison between the filtered output and the fixed-lag smoothed output showed a slight advantage to the former. However, the fixed-lag smoothed output was sometimes characterized as being slightly muffled.

For the computer generated white Gaussian noise we also tested the WEM algorithm, that was designed under the assumption of white Gaussian noise. Our listening tests indicate some advantage to the KEM algorithm over the WEM algorithm.

8.7.6 Evaluation of the UKF

Due to convergence problems, the UKF algorithm is only capable at this stage of working with artificial signals. The performance with real speech signals is still to be determined. However, to demonstrate the potential of the method, we still provide some results.

Time varying Gaussian AR process (4 coefficients) embedded in white Gaussian noise with input SNR level of about 20dB is processed by the *joint* Kalman scheme[3]. The noise level is estimated during non-signal portions of the noisy signal. The tracking ability of the procedure is presented in Fig. 8.8. Although the obtained results are limited to simple problems, they demonstrate the potential of the method.

[3] All Simulations concerning the UKF are implemented by modifying R. van der Merwe *et al.* [54] code, written in Matlab© language.

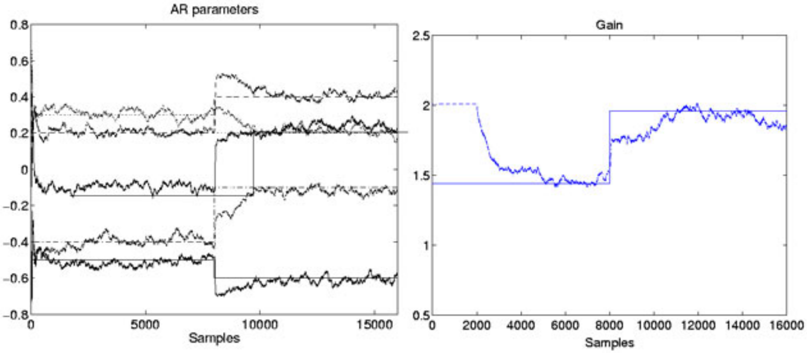


Fig. 8.8. UKF parameter tracking ability for an AR process embedded in white noise.

8.8 Conclusions

A comprehensive survey of Kalman filter based algorithms was given. Two applicable families of estimation algorithms were presented. The first involving an off-line training stage, where representative parameters (e.g. HMM) are extracted from clean speech utterances and then used in the enhancement stage. The second family, consists of online methods in which the signal and the parameters are estimated jointly from the corrupted signal. Usually, members of the second family employ the EM algorithm.

The main difference between the two approaches is that the HMM-based methods constrain the estimated speech (and noise) parameters to some codebook of possible spectra that is obtained from clean speech database. This codebook is in fact a detailed model to the speech signal. The success of the HMM-based methods depends on the accuracy of this model. A mismatch between the database used to construct the speech codebook, and the actual speech signal that needs to be enhanced, might deteriorate the quality of the enhanced signal.

Our main concern in this chapter is the EM based online algorithm. We presented iterative-batch and sequential speech enhancement algorithms, in the presence of colored background noise, and compared the performance of these algorithms with alternative speech enhancement algorithms. The iterative-batch algorithm employs the EM method to estimate the spectral parameters of the speech signal and noise process. Each iteration of the algorithm is comprised of an estimation (E) step and a maximization (M) step. The E-step is implemented by using the Kalman filtering equations. The M-step is implemented by using a non-standard Yule-Walker equation set, in which correlations are replaced by their a posteriori values, that are calculated using the Kalman filtering equations. The enhanced speech is obtained as a byproduct of the E-step. Our development assumes a colored, rather than white, Gaussian noise model. The incremental computational price that is paid for this extension is moderate. However, the realizable improvement in the enhancement performance may be quite significant.

Forth-order cumulant based equations were shown to provide a reliable initialization to the EM algorithm. Alternative initialization methods that we tried, such as third order statistics based equations, were not as effective.

The performance of this algorithm was compared to that of several state-of-the-art alternative speech enhancement algorithms in a series of evaluation tests comprised of both objective (total and weighted segmental SNR, log spectral distance, noise level, and ASR recognition rate) as well as subjective (sonograms and informal listening test) assessment. A distinct advantage of the proposed algorithm, compared to alternative algorithms is that it enhances the quality and SNR of the speech, while preserving its intelligibility and natural sound. Although superior to the HMM based algorithm, the overall performance of the Kalman filter algorithm is inferior to the more modern, MixMax and especially to the OM-LSA algorithms.

We also compared several variants of our method. Fixed-lag Kalman smoothing was superior to Kalman filtering in terms of the objective distance measures. However, our informal speech quality tests suggest the opposite conclusion (i.e., that filtering is slightly superior to fixed-lag smoothing).

In order to reduce the computational load and to eliminate the delay of the iterative-batch algorithm, the sequential algorithm may be used. Although in general, the performance of the iterative-batch algorithm is superior, at low SNR levels, the differences in performance are small.

In the Bayesian framework, the recently proposed UKF was applied to the problem of single-microphone speech enhancement. Results for simple, artificial signals, demonstrate the potential of the method. Nevertheless, for a comprehensive test, it should be further applied to real speech signals embedded in higher noise levels. Performance limitations and optimality issues of the suggested method are still open issues.

Some final remarks to conclude our survey on the use of the Kalman filter in single-microphone speech enhancement tasks.

In spite the fact that it has been used for almost two decades, there is still much to do for improving the performance of Kalman filter based algorithms. The main advantage of these algorithms stems from the fact that the Kalman filter may be continuously updated. The obtained speech has a natural sound and the residual noise is clean from annoying artifacts. However, the obtained noise suppression seems limited. This disadvantage, in our opinion, stems from the linear processing regime in which the Kalman filter is applied in (although some nonlinear extensions were mentioned). Incorporating the nonlinearity, perhaps through the use of the UKF (in conjunction with the HOS-based approach), might yield better noise reduction, while maintaining the low distortion and keeping the computational load sufficiently low.

Appendix A: Derivation of the EM Algorithm

We provide a derivation of the EM algorithm presented in Section 8.3.

Let $\boldsymbol{z}$ defined by (8.8) be the vector of corrupted speech samples (observed data) possessing the probability distribution function (PDF) $f_{\boldsymbol{Z}}(\boldsymbol{z};\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is defined by (8.7).

The maximum likelihood (ML) estimate of $\boldsymbol{\theta}$ is given by

$$\widehat{\boldsymbol{\theta}}_{ML} = \arg\max_{\boldsymbol{\theta}} \log f_{\boldsymbol{Z}}(\boldsymbol{z};\boldsymbol{\theta}). \tag{8.42}$$

Our objective is to estimate the clean speech samples $s(t)$ from the observed data $z(t)$. Such an estimate will be obtained as a byproduct of the ML parameter estimation algorithm. The solution to (8.42) is obtained by using the estimate-maximize (EM) algorithm, [3], which is a general iterative procedure for obtaining the solution to the ML optimization problem. To apply the EM algorithm we need to define a *complete data* vector $\boldsymbol{y}$ which is related to the observed data vector (*incomplete data*) through a (generally non-invertible) transformation, $F(\cdot)$, i.e.,

$$\boldsymbol{z} = F(\boldsymbol{y}). \tag{8.43}$$

The general l-th iteration of the EM algorithm consists of the following estimation (E) step and maximization (M) step,

E-step

$$Q(\boldsymbol{\theta},\widehat{\boldsymbol{\theta}}^{(l)}) = E_{\widehat{\boldsymbol{\theta}}^{(l)}}\left\{\log f_{\boldsymbol{Y}}(\boldsymbol{y};\boldsymbol{\theta})|\boldsymbol{z}\right\}. \tag{8.44}$$

M-step

$$\widehat{\boldsymbol{\theta}}^{(l+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta},\widehat{\boldsymbol{\theta}}^{(l)}).$$

$\widehat{\boldsymbol{\theta}}^{(l)}$ is the estimate of $\boldsymbol{\theta}$ after l iterations of the algorithm. Intuitively, the E-step yields an estimate of the a posteriori *complete data* statistics given the *incomplete data*. The crucial point in any implementation of the EM algorithm is how to define the *complete data* such that the implementation of the maximization required by the M-step is simpler than the maximization required by the original ML criterion, (8.42).

Now consider our noisy speech parameter estimation problem. The observed data vector (*incomplete data*) in the current analysis frame is,

$$\boldsymbol{z} = \left[\, z(0)\ z(2)\ \ldots\ z(N-1)\,\right]^T.$$

The corresponding vectors of speech and noise samples are,

$$\begin{aligned}\boldsymbol{s} &= \left[\, s(-p)\ s(-p+1)\ \ldots\ s(N-1)\,\right]^T,\\ \boldsymbol{v} &= \left[\, v(-q)\ v(-q+1)\ \ldots\ v(N-1)\,\right]^T.\end{aligned}$$

N is the frame length. p and q are the speech and noise AR orders. The *complete data* vector, $\boldsymbol{y}$, is defined to be a concatenation of the clean speech samples $\boldsymbol{s}$, and the noise samples $\boldsymbol{v}$, i.e.

$$\boldsymbol{y}^T = \begin{bmatrix} \boldsymbol{s}^T & \boldsymbol{v}^T \end{bmatrix}.$$

Invoking Bayes's rule,

$$\log f_{\boldsymbol{Y}}(\boldsymbol{y};\boldsymbol{\theta}) = \log f_{\boldsymbol{S}}(\boldsymbol{s};\boldsymbol{\theta}) + \log f_{\boldsymbol{v}}(\boldsymbol{v};\boldsymbol{\theta}),$$

where $\boldsymbol{\theta}$ is the vector of unknown parameters defined in (8.7).

Under the assumption that both the speech innovation sequence, $u_s(t)$, and the noise innovation sequence, $u_v(t)$ are Gaussian (hence, $s(t)$ and $v(t)$ are also assumed Gaussian), and recalling (8.2) and (8.4), one obtains,

$$\begin{aligned}\log f_{\boldsymbol{Y}}(\boldsymbol{y};\boldsymbol{\theta}) = C &+ \log f(\boldsymbol{s}_p(-1)) + \log f(\boldsymbol{v}_q(-1)) - \frac{N}{2}\log g_s - \frac{N}{2}\log g_v \\ &- \frac{1}{2g_s}\sum_{t=0}^{N-1}[s(t) + \boldsymbol{\alpha}^T\boldsymbol{s}_p(t-1)]^2 - \frac{1}{2g_v}\sum_{t=0}^{N-1}[v(t) + \boldsymbol{\beta}^T\boldsymbol{v}_q(t-1)]^2, \quad (8.45)\end{aligned}$$

where C is a constant, independent of the parameter vector $\boldsymbol{\theta}$. Under the assumption that $N >> p, q$, the contributions of $\log f(\boldsymbol{s}_p(-1))$ and $\log f(\boldsymbol{v}_q(-1))$ in (8.45) are negligible. Hence, taking the conditional expectation given the corrupted measurements $\boldsymbol{z}$ at $\widehat{\boldsymbol{\theta}}^{(l)}$ yields,

$$\begin{aligned}Q(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}}^{(l)}) &= E_{\widehat{\boldsymbol{\theta}}^{(l)}}\{\log f_{\boldsymbol{Y}}(\boldsymbol{y};\boldsymbol{\theta})|\boldsymbol{z}\} = C - \frac{N}{2}\log g_s - \frac{N}{2}\log g_v \\ &- \frac{1}{2g_s}\sum_{t=0}^{N-1} \widehat{s^2(t)} + 2\boldsymbol{\alpha}^T\ \widehat{\boldsymbol{s}_p(t-1)s(t)} + \boldsymbol{\alpha}^T\ \widehat{\boldsymbol{s}_p(t-1)\boldsymbol{s}_p^T(t-1)}\ \boldsymbol{\alpha} \\ &- \frac{1}{2g_v}\sum_{t=0}^{N-1} \widehat{v^2(t)} + 2\boldsymbol{\beta}^T\ \widehat{\boldsymbol{v}_q(t-1)v(t)} + \boldsymbol{\beta}^T\ \widehat{\boldsymbol{v}_q(t-1)\boldsymbol{v}_q^T(t-1)}\ \boldsymbol{\beta}, \quad (8.46)\end{aligned}$$

where the notation (8.9) has been used.

Equation (8.46) implies that the maximization of $Q(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}}^{(l)})$ with respect to $\boldsymbol{\theta}$ (**M-step**) is completely decoupled to two separate optimization problems, one with respect to the speech parameters, and the other with respect to the noise parameters. That is a very desirable property of the algorithm. Equations (8.16)–(8.19) are obtained by straightforward differentiation of (8.46).

Appendix B: Derivation of the Sequential Approximation

We provide a derivation of the sequential algorithm presented in Section 8.3.

The suggested recursive algorithm is based on the following gradient descent algorithm for solving the ML optimization problem, (8.42),

$$\hat{\boldsymbol{\theta}}^{(l+1)} = \hat{\boldsymbol{\theta}}^{(l)} + \frac{\rho}{N} \cdot \frac{\partial \log f_{\boldsymbol{Z}}(\boldsymbol{z};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}_{|\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}^{(l)}}, \tag{8.47}$$

where $\hat{\boldsymbol{\theta}}^{(l)}$ is the estimate of $\boldsymbol{\theta}$ after l iteration cycles. The constant ρ is the update step sizes. For sufficiently small step sizes, this algorithm converges to a local maxima of the likelihood function. To compute the partial derivatives in (8.47), we suggest using Fisher's identity [55]. Let the vector $\boldsymbol{y}$ (*complete data*) be some vector, that is related to the measurements vector $\boldsymbol{z}$ by the transformation (8.43). Then Fisher's identity asserts the following:

$$\frac{\partial}{\partial \boldsymbol{\theta}} \log f_{\boldsymbol{Z}}(\boldsymbol{z};\boldsymbol{\theta})_{|\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}^{(l)}} = \frac{\partial}{\partial \boldsymbol{\theta}} E_{\widehat{\boldsymbol{\theta}}^{(l)}}\{\log f_{\boldsymbol{Y}}(\boldsymbol{y};\boldsymbol{\theta})|\boldsymbol{z}\} = \frac{\partial}{\partial \boldsymbol{\theta}} Q(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}}^{(l)})_{|\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}^{(l)}}, \tag{8.48}$$

where we have used the definition (8.44). In order to make this identity useful, $\boldsymbol{y}$ should be chosen such that the differentiation of $Q(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}}^{(l)})$ is simpler to implement than the direct differentiation of $\log f_{\boldsymbol{Z}}(\boldsymbol{z};\boldsymbol{\theta})$.

Differentiating (8.45) with respect to $\boldsymbol{\alpha}$ and invoking (8.47), (8.48) yields,

$$\hat{\boldsymbol{\alpha}}^{(l+1)} = \hat{\boldsymbol{\alpha}}^{(l)} - \frac{\rho_s}{N g_s} \sum_{\tau=0}^{N-1} \left[\widehat{\boldsymbol{s}_p(\tau-1)\boldsymbol{s}_p^T(\tau-1)}\, \hat{\boldsymbol{\alpha}}^{(l)} + \widehat{\boldsymbol{s}_p(\tau-1)s(\tau)} \right].$$

To obtain our sequential algorithm, the iteration index is replaced by the time index. We also incorporate a forgetting factor for calculating the covariance terms. For convenience we define,

$$\begin{aligned} Q^s(t) &\triangleq \begin{bmatrix} Q^s_{11}(t) & Q^s_{12}(t) \\ Q^s_{21}(t) & Q^s_{22}(t) \end{bmatrix} \\ &\triangleq \sum_{\tau=0}^{t} \lambda_s^{t-\tau}\, \widehat{\boldsymbol{s}_{p+1}(\tau)\boldsymbol{s}_{p+1}^T(\tau)} = \lambda_s Q^s(t-1) + \widehat{\boldsymbol{s}_{p+1}(t)\boldsymbol{s}_{p+1}^T(t)}\,. \end{aligned}$$

$Q^s_{11}(t)$ is a $p \times p$ matrix, $Q^s_{12}(t) = (Q^s_{21}(t))^T$ is a $p \times 1$ matrix, and $Q^s_{22}(t)$ is a scalar value. λ_s and λ_v are forgetting factors for the speech and noise, respectively that satisfy $0 \le \lambda_s, \lambda_v \le 1$ and control the update rate. Then our sequential update of $\hat{\boldsymbol{\alpha}}(t)$ (8.21) is

$$\begin{aligned} \hat{\boldsymbol{\alpha}}(t+1) &= \hat{\boldsymbol{\alpha}}(t) - \frac{\rho_s}{g_s} \sum_{\tau=0}^{t} \lambda_s^{t-\tau} \left[\widehat{\boldsymbol{s}_p(\tau-1)\boldsymbol{s}_p^T(\tau-1)}\, \hat{\boldsymbol{\alpha}}(t) + \widehat{\boldsymbol{s}_p(\tau-1)s(\tau)} \right] \\ &= \hat{\boldsymbol{\alpha}}(t) - \frac{\rho_s}{g_s} \left[Q^s_{12}(t) + Q^s_{11}(t)\hat{\boldsymbol{\alpha}}(t) \right]. \end{aligned}$$

g_s may be obtained similarly. Alternatively, we may use the sequential variant of (8.17) (i.e. (8.22)). Similar update equations apply to the noise parameters (8.23)–(8.24).

Appendix C: The Unscented Transform (UT)

Let $\boldsymbol{x}$ be an L-dimensional random vector with mean $\bar{\boldsymbol{x}}$ and covariance matrix P_{xx}. Let, $\boldsymbol{y} = f(\boldsymbol{x})$ be a nonlinear transformation from the random vector $\boldsymbol{x}$ to another random vector $\boldsymbol{y}$. The first- and second-order statistics of the vector $\boldsymbol{y}$ should be calculated. We briefly summarize the method. The mean and covariance of $\boldsymbol{x}$ can be presented by the $2L+1$ σ-points

$$\begin{aligned} \mathcal{X}_0 &= \bar{\boldsymbol{x}}, \\ \mathcal{X}_l &= \bar{\boldsymbol{x}} + \left(\sqrt{(L+\lambda)P_{xx}}\right)_l, \; l = 1, \ldots, L, \\ \mathcal{X}_{l+L} &= \bar{\boldsymbol{x}} - \left(\sqrt{(L+\lambda)P_{xx}}\right)_l, \; l = 1, \ldots, L, \end{aligned}$$

where $\left(\sqrt{(L+\lambda)P_{xx}}\right)_l$ is the l-th row or column of the corresponding matrix square root, and $\lambda = \alpha^2(L+\kappa) - L$. α determines the spread of the sigma points. $\alpha = 1$ was used throughout our simulations . κ is a secondary scaling parameter. The choice $\kappa = 3 - L$ maintains the kurtosis of a Gaussian vector. Throughout our simulations κ is set to 0. β is used to incorporate prior knowledge of the distribution ($\beta = 2$ for Gaussian distributions). A proper choice of these parameters and its influence on the obtainable performance is still an open topic.

Define the weights

$$\begin{aligned} W_0^{(m)} &= \lambda/(L+\lambda), \\ W_0^{(c)} &= \lambda/(L+\lambda) + (1 - \alpha^2 + \beta), \\ W_l^{(m)} &= W_l^{(c)} = 1/2(L+\lambda), \; l = 1, 2, \ldots, 2L. \end{aligned}$$

Then the mean and covariance of the vector $\boldsymbol{y}$ can be calculated using the following procedure,

1. Construct $\boldsymbol{x}$ σ-points: $\mathcal{X}_l, \; l = 0, \ldots, 2L$.
2. Transform each point to the respective $\boldsymbol{y}$ σ-points: $\mathcal{Y}_l = f\left(\mathcal{X}_l\right), \; l = 0, \ldots, 2L$.
3. Use weighted averaging, $\bar{\boldsymbol{y}} \approx \sum_{l=0}^{2L} W_l^{(m)} \mathcal{Y}_l$ to estimate $\boldsymbol{y}$ mean.
4. Use weighted outer product, $P_{yy} \approx \sum_{l=0}^{2L} W_l^{(c)} \left(\mathcal{Y}_l - \bar{\boldsymbol{y}}\right)\left(\mathcal{Y}_l - \bar{\boldsymbol{y}}\right)^T$ to estimate $\boldsymbol{y}$ covariance.

The benefits of using the UT are presented in [30], [31].

References

1. N. Wiener, *The Extrapolation, Interpolation and Smoothing of Stationary Time Series.* John Wiley & Sons, Inc., New York, N.Y., USA, 1949.
2. R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. of the ASME-Journal of Basic Engineering*, 82 (Series D), pp. 35–45, 1960.
3. A. P. Dempster, N. M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc.*, Ser. 3g, pp. 1–38, 1977.
4. J. S. Lim and A. V. Oppenheim, "All-pole modeling of degraded speech," *IEEE Trans. on Acoustic, Speech and Signal Processing*, vol. 26, pp. 197–210, June 1978.
5. J. H. L. Hansen and M. A. Clements, "Constrained iterative speech enhancement with application to automatic speech recognition," in *Proc. IEEE ICASSP*, 1988, pp. 561–564.
6. J. H. L. Hansen and M. A. Clements, "Constrained iterative speech enhancement with application to speech recognition," *IEEE Trans. on Signal Processing*, vol. 39, pp. 795–805, Apr. 1991.
7. B. L. Pellom and J. H. L. Hansen, "An improved constrained iterative speech enhancement for colored noise environments," *IEEE Trans. on Speech and Audio Processing*, vol. 6, pp. 573–579, Nov. 1998.
8. E. Masgrau, J. Salavedra, A. Moreno, and A. Ardanuy, "Speech enhancement by adaptive Wiener filtering based on cumulant AR modeling," in M. Grenie and J. C. Junqua, editors, *Speech Processing in Adverse Conditions*, pp. 143–146. 1992.
9. K. K. Paliwal and A. Basu, "A Speech enhancement method based on Kalman filtering," in *Proc. IEEE ICASSP*, 1987, pp. 177–180.
10. B. Koo, J. D. Gibson, and S. D. Gray, "Filtering of colored noise for speech enhancement and coding," in *Proc. IEEE ICASSP*, 1989, pp. 349–352.
11. J. D. Gibson, B. Koo, and S. D. Gray, "Filtering of colored noise for speech enhancement and coding," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 39, pp. 1732–1742, Aug. 1991.
12. M. Feder, A. V. Oppenheim, and E. Weinstein, "Methods for noise cancellation based on the EM algorithm," in *Proc. IEEE ICASSP*, 1987, pp. 201–204.
13. B. Widrow, J. R. Glover Jr., J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeider, E. Dong Jr., and R. C. Goodlin, "Adaptive noise cancelling: principals and applications," *Proceeding of the IEEE*, vol. 63, 1692–1716, Dec. 1975.
14. M. Feder, A. V. Oppenheim, and E. Weinstein, "Maximum likelihood noise cancellation using the EM algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 204–216, Feb. 1989.
15. E. Weinstein, A. V. Oppenheim, and M. Feder, "Signal enhancement using single and multi-sensor measurements," Technical Report no. 560, M.I.T , Cambridge, MA, Nov. 1990.
16. E. Weinstein, A. V. Oppenheim, M. Feder, and J. R. Buck, "Iterative and sequential algorithms for multisensor signal enhancement," *IEEE Trans. Signal Processing*, vol. 42, pp. 846–859, Apr. 1994.

17. M. Feder, E. Weinstein, and A. V. Oppenheim, “A new class of sequential and adaptive algorithms with application to noise cancellation,” in *Proc. IEEE ICASSP*, 1988, pp. 557–560.
18. A. V. Oppenheim, E. Weinstein, K. C. Zangi, M. Feder, and D. Gauger, “Single-sensor active noise cancellation,” *IEEE Trans. Speech and Audio Processing*, vol. 2, pp. 285–290, Apr. 1994.
19. B.-G. Lee, K. Y. Lee, and S. Ann, “An EM-based approach for parameter enhancement with an application to speech signals,” *Signal Processing*, vol. 46, pp. 1–14, 1995.
20. K. Y. Lee, B.-G. Lee, and S. Ann, “Adaptive filtering for speech enhancement in colored noise,” *IEEE Signal Processing Letters*, vol. 4, pp. 277–279, Oct. 1997.
21. Z. Goh, K.-C. Tan, and B. T. G. Tan, “Kalman-filtering speech enhancement method based on a voiced-unvoiced speech model,” *IEEE Trans. on Speech and Audio Processing*, vol. 7, pp. 510–524, Sept. 1999.
22. M. Gabrea, E. Grivel, and M. Najim, “A single microphone Kalman filter-based noise canceller,” *IEEE Signal Processing Letters*, vol. 6, pp. 55–57, Mar. 1999.
23. S. Gannot, D. Burshtein, and E. Weinstein, “Iterative and sequential Kalman filter-based speech enhancement algorithms,” *IEEE Trans. on Speech and Audio Processing*, vol. 6, pp. 373–385, July 1998.
24. M. Fujimoto and Y. Ariki, “Noisy speech recognition using noise reduction method based on Kalman filter,” in *Proc. IEEE ICASSP*, 2000, pp. 1727–1730.
25. S. F. Boll, “Suppression of acoustic noise in speech using spectral subtraction,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 27, pp. 113–120, Apr. 1979.
26. K. Y. Lee, B.-G. Lee, I. Song, and S. Ann, “Robust estimation of AR parameters and its application for speech enhancement,” in *Proc. IEEE ICASSP*, 1992, pp. 309–312.
27. N. Ma, M. Bouchard, and R. A. Goubran, “Perceptual Kalman filtering for speech enhancement in colored noise,” in *Proc. IEEE ICASSP*, vol. 1, 2004, pp. 717–720.
28. X. Shen and L. Deng, “A dynamic system approach to speech enhancement using the H_∞ filtering algorithm,” *IEEE Trans. on Speech and Audio Processing*, vol. 27, pp. 391–399, July 1999.
29. E. A. Wan and A. T. Nelson, “Removal of noise from speech using the dual EKF algorithm,” in *Proc. IEEE ICASSP*, 1998.
30. S. J. Julier and J. K. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 92, pp. 401–422, Mar. 2004.
31. E. A. Wan and R. van der Merwe, “The unscented Kalman filter for nonlinear estimation,” in *Proc. IEEE Symposium on Adaptive Systems for Signal Processing, Communication and Control (AS-SPCC)*, 2000.
32. S. Gannot and M. Moonen, “On the application of the unscented Kalman filter to speech processing,” in *Proc. IWAENC*, 2003, pp. 27–30.
33. W. Fong and S. Godsill, “Monte Carlo smoothing with application to audio signal enhancement,” in *Proc. IEEE SSP Workshop*, 2001, pp. 18–210.
34. Y. Ephraim, D. Malah, and B. H. Juang, “On the application of hidden Markov models for enhancing noisy speech,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1846–1856, 1989.

35. Y. Ephraim, “A Bayesian estimation approach for speech enhancement using hidden Markov models,” *IEEE Trans. Signal Processing*, vol. 40, pp. 725–735, 1992.
36. Y. Ephraim, “Speech enhancement using state dependent dynamical system model,” in *Proc. IEEE ICASSP*, 1992, pp. 289–292.
37. K. Y. Lee and K. Shirai, “Efficient recursive estimation for speech enhancement in colored noise,” *IEEE Signal Processing Letters*, vol. 3, pp. 196–199, 1996.
38. K. Y. Lee and S. Jung, “Time-domain approach using multiple Kalman filters and EM algorithm to speech enhancement with nonstationary noise,” *IEEE Trans. Speech and Audio Proc.*, vol. 8, pp. 373–385, May 2000.
39. J. B. Kim, K. Y. Lee, and C. W. Lee, “On the applications of the interacting multiple model algorithm for enhancing noisy speech,” *IEEE Trans. Speech and Audio Processing*, vol. 8, pp. 349–352, May 2000.
40. K. Y. Lee, S. McLaughlin, and K. Shirai, “Speech enhancement based on extended Kalman filter and neural predictive hidden Markov model,” in *Proc. IEEE Int. Workshop Neural Networks for Signal Processing*, 1996, pp. 302–310.
41. Y. Ephraim and D. Malah, “Speech enhancement using a minimum mean square error log-spectral amplitude estimator,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, pp. 443–445, Apr. 1985.
42. I. Cohen and B. Berdugo, “Speech enhancement for non-stationary noise environments *Signal Processing*, vol. 81, pp. 2403–2418, Oct. 2001.
43. D. Burshtein and S. Gannot, “Speech enhancement using a mixture-maximum model,” *IEEE Trans. Speech and Audio Processing*, vol. 10, pp. 341–351, Sept. 2002.
44. D. Burshtein, “Joint modeling and maximum-likelihood estimation of pitch and linear prediction coefficient parameters,” *J. Acoustic Society of America*, vol. 3, pp. 1531–1537, Mar. 1992.
45. R. H. Shumway and D. S. Stoffer, “An approach to time series smoothing and forecasting using the EM algorithm,” *J. Time Series Anal.*, vol. 3, no. 7, pp. 253–264, 1982.
46. C. L. Nikias and A. P. Petropulu, *Higher-Order Spectra Analysis.* Pearson Education POD, 1st edition, 1993.
47. K. K. Paliwal and M. M. Sondhi, “Recognition of noisy speech using cumulant based linear prediction analysis,” in *Proc. IEEE ICASSP*, 1991, pp. 429–432.
48. J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, “Acoustic-phonetic continuous speech corpus (timit),” CD-ROM, Oct. 1991.
49. R. G. Leonard and G. Doddington, “A database for speaker independent digit recognition (tidigits),” CD-ROM, Oct. 1984.
50. A. Varga and H. J. M. Steeneken, “Assessment for automatic speech recognition: II. NOISEX-92: a database and an axperiment to study the effect of additive noise on speech recognition systems,” *Speech Communication*, vol. 12, pp. 247–251, July 1993.
51. ANSI, “Specifications for octave-band and fractional-octave-band analog and digital filters,” S1.1-1986 (ASA 65-1986), 1993.
52. S. R. Quackenbush, T. P. Barnwell, and M. A. Clements, *Objective Measures of Speech Quality.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1988.
53. S. Gannot, “Audio sample files,” `http://www.biu.ac.il/~gannot`, Oct. 2004.

54. R. van der Merwe, “Recursive Bayesian estimation library (ReBEL),” `http://cslu.ece.ogi.edu/mlsp/rebel/`, 2002.
55. R. A. Fisher, “Theory of statistical estimation,” *Proc. of the Cambridge Philosophical Society*, vol. 22, pp. 700–725, 1925.

9 Speech Distortion Weighted Multichannel Wiener Filtering Techniques for Noise Reduction

Simon Doclo[1], Ann Spriet[1,2], Jan Wouters[2], and Marc Moonen[1]

[1] Katholieke Universiteit Leuven, Dept. of Electrical Engineering (ESAT-SCD)
Leuven 3001, Belgium
E-mail: {doclo, spriet, moonen}@esat.kuleuven.ac.be

[2] Katholieke Universiteit Leuven, Laboratory for Exp. ORL
Leuven 3000, Belgium
E-mail: jan.wouters@uz.kuleuven.ac.be

Abstract. In many speech communication applications a microphone array is available nowadays, such that multi-microphone speech enhancement techniques can be used instead of single-microphone speech enhancement techniques. A well-known multi-microphone speech enhancement technique is the generalized sidelobe canceller (GSC), which is however quite sensitive to signal model errors, such as microphone mismatch. This chapter discusses a more robust technique called the spatially pre-processed speech distortion weighted multichannel Wiener filter (SP-SDW-MWF), which takes speech distortion due to signal model errors explicitly into account in its design criterion, and which encompasses the standard GSC as a special case. In addition, a novel frequency-domain criterion for the SDW-MWF is presented, from which several – existing and novel – adaptive frequency-domain algorithms can be derived for implementing the SDW-MWF. The noise reduction performance and the robustness of these adaptive algorithms is investigated for a hearing aid application. Using experimental results with a small-sized microphone array, it is shown that the SP-SDW-MWF is more robust against signal model errors than the GSC, both in stationary and in changing noise scenarios.

9.1 Introduction

In many speech communication applications, such as hands-free mobile telephony, hearing aids and voice-controlled systems, the recorded speech signals are often corrupted by a considerable amount of acoustic background noise. Generally speaking, background noise is broadband and non-stationary, and the signal-to-noise ratio (SNR) of the microphone signals may be quite low. Background noise causes a signal degradation that can lead to total unintelligibility of the speech signal and that substantially decreases the performance of speech coding and speech recognition systems. Therefore, efficient speech enhancement techniques are required.

Since the desired speech signal and the undesired noise signal usually occupy overlapping frequency bands, single-microphone speech enhancement techniques, such as spectral subtraction, Kalman filtering, and signal subspace-based techniques, may encounter problems to reduce the background noise without introducing noticeable artifacts (e.g. musical noise) or speech distortion. When the speech and the noise sources are physically located at different positions, it is possible to exploit this spatial diversity by using a microphone array, such that both the spectral and the spatial characteristics can be used in the speech enhancement algorithm.

Well-known multi-microphone speech enhancement techniques are fixed and adaptive beamforming [1]. In a minimum variance distortionless response (MVDR) beamformer [2], the energy of the output signal is minimized under the constraint that signals arriving from the look direction, i.e. the direction of the speech source, are processed without distortion. A widely studied adaptive implementation of this beamformer is the generalized sidelobe canceller (GSC) [3,4], which consists of a fixed spatial pre-processor, i.e. a fixed beamformer, creating a so-called speech reference, and a blocking matrix, creating so-called noise references; and a multichannel adaptive noise canceller, eliminating the noise components in the speech reference which are correlated with the noise references. Multi-microphone noise reduction techniques based on the GSC have already been successfully implemented for various speech applications [5–10].

Due to room reverberation, microphone mismatch, look direction error and spatially distributed sources, undesired speech components however leak into the noise references of the standard GSC, giving rise to speech distortion and cancellation. Several techniques have already been proposed for limiting the speech distortion that results from this speech leakage, by

- reducing the speech leakage components in the noise references, e.g. by using a robust fixed spatial filter designed blocking matrix [5,11–13], by using an adaptive blocking matrix [14–17], or by constructing the blocking matrix based on estimating the acoustic transfer functions from the speech source to the microphone array [18,19];
- limiting the distorting effect of the present speech leakage by
  - updating the adaptive filter only during periods (and for frequencies) where the noise component is dominant, i.e. low SNR [5–7,10,14–17,20]; and
  - constraining the update formula of the adaptive filter, e.g. by imposing a quadratic inequality constraint (QIC) [21–24], by using the leaky least-mean-square (LMS) algorithm [11,12], by using coefficient constraints [15], or by taking speech distortion due to speech leakage directly into account using *speech distortion weighted multichannel Wiener filtering* (SDW-MWF) [25–27].

In [25], a generalized noise reduction scheme, called the spatially preprocessed speech distortion weighted multichannel Wiener filter (SP-SDW-MWF), has been presented, which consists of a fixed spatial pre-processor and an adaptive SDW-MWF stage. By taking speech distortion explicitly into account in the design criterion of the adaptive stage, the SP-SDW-MWF adds robustness to the GSC. In [25], it has been shown that, compared to the widely studied QIC-GSC, the SP-SDW-MWF achieves a better noise reduction performance for a given maximum speech distortion level.

In [28–31], recursive matrix-based implementations for the SDW-MWF have been proposed based on the generalized singular value decomposition or the QR decomposition, which are computationally quite expensive. In [26,32] cheaper (time-domain and frequency-domain) stochastic gradient algorithms have been presented. These algorithms however require large circular data buffers, resulting in a large memory usage. In [27,33] adaptive frequency-domain algorithms for the SDW-MWF have been presented using (block-)diagonal correlation matrices, reducing the memory usage and the computational complexity.

In this chapter, we present a novel *frequency-domain criterion* for the SDW-MWF, trading off noise reduction and speech distortion. This frequency-domain criterion for multichannel speech enhancement is in fact an extension of the criterion used in [34–36] for multichannel acoustic echo cancellation. Using the proposed criterion, several existing [27,33] and novel adaptive frequency-domain algorithms for the SDW-MWF can be derived. The main difference between these algorithms consists in the calculation of the step size matrix in the update formula for the adaptive filter and in the calculation of the regularization term (cf. Sects. 9.3 and 9.4).

The chapter is organized as follows. In Sect. 9.2, the GSC and the spatially pre-processed SDW-MWF are briefly reviewed. In Sect. 9.3, the novel frequency-domain criterion for the SDW-MWF is presented. An adaptive algorithm is derived for optimizing this criterion and it is shown how this adaptive algorithm can be implemented in practice. In Sect. 9.4, several approximations are proposed for reducing the computational complexity, where some of these approximations lead to already existing frequency-domain algorithms for the SDW-MWF [27,33]. In Sect. 9.5, the noise reduction performance, the robustness against signal model errors, and the tracking behaviour of the proposed algorithms are illustrated using experimental results for a small-sized microphone array in a hearing aid.

9.2 GSC and Spatially Pre-Processed SDW-MWF

9.2.1 Notation and General Structure

Consider a microphone array with M microphones, where each microphone signal $u_i[k]$, $i = 1 \dots M$, at time k, consists of a filtered version of the clean speech signal $s[k]$ and additive noise, i.e.

$$u_i[k] = h_i[k] \otimes s[k] + u_i^v[k],\ i = 1 \dots M, \tag{9.1}$$

with $h_i[k]$ the acoustic impulse response between the speech source and the ith microphone and $\otimes$ denoting convolution. The additive noise $u_i^v[k]$ can be colored and is assumed to be uncorrelated with the clean speech signal.

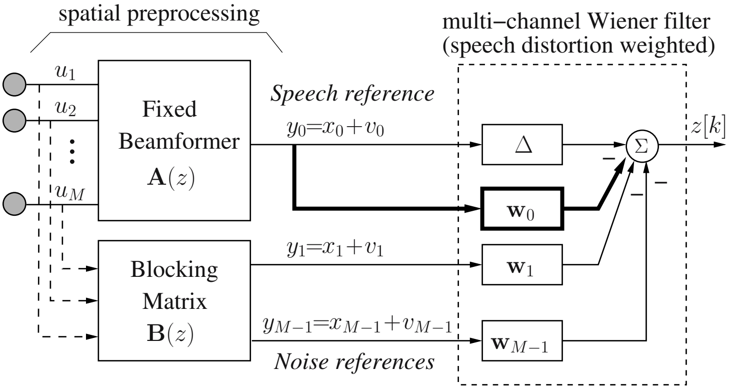


Fig. 9.1. Structure of the spatially pre-processed speech distortion weighted multichannel Wiener filter (SP-SDW-MWF).

The spatially pre-processed speech distortion weighted multichannel Wiener Filter (SP-SDW-MWF) [25] is depicted in Fig. 9.1. It consists of a fixed spatial pre-processor, i.e. a fixed beamformer and a blocking matrix, and an adaptive stage. Note that the structure of the SP-SDW-MWF strongly resembles the standard GSC, where the difference lies in the fact that an adaptive SDW-MWF is used in the adaptive stage and that it is possible to include an extra filter $\mathbf{w}_0$ on the speech reference.

The *fixed beamformer* $\mathbf{A}(z)$ creates a so-called speech reference

$$y_0[k] = x_0[k] + v_0[k], \tag{9.2}$$

with $x_0[k]$ and $v_0[k]$ respectively the speech and the noise component of the speech reference, by steering a beam towards the direction of the speaker. The fixed beamformer should be designed such that the distortion of the speech component $x_0[k]$, due to possible errors in the assumed signal model (e.g. look direction error, microphone mismatch) is small. A delay-and-sum beamformer, which time-aligns the microphone signals, offers sufficient robustness against signal model errors since it minimizes the noise sensitivity. However, in order to achieve a better spatial selectivity while preserving robustness, the fixed beamformer can be optimized, e.g. using statistical knowledge about the signal model errors that occur in practice [13].

The *blocking matrix* $\mathbf{B}(z)$ creates $M-1$ so-called noise references

$$y_n[k] = x_n[k] + v_n[k],\ n = 1 \dots M-1, \tag{9.3}$$

by steering zeroes towards the direction of the speaker, such that it is anticipated that the noise components $v_n[k]$ are dominant compared to the speech components $x_n[k]$. A simple technique to create the noise references consists of pair-wisely subtracting the time-aligned microphone signals. Under ideal conditions (i.e. no reverberation, point speech source, no look direction error, no microphone mismatch), the noise references only contain noise components $v_n[k]$. Since these conditions are practically never fulfilled, undesired speech components $x_n[k]$, i.e. so-called *speech leakage*, are present in the noise references. Several techniques have already been proposed for reducing the speech leakage components in the noise references [5,11–19].

During *speech-periods* the speech and the noise references consist of speech+noise, i.e. $y_n[k] = x_n[k] + v_n[k]$, whereas during *noise-only-periods* only the noise components $v_n[k]$ are observed. We assume that the (spatial and/or temporal) second-order statistics of the noise are sufficiently stationary such that they can be estimated during noise-only-periods and used during subsequent speech-periods. This requires the use of a voice activity detection (VAD) mechanism [37–39] or an on-line procedure for estimating the SNR [40].

The goal of the *adaptive stage* is to make an estimate of the noise component in the speech reference and to subtract this noise estimate from the speech reference in order to obtain an enhanced output signal $z[k]$. Let N be the number of input channels to the multichannel filter ($N = M$ if the filter $\mathbf{w}_0$ on the speech reference is present, $N = M - 1$ otherwise). Let the FIR filters $\mathbf{w}_n[k]$, $n = M - N \dots M - 1$, have length L, and consider the L-dimensional data vectors $\mathbf{y}_n[k]$, the NL-dimensional stacked data vector $\mathbf{y}[k]$, and the NL-dimensional stacked filter $\mathbf{w}[k]$, defined as

$$\mathbf{y}_n[k] = \left[y_n[k] \; y_n[k-1] \dots y_n[k-L+1] \right]^T, \quad n = M-N \dots M-1, \tag{9.4}$$

$$\mathbf{y}[k] = \left[\mathbf{y}_{M-N}^T[k] \; \mathbf{y}_{M-N+1}^T[k] \dots \mathbf{y}_{M-1}^T[k] \right]^T, \tag{9.5}$$

$$\mathbf{w}[k] = \left[\mathbf{w}_{M-N}^T[k] \; \mathbf{w}_{M-N+1}^T[k] \dots \mathbf{w}_{M-1}^T[k] \right]^T, \tag{9.6}$$

with ^{T} denoting transpose of a vector or a matrix. The stacked data vector can be decomposed into a speech and a noise component, i.e. $\mathbf{y}[k] = \mathbf{x}[k] + \mathbf{v}[k]$, where $\mathbf{x}[k]$ and $\mathbf{v}[k]$ are defined similarly as in (9.5). The goal of the filter $\mathbf{w}[k]$ is to make an estimate of the delayed noise component $v_0[k - \Delta]$ in the speech reference[1]. This noise estimate is then subtracted from the speech reference in order to obtain the enhanced output signal $z[k]$, i.e.

$$z[k] = y_0[k - \Delta] - \mathbf{w}^T[k]\mathbf{y}[k] \tag{9.7}$$

$$= x_0[k - \Delta] + \underbrace{(v_0[k - \Delta] - \mathbf{w}^T[k]\mathbf{v}[k])}_{e_v[k]} - \underbrace{\mathbf{w}^T[k]\mathbf{x}[k]}_{e_x[k]}. \tag{9.8}$$

Hence, the output signal $z[k]$ consists of 3 terms: the delayed speech component $x_0[k - \Delta]$ in the speech reference, residual noise $e_v[k]$, and (linear) speech distortion $e_x[k]$. The goal of any speech enhancement algorithm is to

[1] The delay Δ is applied to the speech reference in order to allow for non-causal filter taps. This delay is usually set equal to $\lceil L/2 \rceil$, where $\lceil x \rceil$ denotes the smallest integer larger than or equal to x.

reduce the residual noise as much as possible, while simultaneously limiting the speech distortion. The speech distortion can e.g. be limited by reducing the speech leakage components $\mathbf{x}[k]$ in the noise references and/or by constraining the filter $\mathbf{w}[k]$.

9.2.2 Generalized Sidelobe Canceller

The standard generalized sidelobe canceller (GSC) minimizes the residual noise energy without taking into account speech distortion, i.e.

$$J_{GSC}(\mathbf{w}[k]) = \varepsilon_v^2[k] = \mathcal{E}\Big\{\big|v_0[k-\Delta] - \mathbf{w}^T[k]\mathbf{v}[k]\big|^2\Big\}, \tag{9.9}$$

with $\mathcal{E}$ denoting the expected value operator. The filter $\mathbf{w}[k]$ minimizing this cost function is equal to

$$\mathbf{w}[k] = \mathcal{E}\{\mathbf{v}[k]\mathbf{v}^T[k]\}^{-1}\,\mathcal{E}\{\mathbf{v}[k]v_0[k-\Delta]\}, \tag{9.10}$$

where the noise correlation matrix $\mathcal{E}\{\mathbf{v}[k]\mathbf{v}^T[k]\}$ and the noise cross-correlation vector $\mathcal{E}\{\mathbf{v}[k]v_0[k-\Delta]\}$ are estimated during noise-only-periods. Hence, in an adaptive implementation, the filter $\mathbf{w}[k]$ is also allowed to be updated only during noise-only-periods [5–7,10,14–17,20], since adaptation during speech-periods would lead to an incorrect solution and signal cancellation. Note however that signal distortion due to speech leakage still occurs even when the adaptive filter is updated only during noise-only-periods, since the speech distortion term $e_x[k]$ is still present in the output signal $z[k]$.

A commonly used approach to increase the robustness against signal model errors is to apply a quadratic inequality constraint (QIC) [21–24], i.e.

$$\mathbf{w}^T[k]\mathbf{w}[k] \leq \beta^2. \tag{9.11}$$

The QIC avoids excessive growth of the filter coefficients $\mathbf{w}[k]$, and hence limits speech distortion $\mathbf{w}^T[k]\mathbf{x}[k]$ due to speech leakage. The QIC can be implemented using the scaled projection algorithm [22] or by using variable loading [24]. Similar approaches for constraining the filter coefficients consist in using the leaky LMS algorithm [11,12] or using coefficient constraints [15].

In the GSC the number of input channels to the adaptive filter is typically equal to $N = M - 1$. It is not possible to include the filter $\mathbf{w}_0$ on the speech reference, since in this case the filter $\mathbf{w}[k]$ in (9.10) would be equal to

$$\mathbf{w}_0[k] = \mathbf{u}_{\Delta+1}, \quad \mathbf{w}_n[k] = \mathbf{0},\, n = 1 \ldots M-1\,, \tag{9.12}$$

with $\mathbf{u}_l$ the lth canonical L-dimensional vector, whose lth element is equal to 1 and all other elements are equal to 0, such that the output signal $z[k] = 0$.

9.2.3 Speech Distortion Weighted Multichannel Wiener Filter

The speech distortion weighted multichannel Wiener filter (SDW-MWF) takes speech distortion due to speech leakage explicitly into account in the design criterion of the filter $\mathbf{w}[k]$ and minimizes the weighted sum of the residual noise energy $\varepsilon_v^2[k]$ and the speech distortion energy $\varepsilon_x^2[k]$, i.e.

$$J(\mathbf{w}[k]) = \varepsilon_v^2[k] + \frac{1}{\mu}\varepsilon_x^2[k] \tag{9.13}$$

$$= \mathcal{E}\Big\{\big|v_0[k-\Delta] - \mathbf{w}^T[k]\mathbf{v}[k]\big|^2\Big\} + \frac{1}{\mu}\mathcal{E}\Big\{\big|\mathbf{w}^T[k]\mathbf{x}[k]\big|^2\Big\}, \tag{9.14}$$

where the parameter $\mu \in [0, \infty]$ provides a trade-off between noise reduction and speech distortion [25,29,41]. If $\mu = 1$, the minimum mean square error (MMSE) criterion is obtained. If $\mu < 1$, speech distortion is reduced at the expense of increased residual noise energy. On the other hand, if $\mu > 1$, residual noise is reduced at the expense of increased speech distortion.

The filter $\mathbf{w}[k]$ minimizing the cost function in (9.14) is equal to

$$\mathbf{w}[k] = \left[\mathcal{E}\big\{\mathbf{v}[k]\mathbf{v}^T[k]\big\} + \frac{1}{\mu}\mathcal{E}\big\{\mathbf{x}[k]\mathbf{x}^T[k]\big\}\right]^{-1} \mathcal{E}\big\{\mathbf{v}[k]v_0[k-\Delta]\big\}, \tag{9.15}$$

where, using the independence assumption between speech and noise, the correlation matrix $\mathcal{E}\{\mathbf{x}[k]\mathbf{x}^T[k]\}$ can be computed as

$$\mathcal{E}\big\{\mathbf{x}[k]\mathbf{x}^T[k]\big\} = \mathcal{E}\big\{\mathbf{y}[k]\mathbf{y}^T[k]\big\} - \mathcal{E}\big\{\mathbf{v}[k]\mathbf{v}^T[k]\big\}. \tag{9.16}$$

The speech correlation matrix $\mathcal{E}\{\mathbf{y}[k]\mathbf{y}^T[k]\}$ is estimated during speech-periods and the noise correlation matrix $\mathcal{E}\{\mathbf{v}[k]\mathbf{v}^T[k]\}$ is estimated during noise-only-periods. We assume that the second-order statistics of the noise are sufficiently stationary such that they can be estimated during noise-only-periods and used during subsequent speech-periods.

Since the SDW-MWF takes speech distortion explicitly into account in its optimization criterion, it is possible to include the filter $\mathbf{w}_0$ on the speech reference. Depending on the setting of the parameter μ and the presence/absence of the filter $\mathbf{w}_0$, different algorithms are obtained:

- Without filter $\mathbf{w}_0$ ($N = M - 1$), we obtain the *speech distortion regularized GSC* (SDR-GSC), where the standard optimization criterion of the GSC in (9.9) is supplemented with a regularization term $1/\mu\,\varepsilon_x^2$. For $\mu = \infty$, speech distortion is completely ignored, which corresponds to the standard GSC. For $\mu = 0$, all emphasis is put on speech distortion, such that $\mathbf{w}[k] = \mathbf{0}$ and the output signal $z[k]$ is equal to the delayed speech reference $y_0[k-\Delta]$. Compared to the QIC-GSC, the SDR-GSC is less conservative, since the regularization term in the SDR-GSC is proportional to the actual amount of speech leakage present in the noise references. On the other hand, the constraint value β^2 in the QIC-GSC, cf. (9.11), needs to be chosen based on the largest signal model errors that may occur, such that the noise reduction performance is compromised even when no or a small amount of speech leakage is present. In the absence of speech leakage, the regularization term in the SDR-GSC equals 0, such that the GSC solution is obtained and hence the noise reduction performance is not compromised. In [25], it has been shown that in comparison with the QIC-GSC, the SDR-GSC obtains a better noise reduction for small model errors, while guaranteeing robustness against large model errors.

- With filter $\mathbf{w}_0$ $(N = M)$, we obtain the *spatially pre-processed speech distortion weighted multichannel Wiener filter* (SP-SDW-MWF). For $\mu = 1$, the output signal $z[k]$ is the MMSE estimate of the delayed speech component $x_0[k - \Delta]$ in the speech reference. In [25], it has been shown that, for infinite filter lengths, the performance of the SP-SDW-MWF is not affected by microphone mismatch as long as the speech component in the speech reference remains unaltered by the microphone mismatch. Hence, the extra filter on the speech reference further improves the performance.

In [28–31], recursive matrix-based implementations for the SDW-MWF have been proposed based on the generalized singular value decomposition or the QR decomposition, which are computationally quite expensive. Starting from the cost function in (9.14), a cheaper time-domain stochastic gradient algorithm has been derived in [26][2]. In order to speed up convergence and reduce the computational complexity, this algorithm has been implemented in the frequency domain. It has been shown in [26] that for highly non-stationary noise, this stochastic gradient algorithm suffers from a large excess error, which can be reduced by low-pass filtering the regularization term, i.e. the part of the gradient estimate that limits speech distortion. The computation of this regularization term however requires the storage of circular data buffers for the speech-and-noise samples and for the noise-only samples. Using these circular data buffers, the filter coefficients $\mathbf{w}[k]$ can be updated both during speech-periods and during noise-only-periods, but the storage of the data buffers unfortunately gives rise to a large memory usage. In [27], the regularization term has been approximated in the frequency domain, using (diagonal) speech and noise correlation matrices in the frequency domain. This approximation leads to a drastic decrease in memory usage, while also further reducing the computational complexity but not compromising the noise reduction performance and the robustness against signal model errors.

In the following section, a novel *frequency-domain criterion* for the SDW-MWF is proposed, similar to the cost function in (9.14). This frequency-domain criterion is an extension of the criterion used in [34–36] for multichannel echo cancellation to the problem of multichannel speech enhance-

[2] In [32] a similar time-domain stochastic gradient algorithm has been presented, which however invokes some independence assumptions that result in a significant performance degradation compared to the algorithm in [26].

ment. Furthermore, it provides a way for linking existing adaptive frequency-domain algorithms for the SDW-MWF [27,33] and for deriving novel adaptive algorithms, as will be shown in Sect. 9.4.

9.3 Frequency-Domain Criterion for SDW-MWF

9.3.1 Frequency-Domain Notation

We define the L-dimensional block signals $\mathbf{e}_v[m]$ and $\mathbf{e}_x[m]$ as

$$\mathbf{e}_v[m] = \left[\, e_v[mL]\; e_v[mL+1] \ldots e_v[mL+L-1] \,\right]^T , \tag{9.17}$$
$$\mathbf{e}_x[m] = \left[\, e_x[mL]\; e_x[mL+1] \ldots e_x[mL+L-1] \,\right]^T , \tag{9.18}$$

with m the block time index. The block signal $\mathbf{e}_v[m]$, representing the residual noise, can be written as

$$\mathbf{e}_v[m] = \mathbf{d}[m] - \sum_{n=M-N}^{M-1} \mathbf{V}_n^T[m]\, \mathbf{w}_n , \tag{9.19}$$

with the L-dimensional block signal $\mathbf{d}[m]$, and the $L \times L$-dimensional Toeplitz matrices $\mathbf{V}_n[m]$, $n = M-N \ldots M-1$, equal to

$$\mathbf{d}[m] = \left[\, v_0[mL-\Delta]\; v_0[mL-\Delta+1] \ldots v_0[mL-\Delta+L-1] \,\right]^T , \tag{9.20}$$
$$\mathbf{V}_n[m] = \left[\, \mathbf{v}_n[mL]\; \mathbf{v}_n[mL+1] \ldots \mathbf{v}_n[mL+L-1] \,\right] . \tag{9.21}$$

It is well known that the filtering operation $\mathbf{V}_n^T[m]\,\mathbf{w}_n$ can be calculated in the frequency domain as [35,42]

$$\mathbf{V}_n^T[m]\,\mathbf{w}_n = \left[\, \mathbf{0}_L\; \mathbf{I}_L \,\right] \mathbf{F}_{2L}^{-1}\, \mathbf{D}_{v,n}[m]\, \mathbf{F}_{2L} \begin{bmatrix} \mathbf{I}_L \\ \mathbf{0}_L \end{bmatrix} \mathbf{w}_n , \tag{9.22}$$

where $\mathbf{0}_L$ represents the $L \times L$-dimensional zero matrix, $\mathbf{I}_L$ represents the $L \times L$-dimensional unity matrix, $\mathbf{F}_{2L}$ is the $2L \times 2L$-dimensional Fourier matrix and $\mathbf{D}_{v,n}[m]$ is a $2L \times 2L$-dimensional diagonal matrix whose elements are the discrete Fourier transform of the first column of

$$\begin{bmatrix} \mathbf{V}_n^T[m-1] \\ \mathbf{V}_n^T[m] \end{bmatrix} , \tag{9.23}$$

i.e. the $2L$-dimensional vector

$$\left[\, v_n[mL-L] \ldots v_n[mL-1]\; v_n[mL] \ldots v_n[mL+L-1] \,\right]^T . \tag{9.24}$$

Hence, combining (9.19) and (9.22), the block signal $\mathbf{e}_v[m]$ can be written as

$$\begin{aligned} \mathbf{e}_v[m] &= \mathbf{d}[m] - \left[\, \mathbf{0}_L\; \mathbf{I}_L \,\right] \mathbf{F}_{2L}^{-1} \sum_{n=M-N}^{M-1} \mathbf{D}_{v,n}[m]\, \mathbf{F}_{2L} \begin{bmatrix} \mathbf{I}_L \\ \mathbf{0}_L \end{bmatrix} \mathbf{w}_n && (9.25) \\ &= \mathbf{d}[m] - \left[\, \mathbf{0}_L\; \mathbf{I}_L \,\right] \mathbf{F}_{2L}^{-1}\, \mathbf{U}_v[m]\, \mathbf{w} , && (9.26) \end{aligned}$$

with the $2L \times NL$-dimensional matrix $\mathbf{U}_v[m]$ defined as

$$\mathbf{U}_v[m] = \left[\mathbf{D}_{v,M-N}[m]\,\mathbf{F}_{2L}\begin{bmatrix}\mathbf{I}_L\\ \mathbf{0}_L\end{bmatrix} \ldots \mathbf{D}_{v,M-1}[m]\,\mathbf{F}_{2L}\begin{bmatrix}\mathbf{I}_L\\ \mathbf{0}_L\end{bmatrix}\right] \tag{9.27}$$

$$= \mathbf{D}_v[m]\,\mathbf{F}^{10}_{2NL\times NL}, \tag{9.28}$$

and the $2L \times 2NL$-dimensional matrix $\mathbf{D}_v[m]$ and the $2NL \times NL$-dimensional diagonal block-matrix $\mathbf{F}^{10}_{2NL\times NL}$ equal to

$$\mathbf{D}_v[m] = \left[\mathbf{D}_{v,M-N}[m] \ldots \mathbf{D}_{v,M-1}[m]\right], \tag{9.29}$$

$$\mathbf{F}^{10}_{2NL\times NL} = \operatorname{diag}\left[\mathbf{F}_{2L}\begin{bmatrix}\mathbf{I}_L\\ \mathbf{0}_L\end{bmatrix} \ldots \mathbf{F}_{2L}\begin{bmatrix}\mathbf{I}_L\\ \mathbf{0}_L\end{bmatrix}\right]. \tag{9.30}$$

Similarly, the block signal $\mathbf{e}_x[m]$, representing the speech distortion, can be written as

$$\mathbf{e}_x[m] = \left[\mathbf{0}_L\ \mathbf{I}_L\right]\mathbf{F}^{-1}_{2L}\,\mathbf{U}_x[m]\,\mathbf{w} = \left[\mathbf{0}_L\ \mathbf{I}_L\right]\mathbf{F}^{-1}_{2L}\,\mathbf{D}_x[m]\,\mathbf{F}^{10}_{2NL\times NL}\,\mathbf{w}, \tag{9.31}$$

with $\mathbf{U}_x[m]$ and $\mathbf{D}_x[m]$ defined similarly as $\mathbf{U}_v[m]$ and $\mathbf{D}_v[m]$ for the speech component instead as for the noise component.

If we multiply the block signals in (9.26) and (9.31) with the $L \times L$-dimensional Fourier matrix $\mathbf{F}_L$, we obtain the error signals in the frequency domain (denoted by underbars), i.e.

$$\underline{\mathbf{e}}_v[m] = \mathbf{F}_L\,\mathbf{e}_v[m] = \underline{\mathbf{d}}[m] - \mathbf{G}^{01}_{L\times 2L}\mathbf{U}_v[m]\,\mathbf{w}, \tag{9.32}$$

$$\underline{\mathbf{e}}_x[m] = \mathbf{F}_L\,\mathbf{e}_x[m] = \mathbf{G}^{01}_{L\times 2L}\mathbf{U}_x[m]\,\mathbf{w}, \tag{9.33}$$

with $\underline{\mathbf{d}}[m] = \mathbf{F}_L\,\mathbf{d}[m]$ and

$$\mathbf{G}^{01}_{L\times 2L} = \mathbf{F}_L\,\left[\mathbf{0}_L\ \mathbf{I}_L\right]\mathbf{F}^{-1}_{2L}\,. \tag{9.34}$$

We now define a *frequency-domain criterion* similar to (9.14), minimizing the weighted sum of residual noise energy and speech distortion energy, as

$$J_f[m] = (1-\lambda_v)\sum_{i=0}^{m}\lambda_v^{m-i}\underline{\mathbf{e}}_v^H[i]\,\underline{\mathbf{e}}_v[i] + \frac{1}{\mu}(1-\lambda_x)\sum_{i=0}^{m}\lambda_x^{m-i}\underline{\mathbf{e}}_x^H[i]\,\underline{\mathbf{e}}_x[i], \tag{9.35}$$

where ^{H} denotes complex conjugate of a vector or a matrix, λ_v and λ_x are exponential forgetting factors respectively for noise and speech ($0 < \lambda_v < 1$, $0 < \lambda_x < 1$), and $1/\mu$ is the trade-off parameter between noise reduction and speech distortion.

9.3.2 Normal Equations

The cost function $J_f[m]$ can be minimized by setting its derivative with respect to the (time-domain) filter coefficients $\mathbf{w}[m]$ equal to zero. Using

(9.32) and (9.33), the derivative is equal to

$$\frac{\partial J_f[m]}{\partial \mathbf{w}[m]} = (1-\lambda_v)\sum_{i=0}^{m}\lambda_v^{m-i}\Big(\mathbf{U}_v^H[i]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_v[i]\mathbf{w}[m] - \mathbf{U}_v^H[i]\,\underline{\mathbf{d}}_{2L}[i]\Big)$$

$$+\frac{1}{\mu}(1-\lambda_x)\sum_{i=0}^{m}\lambda_x^{m-i}\,\mathbf{U}_x^H[i]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_x[i]\mathbf{w}[m], \tag{9.36}$$

with

$$\underline{\mathbf{d}}_{2L}[m] = 2(\mathbf{G}_{L\times 2L}^{01})^H\underline{\mathbf{d}}[m] = \mathbf{F}_{2L}\begin{bmatrix}\mathbf{0}_L\\ \mathbf{I}_L\end{bmatrix}\mathbf{d}[m], \tag{9.37}$$

$$\mathbf{G}_{2L\times 2L}^{01} = 2(\mathbf{G}_{L\times 2L}^{01})^H\mathbf{G}_{L\times 2L}^{01}, = \mathbf{F}_{2L}\begin{bmatrix}\mathbf{0}_L & \mathbf{0}_L\\ \mathbf{0}_L & \mathbf{I}_L\end{bmatrix}\mathbf{F}_{2L}^{-1}. \tag{9.38}$$

Hence, the *normal equations* can be written as

$$\left[\mathbf{S}_v[m] + \frac{1}{\mu}\mathbf{S}_x[m]\right]\mathbf{w}[m] = \mathbf{s}[m], \tag{9.39}$$

with the $NL \times NL$-dimensional correlation matrices $\mathbf{S}_v[m]$ and $\mathbf{S}_x[m]$, and the NL-dimensional cross-correlation vector $\mathbf{s}[m]$ defined as

$$\mathbf{S}_v[m] = (1-\lambda_v)\sum_{i=0}^{m}\lambda_v^{m-i}\,\mathbf{U}_v^H[i]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_v[i] \tag{9.40}$$

$$= \lambda_v\mathbf{S}_v[m-1] + (1-\lambda_v)\,\mathbf{U}_v^H[m]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_v[m], \tag{9.41}$$

$$\mathbf{S}_x[m] = (1-\lambda_x)\sum_{i=0}^{m}\lambda_x^{m-i}\,\mathbf{U}_x^H[i]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_x[i] \tag{9.42}$$

$$= \lambda_x\mathbf{S}_x[m-1] + (1-\lambda_x)\,\mathbf{U}_x^H[m]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_x[m], \tag{9.43}$$

$$\mathbf{s}[m] = (1-\lambda_v)\sum_{i=0}^{m}\lambda_v^{m-i}\,\mathbf{U}_v^H[i]\,\underline{\mathbf{d}}_{2L}[i] \tag{9.44}$$

$$= \lambda_v\mathbf{s}[m-1] + (1-\lambda_v)\,\mathbf{U}_v^H[m]\,\underline{\mathbf{d}}_{2L}[m]. \tag{9.45}$$

The optimal Wiener filter is the solution of these normal equations.

9.3.3 Adaptive Algorithm

An algorithm for adapting $\mathbf{w}[m]$ can be found by enforcing the normal equations (9.39) at block time m and $m-1$, i.e.

$$\begin{aligned}
&\left[\mathbf{S}_v[m] + \frac{1}{\mu}\mathbf{S}_x[m]\right]\mathbf{w}[m] \\
&\quad = \lambda_v \left[\mathbf{S}_v[m-1] + \frac{1}{\mu}\mathbf{S}_x[m-1]\right]\mathbf{w}[m-1] + (1-\lambda_v)\,\mathbf{U}_v^H[m]\,\underline{\mathbf{d}}_{2L}[m] \\
&\quad = \left[\frac{1}{\mu}\frac{\lambda_v}{\lambda_x}\Big(\mathbf{S}_x[m] - (1-\lambda_x)\,\mathbf{U}_x^H[m]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_x[m]\Big) + \mathbf{S}_v[m] - \right. \\
&\qquad \left. (1-\lambda_v)\,\mathbf{U}_v^H[m]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_v[m]\right]\mathbf{w}[m-1] + (1-\lambda_v)\,\mathbf{U}_v^H[m]\,\underline{\mathbf{d}}_{2L}[m],
\end{aligned}$$

such that the update formula for $\mathbf{w}[m]$ can be written as

$$\underline{\mathbf{e}}_{v,2L}[m] = \mathbf{F}_{2L}\begin{bmatrix}\mathbf{0}_L\\ \mathbf{I}_L\end{bmatrix}\mathbf{e}_v[m] = \underline{\mathbf{d}}_{2L}[m] - \mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_v[m]\,\mathbf{w}[m-1], \tag{9.46}$$

$$\underline{\mathbf{e}}_{x,2L}[m] = \mathbf{F}_{2L}\begin{bmatrix}\mathbf{0}_L\\ \mathbf{I}_L\end{bmatrix}\mathbf{e}_x[m] = \mathbf{G}_{2L\times 2L}^{01}\mathbf{U}_x[m]\,\mathbf{w}[m-1], \tag{9.47}$$

$$\begin{aligned}
\mathbf{w}[m] = \left[\mathbf{S}_v[m] + \frac{1}{\mu}\mathbf{S}_x[m]\right]^{-1}\Bigg\{&\left[\mathbf{S}_v[m] + \frac{1}{\mu}\frac{\lambda_v}{\lambda_x}\mathbf{S}_x[m]\right]\mathbf{w}[m-1] + \\
&(1-\lambda_v)\mathbf{U}_v^H[m]\underline{\mathbf{e}}_{v,2L}[m] - \frac{1}{\mu}\frac{\lambda_v}{\lambda_x}(1-\lambda_x)\mathbf{U}_x^H[m]\underline{\mathbf{e}}_{x,2L}[m]\Bigg\}.
\end{aligned} \tag{9.48}$$

For convenience, we now define the $2NL \times 2NL$-dimensional correlation matrices $\mathbf{Q}_v[m]$ and $\mathbf{Q}_x[m]$ as

$$\mathbf{S}_v[m] = \left(\mathbf{F}_{2NL\times NL}^{10}\right)^H \mathbf{Q}_v[m]\,\mathbf{F}_{2NL\times NL}^{10}, \tag{9.49}$$

$$\mathbf{S}_x[m] = \left(\mathbf{F}_{2NL\times NL}^{10}\right)^H \mathbf{Q}_x[m]\,\mathbf{F}_{2NL\times NL}^{10}, \tag{9.50}$$

such that

$$\mathbf{Q}_v[m] = \lambda_v\mathbf{Q}_v[m-1] + (1-\lambda_v)\,\mathbf{D}_v^H[m]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{D}_v[m], \tag{9.51}$$

$$\mathbf{Q}_x[m] = \lambda_x\mathbf{Q}_x[m-1] + (1-\lambda_x)\,\mathbf{D}_x^H[m]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{D}_x[m]. \tag{9.52}$$

In addition, we define the $2NL$-dimensional frequency-domain filter $\underline{\mathbf{w}}_{2NL}[m]$ as

$$\underline{\mathbf{w}}_{2NL}[m] = \mathbf{F}_{2NL\times NL}^{10}\mathbf{w}[m] \tag{9.53}$$

$$\qquad = \left[\underline{\mathbf{w}}_{M-N,2L}^T[m]\ \ldots\ \underline{\mathbf{w}}_{M-1,2L}^T[m]\right]^T, \tag{9.54}$$

with

$$\underline{\mathbf{w}}_{n,2L}[m] = \mathbf{F}_{2L}\begin{bmatrix}\mathbf{I}_L\\ \mathbf{0}_L\end{bmatrix}\mathbf{w}_n[m]. \tag{9.55}$$

By pre-multiplying both sides of (9.48) with $\mathbf{F}^{10}_{2NL\times NL}$, and by using (9.49) and (9.50), we obtain

$$\underline{\mathbf{e}}_{v,2L}[m] = \underline{\mathbf{d}}_{2L}[m] - \mathbf{G}^{01}_{2L\times 2L}\mathbf{D}_v[m]\,\underline{\mathbf{w}}_{2NL}[m-1], \tag{9.56}$$

$$\underline{\mathbf{e}}_{x,2L}[m] = \mathbf{G}^{01}_{2L\times 2L}\mathbf{D}_x[m]\,\underline{\mathbf{w}}_{2NL}[m-1], \tag{9.57}$$

$$\begin{aligned}\underline{\mathbf{w}}_{2NL}[m] = \mathbf{F}^{10}_{2NL\times NL}\left[\mathbf{S}_v[m] + \frac{1}{\mu}\mathbf{S}_x[m]\right]^{-1}(\mathbf{F}^{10}_{2NL\times NL})^H\,\cdot\\ \Bigg\{\left[\mathbf{Q}_v[m]+\frac{1}{\mu}\frac{\lambda_v}{\lambda_x}\mathbf{Q}_x[m]\right]\underline{\mathbf{w}}_{2NL}[m-1] + (1-\lambda_v)\mathbf{D}^H_v[m]\underline{\mathbf{e}}_{v,2L}[m]\\ -\frac{1}{\mu}\frac{\lambda_v}{\lambda_x}(1-\lambda_x)\,\mathbf{D}^H_x[m]\,\underline{\mathbf{e}}_{x,2L}[m]\Bigg\}.\end{aligned} \tag{9.58}$$

In [35], it has been shown that

$$\mathbf{F}^{10}_{2NL\times NL}\,\mathbf{S}^{-1}_v[m]\,(\mathbf{F}^{10}_{2NL\times NL})^H = \mathbf{G}^{10}_{2NL\times 2NL}\,\mathbf{Q}^{-1}_v[m], \tag{9.59}$$

with the $2NL \times 2NL$-dimensional diagonal block-matrix $\mathbf{G}^{10}_{2NL\times 2NL}$ defined as

$$\mathbf{G}^{10}_{2NL\times 2NL} = \mathrm{diag}\left[\,\mathbf{G}^{10}_{2L\times 2L}\ \ldots\ \mathbf{G}^{10}_{2L\times 2L}\,\right], \tag{9.60}$$

with

$$\mathbf{G}^{10}_{2L\times 2L} = \mathbf{F}_{2L}\begin{bmatrix}\mathbf{I}_L & \mathbf{0}_L\\ \mathbf{0}_L & \mathbf{0}_L\end{bmatrix}\mathbf{F}^{-1}_{2L}, \tag{9.61}$$

such that (9.58) can be written as

$$\begin{aligned}\underline{\mathbf{w}}_{2NL}[m] = \mathbf{G}^{10}_{2NL\times 2NL}\left[\mathbf{Q}_v[m] + \frac{1}{\mu}\mathbf{Q}_x[m]\right]^{-1}\\ \Bigg\{\left[\mathbf{Q}_v[m]+\frac{1}{\mu}\frac{\lambda_v}{\lambda_x}\mathbf{Q}_x[m]\right]\underline{\mathbf{w}}_{2NL}[m-1] + (1-\lambda_v)\mathbf{D}^H_v[m]\underline{\mathbf{e}}_{v,2L}[m]\\ -\frac{1}{\mu}\frac{\lambda_v}{\lambda_x}(1-\lambda_x)\,\mathbf{D}^H_x[m]\,\underline{\mathbf{e}}_{x,2L}[m]\Bigg\}.\end{aligned} \tag{9.62}$$

In the following, we will assume that the exponential forgetting factors $\lambda_x = \lambda_v = \lambda$, such that (9.62) reduces to

$$\begin{aligned}\underline{\mathbf{w}}_{2NL}[m] = \underline{\mathbf{w}}_{2NL}[m-1] + (1-\lambda)\,\mathbf{G}^{10}_{2NL\times 2NL}\left[\mathbf{Q}_v[m] + \frac{1}{\mu}\mathbf{Q}_x[m]\right]^{-1}\cdot\\ \left\{\mathbf{D}^H_v[m]\,\underline{\mathbf{e}}_{v,2L}[m] - \frac{1}{\mu}\mathbf{D}^H_x[m]\,\underline{\mathbf{e}}_{x,2L}[m]\right\}.\end{aligned} \tag{9.63}$$

When the trade-off parameter $1/\mu = 0$, this algorithm is equal to the multichannel frequency-domain adaptive filtering algorithm derived in [35], applied to the GSC. For $1/\mu > 0$, the $2NL$-dimensional additional *regularization term*

$$\underline{\mathbf{r}}_{2NL}[m] = \frac{1}{\mu}\mathbf{D}_x^H[m]\,\underline{\mathbf{e}}_{x,2L}[m] \tag{9.64}$$

$$= \frac{1}{\mu}\mathbf{D}_x^H[m]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{D}_x[m]\,\underline{\mathbf{w}}_{2NL}[m-1], \tag{9.65}$$

limits speech distortion due to speech leakage in the noise references.

9.3.4 Practical Implementation

Since the SP-SDW-MWF takes speech leakage explicitly into account, it should in principle be possible to adapt the filter $\mathbf{w}[m]$ both during speech-periods and during noise-only-periods, unlike the standard GSC. This approach has been taken in [26], where a stochastic gradient based implementation of the SP-SDW-MWF has been proposed using 2 circular data buffers, i.e. a speech+noise buffer storing speech+noise vectors $\mathbf{y}[k]$ during speech-periods, and a noise-only buffer storing noise-only vectors $\mathbf{v}[k]$ during noise-only-periods. This implementation using circular data buffers however gives rise to a large memory usage.

If we take a closer look at (9.63), we notice that $\mathbf{D}_v[m]$ and $\underline{\mathbf{e}}_{v,2L}[m]$ can only be computed during noise-only-periods, whereas $\mathbf{D}_x[m]$ and $\underline{\mathbf{e}}_{x,2L}[m]$ can only be computed during speech-periods. Hence, we will take a similar approach as in the standard GSC, i.e. *updating the filter coefficients only during noise-only-periods*. Since during noise-only-periods the (instantaneous) correlation matrix $\mathbf{D}_x^H[m]\mathbf{G}_{2L\times 2L}^{01}\mathbf{D}_x[m]$ of the clean speech signal, required in the computation of the regularization term $\underline{\mathbf{r}}_{2NL}[m]$, is not available, we will approximate this term by the (average) correlation matrix $\mathbf{Q}_x[m]$[3], such that the regularization term can be computed as

$$\underline{\mathbf{r}}_{2NL}[m] = \frac{1}{\mu}\mathbf{Q}_x[m]\,\underline{\mathbf{w}}_{2NL}[m-1]. \tag{9.66}$$

In fact, using the correlation matrix $\mathbf{Q}_x[m]$ instead of $\mathbf{D}_x^H[m]\mathbf{G}_{2L\times 2L}^{01}\mathbf{D}_x[m]$ is quite similar to the low-pass filtering of the time-domain regularization term, which has been proposed in [26] to improve the performance in highly non-stationary noise. In practice, using the assumption that speech and noise are uncorrelated, the speech correlation matrix is approximated as

$$\mathbf{Q}_x[m] = \mathbf{Q}_y[m] - \mathbf{Q}_v[m], \tag{9.67}$$

[3] Note that a similar reasoning for computing the term $\mathbf{D}_v^H[m]\,\underline{\mathbf{e}}_{v,2L}[m]$ during speech-periods is not possible, since

$$\mathbf{D}_v^H[m]\,\underline{\mathbf{e}}_{v,2L}[m] = \mathbf{D}_v^H[m]\,\underline{\mathbf{d}}_{2L}[m] - \mathbf{D}_v^H[m]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{D}_v[m]\,\underline{\mathbf{w}}_{2NL}[m-1]$$

cannot be easily approximated, because of the term $\mathbf{D}_v^H[m]\,\underline{\mathbf{d}}_{2L}[m]$.

where $\mathbf{Q}_y[m]$ is the $2NL \times 2NL$-dimensional correlation matrix updated during speech-periods, i.e.

$$\mathbf{Q}_y[m] = \lambda \mathbf{Q}_y[m-1] + (1-\lambda)\,\mathbf{D}_y^H[m]\,\mathbf{G}_{2L\times 2L}^{01}\mathbf{D}_y[m], \tag{9.68}$$

with $\mathbf{D}_y[m]$ defined similarly as $\mathbf{D}_x[m]$. For a speech source at a fixed position, it may also be possible to determine $\mathbf{Q}_x[m]$ using calibration signals [43,44] or by additionally incorporating geometric constraints [45,46]. In conclusion, the total frequency-domain adaptive algorithm for the SDW-MWF is summarized in Algorithm 1.

9.4 Approximations for Reducing the Complexity

Algorithm 1 presents a general framework from which different adaptive algorithms can be derived. Some of these algorithms have already been presented in the literature [27,33], while other algorithms represent novel techniques for implementing the speech distortion weighted multichannel Wiener filter in the frequency domain.

9.4.1 Block-Diagonal Correlation Matrices

Since the correlation matrices $\mathbf{Q}_v[m]$ and $\mathbf{Q}_y[m]$ have no special structure, the update of these correlation matrices according to (9.51) and (9.68), and the matrix inverse required in (9.63) are computationally quite complex operations, such that in fact Algorithm 1 is not very useful in practice. However, in [34–36] it has been shown that the matrix $\mathbf{G}_{2L\times 2L}^{01}$ may be well approximated by $\mathbf{I}_{2L}/2$, because – for large L – the off-diagonal elements of $\mathbf{G}_{2L\times 2L}^{01}$ are small compared to its diagonal elements.

When using this approximation, we obtain the following update formula for the block-diagonal correlation matrices $\tilde{\mathbf{Q}}_v[m]$ and $\tilde{\mathbf{Q}}_y[m]$,

$$\tilde{\mathbf{Q}}_v[m] = \lambda \tilde{\mathbf{Q}}_v[m] + (1-\lambda)\,\mathbf{D}_v^H[m]\,\mathbf{D}_v[m]/2, \tag{9.69}$$

$$\tilde{\mathbf{Q}}_y[m] = \lambda \tilde{\mathbf{Q}}_y[m] + (1-\lambda)\,\mathbf{D}_y^H[m]\,\mathbf{D}_y[m]/2, \tag{9.70}$$

which consist of N^2 $2L \times 2L$-dimensional diagonal sub-matrices $\tilde{\mathbf{Q}}_{v,np}[m]$ and $\tilde{\mathbf{Q}}_{y,np}[m]$, $n = M-N \ldots M-1$, $p = M-N \ldots M-1$. In addition, we obtain the following update formula for the filter coefficients,

$$\underline{\mathbf{w}}_{2NL}[m] = \underline{\mathbf{w}}_{2NL}[m-1] + \rho(1-\lambda)\,\mathbf{G}_{2NL\times 2NL}^{10}\left[\tilde{\mathbf{Q}}_v[m] + \frac{1}{\mu}\tilde{\mathbf{Q}}_x[m]\right]^{-1} \cdot \\ \left\{\mathbf{D}_v^H[m]\,\underline{\mathbf{e}}_{v,2L}[m] - \underline{\mathbf{r}}_{2NL}[m]\right\}, \tag{9.71}$$

where the step size parameter ρ is assumed to be in the range $0 < \rho \leq 1$. This equation requires the computation of the inverse of the block-diagonal matrix

Algorithm 1 Frequency-domain implementation of SDW-MWF.

Matrix definitions:

$\mathbf{F}_{2L} = 2L \times 2L$-dimensional DFT matrix

$\mathbf{0}_L = L \times L$-dimensional zero matrix, $\quad \mathbf{I}_L = L \times L$-dimensional identity matrix

$$\mathbf{G}^{01}_{2L\times 2L} = \mathbf{F}_{2L} \begin{bmatrix} \mathbf{0}_L & \mathbf{0}_L \\ \mathbf{0}_L & \mathbf{I}_L \end{bmatrix} \mathbf{F}^{-1}_{2L}, \quad \mathbf{G}^{10}_{2L\times 2L} = \mathbf{F}_{2L} \begin{bmatrix} \mathbf{I}_L & \mathbf{0}_L \\ \mathbf{0}_L & \mathbf{0}_L \end{bmatrix} \mathbf{F}^{-1}_{2L}$$

$$\mathbf{G}^{10}_{2NL\times 2NL} = \mathrm{diag}\left[\, \mathbf{G}^{10}_{2L\times 2L} \ \ldots \ \mathbf{G}^{10}_{2L\times 2L} \,\right]$$

For each new block of L samples:

$$\mathbf{d}[m] = \left[\, y_0[mL-\Delta] \ y_0[mL-\Delta+1] \ \ldots \ y_0[mL-\Delta+L-1] \,\right]^T$$

$$\mathbf{D}_{y,n}[m] = \mathrm{diag}\left\{ \mathbf{F}_{2L} \left[\, y_n[mL-L] \ \ldots \ y_n[mL+L-1] \,\right]^T \right\}, \qquad n = M-N \ldots M-1$$

$$\mathbf{D}_y[m] = \left[\, \mathbf{D}_{y,M-N}[m] \ \ldots \ \mathbf{D}_{y,M-1}[m] \,\right]$$

Output signal:

$$\mathbf{e}[m] = \mathbf{d}[m] - \left[\, \mathbf{0}_L \ \mathbf{I}_L \,\right] \mathbf{F}^{-1}_{2L} \mathbf{D}_y[m] \, \underline{\mathbf{w}}_{2NL}[m-1]$$

If speech detected:

$$\mathbf{Q}_y[m] = \lambda \mathbf{Q}_y[m-1] + (1-\lambda) \, \mathbf{D}^H_y[m] \, \mathbf{G}^{01}_{2L\times 2L} \mathbf{D}_y[m]$$

If noise detected: $\mathbf{D}_v[m] = \mathbf{D}_y[m]$

$$\mathbf{Q}_v[m] = \lambda \mathbf{Q}_v[m-1] + (1-\lambda) \, \mathbf{D}^H_v[m] \, \mathbf{G}^{01}_{2L\times 2L} \mathbf{D}_v[m]$$

Update formula (only during noise-only-periods):

$$\underline{\mathbf{e}}_{v,2L}[m] = \mathbf{F}_{2L} \begin{bmatrix} \mathbf{0}_L \\ \mathbf{I}_L \end{bmatrix} \mathbf{e}[m]$$

$$\mathbf{Q}_x[m] = \mathbf{Q}_y[m] - \mathbf{Q}_v[m]$$

$$\underline{\mathbf{r}}_{2NL}[m] = \tfrac{1}{\mu} \mathbf{Q}_x[m] \, \underline{\mathbf{w}}_{2NL}[m-1]$$

$$\underline{\mathbf{w}}_{2NL}[m] = \underline{\mathbf{w}}_{2NL}[m-1] + (1-\lambda) \, \mathbf{G}^{10}_{2NL\times 2NL} \left[\mathbf{Q}_v[m] + \tfrac{1}{\mu} \mathbf{Q}_x[m] \right]^{-1} \cdot \left\{ \mathbf{D}^H_v[m] \, \underline{\mathbf{e}}_{v,2L}[m] - \underline{\mathbf{r}}_{2NL}[m] \right\}$$

$\tilde{\mathbf{Q}}_v[m] + 1/\mu \, \tilde{\mathbf{Q}}_x[m]$. It is well known that the inverse of a block-diagonal matrix $\mathbf{Q}$, consisting of N^2 $2L \times 2L$-dimensional diagonal sub-matrices $\mathbf{Q}_{np}$, i.e.

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{M-N,M-N} & \cdots & \mathbf{Q}_{M-N,M-1} \\ \vdots & & \vdots \\ \mathbf{Q}_{M-1,M-N} & \cdots & \mathbf{Q}_{M-1,M-1} \end{bmatrix}, \tag{9.72}$$

is also a block-diagonal matrix. Its computation corresponds to inverting $2L$ $N \times N$-dimensional matrices, which is attractive from a complexity point of view. The block-diagonal matrix $\mathbf{Q}$ can be easily transformed to the diagonal block-matrix $\bar{\mathbf{Q}}$,

$$\bar{\mathbf{Q}} = \operatorname{diag}\left[\,\bar{\mathbf{Q}}_0 \,\ldots\, \bar{\mathbf{Q}}_{2L-1}\,\right], \tag{9.73}$$

consisting of $2L$ $N \times N$-dimensional sub-matrices $\bar{\mathbf{Q}}_l$, $l = 0 \ldots 2L-1$, on its diagonal, by the transformation

$$\bar{\mathbf{Q}} = \mathbf{A}^T\, \mathbf{Q}\, \mathbf{A}. \tag{9.74}$$

The matrix $\mathbf{A}$ is a $2NL \times 2NL$-dimensional column permutation matrix (and hence $\mathbf{A}^T$ is a row permutation matrix), consisting of $2NL$ $2L \times N$-dimensional sub-matrices $\mathbf{A}_{nl}$, $n = M-N \ldots M-1$, $l = 0 \ldots 2L-1$, where the (l, n)-th element of $\mathbf{A}_{nl}$ is equal to 1. It readily follows that

$$\mathbf{Q}^{-1} = \mathbf{A}\, \bar{\mathbf{Q}}^{-1}\, \mathbf{A}^T, \tag{9.75}$$

where $\bar{\mathbf{Q}}^{-1}$ can be easily computed by inverting the $N \times N$-dimensional sub-matrices $\bar{\mathbf{Q}}_l$ on its diagonal, i.e.

$$\bar{\mathbf{Q}}^{-1} = \operatorname{diag}\left[\,\bar{\mathbf{Q}}_0^{-1} \,\ldots\, \bar{\mathbf{Q}}_{2L-1}^{-1}\,\right]. \tag{9.76}$$

In addition, one should take care that the matrix $\tilde{\mathbf{Q}}_v[m] + 1/\mu\, \tilde{\mathbf{Q}}_x[m]$ in (9.71) is positive definite. When this matrix is not positive definite, this actually has the same effect as a negative step size ρ, i.e. divergence of the filter coefficients. The noise correlation matrix $\tilde{\mathbf{Q}}_v[m]$ is always positive definite, but the speech correlation matrix $\tilde{\mathbf{Q}}_x[m]$ may not always be positive definite (especially for non-stationary signals), since it is computed as $\tilde{\mathbf{Q}}_x[m] = \tilde{\mathbf{Q}}_y[m] - \tilde{\mathbf{Q}}_v[m]$, where $\tilde{\mathbf{Q}}_y[m]$ and $\tilde{\mathbf{Q}}_v[m]$ are estimated during (different) speech-periods and noise-only-periods. Checking the positive definiteness of a matrix comes down to computing its eigenvalues.

It can be easily shown that the eigenvalues γ of a block-diagonal matrix $\mathbf{Q}$ are equal to the set of eigenvalues of its $N \times N$-dimensional sub-matrices $\bar{\mathbf{Q}}_l$, $l = 0 \ldots 2L-1$, since using (9.74) and the fact that $\mathbf{A}^T\mathbf{A} = \mathbf{I}_{2NL}$ and $\det(\mathbf{A}) = \pm 1$, it follows that

$$\det(\mathbf{Q} - \gamma \mathbf{I}_{2NL}) = \det\left(\mathbf{A}(\bar{\mathbf{Q}} - \gamma \mathbf{I}_{2NL})\mathbf{A}^T\right) = \det(\bar{\mathbf{Q}} - \gamma \mathbf{I}_{2NL}) \tag{9.77}$$

$$= \prod_{l=0}^{2L-1} \det(\bar{\mathbf{Q}}_l - \gamma \mathbf{I}_N). \tag{9.78}$$

Hence, before computing the inverse of the matrix $\tilde{\mathbf{Q}}_v[m] + 1/\mu\, \tilde{\mathbf{Q}}_x[m]$ in (9.71), we first compute the eigenvalues of the matrix $\tilde{\mathbf{Q}}_x[m]$. We will then use the inverse of the regularized matrix

$$\tilde{\mathbf{Q}}_v[m] + \frac{1}{\mu}\left[\tilde{\mathbf{Q}}_x[m] - \min(\gamma_{min}, 0)\, \mathbf{I}_{2NL}\right] + \delta\, \mathbf{I}_{2NL} \tag{9.79}$$

in (9.71), with γ_{min} the smallest eigenvalue of $\tilde{\mathbf{Q}}_x[m]$ and δ a small positive regularization factor. Whereas in general computing the smallest eigenvalue of an $N \times N$-dimensional Hermitian matrix is computationally quite complex, for $N = 2$ the smallest eigenvalue $\gamma_{l,min}$ of the sub-matrix

$$\bar{\mathbf{Q}}_l = \begin{bmatrix} \bar{q}_{l,11} & \bar{q}_{l,12} \\ \bar{q}^*_{l,12} & \bar{q}_{l,22} \end{bmatrix}, \tag{9.80}$$

with $\bar{q}_{l,11}$ and $\bar{q}_{l,22}$ real-valued, can be easily computed as

$$\gamma_{l,min} = \frac{(\bar{q}_{l,11} + \bar{q}_{l,22}) - \sqrt{(\bar{q}_{l,11} - \bar{q}_{l,22})^2 + 4|\bar{q}_{l,12}|^2}}{2}. \tag{9.81}$$

9.4.2 Diagonal Correlation Matrices

In a further approximation, we can decouple the updates for the N filters $\underline{\mathbf{w}}_{n,2L}[m]$ in (9.71) by neglecting the off-diagonal elements of the matrix $\tilde{\mathbf{Q}}_v[m] + 1/\mu\, \tilde{\mathbf{Q}}_x[m]$, which represent the inter-channel correlation. Hence, the update formula for the filter coefficients $\underline{\mathbf{w}}_{n,2L}[m]$, $n = M - N \dots M - 1$ becomes

$$\begin{aligned}\underline{\mathbf{w}}_{n,2L}[m] = \underline{\mathbf{w}}_{n,2L}[m-1] + \rho(1-\lambda)\, \mathbf{G}^{10}_{2L\times 2L} \left[\tilde{\mathbf{Q}}_{v,nn}[m] + \frac{1}{\mu}\tilde{\mathbf{Q}}_{x,nn}[m]\right]^{-1} \cdot \\ \left\{\mathbf{D}^H_{v,n}[m]\, \underline{\mathbf{e}}_{v,2L}[m] - \underline{\mathbf{r}}_{n,2L}[m]\right\},\end{aligned} \tag{9.82}$$

with $\tilde{\mathbf{Q}}_{v,nn}[m]$ and $\tilde{\mathbf{Q}}_{x,nn}[m]$ the $2L \times 2L$-dimensional diagonal sub-matrices on the diagonal of $\tilde{\mathbf{Q}}_v[m]$ and $\tilde{\mathbf{Q}}_x[m]$, and $\underline{\mathbf{r}}_{n,2L}[m]$ a $2L$-dimensional sub-vector of $\underline{\mathbf{r}}_{2NL}[m]$[4]. Ensuring the positive definiteness of $\tilde{\mathbf{Q}}_{x,nn}[m]$ now is very easy, since the eigenvalues of $\tilde{\mathbf{Q}}_{x,nn}[m]$ are equal to the diagonal elements. We expect that updating the filter coefficients using these diagonal correlation matrices will be slower than using the block-diagonal correlation matrices, since the inter-channel correlation is not taken into account any more.

Where in (9.82) a different step size matrix $\tilde{\mathbf{Q}}_{v,nn}[m] + 1/\mu\tilde{\mathbf{Q}}_{x,nn}[m]$ is used for each channel n, it is also possible to use a common step size matrix $\tilde{\mathbf{Q}}_c$, e.g. the sum or average over all channels, i.e.

$$\begin{aligned}\underline{\mathbf{w}}_{n,2L}[m] = \underline{\mathbf{w}}_{n,2L}[m-1] + \rho(1-\lambda)\, \mathbf{G}^{10}_{2L\times 2L} \tilde{\mathbf{Q}}_c^{-1}[m] \cdot \\ \left\{\mathbf{D}^H_{v,n}[m]\, \underline{\mathbf{e}}_{v,2L}[m] - \underline{\mathbf{r}}_{n,2L}[m]\right\},\end{aligned} \tag{9.83}$$

$$\tilde{\mathbf{Q}}_c[m] = \left(\frac{1}{N}\right) \sum_{n=M-N}^{M-1} \tilde{\mathbf{Q}}_{v,nn}[m] + \frac{1}{\mu}\tilde{\mathbf{Q}}_{x,nn}[m]. \tag{9.84}$$

[4] Note that we still use the off-diagonal elements of $\tilde{\mathbf{Q}}_x[m]$ for computing the regularization term $\underline{\mathbf{r}}_{2NL}[m]$.

In fact, this algorithm is very similar to the algorithm already presented in [27]. Note however that the algorithm in [27] has been derived as a frequency-domain implementation of a time-domain stochastic gradient algorithm for minimizing the (time-domain) cost function (9.14).

9.4.3 Unconstrained Algorithms

In Sect. 9.4.1, the term $\mathbf{G}^{01}_{2L\times 2L}$ in the calculation of the correlation matrices has been approximated by $\mathbf{I}_{2L}/2$. It is also possible to use the same approximation for the term $\mathbf{G}^{10}_{2L\times 2L}$ and hence approximate $\mathbf{G}^{10}_{2NL\times 2NL}$ in the update formula of the filter coefficients in (9.63) by

$$\mathbf{G}^{10}_{2NL\times 2NL} \approx \text{diag}\left[\mathbf{I}_{2L}/2 \ldots \mathbf{I}_{2L}/2\right] = \mathbf{I}_{2NL}/2, \tag{9.85}$$

resulting in the following so-called unconstrained update formula,

$$\underline{\mathbf{w}}_{2NL}[m] = \underline{\mathbf{w}}_{2NL}[m-1] + \frac{(1-\lambda)}{2}\left[\mathbf{Q}_v[m] + \frac{1}{\mu}\mathbf{Q}_x[m]\right]^{-1} \cdot \\ \left\{\mathbf{D}_v^H[m]\,\underline{\mathbf{e}}_{v,2L}[m] - \underline{\mathbf{r}}_{2NL}[m]\right\}. \tag{9.86}$$

This update formula gives rise to a lower computational complexity, since it requires $2N$ less FFT operations. However, using this update formula it cannot be guaranteed that the second half of $\mathbf{F}^{-1}_{2L}\underline{\mathbf{w}}_{n,2L}[m]$, $n = M-N \ldots M-1$, is equal to zero, cf. (9.55).

In fact, this update formula can also be derived by setting the derivative of the cost function $J_f[m]$ in (9.35) with respect to the (frequency-domain) filter coefficients $\underline{\mathbf{w}}_{2NL}[m]$ equal to zero. Since from (9.32), (9.33), (9.46) and (9.47), it readily follows that $\underline{\mathbf{e}}^H_{v,2L}[i]\,\underline{\mathbf{e}}_{v,2L}[i] = 2\,\underline{\mathbf{e}}^H_v[i]\,\underline{\mathbf{e}}_v[i]$ and $\underline{\mathbf{e}}^H_{x,2L}[i]\,\underline{\mathbf{e}}_{x,2L}[i] = 2\,\underline{\mathbf{e}}^H_x[i]\,\underline{\mathbf{e}}_x[i]$, the cost function $J_f[m]$ can be written as

$$J_f[m] = \frac{(1-\lambda_v)}{2}\sum_{i=0}^{m}\lambda_v^{m-i}\underline{\mathbf{e}}^H_{v,2L}[i]\underline{\mathbf{e}}_{v,2L}[i] + \frac{1}{\mu}\frac{(1-\lambda_x)}{2}\sum_{i=0}^{m}\lambda_x^{m-i}\underline{\mathbf{e}}^H_{x,2L}[i]\underline{\mathbf{e}}_{x,2L}[i].$$

Hence, using (9.56) and (9.57) and the fact that $(\mathbf{G}^{01}_{2L\times 2L})^H\mathbf{G}^{01}_{2L\times 2L} = \mathbf{G}^{01}_{2L\times 2L}$ and $(\mathbf{G}^{01}_{2L\times 2L})^H\underline{\mathbf{d}}_{2L}[i] = \underline{\mathbf{d}}_{2L}[i]$, the derivative of $J_f[m]$ with respect to $\underline{\mathbf{w}}_{2NL}[m]$ is equal to

$$\frac{\partial J_f[m]}{\partial \underline{\mathbf{w}}_{2NL}[m]} = (1-\lambda_v)\sum_{i=0}^{m}\lambda_v^{m-i}\Big(\mathbf{D}_v^H[i]\mathbf{G}^{01}_{2L\times 2L}\mathbf{D}_v[i]\underline{\mathbf{w}}_{2NL}[m] - \mathbf{D}_v^H[i]\underline{\mathbf{d}}_{2L}[i]\Big) \\ + \frac{1}{\mu}(1-\lambda_x)\sum_{i=0}^{m}\lambda_x^{m-i}\,\mathbf{D}_x^H[i]\,\mathbf{G}^{01}_{2L\times 2L}\mathbf{D}_x[i]\underline{\mathbf{w}}_{2NL}[m]. \tag{9.87}$$

Setting the derivative equal to zero, one obtains the normal equations

$$\left[\mathbf{Q}_v[m] + \frac{1}{\mu}\mathbf{Q}_x[m]\right]\underline{\mathbf{w}}_{2NL}[m] = \mathbf{q}[m]\;, \tag{9.88}$$

Table 9.1 Step size matrix $\mathbf{\Lambda}[m]$ for different algorithms (C: constrained, U: unconstrained, BD: block-diagonal, D1: diagonal - channel, D2: diagonal - common).

Algorithm	**Step size matrix**
Algo 1 (C-BD)	$\mathbf{G}^{10}_{2NL\times 2NL}\left[\tilde{\mathbf{Q}}_v[m]+\frac{1}{\mu}\tilde{\mathbf{Q}}_x[m]\right]^{-1}$
Algo 2 (U-BD)	$\frac{1}{2}\left[\tilde{\mathbf{Q}}_v[m]+\frac{1}{\mu}\tilde{\mathbf{Q}}_x[m]\right]^{-1}$
Algo 3 (C-D1)	$\mathbf{G}^{10}_{2NL\times 2NL}\mathrm{diag}\{\left[\tilde{\mathbf{Q}}_{v,nn}[m]+\frac{1}{\mu}\tilde{\mathbf{Q}}_{x,nn}[m]\right]^{-1}\}$
Algo 4 (U-D1)	$\frac{1}{2}\,\mathrm{diag}\{\left[\tilde{\mathbf{Q}}_{v,nn}[m]+\frac{1}{\mu}\tilde{\mathbf{Q}}_{x,nn}[m]\right]^{-1}\}$
Algo 5 (C-D2)	$\mathbf{G}^{10}_{2NL\times 2NL}\mathrm{diag}\{\left[(1/N)\sum_{n=M-N}^{M-1}\tilde{\mathbf{Q}}_{v,nn}[m]+\frac{1}{\mu}\tilde{\mathbf{Q}}_{x,nn}[m]\right]^{-1}\}$
Algo 6 (U-D2)	$\frac{1}{2}\,\mathrm{diag}\{\left[(1/N)\sum_{n=M-N}^{M-1}\tilde{\mathbf{Q}}_{v,nn}[m]+\frac{1}{\mu}\tilde{\mathbf{Q}}_{x,nn}[m]\right]^{-1}\}$

with $\mathbf{Q}_v[m]$ and $\mathbf{Q}_x[m]$ defined in (9.51) and (9.52), and the $2NL$-dimensional vector $\mathbf{q}[m]$ equal to

$$\mathbf{q}[m]=\lambda_v\mathbf{q}[m-1]+(1-\lambda_v)\,\mathbf{D}_v^H[m]\,\underline{\mathbf{d}}_{2L}[m]. \tag{9.89}$$

Hence, by enforcing the normal equations at block time m and $m-1$ (and assuming $\lambda_x=\lambda_v=\lambda$), we obtain the update formula (9.86). In addition, for the unconstrained algorithms one can also approximate the correlation matrices $\mathbf{Q}_v[m]$ and $\mathbf{Q}_x[m]$ by block-diagonal or diagonal matrices, cf. Sects. 9.4.1 and 9.4.2.

9.4.4 Summary

Summarizing all presented algorithms in Sect. 9.4, the update formula for the filter coefficients $\underline{\mathbf{w}}_{2NL}[m]$ can be written as

$$\underline{\mathbf{w}}_{2NL}[m]=\underline{\mathbf{w}}_{2NL}[m-1]+\rho(1-\lambda)\,\mathbf{\Lambda}[m]\Big\{\mathbf{D}_v^H[m]\,\underline{\mathbf{e}}_{v,2L}[m]-\underline{\mathbf{r}}_{2NL}[m]\Big\}, \tag{9.90}$$

where the $2NL\times 2NL$-dimensional step size matrix $\mathbf{\Lambda}[m]$ is summarized in Table 9.1. For all algorithms, the matrix $\tilde{\mathbf{Q}}_x[m]$ needs to be regularized in order to ensure that it is positive definite. The algorithm presented in [27] corresponds to **Algo 5**, while in [33] a multichannel frequency-domain algorithm for speech enhancement has been presented that bears quite some similarities to **Algo 2**. However, in [33] the speech component instead of the noise component is estimated, and the parameter $1/\mu$ is only effectively applied to the step size matrix (i.e. no regularization term $\underline{\mathbf{r}}_{2NL}[m]$ is present), such that the algorithm in [33] comes down to the MMSE estimation of the speech component.

9.5 Experimental Results

In this section experimental results are presented for a hearing aid application. For small-sized microphone arrays as typically used in hearing aids, robustness is very important, since small-sized microphone arrays exhibit a large sensitivity to signal model errors [47]. Sect. 9.5.1 describes the setup and defines the used performance measures. In Sect. 9.5.2 the performance of the adaptive algorithms is analyzed, and the impact of different parameter settings for the SP-SDW-MWF (i.e. filter $\mathbf{w}_0$ and $1/\mu$) on the performance and the robustness against signal model errors is evaluated. In Sect. 9.5.3 the tracking performance is investigated for a changing noise scenario.

9.5.1 Setup and Performance Measures

A behind-the-ear hearing aid with $M = 3$ omni-directional microphones (Knowles FG-3452) in an end-fire configuration has been mounted on a dummy head in an office room. The inter-spacing between the first and the second microphone is about 1 cm and the inter-spacing between the second and the third microphone is about 1.5 cm. The reverberation time $T_{60\text{dB}}$ of the office room is about 700 ms. The speech source and the noise sources are positioned at a distance of 1 m from the head: the speech source in front of the head (0°), and the noise sources at an angle θ with respect to the speech source. Both the speech signal and the noise signal have a level of 70 dB at the center of the head. For evaluation purposes, the speech and the noise signal have been recorded separately. The sampling frequency is equal to 16 kHz.

The microphone signals are pre-whitened prior to processing in order to improve the intelligibility, and the output signal $z[k]$ is accordingly dewhitened [48]. In the experiments, the microphones have been calibrated using anechoic recordings of a speech-weighted noise signal at 0° with the microphone array mounted on the head. A delay-and-sum beamformer is used for the fixed beamformer $\mathbf{A}(z)$, since – in the case of small microphone inter-spacing – this beamformer is quite robust to signal model errors. The blocking matrix $\mathbf{B}(z)$ pair-wisely subtracts the time-aligned calibrated microphone signals.

To assess the performance of the different algorithms, the broadband intelligibility weighted signal-to-noise ratio improvement $\varDelta\text{SNR}_{\text{intellig}}$ is used, which is defined as [49]

$$\varDelta\text{SNR}_{\text{intellig}} = \sum_i I_i \left(\text{SNR}_{i,\text{out}} - \text{SNR}_{i,\text{in}}\right), \tag{9.91}$$

where the band importance function I_i expresses the importance of the ith one-third octave band with center frequency f_i^c for intelligibility, and where $\text{SNR}_{i,\text{out}}$ and $\text{SNR}_{i,\text{in}}$ are respectively the output SNR and the input SNR (in dB) in this band. The center frequencies f_i^c and the values I_i are defined in [50]. The intelligibility weighted SNR improvement reflects how much the speech intelligibility is improved by the noise reduction algorithms, but does not take into account speech distortion.

In order to measure the amount of (linear) speech distortion, we similarly define an intelligibility weighted spectral distortion measure $\mathrm{SD}_{\mathrm{intellig}}$,

$$\mathrm{SD}_{\mathrm{intellig}} = \sum_i I_i \, \mathrm{SD}_i, \tag{9.92}$$

with SD_i the average spectral distortion (dB) in the ith one-third octave band, calculated as

$$\mathrm{SD}_i = \frac{1}{\left(2^{1/6} - 2^{-1/6}\right) f_i^c} \int_{2^{-1/6} f_i^c}^{2^{1/6} f_i^c} \left|10 \log_{10} G_x(f)\right| df, \tag{9.93}$$

with $G_x(f)$ the power transfer function for the speech component from the input to the output of the noise reduction algorithm.

All algorithms are evaluated for a filter length $L = 32$, and the input SNR of the microphone signals is equal to 0 dB. The speech-periods and the noise-only-periods, used for updating the correlation matrices $\tilde{\mathbf{Q}}_y[m]$ and $\tilde{\mathbf{Q}}_v[m]$ and the adaptive filter, have been marked manually. In order to exclude the effect of the spatial pre-processor, the performance measures (9.91) and (9.92) are calculated with respect to the output of the fixed beamformer, i.e. the speech reference $y_0[k]$. In some experiments, a microphone gain mismatch of 4 dB is applied to the second microphone to illustrative the sensitivity to signal model errors. Among the different possible signal model errors, microphone mismatch has been found to be quite harmful to the performance of the GSC in a hearing aid application [47]. In hearing aids, microphones are rarely matched in gain and phase, with gain and phase differences between microphone characteristics of up to 6 dB and 10° [51].

9.5.2 SNR Improvement and Robustness Against Microphone Mismatch

For the experiments in this section, the desired speech source at 0° consists of sentences from the HINT-database [52] spoken by a male speaker, and a complex noise scenario consisting of 5 spectrally non-stationary multi-talker babble noise sources at 75°, 120°, 180°, 240° and 285°, has been used.

Figure 9.2 depicts the convergence of the SNR improvement for different adaptive algorithms (constrained vs. unconstrained, block-diagonal vs. diagonal step size matrix) for different values of the the step size parameter ρ and the exponential forgetting factor λ. The exponential forgetting factor $\lambda = 0.995$ corresponds to an averaging of the correlation matrices over approximately $1/(1-\lambda) = 200$ blocks of $L = 32$ samples, i.e. 0.4 seconds, whereas the factor $\lambda = 0.99875$ corresponds to an averaging over 800 blocks,

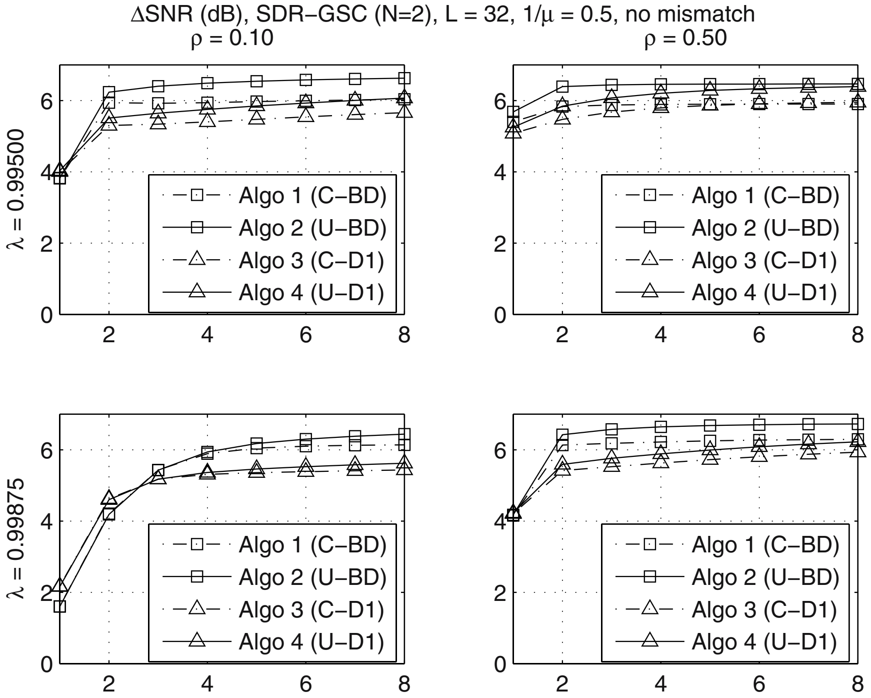


Fig. 9.2. Effect of the step size parameter ρ and the exponential forgetting factor λ on the convergence of the SNR improvement for different adaptive algorithms (SDR-GSC, $L = 32$, $1/\mu = 0.5$, no microphone mismatch).

i.e. 1.6 seconds. In this experiment, we have used the SDR-GSC ($N = 2$) with trade-off parameter $1/\mu = 0.5$ and no microphone mismatch present. Of course, similar plots can be obtained for the SP-SDW-MWF ($N = 3$), for different values of the trade-off parameter and when microphone mismatch is present. From Fig. 9.2 it can be seen that using a block-diagonal step size matrix gives rise to a faster convergence than using a diagonal step size matrix. In addition, the larger the step size parameter ρ, the faster the convergence (of course, taking ρ too large will give rise to divergence). From Fig. 9.2 it can also be seen that the larger the exponential forgetting factor λ, the slower the convergence, but the larger the SNR improvement after convergence. This can be explained by the fact that for spectrally and/or spatially stationary sources a better estimate of the correlation matrices is obtained for larger λ.

Figure 9.3 plots the SNR improvement and the speech distortion of the SDR-GSC ($N = 2$), using the unconstrained update formula (with block-diagonal and diagonal step size matrix), as a function of the trade-off parameter $1/\mu$. This figure also depicts the effect of a gain mismatch of 4 dB at the second microphone. In the absence of microphone mismatch, the amount of

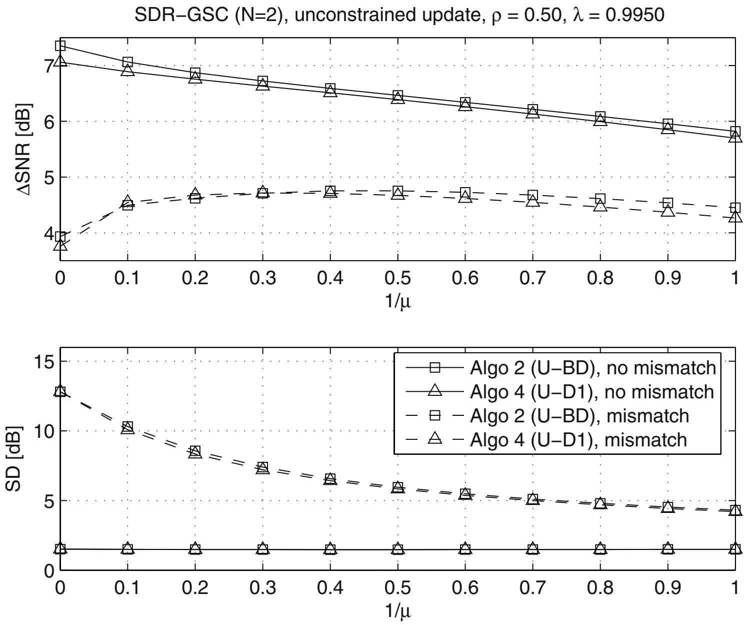


Fig. 9.3. SNR improvement and speech distortion of SDR-GSC in function of trade-off parameter $1/\mu$, without and with gain mismatch (unconstrained update formula, $\rho = 0.5$, $\lambda = 0.995$).

speech leakage into the noise references is limited, such that the speech distortion is small for all $1/\mu$. However, since there is some speech leakage present due to reverberation, the SNR improvement decreases for increasing $1/\mu$. In the presence of microphone mismatch, the amount of speech leakage into the noise references grows. For the standard GSC, i.e. $1/\mu = 0$, significant speech distortion now occurs and the SNR improvement is seriously degraded. Setting $1/\mu > 0$ improves the performance of the GSC in the presence of signal model errors, i.e. speech distortion decreases and the SNR degradation becomes smaller. For the given setup, a value $1/\mu = 0.5$ seems appropriate for guaranteeing good performance for a gain mismatch up to 4 dB.

For the same setup, Fig. 9.4 plots the SNR improvement and the speech distortion of the SP-SDW-MWF ($N = 3$) as a function of $1/\mu$. This figure shows that the speech distortion and the SNR improvement decreases for increasing $1/\mu$. This figure also shows that the speech distortion for the SP-SDW-MWF is larger than for the SDR-GSC[5], but that both the SNR improvement and the speech distortion are hardly affected by microphone mismatch.

[5] In [25], it has been shown that the SP-SDW-MWF can be interpreted as an SDR-GSC with a single-channel post-filter in the absence of speech leakage.

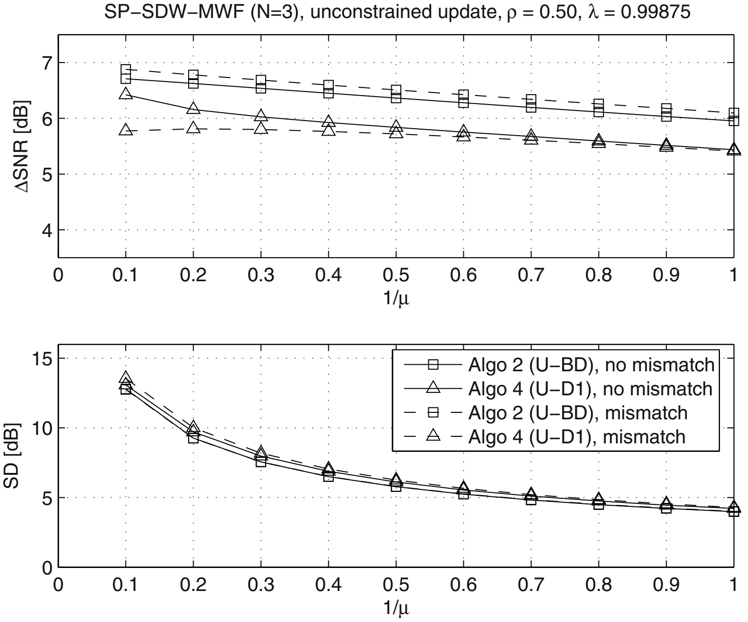


Fig. 9.4. SNR improvement and speech distortion of SP-SDW-MWF in function of trade-off parameter $1/\mu$, without and with gain mismatch (unconstrained update formula, $\rho = 0.5$, $\lambda = 0.99875$).

9.5.3 Tracking Performance

In order to investigate the tracking performance, we have used a stationary speech-weighted noise signal both for the desired speech source and for the noise sources. We consider 2 noise scenarios: scenario 1 with five noise sources at 75°, 120°, 180°, 240° and 285°, and scenario 2 with three noise sources at 120°, 240° and 285°. The noise scenario suddenly changes from scenario 1 to scenario 2 after 45 seconds, and then changes back to scenario 1 after 90 seconds. The speech component consists of alternating segments of silence and signal, each with a length of 1600 samples. For both noise scenarios, the signals received at the microphones have been normalized, such that for both scenarios the input SNR is equal to 0 dB.

Figure 9.5 plots the SNR improvement, the speech distortion and the residual noise energy ε_v^2 for the GSC ($1/\mu = 0$) and the SDR-GSC ($1/\mu = 0.5$) using the unconstrained update formula with block-diagonal step size matrix (Algo 2), in the case of a gain mismatch of 4 dB. These performance measures have been calculated per segment of 3200 samples. Again, this figure shows that the SDR-GSC is more robust to signal model errors than the GSC, since the SDR-GSC gives rise to a larger SNR improvement and a smaller speech distortion than the GSC, although the residual noise energy is larger (for

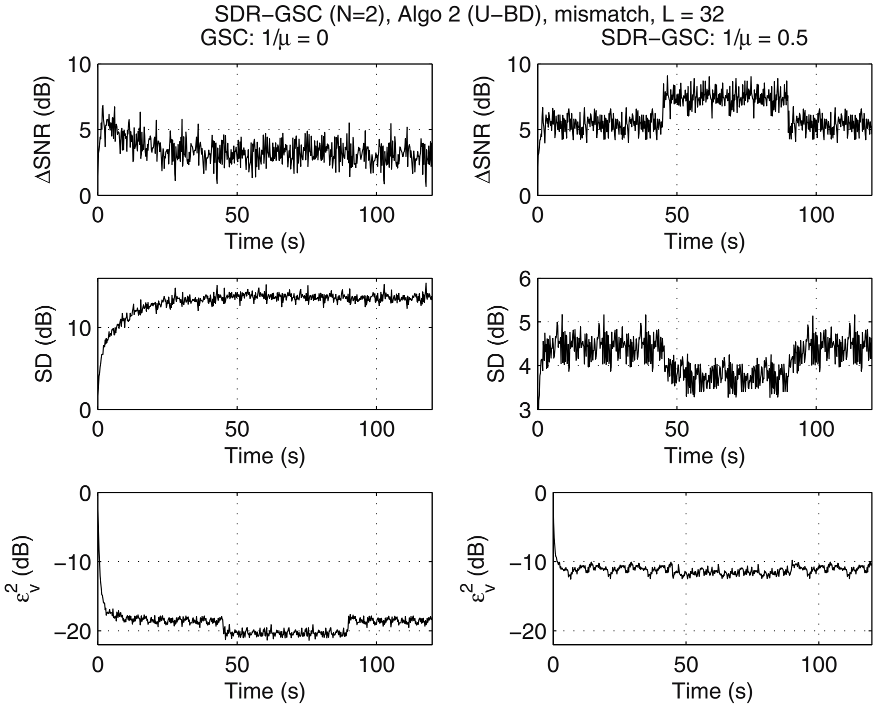


Fig. 9.5. Tracking performance of GSC and SDR-GSC for a changing noise scenario.

both noise scenarios). This figure also shows that the performance of the SDR-GSC for noise scenario 2 is better than for noise scenario 1, and that the SDR-GSC is able to track sudden changes in noise scenarios.

9.6 Conclusions

In this chapter, we have discussed a robust multi-microphone speech enhancement technique, called the spatially pre-processed speech distortion weighted multichannel Wiener filter (SP-SDW-MWF). The SP-SDW-MWF takes speech distortion due to speech leakage explicitly into account in the design criterion of the adaptive filter, and is hence more robust against signal model errors than the standard GSC. Depending on its parameter setting, the SP-SDW-MWF encompasses the standard GSC and the speech distortion regularized GSC (SDR-GSC) as special cases. We have presented a frequency-domain criterion for the SDW-MWF, which provides a way for linking existing adaptive frequency-domain algorithms and for deriving novel adaptive algorithms for implementing the SDW-MWF. The main difference between these adaptive algorithms consists in the calculation of the step size matrix (constrained vs. unconstrained, block-diagonal vs. diagonal) used in the update formula for the adaptive filter. Experimental results using a small-sized microphone array show that setting the trade-off parameter between noise reduction and speech distortion larger than 0 in the SDR-GSC improves the performance in the presence of signal model errors, i.e. the speech distortion decreases and the SNR degradation due to the signal model errors becomes smaller. Moreover, in the SP-SDW-MWF both the SNR improvement and the speech distortion are hardly affected by signal model errors. Experimental results also show that using a block-diagonal step size matrix gives rise to a faster convergence than using a diagonal step size matrix and that the SP-SDW-MWF is able to track changing noise scenarios.

References

1. B. D. Van Veen and K. M. Buckley, “Beamforming: a versatile approach to spatial filtering,” *IEEE ASSP Magazine*, vol. 5, pp. 4–24, Apr. 1988.
2. O. L. Frost III, “An algorithm for linearly constrained adaptive array processing,” *Proc. of the IEEE*, vol. 60, pp. 926–935, Aug. 1972.
3. L. J. Griffiths and C. W. Jim, “An alternative approach to linearly constrained adaptive beamforming,” *IEEE Trans. Antennas Propagat.*, vol. 30, pp. 27–34, Jan. 1982.
4. K. M. Buckley, “Broad-band beamforming and the generalized sidelobe canceller,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, pp. 1322–1323, Oct. 1986.
5. S. Nordholm, I. Claesson, and B. Bengtsson, “Adaptive array noise suppression of handsfree speaker input in cars,” *IEEE Trans. Veh. Technol.*, vol. 42, pp. 514–518, Nov. 1993.
6. J. E. Greenberg and P. M. Zurek, “Evaluation of an adaptive beamforming method for hearing aids,” *Journal of the Acoustical Society of America*, vol. 91, pp. 1662–1676, Mar. 1992.
7. J. Vanden Berghe and J. Wouters, “An adaptive noise canceller for hearing aids using two nearby microphones,” *Journal of the Acoustical Society of America*, vol. 103, pp. 3621–3626, June 1998.
8. J.-B. Maj, J. Wouters, and M. Moonen, “Noise reduction results of an adaptive filtering technique for dual-microphone behind-the-ear hearing aids,” *Ear and Hearing*, vol. 25, pp. 215–229, June 2004.
9. J. Bitzer, K. U. Simmer, and K.-D. Kammeyer, “Multi-microphone noise reduction techniques as front-end devices for speech recognition,” *Speech Communication*, vol. 34, pp. 3–12, Apr. 2001.
10. W. Herbordt, H. Buchner, and W. Kellermann, “An acoustic human-machine front-end for multimedia applications,” *EURASIP Journal on Applied Signal Processing*, vol. 2003, pp. 21–31, Jan. 2003.
11. I. Claesson and S. Nordholm, “A spatial filtering approach to robust adaptive beamforming,” *IEEE Trans. Antennas Propagat.*, vol. 40, pp. 1093–1096, Sept. 1992.
12. S. Nordebo, I. Claesson, and S. Nordholm, “Adaptive beamforming: spatial filter designed blocking matrix,” *IEEE Journal of Oceanic Engineering*, vol. 19, pp. 583–590, Oct. 1994.

13. S. Doclo and M. Moonen, “Design of broadband beamformers robust against gain and phase errors in the microphone array characteristics,” *IEEE Trans. Signal Processing*, vol. 51, pp. 2511–2526, Oct. 2003.
14. D. Van Compernolle, “Switching adaptive filters for enhancing noisy and reverberant speech from microphone array recordings,” in *Proc. IEEE ICASSP*, 1990, pp. 833–836.
15. 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 Processing*, vol. 47, pp. 2677–2684, Oct. 1999.
16. W. Herbordt and W. Kellermann, “Computationally efficient frequency-domain robust generalized sidelobe canceller,” in *Proc. IWAENC*, 2001, pp. 51–54.
17. W. Herbordt and W. Kellermann, “Adaptive beamforming for audio signal acquisition,” chapter 6 in *Adaptive Signal Processing: Applications to Real-World Problems*, J. Benesty and Y. Huang, Eds.), pp. 155–194, Springer-Verlag, 2003.
18. S. Gannot, D. Burshtein, and E. Weinstein, “Signal enhancement using beamforming and non-stationarity with applications to speech,” *IEEE Trans. Signal Processing*, vol. 49, pp. 1614–1626, Aug. 2001.
19. S. Gannot and I. Cohen, “Speech enhancement based on the general transfer function GSC and postfiltering,” *IEEE Trans. Speech and Audio Processing*, vol. 12, pp. 561–571, Nov. 2004.
20. O. Hoshuyama, B. Begasse, and A. Sugiyama, “A new adaptation-mode control based on cross correlation for a robust adaptive microphone array,” *IEICE Trans. Fundamentals*, vol. E84-A, pp. 406–413, Feb. 2001.
21. N. K. Jablon, “Adaptive beamforming with the generalized sidelobe canceller in the presence of array imperfections,” *IEEE Trans. Antennas Propagat.*, vol. 34, pp. 996–1012, Aug. 1986.
22. H. Cox, R. M. Zeskind, and M. M. Owen, “Robust adaptive beamforming,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 35, pp. 1365–1376, Oct. 1987.
23. M. W. Hoffman and K. M. Buckley, “Robust time-domain processing of broadband microphone array data,” *IEEE Trans. Speech and Audio Processing*, vol. 3, pp. 193–203, May 1995.
24. Z. Tian, K. L. Bell, and H. L. Van Trees, “A recursive least squares implementation for LCMP beamforming under quadratic constraint,” *IEEE Trans. Signal Processing*, vol. 49, pp. 1138–1145, June 2001.
25. A. Spriet, M. Moonen, and J. Wouters, “Spatially pre-processed speech distortion weighted multi-channel Wiener filtering for noise reduction,” *Signal Processing*, vol. 84, pp. 2367–2387, Dec. 2004.
26. A. Spriet, M. Moonen, and J. Wouters, “Stochastic gradient implementation of spatially pre-processed multi-channel Wiener filtering for noise reduction in hearing aids,” in *Proc. IEEE ICASSP*, 2004, pp. 57–60.
27. S. Doclo, A. Spriet, and M. Moonen, “Efficient frequency-domain implementation of speech distortion weighted multi-channel Wiener filtering for noise reduction,” in *Proc. EUSIPCO*, 2004, pp. 2007–2010.
28. S. Doclo and M. Moonen, “GSVD-based optimal filtering for multi-microphone speech enhancement,” chapter 6 in *Microphone Arrays: Signal Processing Techniques and Applications*, (M. S. Brandstein and D. B. Ward, Eds.), pp. 111–132, Springer-Verlag, 2001.

29. S. Doclo and M. Moonen, “GSVD-based optimal filtering for single and multi-microphone speech enhancement,” *IEEE Trans. Signal Processing*, vol. 50, pp. 2230–2244, Sept. 2002.
30. A. Spriet, M. Moonen, and J. Wouters, “A multi-channel subband generalized singular value decomposition approach to speech enhancement,” *European Transactions on Telecommunications, special issue on Acoustic Echo and Noise Control*, vol. 13, pp. 149–158, Mar.-Apr. 2002.
31. G. Rombouts and M. Moonen, “QRD-based unconstrained optimal filtering for acoustic noise reduction,” *Signal Processing*, vol. 83, pp. 1889–1904, Sept. 2003.
32. D. A. Florêncio and H. S. Malvar, “Multichannel filtering for optimum noise reduction in microphone arrays,” in *Proc. IEEE ICASSP*, 2001, pp. 197–200.
33. R. Aichner, W. Herbordt, H. Buchner, and W. Kellermann, “Least-squares error beamforming using minimum statistics and multichannel frequency-domain adaptive filtering,” in *Proc. IWAENC*, 2003, pp. 223–226.
34. J. Benesty and D. R. Morgan, “Frequency-domain adaptive filtering revisited, generalization to the multi-channel case, and application to acoustic echo cancellation,” in *Proc. IEEE ICASSP*, 2000, pp. 789–792.
35. J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, “General derivation of frequency-domain adaptive filtering,” chapter 8 in *Advances in Network and Acoustic Echo Cancellation*, pp. 157–176, Springer-Verlag, 2001.
36. H. Buchner, J. Benesty, and W. Kellermann, “Multichannel frequency-domain adaptive filtering with application to multichannel acoustic echo cancellation,” chapter 4 in *Adaptive Signal Processing: Applications to Real-World Problems* (J. Benesty and Y. Huang, Eds.), pp. 95–128, Springer-Verlag, 2003.
37. S. Van Gerven and F. Xie, “A comparative study of speech detection methods,” in *Proc. EUROSPEECH*, 1997, vol. 3, pp. 1095–1098.
38. J. Sohn, N. S. Kim, and W. Sung, “A statistical model-based voice activity detection,” *IEEE Signal Processing Lett.*, vol. 6, pp. 1–3, Jan. 1999.
39. S. G. Tanyer and H. Özer, “Voice activity detection in nonstationary noise,” *IEEE Trans. Speech and Audio Processing*, vol. 8, pp. 478–482, July 2000.
40. W. Herbordt, T. Trini, and W. Kellermann, “Robust spatial estimation of the signal-to-interference ratio for non-stationary mixtures,” in *Proc. IWAENC*, 2003, pp. 247–250.
41. Y. Ephraim and H. L. Van Trees, “A signal subspace approach for speech enhancement,” *IEEE Trans. Speech and Audio Processing*, vol. 3, pp. 251–266, July 1995.
42. J. J. Shynk, “Frequency-domain and multirate adaptive filtering,” *IEEE Signal Processing Magazine*, vol. 9, pp. 14–37, Jan. 1992.
43. S. Nordholm, I. Claesson, and M. Dahl, “Adaptive microphone array employing calibration signals: an analytical evaluation,” *IEEE Trans. Speech and Audio Processing*, vol. 7,pp. 241–252, May 1999.
44. S. Nordholm, I. Claesson, and N. Grbić, “Optimal and adaptive microphone arrays for speech input in automobiles,” chapter 14 in *Microphone Arrays: Signal Processing Techniques and Applications*, (M. S. Brandstein and D. B. Ward, Eds.), pp. 307–330, Springer-Verlag, 2001.
45. H. Q. Dam, S. Nordholm, N. Grbić, and H. H. Dam, “Speech enhancement employing adaptive beamformer with recursively updated soft constraints,” in *Proc. IWAENC*, 2003, pp. 307–310.

46. H. Q. Dam, S. Y. Low, H. H. Dam, and S. Nordholm, “Space constrained beamforming with source PSD updates,” in *Proc. IEEE ICASSP*, 2004, pp. 93–96.
47. A. Spriet, M. Moonen, and J. Wouters, “Robustness analysis of GSVD based optimal filtering and generalized sidelobe canceller for hearing aid applications,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2001, pp. 31–34.
48. M. J. Link and K. M. Buckley, “Prewhitening for intelligibility gain in hearing aids arrays,” *Journal of the Acoustical Society of America*, vol. 93, pp. 2139–2140, Apr. 1993.
49. J. E. Greenberg, P. M. Peterson, and P. M. Zurek, “Intelligibility-weighted measures of speech-to-interference ratio and speech system performance,” *Journal of the Acoustical Society of America*, vol. 94, pp. 3009–3010, Nov. 1993.
50. Acoustical Society of America, *ANSI S3.5-1997 American National Standard Methods for Calculation of the Speech Intelligibility Index*. June 1997.
51. L. B. Jensen, “Hearing aid with adaptive matching of input transducers,” United States Patent, no. 6,741,714, May 25, 2004.
52. M. Nilsson, S. D. Soli, and A. Sullivan, “Development of the hearing in noise test for the measurement of speech reception thresholds in quiet and in noise,” *Journal of the Acoustical Society of America*, vol. 95, pp. 1085–1099, Feb. 1994.

10 Adaptive Microphone Array Employing Spatial Quadratic Soft Constraints and Spectral Shaping

Sven Nordholm[1], Hai Quang Dam[1], Nedelko Grbić[2], and Siow Yong Low[1]

[1] Western Australian Telecommunications Research Institute (WATRI)
Crawley, WA 6009, Australia
E-mail: {sven, damhai, siowyong}@watri.org.au

[2] Blekinge Institute of Technology
Department of Telecommunications and Signal Processing
Ronneby, SE 37225, Sweden
E-mail: nedelko.grbic@bth.se

Abstract. The convenience and the ease of use provided by hands-free operation of speech communication devices mean that speech enhancement schemes are becoming indispensable. In this chapter, two subband adaptive microphone array schemes are presented, which aim to provide good speech enhancement capability in poor signal to noise ratio situations. The basic commonality of the adaptive microphone array schemes is that they approximate the Wiener solution in an adaptive manner as new data comes in. Furthermore, both schemes include a quadratic constraint to prevent the trivial zero solution of the weights and to avoid suppression of the source of interest. The constraint is included to provide robustness against model mismatch and good spatial capture of the target signal. Furthermore, by using a subband structure the processing allows a time-frequency operation for each channel. As such, both schemes utilize the spatial, spectral, and temporal domains in an efficient and concise manner allowing a computational effective processing while maintaining high performance speech enhancement. Evaluations on the same data set, gathered from a car, show that the proposed schemes achieve good noise suppression up to 20 dB while experiencing very low levels of speech distortion.

10.1 Introduction

The comfort and flexibility provided by hands-free communication systems have spurred the integration of hands-free voice interface into everyday essentials such as personal digital assistants (PDAs), mobile phones, speech recognition devices, etc. With such a great demand, speech enhancement with regard to hands-free communications particularly in adverse environments has been an area of intensive research [1], [2], [3], [4], [5], [6]. Numerous speech enhancement schemes have been presented over the years with microphone array based techniques dominating the field. This is because microphone arrays offer the invaluable spatial diversity to spatially extract (or form a beam towards) the source of interest (SOI) [7]. In particular, adaptive microphone arrays are reported to have good interference suppression capability [8], [9], [10]. However, adaptive microphone array such as the generalized sidelobe canceller (GSC) succumbs to target signal cancellation in the presence of steering vector errors (e.g. microphone positions, reverberation, etc) [11], [12]. One solution to overcome the problem is to employ a voice activity detector (VAD) or an energy detector in which the GSC is only adapted when there is no target signal (or the signal-to-interference ratio (SIR) is low). Another more straightforward approach to address the problem is to calibrate the microphone array in the actual environment [13], [14]. By doing so, all the information on the array geometry and imperfections will be reflected in the final solution. This approach seems efficient and robust at first glance, but the need for calibration makes its use rather limited in consumer applications. For instance, when the SOI spatially moves or the environment changes, it requires a re-calibration. Such inflexibility may not be practically viable.

In this chapter, we will present two subband based schemes, namely robust soft constrained adaptive microphone array (RSCAMA) and noise statistics updated adaptive microphone array (NSUAMA). Both schemes have their roots in the calibrated microphone array [13], [14] but circumvent the calibration phase which makes them considerably more versatile. Instead, a source model is carefully embedded in the solution whereas the noise statistics is estimated on-line. To complement the source model, the SOI power spectral density (PSD) is also estimated from the data to preserve the spectral shape of the SOI. The real objective is to achieve a solution that is close to the optimal Wiener solution [15], [16] whilst incorporating a tracking capability to handle non-stationary noise. Both structures differ in the way the SOI power spectral density and noise statistics are incorporated in the solution but share the commonality of having a 2-D space constrained source model. Unlike a point source model, the 2-D space model (the physical area of the SOI e.g. a person's mouth) effectively compensates for the large radial vector errors in the source location caused by the presence of erroneous steering vector in real life situations, making both proposed structures robust against errors.

The RSCAMA scheme is constrained to extract the SOI in a pre-defined area (as modelled by the 2-D space constraints). Basically, the idea is originally derived from [4] with the assumption that the power spectral density (PSD) of the source is constant over time and frequency range. However, speech signal is short-term stationary and this implies that the spectrum varies over time. Therefore, to better utilize the time-frequency information of the SOI, its PSD is recursively updated in the constraints using the most current time-frequency content of the output signal from the beamformer. The motivation behind the use of the output signal in the update comes from the fact that the optimum beamformer output in each subband, is an enhanced version of the spectral information of the SOI. In other words, the feedback from the beamformer output continuously shapes the SOI spectrum, thus providing a spectrally improved constraint at each time instant. The noise statistics on the other hand, are estimated recursively from the received data without the need of a VAD as the solution has been constrained to preserve any signal from the desired region. Needless to say, the performance will be very much improved if a VAD is used, however at the expense of a higher computational complexity.

As the name suggests, the NSUAMA scheme includes a noise statistics update to track variations in the background noise. Simply, the adaptive microphone array estimates the covariance information and decides if the estimated information can be used to update the noise statistics in the solution i.e. the noise covariance information. A modified VAD or a noise covariance detector which includes spatial information is introduced to ensure that “only noise covariance” information is used in the update. With the incorporation of the criterion, the microphone array behaves like a “noise only” detector which uses only the noise information to update its solution. This results in an efficient and fast converging adaptive microphone array even in highly non-stationary environment. Similar to RSCAMA, the source PSD is embedded in the optimum Wiener solution in each subband to fully utilize the time-frequency information of the target signal. However unlike RSCAMA, the source PSD is updated using a least-squares criterion [17]. As such, it tracks the variation in the spectral content of the target signal continuously, yielding a statistically optimized constraint for each time instant.

Clearly, the major difference between the RSCAMA and NSUAMA schemes is their computational complexities. The RSCAMA structure offers simplicity and is straightforward to implement in real-time. Naturally, the downside of it is less suppression capability when compared to the NSUAMA scheme. Evaluations in a real car hands-free scenario reveal that the NSUAMA scheme manages to achieve an impressive noise suppression level of 20 dB whilst the simpler RSCAMA performs around 16-17 dB. Most importantly, both schemes maintain negligible distortion on the target signal.

10.2 Signal Modelling and Problem Formulation

Consider a linear microphone array with I microphones. The target signal in this case is a person speaking, which can be modelled as an infinite number of point sources clustered closely in space. This space is modelled as a circular area $\mathbf{A}$ with radius r and a distance h from the array, see Fig. 10.1. Alternatively, the source constrained region can be modelled as a pie sliced area defined by radii $[R_a, R_b]$ and angles $[\theta_a, \theta_b]$ [8]. As mentioned previously, the advantage of the source constrained region in Fig. 10.1 as opposed to a point source is consistent with the fact that errors in the response vector cause large radial errors in the corresponding source location [11]. These errors are typically due to sensor misplacement and gain variations in the microphones. With the inclusion of the constrained area, the structure is made more robust

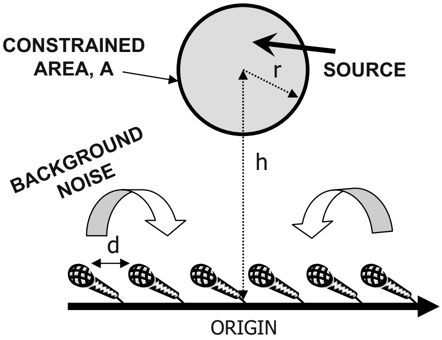


Fig. 10.1. Configuration of the linear microphone array with the inter-element distance d and the source constrained area defined by radius r and distance h.

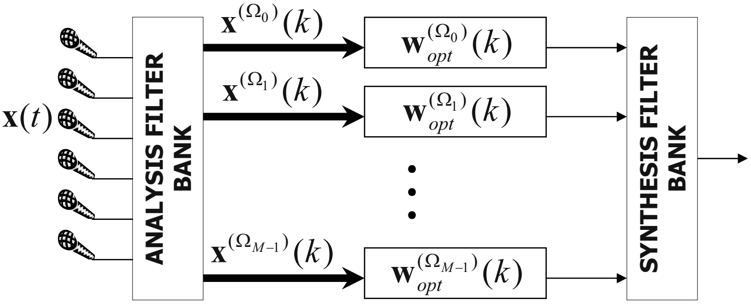

$\mathbf{x}(t)$
ANALYSIS FILTER BANK
$\mathbf{x}^{(\Omega_0)}(k)$
$\mathbf{x}^{(\Omega_1)}(k)$
$\mathbf{x}^{(\Omega_{M-1})}(k)$
$\mathbf{w}_{opt}^{(\Omega_0)}(k)$
$\mathbf{w}_{opt}^{(\Omega_1)}(k)$
$\mathbf{w}_{opt}^{(\Omega_{M-1})}(k)$
SYNTHESIS FILTER BANK

Fig. 10.2. Structure of the RSCAMA subband beamformer.

and more closely related to a real situation. Throughout the chapter, the SOI is assumed to be in the constrained region as shown in Fig. 10.1.

Figures 10.2 and 10.3 show the block diagrams of the proposed RSCAMA and NSUAMA respectively. Irrespective of the different structures, both the subband based RSCAMA and NSUAMA aim to extract the SOI in the constrained region. From the figures, the received signal is initially decomposed into M subbands by using an analysis filterbank. After the relevant processing independently (in each structure), the processed subband signals are then reconstructed by the synthesis filterbank into fullband representation.

10.2.1 Analysis and Synthesis Filterbanks

The main consideration in the design is to minimize aliasing in the subband signals as well as minimizing magnitude, phase and aliasing distortion in the reconstructed output. Literature associated with filterbanks can be found in the following references [18], [19]. In this work, an oversampled uniform analysis DFT filterbank is employed to decompose each of the I microphone input signals into M subbands with an oversampling decimation factor of

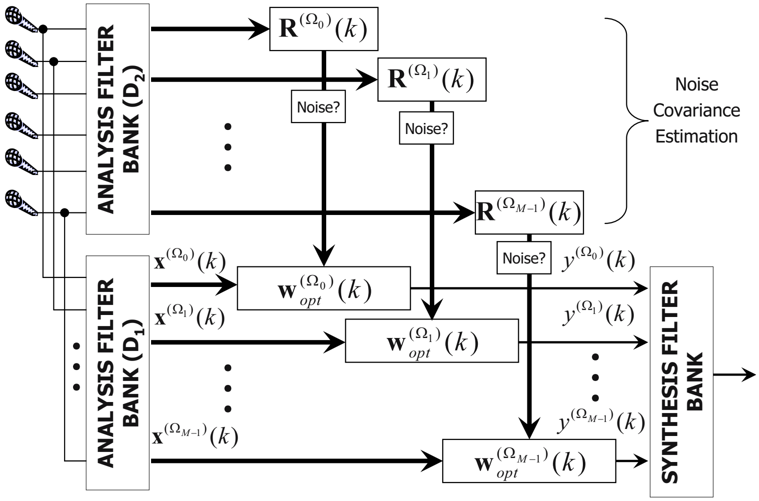


Fig. 10.3. Structure of the NSUAMA subband beamformer. D_1 and D_2 are the different decimation factors with $D_1 \geq D_2$.

$D_1 = M/2$ unless otherwise stated. By oversampling, the inband aliasing effects is greatly reduced. The analysis and synthesis prototype filters are designed using a Hamming window with a cut off frequency π/M. The Hamming window has side-lobes that are 50 dB below its mainlobe and by using a factor two over-sampling, the overall distortion and aliasing will be kept small. Note that the noise covariance estimation in Fig. 10.3 has its own requirement, as it is decimated at a lower rate of D_2, where $D_1 > D_2$. This is to ensure the sufficiency of data in estimating the noise covariance matrix (i.e. to achieve low variance estimate for a better tracking in the noise statistics).

10.2.2 The Wiener Solution

In this section, the multichannel Wiener filter in each subband is formulated. To begin, let $\mathbf{w}_{opt}^{(\Omega)}(k)$ be the optimum weight vector at time index k for each frequency $\Omega \in [\Omega_0, \cdots, \Omega_{M-1}]$ as

$$\mathbf{w}_{opt}^{(\Omega)}(k) = [w_1^{(\Omega)}(k), w_2^{(\Omega)}(k), \cdots, w_I^{(\Omega)}(k)]^T. \tag{10.1}$$

The optimal weight vector at time point k above can be readily found from the Wiener solution as follows,

$$\mathbf{w}_{opt}^{(\Omega)}(k) = [\mathbf{R}_s^{(\Omega)}(k) + \mathbf{R}_n^{(\Omega)}(k)]^{-1}\mathbf{r}_s^{(\Omega)}(k), \tag{10.2}$$

where $\mathbf{R}_s^{(\Omega)}(k)$ and $\mathbf{r}_s^{(\Omega)}(k)$ are the covariance matrix and the cross-covariance vector for the SOI for frequency band Ω, respectively. The covariance matrix

$\mathbf{R}_s^{(\Omega)}(k)$ can be resolved into a normalized spatial covariance matrix $\bar{\mathbf{R}}_s^{(\Omega)}(k)$ and a non-negative spectral weighting as

$$\mathbf{R}_s^{(\Omega)}(k) = S^{(\Omega)}(k)\bar{\mathbf{R}}_s^{(\Omega)}. \tag{10.3}$$

Likewise, the cross-covariance vector $\mathbf{r}_s^{(\Omega)}$ can be decomposed as

$$\mathbf{r}_s^{(\Omega)}(k) = S^{(\Omega)}(k)\bar{\mathbf{r}}_s^{(\Omega)}, \tag{10.4}$$

where $\bar{\mathbf{r}}_s^{(\Omega)}$ is the normalized spatial cross-covariance vector. Substituting (10.3) and (10.4) into (10.2) yields,

$$\mathbf{w}_{opt}^{(\Omega)}(k) = [S^{(\Omega)}(k)\bar{\mathbf{R}}_s^{(\Omega)} + \mathbf{R}_n^{(\Omega)}(k)]^{-1}S^{(\Omega)}(k)\bar{\mathbf{r}}_s^{(\Omega)}. \tag{10.5}$$

Equation (10.5) forms the basis for the development of both microphone array schemes RSCAMA and NSUAMA. From (10.5), it is clear that there are two varying parameters that need to be estimated continuously i.e. $S^{\Omega}(k)$ and $\mathbf{R}_n^{(\Omega)}(k)$. The former functions as the source spectral moulder (to reduce spectral distortion) and the latter is to track the noise statistics for optimal noise suppression. The SOI spatial covariance matrix $\bar{\mathbf{R}}_s^{(\Omega)}$ and the spatial cross-covariance vector $\bar{\mathbf{r}}_s^{(\Omega)}$ on the other hand, are determined by the spatial location of the SOI. For many applications such as internet telephony, hands-free mobile telephony, etc, the SOI is typically located more or less in a fixed position (in front of the array). In keeping with this, the SOI is assumed to be spatially stationary in a pre-defined region and a constraint called the space constraint is used to model it. Both the RSCMA and NSUAMA schemes employ the space constraint to model the SOI spatial information. In the following section, the space constraint is explained.

10.2.3 The Space Constrained Source Covariance Information

Let us denote $S^{(\Omega)}$ as the PSD of the source at frequency Ω. Note that the PSD will be time varying and can be thought of as short-term stationary. However, for the following model it is kept constant. As mentioned previously, the source is assumed to be in the pre-defined area $\mathbf{A}$ afore-mentioned (see Fig. 10.1). Thus, the spatio-temporal covariance matrix of source in the spectral band $[\Omega_a, \Omega_b]$ can be computed as

$$\mathbf{R}_s = \int_{\Omega_a}^{\Omega_b} \iint\limits_{\mathbf{A}} S^{(\Omega)}\mathbf{d}^{(\Omega)}(\overrightarrow{\mathbf{a}})(\mathbf{d}^{(\Omega)}(\overrightarrow{\mathbf{a}}))^H d\overrightarrow{\mathbf{a}}d\Omega, \tag{10.6}$$

where $\overrightarrow{\mathbf{a}}$ is the point source localization vector and $(\cdot)^H$ denotes the Hermitian transposition. The response vector $\mathbf{d}^{(\Omega)}(\overrightarrow{\mathbf{a}})$ is defined as

$$\mathbf{d}^{(\Omega)}(\overrightarrow{\mathbf{a}}) = \left[\frac{1}{R_1}e^{-j\Omega\tau_1(\overrightarrow{\mathbf{a}})}, \frac{1}{R_2}e^{-j\Omega\tau_2(\overrightarrow{\mathbf{a}})}, \cdots, \frac{1}{R_I}e^{-j\Omega\tau_I(\overrightarrow{\mathbf{a}})}\right]^T, \tag{10.7}$$

where $\tau_i(\overrightarrow{\mathbf{a}})$, $1 \leq i \leq I$ is the time delay from a point source in the predefined area to sensor i, R_i is the distance between the source and sensor i and $[\cdot]^T$ denotes the transposition operator. The reference point for the microphone array response is defined at the origin of the coordinates.

Therefore for a frequency Ω, the spatial covariance matrix in (10.3) is,

$$\mathbf{R}_s^{(\Omega)} = S^{(\Omega)} \bar{\mathbf{R}}_s^{(\Omega)}, \tag{10.8}$$

where the normalized spatial covariance matrix is defined from (10.6) as,

$$\bar{\mathbf{R}}_s^{(\Omega)} = \iint_{\mathbf{A}} \mathbf{d}^{(\Omega)}(\overrightarrow{\mathbf{a}}) \mathbf{d}^{(\Omega)}(\overrightarrow{\mathbf{a}})^H d\overrightarrow{\mathbf{a}}. \tag{10.9}$$

The spatial cross covariance vector is given by

$$\mathbf{r}_s^{(\Omega)} = S^{(\Omega)} \bar{\mathbf{r}}_s^{(\Omega)}, \tag{10.10}$$

where

$$\bar{\mathbf{r}}_s^{(\Omega)} = \iint_{\mathbf{A}} \mathbf{d}^{(\Omega)}(\overrightarrow{\mathbf{a}}) d\overrightarrow{\mathbf{a}}. \tag{10.11}$$

With the space constrained model readily available, the task at hand is to efficiently estimate the varying parameters $S^{(\Omega)}(k)$ and $\mathbf{R}_n^{(\Omega)}(k)$ in (10.5). This is where the distinction between the RSCAMA and NSUAMA structures comes in. The simpler RSCAMA estimates the information directly irrespective of whether the SOI is active or inactive whereas the NSUAMA performs otherwise. Sections 10.3 and 10.4 explain both the RSCAMA and NSUAMA structures in detail.

10.3 Robust Soft Constrained Adaptive Microphone Array (RSCAMA)

10.3.1 Problem Formulation

Let $\mathbf{w}_{opt}^{(\Omega)}$ be the optimum weight vector for frequency Ω,

$$\mathbf{w}_{opt}^{(\Omega)} = [w_1^{(\Omega)}, w_2^{(\Omega)}, \ldots, w_I^{(\Omega)}]^T, \tag{10.12}$$

where $w_i^{(\Omega)}$ is the optimum coefficient for the i^{th} sensor. The optimum weight vector is then calculated as

$$\mathbf{w}_{opt}^{(\Omega)} = \left[\mathbf{R}_s^{(\Omega)} + \mathbf{R}_n^{(\Omega)}\right]^{-1} \mathbf{r}_s^{(\Omega)}, \tag{10.13}$$

where $\mathbf{R}_n^{(\Omega)}$ is the noise covariance matrix. Suppose that we have knowledge of the PSD of the SOI $S^{(\Omega)}$, then (10.13) can be rewritten as

$$\begin{aligned}\mathbf{w}_{opt}^{(\Omega)} &= \left[\mathbf{R}_s^{(\Omega)}/S(\Omega) + \mathbf{R}_n^{(\Omega)}/S(\Omega)\right]^{-1} \left(\mathbf{r}_s^{(\Omega)}/S(\Omega)\right) \\ &= \left[\bar{\mathbf{R}}_s^{(\Omega)} + \bar{\mathbf{R}}_n^{(\Omega)}\right]^{-1} \bar{\mathbf{r}}_s^{(\Omega)},\end{aligned} \tag{10.14}$$

where $\bar{\mathbf{R}}_s^{(\Omega)}$ is the normalized spatial covariance matrix given in (10.9) and $\bar{\mathbf{r}}_s^{(\Omega)}$ is the normalized spatial cross covariance vector as defined in (10.11). The implication of (10.14) is that the SOI PSD $S^{(\Omega)}$ is incorporated in the solution and both $\bar{\mathbf{R}}_s^{(\Omega)}$ and $\bar{\mathbf{r}}_s^{(\Omega)}$ can be calculated for a given constraint region without the knowledge of the PSD of the source, given that they are spatially invariant.

The remaining issue is to recursively estimate the noise parameters $\bar{\mathbf{R}}_n^{(\Omega)}$. Since data containing only the active noise is not available, the noise covariance matrix $\bar{\mathbf{R}}_n^{(\Omega)}$ is estimated by using K samples of the received data $\mathbf{x}^{(\Omega)}(k)$, where K is a fixed positive number and the index k is the subband time index. Moreover, the exact PSD of the source $S^{(\Omega)}(k)$ is not available, particularly in a car environment where strong speech masking components of noise exists. Thus, we propose to use the previous microphone array outputs for the estimation of $S^{(\Omega)}(k)$ as

$$\mathbf{z}^{(\Omega)}(k) = \frac{\mathbf{x}^{(\Omega)}(k)}{|\mathbf{w}_{opt}^{(\Omega)}(k-1)^H \mathbf{x}^{(\Omega)}(k-1)| + \delta}, \tag{10.15}$$

where $|.|$ is the absolute value operator and δ is a positive number to avoid zero division. At iteration k, $\bar{\mathbf{R}}_n^{(\Omega)}(k)$ can be estimated based on $\mathbf{z}^{(\Omega)}(m)$ where $\max(0, k-K) \leq m \leq k$ as follows,

- if $k \leq K$ then

$$\bar{\mathbf{R}}_n^{(\Omega)}(k) = \frac{1}{k} \sum_{m=1}^{k} \mathbf{z}^{(\Omega)}(m) \mathbf{z}^{(\Omega)}(m)^H, \tag{10.16}$$

- if $k > K$ then

$$\bar{\mathbf{R}}_n^{(\Omega)}(k) = \frac{1}{K} \sum_{m=k-K+1}^{k} \mathbf{z}^{(\Omega)}(m) \mathbf{z}^{(\Omega)}(m)^H. \tag{10.17}$$

In the next section, a recursive algorithm is developed to efficiently update the beamforming weights according to (10.16) and (10.17) based on the received data.

10.3.2 A Recursive Algorithm for the RSCAMA

The algorithm runs in parallel/sequentially for each subband with mid-frequency $\Omega = 2\pi f_s m/M$, $0 \leq m \leq M-1$, where f_s is the sampling frequency. Let

$$\bar{\mathbf{R}}^{(\Omega)}(k) = \bar{\mathbf{R}}_s^{(\Omega)} + \bar{\mathbf{R}}_n^{(\Omega)}(k) \tag{10.18}$$

and

$$\mathbf{P}^{(\Omega)}(k) = [\bar{\mathbf{R}}^{(\Omega)}(k)]^{-1}. \tag{10.19}$$

The optimal weight vector for the iteration k is then reduced to

$$\mathbf{w}_{opt}^{(\Omega)}(k) = \mathbf{P}^{(\Omega)}(k)\bar{\mathbf{r}}_s^{(\Omega)}. \tag{10.20}$$

It follows from (10.17) that for $k > K$, $\bar{\mathbf{R}}^{(\Omega)}(k)$ can be obtained from the previous estimate as

$$\bar{\mathbf{R}}^{(\Omega)}(k) = \bar{\mathbf{R}}^{(\Omega)}(k-1) + \frac{1}{K}\mathbf{z}^{(\Omega)}(k)\mathbf{z}^{(\Omega)}(k)^H - \frac{1}{K}\mathbf{z}^{(\Omega)}(k-K)\mathbf{z}^{(\Omega)}(k-K)^H. \tag{10.21}$$

Thus, the inverse matrix $\mathbf{P}^{(\Omega)}(k)$ for $k > K$ can be updated efficiently by using the matrix inversion lemma

$$\mathbf{D} = \mathbf{P}^{(\Omega)}(k-1) - \frac{\mathbf{P}^{(\Omega)}(k-1)\mathbf{z}^{(\Omega)}(k)\mathbf{z}^{(\Omega)}(k)^H\mathbf{P}^{(\Omega)}(k-1)}{\left(K + \mathbf{z}^{(\Omega)}(k)^H\mathbf{P}^{(\Omega)}(k-1)\mathbf{z}^{(\Omega)}(k)\right)} \tag{10.22}$$

and

$$\mathbf{P}^{(\Omega)}(k) = \mathbf{D} + \frac{\mathbf{D}\mathbf{z}^{(\Omega)}(k-K)\mathbf{z}^{(\Omega)}(k-K)^H\mathbf{D}}{\left(K - \mathbf{z}^{(\Omega)}(k-K)^H\mathbf{D}\mathbf{z}^{(\Omega)}(k-K)\right)}, \tag{10.23}$$

where $\mathbf{D}$ in this case is an intermediate matrix of the same size as $\mathbf{P}^{(\Omega)}(k)$. The recursive algorithm is now summarized in the following steps,

- *Step 1: Choose a number of subbands M, a block size K and a weight smoothing factor λ*[1]*.*
- *Step 2: Initialize $k = 1$ and the weight vector $\mathbf{w}_{opt}^{(\Omega)}(0)$ as an $I \times 1$ zero vector.*
- *Step 3: Calculate the matrix $\bar{\mathbf{R}}_s^{(\Omega)}$ and the vector $\bar{\mathbf{r}}_s^{(\Omega)}$ according to (10.9) and (10.11), respectively.*

[1] The factor λ is employed because the target speech signal adds spatial coherent power to the pre-calculated covariance matrix, and this in turn leads to small weight power fluctuations.

- *Step 4: If $k \leq K$, the matrix $\mathbf{P}^{(\Omega)}(k)$ is calculated according to (10.15), (10.16) and (10.19) by using pseudo-inverse operation instead of the conventional matrix inverse operation due to rank deficiency. Otherwise, the matrix $\mathbf{P}^{(\Omega)}(k)$ is updated recursively by using (10.22) and (10.23). The weight vector is then updated as*

$$\mathbf{w}_{opt}^{(\Omega)}(k) = \lambda \mathbf{w}_{opt}^{(\Omega)}(k-1) + (1-\lambda)\mathbf{P}^{(\Omega)}(k)\bar{\mathbf{r}}_s^{(\Omega)},$$

and the output is given by

$$y^{(\Omega)}(k) = \mathbf{w}_{opt}^{(\Omega)}(k)^H \mathbf{x}^{(\Omega)}(k).$$

- *Step 5: Set $k = k+1$ and return to Step 4 until the end of the data.*

10.4 Noise Statistics Updated Adaptive Microphone Array (NSUAMA)

10.4.1 Problem Formulation

In the formulation of the RSCAMA scheme, the update of the noise covariance matrix estimate $\bar{\mathbf{R}}_n^{(\Omega)}(k)$ is performed continuously sample by sample. This means that the covariance information also contains the SOI. Naturally, if the noise covariance estimation is free from the SOI, the advantages will be twofold i.e. better noise suppression and consequently better source PSD estimate. Here, the NSUAMA scheme employs a “noise covariance detector” to avoid the inclusion of the SOI in the noise covariance matrix.

In order to explain this formulation, we consider the Wiener solution [eq. (10.5)] again

$$\mathbf{w}_{opt}^{(\Omega)}(k) = [S^{(\Omega)}(k)\bar{\mathbf{R}}_s^{(\Omega)} + \mathbf{R}_n^{(\Omega)}(k)]^{-1} S^{(\Omega)}(k)\bar{\mathbf{r}}_s^{(\Omega)}. \tag{10.24}$$

As before, both the $\bar{\mathbf{R}}_s^{(\Omega)}$ and $\bar{\mathbf{r}}_s^{(\Omega)}$ can be precalculated according to (10.9) and (10.11) respectively as long as the SOI is spatially invariant. Similar to the RSCAMA scheme, the objective is to calculate the Wiener solution above by efficiently estimating the power spectrum of the SOI $S^{(\Omega)}(k)$ and the noise covariance matrix $\mathbf{R}_n^{(\Omega)}(k)$.

10.4.2 The Noise Covariance Detector

From the pre-defined source area model $\mathbf{A}$, the matrix $\mathbf{R}_s^{(\Omega)}$ in (10.9) for frequency Ω has non-zero determinant and is therefore a full rank matrix[2]. Thus, this matrix can be decomposed as follows,

$$\bar{\mathbf{R}}_s^{(\Omega)} = \mathbf{V}^{(\Omega)}\mathbf{\Lambda}^{(\Omega)}\mathbf{V}^{(\Omega)H}, \tag{10.25}$$

[2] Depending on how much of the space it spans, it will have a few dominating eigenvalues.

where

$$\mathbf{V}^{(\Omega)} = [\mathbf{v}_1^{(\Omega)}, \cdots, \mathbf{v}_I^{(\Omega)}] \tag{10.26}$$

is a matrix that contains the eigenvectors and

$$\mathbf{\Lambda}^{(\Omega)} = \mathrm{diag}\{\lambda_1^{(\Omega)}, \cdots, \lambda_I^{(\Omega)}\} \tag{10.27}$$

is a diagonal matrix that consists of the eigenvalues. Since the SOI and noise are assumed to be uncorrelated and by using the proposed source covariance model, the total covariance matrix can be written as

$$\mathbf{R}^{\Omega}(k) = S^{\Omega}(k)\bar{\mathbf{R}}_s^{\Omega} + \mathbf{R}_n^{\Omega}(k), \tag{10.28}$$

where the total covariance matrix $\mathbf{R}^{\Omega}(k)$ can be calculated from the received signal $\mathbf{x}^{(\Omega)}(k)$ by K of its samples as follows

$$\mathbf{R}^{(\Omega)}(k) = \frac{1}{K} \sum_{m=k-K+1}^{k} \mathbf{x}^{(\Omega)}(m)\mathbf{x}^{(\Omega)}(m)^H. \tag{10.29}$$

By multiplying the left and right side of (10.28) with the eigenvector $\mathbf{v}_{max}^{(\Omega)}$ that corresponds to the largest eigenvalue of $\bar{\mathbf{R}}_s^{(\Omega)}$, $\lambda_{max}^{(\Omega)}$, we have the following equation

$$\begin{aligned} &{\mathbf{v}_{max}^{(\Omega)}}^H \mathbf{R}^{(\Omega)}(k)\mathbf{v}_{max}^{(\Omega)} = \\ &S^{(\Omega)}(k)\ {\mathbf{v}_{max}^{(\Omega)}}^H \bar{\mathbf{R}}_s^{(\Omega)}\mathbf{v}_{max}^{(\Omega)} + {\mathbf{v}_{max}^{(\Omega)}}^H \mathbf{R}_n^{(\Omega)}(k)\mathbf{v}_{max}^{(\Omega)}. \end{aligned} \tag{10.30}$$

The right hand side of (10.30) can be simplified to

$$\begin{aligned} &{\mathbf{v}_{max}^{(\Omega)}}^H \mathbf{R}^{(\Omega)}(k)\mathbf{v}_{max}^{(\Omega)} = \\ &S^{(\Omega)}(k)\ \lambda_{max}^{(\Omega)} + {\mathbf{v}_{max}^{(\Omega)}}^H \mathbf{R}_n^{(\Omega)}(k)\mathbf{v}_{max}^{(\Omega)}. \end{aligned} \tag{10.31}$$

The purpose of using $\mathbf{v}_{max}^{(\Omega)}$ is consistent with the fact that it represents the strongest component in the target signal subspace. By denoting

$$F^{(\Omega)}(k) = {\mathbf{v}_{max}^{(\Omega)}}^H \mathbf{R}^{(\Omega)}(k)\mathbf{v}_{max}^{(\Omega)}, \tag{10.32}$$

(10.31) can be rewritten as,

$$F^{(\Omega)}(k) = S^{(\Omega)}(k)\lambda_{max}^{(\Omega)} + {\mathbf{v}_{max}^{(\Omega)}}^H \mathbf{R}_n^{(\Omega)}(k)\mathbf{v}_{max}^{(\Omega)}. \tag{10.33}$$

In the following, we will propose a criterion for the case when the noise is assumed to be long-term stationary (such as in a car or helicopter environment) whereas the speech signal is short-term stationary. This means that the statistics of the noise remain unchanged for at least 1 second. Using this assumption, there exists a number of sample points $L >> K$ which corresponds to 1 second in time, where the noise is stationary during the interval $[k - L, k]$. As such, if the SOI is silent, then the number of K sample points in (10.29) will be sufficient to capture the noise statistics for that particular period.

It follows from (10.33) that when there is no SOI at sample instant k, the first term $S^{(\Omega)}(k)\lambda_{max}^{(\Omega)}$ will be approximately zero. The second term ${\mathbf{v}_{max}^{(\Omega)}}^H \mathbf{R}_n^{(\Omega)}(k)\mathbf{v}_{max}^{(\Omega)}$ will reduce to a minimum value for $F^{(\Omega)}(k)$ during $[k - L, k]$ period due to the stationarity of the noise in that time frame. This term essentially represents the lower bound for the function in (10.33). Naturally, if speech (i.e SOI is active) is present, then the value in $F^{(\Omega)}(k)$ will be higher than its lower bound. Strictly speaking, it is a function that contains information on the periods of "speech-silence" in the constrained area. Having said so, a criterion can be formulated as follows,

$$F^{(\Omega)}(k) - \min_{[k-L,k]} F^{(\Omega)} < \lambda_{max}^{(\Omega)}\varepsilon, \forall \Omega, \tag{10.34}$$

where $\min_{[k-L,k]}(F^{(\Omega)})$ denotes the minimum of $F^{(\Omega)}$ over the period[3] $[k - L, k]$. The parameter ε in this case is the threshold in the detector. If the criterion in (10.34) is met for all frequency bands, then the covariance matrix $\mathbf{R}^{(\Omega)}(k)$ is used to update the estimated noise covariance matrix $\mathbf{R}_n^{(\Omega)}(k)$, through a first order smoothing function given by,

$$\mathbf{R}_n^{(\Omega)}(k) = (1 - \lambda)\mathbf{R}_n^{(\Omega)}(k - 1) + \lambda\mathbf{R}^{(\Omega)}(k). \tag{10.35}$$

The constant λ in this case is the smoothing factor. If the condition in (10.34) is not met, then

$$\mathbf{R}_n^{(\Omega)}(k) = \mathbf{R}_n^{(\Omega)}(k - 1). \tag{10.36}$$

Since the speech signal has most of its energy in the frequency range 500 Hz to 2000 Hz, the criterion in (10.34) can be performed only in the speech dominant subbands. In other words, the detector can be implemented in the frequency range where the speech energy mainly concentrates.

10.4.3 Estimation of Power Spectrum of SOI

The SOI PSD can be estimated by using the least-squares approach given as

$$S^{(\Omega)}(k) = \arg \min_{S^{(\Omega)}, S^{(\Omega)} > 0} \parallel \mathbf{R}^{(\Omega)}(k) - \mathbf{R}_n^{(\Omega)}(k) - S^{(\Omega)}\bar{\mathbf{R}}_s^{(\Omega)} \parallel_{\mathcal{F}}^2, \tag{10.37}$$

[3] This period is the interval in which the noise statistics remains unchanged. Since the noise considered is long-term stationary, a suitable duration will be around one second.

where $\| \cdot \|_{\mathcal{F}}$ is the Frobenius norm operator. For ease of computation, (10.37) can be efficiently solved by stacking the columns of each matrix to form a I^2 long vector. This problem can then be reduced to a quadratic optimization problem with I^2 variables. By setting the first derivative of (10.37) to zero, the optimum $S^{(\Omega)}(k)$ can be obtained. This PSD is estimated at every iteration of the received signal covariance matrix to provide a spectrally optimized constraint on the source. In simple terms, it attempts to preserve the spectrum of the source like a spectrum moulder.

10.4.4 The NSUAMA Algorithm

For simplicity, the noise covariance matrix can be updated in the algorithm only every one second due to the assumption on the long-term stationarity for the noise (otherwise, the noise covariance matrix can be re-evaluated by (10.35) in every iteration). Equations (10.24) and (10.37) can be reformulated as follows

$$\mathbf{w}_{opt}^{(\Omega)}(k) = [S^{(\Omega)}(k)\bar{\mathbf{R}}_s^{(\Omega)} + \mathbf{R}_n^{(\Omega)}]^{-1} S^{(\Omega)}(k)\bar{\mathbf{r}}_s^{(\Omega)} \tag{10.38}$$

and

$$S^{(\Omega)}(k) = \\ \arg \min_{S^{(\Omega)}, S^{(\Omega)}>0} \| \mathbf{R}^{(\Omega)}(k) - \mathbf{R}_n^{(\Omega)} - S^{(\Omega)}\bar{\mathbf{R}}_s^{(\Omega)} \|_{\mathcal{F}}^2, \tag{10.39}$$

where $\mathbf{R}_n^{(\Omega)}$ is the most current evaluated noise covariance matrix from the noise detector during this time. The NSUAMA algorithm can be summarized in the following steps.

- *Step 1: Choose the number of subbands M, decimation factors D_1 and D_2 (in our algorithm $D_1 = M/2$ and $D_2 = M/4$), a block size K, a length of noise evaluation period L and a weight smoothing factor λ.*
- *Step 2: Initialize $k = 1$, the weight vector $\mathbf{w}_{opt}^{(\Omega)}(0)$ as an $I \times 1$ zero vector, the noise covariance matrix $\mathbf{R}_n^{(\Omega)}$ to an $I \times I$ identity matrix.*
- *Step 3: Calculate the matrix $\bar{\mathbf{R}}_s^{(\Omega)}$ and the vector $\bar{\mathbf{r}}_s^{(\Omega)}$ according to (10.9) and (10.11), respectively and the eigenvector $\mathbf{v}_{max}^{(\Omega)}$ that corresponds to the largest eigenvalue $\lambda_{max}^{(\Omega)}$ of $\bar{\mathbf{R}}_s^{(\Omega)}$.*
- *Step 4: Calculate $\mathbf{x}^{(\Omega)}(k)$ with D_1 decimation factor and $\mathbf{R}^{(\Omega)}(k)$ using the samples with D_2 decimation factor. The SOI PSD $S^{(\Omega)}(k)$ and the weight vector $\mathbf{w}_{opt}^{(\Omega)}(k)$ are calculated by using (10.38) and (10.39). The output is given by*

  $$y^{(\Omega)}(k) = \mathbf{w}_{opt}^{(\Omega)}(k)^H \mathbf{x}^{(\Omega)}(k).$$

- *Step 5: Update $\mathbf{R}_n^{(\Omega)}(k)$ by checking the criterion (10.34) using (10.35) or (10.36). If k is within L, set $\mathbf{R}_n^{(\Omega)} = \mathbf{R}_n^{(\Omega)}(k)$.*
- *Step 6: Set $k = k + 1$ and return to Step 4 until the end of the data.*

10.5 Evaluations

10.5.1 The Simulation Scenario

The performance evaluation of the proposed microphone arrays was made in a real car hands-free situation. A six-sensor array with an inter-element distance of 5 cm was mounted on the visor at the passenger side in a Volvo station wagon. Data were gathered on a multichannel DAT-recorder with a sampling rate of 12 kHz and bandlimited to 300-3400 Hz. The car was moving at the speed of 110 km/h on a paved road.

For all the evaluations, the length of the speech signal (female) was 4 seconds long and the matrix in (10.9) and the vector (10.11) were calculated by using numerical integration according to the constrained region given in Fig. 10.1. Here, the circular constrained area of the SOI was set to be 30 cm from the center of the array with a radius of 10 cm. The only parameter in the RSCAMA structure, the weight smoothing factor λ was chosen to be $\lambda = 0.99$. As for the NSUAMA scheme, the detector threshold was set to $\varepsilon = 0.01$ and both the K and L number of samples were chosen to be 30 ms and 1 s long respectively. The decimation factor for D_1 was made oversampled and set to $M/2$ in order to reduce the aliasing effects between the adjacent subbands. The decimation factor D_2 for the covariance estimation on the other hand, was chosen to be $D_2 = M/4$ to ensure the sufficiency of data.

10.5.2 Results for RSCAMA and NSUAMA Beamformers

Figure 10.4 shows the time-domain plots of the original speech, the noisy speech at the 4^{th} microphone and the microphone array outputs for RSCAMA and NSUAMA beamformers respectively. The SNR is -7 dB and the noise level of the signal at other microphones is approximately the same as the 4^{th} microphone. Clearly, Figs. 10.4(c) and 10.4(d) show that the background noise is suppressed significantly by both beamformers respectively. The plots also suggest good timbre of the output signal as the envelope of the SOI follows that of the original SOI [Fig. 10.4(a)].

To quantify the performance of the beamformers, the following noise suppression measure is defined as,

$$NS = 10\log_{10}\left(\frac{\int_{-\pi}^{\pi}\hat{P}_{in,n}(\omega)d\omega}{\int_{-\pi}^{\pi}\hat{P}_{out,n}(\omega)d\omega}\right) - 10\log_{10}(C_d), \tag{10.40}$$

where $\hat{P}_{in,n}(\omega)$ and $\hat{P}_{out,n}(\omega)$ are the spectral power estimates of the reference sensor observation and the output respectively, when the noise is active alone. The constant C_d normalizes the performance measure such that if the SOI is attenuated by the beamformer, the measure is reduced correspondingly (i.e. normalizes the noise suppression to unity SOI gain). Table 10.1 presents

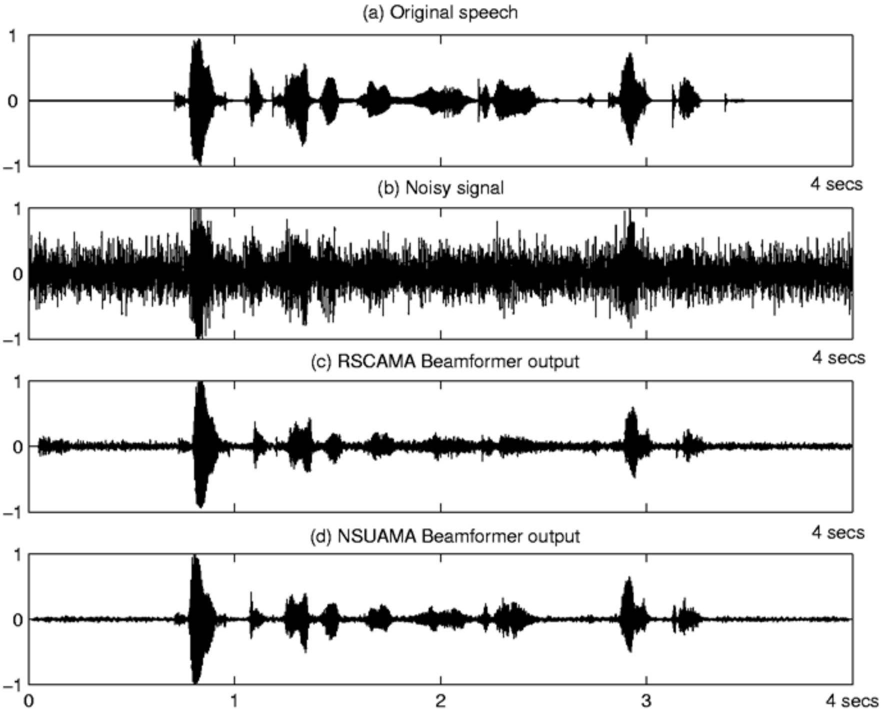


Fig. 10.4. Plots of the RSCAMA and NSUAMA data (a) clean target signal, (b) received signal, (c) RSCAMA beamformer output, and (d) NSUAMA beamformer output.

Table 10.1 Noise suppression (NS) for the RSCAMA and NSUAMA beamformers with different number of subbands.

Subbands M	NS for RSCAMA (dB)	NS for NSUAMA (dB)
16	12.8	15.7
32	14.1	18.3
64	15.2	20.1

the noise suppression levels with the number of subbands increases from 16 to 64 for both the RSCAMA and the NSUAMA schemes. The suppression levels for both the beamformers improve as the number of subbands increases. Evidently, the NSUAMA achieves $4 - 5$ dB noise suppression improvement over the RSCAMA structure irrespective of the number of subbands, yielding an impressive noise suppression level of 20.1 dB for the case of $M = 64$ subbands.

For completeness, Figs. 10.5(a) and 10.5(b) show the normalized output power plots of both the source and noise before and after the processing for both beamformers. From the power spectral plots, it is evident that the signal integrity of the source is maintained whilst the noise is suppressed uniformly across the frequency for both schemes. As mentioned previously, the NSUAMA algorithm achieves better noise suppression compared to the RSCAMA algorithm. More significantly, Fig. 10.5(a) reveals that the NSUAMA offers less spectral distortion to the SOI than the RSCAMA structure. This is attributed to the noise detector, which prevents the inclusion of the SOI in the update of the noise information. Nevertheless, the RSCAMA scheme has its merits as far as computational burden is concerned. For instance, in the RSCAMA algorithm, the update routine uses the matrix inversion lemma only twice (see Section 10.3.2). The NSUAMA algorithm on the other hand, requires the use of matrix inversion lemma to update all the eigenvectors of the SOI (see Section 10.4.4). However, depending on how much the space the SOI spans, it will have a few dominating eigenvectors. Therefore only half of the eigenvectors are updated in this evaluation and thus the NSUAMA scheme requires more computational requirements than the RSCAMA structure. Informal listening tests suggest good quality outputs from both the RSCAMA and the NSUAMA beamformers, with the NSUAMA offering more superior sound quality.

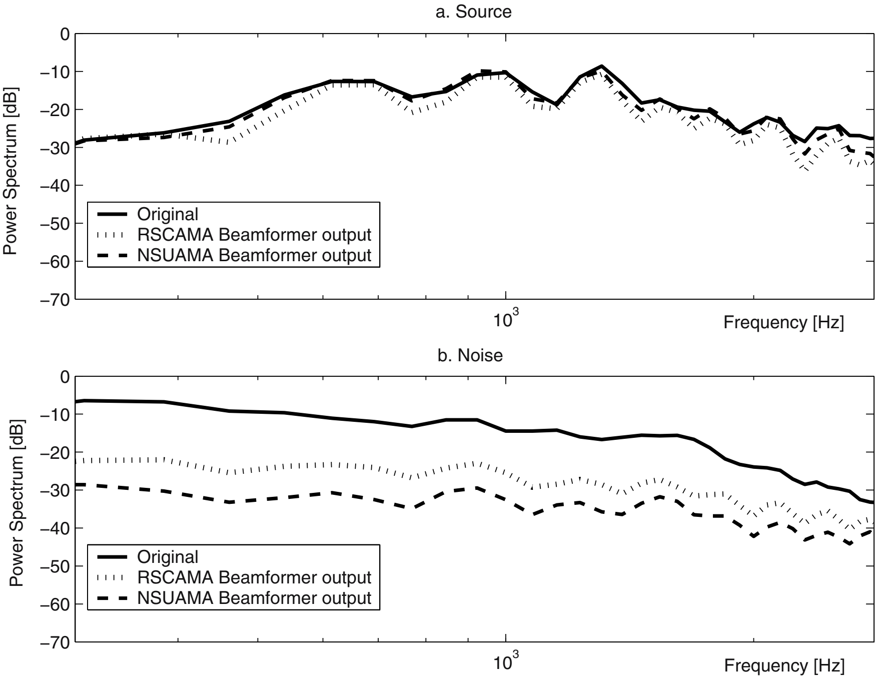


Fig. 10.5. Normalized output PSD plots for the RSCAMA and NSUAMA before and after processing of (a) source and (b) noise.

10.6 Conclusions

Two new space constrained adaptive microphone arrays with noise statistics updates have been presented. The novelty of both the structures lies in their space constraints, SOI spectral information and noise information updates. The space constraints provide robustness against steering vector errors and the update allows the noise statistics to be efficiently tracked in the Wiener solution. Also, the inclusion of the SOI PSD update in the solution offers a spectrally optimized constraint on the target signal integrity. The combination of both the PSD and space in the constraints makes full use of the available spatio-temporal domain. The major difference between the RSCAMA and NSUAMA algorithms is the manner that the SOI PSD and noise information updates are estimated. Whilst the RSCAMA is more computationally straightforward compared to the NSUAMA scheme, the NSUAMA achieves higher noise suppression capability. Results in a real hands-free car scenario show that the RSCAMA manages to achieve a good noise suppression level up to 15 dB and an impressive noise suppression of 20 dB for the NSUAMA.

References

1. Y. Grenier, "A microphone array for car environment," *Speech Communication*, vol. 12, pp. 25–39, Dec. 1993.
2. S. Nordholm, I. Claesson, and B. Bengtsson, "Adaptive array noise suppression of handsfree speaker input in cars," *IEEE Trans. on Vehicular Technology*, vol. 42, pp. 514–518, Nov. 1993.
3. N. Grbić, S. Nordholm, and A. Johansson, "Speech enhancement for hands-free terminals," in *Proc. IEEE Int. Sym. on Image and Signal Process. and Analysis*, 2001, pp. 435–440.
4. N. Grbić and S. Nordholm, "Soft constrained subband beamforming for hands-free speech enhancement," in *Proc. IEEE ICASSP*, 2002, vol. 1, pp. 885–888.
5. E. Jan and J. Flanagan, "Microphone arrays for speech processing," in *Proc. IEEE Int. Sym. on Signals, Systems, and Electronics*, 1995, pp. 373–376.
6. M. Brandstein and D. B. Ward, Editors, *Microphone Arrays: Signal Processing Techniques and Applications*, Ch. 3, pp. 39–60, Springer-Verlag, 2001.
7. B. D. Van Veen and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE Acoust., Speech and Signal Process. Magazine*, vol. 5, pp. 4–24, Apr. 1988.
8. H. Q. Dam, S. Nordholm, N. Grbic, and H. H. Dam, "Speech enhancement employing adaptive beamformer with recursively updated soft constraints," in *Proc. IWAENC*, 2003, pp. 307–310.
9. S. Y. Low, S. Nordholm, and N Grbić, "Subband generalized sidelobe canceller - a constrained region approach," in *Proc. IEEE Int. Workshop on Apps. of Signal Process. to Audio and Acoust.*, 2003, pp. 41–44.
10. H. Q. Dam, S. Y. Low, S. Nordholm, and H. H. Dam, "Adaptive microphone array with noise statistics updates," in *Proc. IEEE Int. Sym. on Circuits and Systems*, 2004, vol. 3, pp. 433–436.

11. O. Hoshuyama, A. Sugiyama, and A. Hirano, “A robust adaptive beamformer for microphone arrays with a blocking matrix using constrained adaptive filters,” *IEEE Trans. on Signal Process.*, vol. 47, pp. 2677–2684, June 1999.
12. I. Claesson and S. Nordholm, “A spatial filtering approach to robust beamforming,” *IEEE Trans. on Antennas and Propagation*, vol. 40, pp. 1093–1096, Sept. 1992.
13. S. Nordholm, I. Claesson, and M. Dahl, “Adaptive microphone array employing calibration signals: analytical evaluation,” *IEEE Trans. on Speech and Audio Process.*, vol. 7, pp. 241–252, May 1999.
14. M. Dahl and I. Claesson, “Acoustic noise and echo cancelling with microphone array,” *IEEE Trans. on Speech and Audio Process.*, vol. 48, pp. 1518–1526, Sept. 1999.
15. N. Grbić, S. Nordholm, and A. Cantoni, “Optimal FIR subband beamforming for speech enhancement in multipath environments,” *IEEE Signal Process. Letters*, vol. 10, pp. 335–338, Nov. 2003.
16. N. Grbić, S. Nordholm, and A. Cantoni, “Limits in FIR subband beamforming for spatially spread near-field speech sources,” in *Proc. IEEE Int. Sym. on Circuits and Systems*, 2003, vol. 2, pp. 516–519.
17. H. Q. Dam, S. Y. Low, H. H. Dam, and S. Nordholm, “Space constrained beamforming with source PSD updates,” in *Proc. IEEE ICASSP*, 2004, vol. 4, pp. 93–96.
18. J. M. de Haan, N. Grbić, I. Claesson, and S. Nordholm, “Filter bank design for subband adaptive microphone arrays,” *IEEE Trans. on Speech and Audio Process.*, vol. 11 , pp. 14–23, Jan. 2003.
19. K. F. C. Yiu, N. Grbić, S. Nordholm, and K. L. Teo, “Multicriteria design of oversampled uniform DFT filter banks,” *IEEE Signal Process. Letters*, vol. 11, pp. 541–544, June 2004.

11 Single-Microphone Blind Dereverberation

Tomohiro Nakatani, Masato Miyoshi, and Keisuke Kinoshita

NTT Communication Science Laboratories
Soraku-gun, Kyoto 619-0237, Japan
E-mail: {nak, miyo, kinoshita}@cslab.kecl.ntt.co.jp

Abstract. Although a number of dereverberation methods have been studied, dereverberation is still a challenging problem especially when using a single microphone. An important aspect of single-channel speech enhancement is the characteristic feature of speech signals that allows us to restore the quality of source signals. In this chapter, we describe a single-channel speech dereverberation method based on the harmonicity of speech signals. We show that a filter that enhances the harmonic structure of reverberant speech signals approximates the inverse filter of the reverberation process, thus enabling us to achieve high quality blind dereverberation. The presented method is referred to as the harmonicity based dereverberation method, HERB. Simulation experiments show that HERB can work effectively to dereverberate speech signals in terms of energy decay curves of room impulse responses and automatic speech recognition performance even when the reverberation time is as long as 1.0 sec provided a sufficiently large number of observed signals are available. Further discussions on several future directions are also provided with a view to extending HERB so that it can cope with more realistic situations.

11.1 Introduction

In the real world, reverberation is one of the primary factors that degrade the quality of speech signals when captured by a distant microphone. It makes sounds unintelligible, and prevents computers from adequately extracting any speech features. This problem becomes more severe as the reverberation time becomes longer. For example, when the reverberation time is longer than 0.5 sec, the performance of an automatic speech recognition system does not improve sufficiently even when the recognizer is trained on reverberant signals captured in the same environment [1].

We need to develop a way to restore the quality of speech signals from the reverberant signals in order to overcome this problem. In particular, this must be accomplished based solely on the observed signals when we have no prior information about the room's acoustic properties. This operation is known as blind dereverberation.

In general, an observed reverberant signal, $x(n)$, can be modeled as a convolution of its source signal, $s(n)$, and a room impulse response, $h(n)$, as

$$x(n) = \sum_{m=0}^{\infty} h(m)s(n-m), \tag{11.1}$$

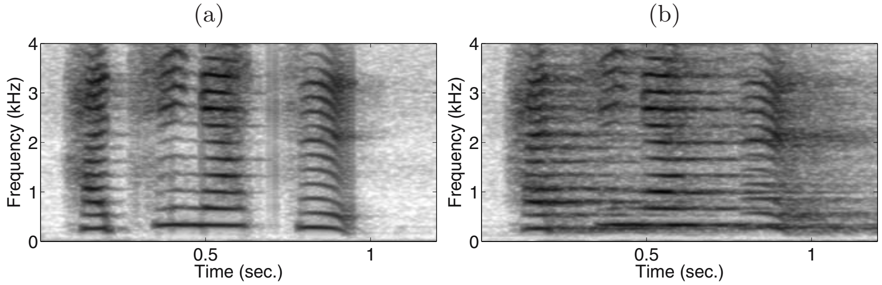


Fig. 11.1. Example spectrograms of (a) a clean speech signal and (b) a reverberant speech signal when the reverberation time is 1.0 sec.

where n and m are time indices of sampled signals. This equation means that speech signals in different time regions are added up to the reverberation with the weight of the room impulse response. Since features of a speech signal change with time, different features are mixed in the reverberation, which degrades the quality of the signals. Figure 11.1 shows an example of spectrograms of clean and reverberant speech signals. As seen in the figure, the time and spectral structures of the speech are unclear in the reverberant signal.

Inverse filtering is a method that is frequently used to achieve speech dereverberation [2]. With this approach, reverberant signals are dereverberated by estimating the inverse filter, $w(n)$, that cancels out the reverberation effect by applying it to the signals as

$$y(n) = \sum_{m=-\infty}^{\infty} w(m)x(n-m). \tag{11.2}$$

Ideally, $y(n)$ is identical to $s(n)$ multiplied with a certain scalar constant c, that is,

$$y(n) = cs(n). \tag{11.3}$$

Then, $w(n)$ should satisfy

$$\sum_{m=0}^{\infty} w(n-m)h(m) = c\delta_{n,0}, \tag{11.4}$$

where $\delta_{i,j}$ is the Kronecker delta function. Because neither $s(n)$ nor $h(n)$ are known in the case of blind dereverberation, we have to estimate $w(n)$ without directly examining (11.3) or (11.4). Therefore, further assumptions regarding $h(n)$ or $s(n)$ are indispensable to blind dereverberation.

In this chapter, we describe blind dereverberation technologies for single-channel speech signals. We consider the investigation of this technology to be very important for the following reasons:

1. Single-channel processing requires only one microphone, so it is supposed to be making it potentially useful for a variety of applications. Most recently developed speech applications such as automatic speech recognition (ASR) are based on single-channel processing.
2. It may provide a basic technology that can utilize the features of source signals for dereverberation. This technology is not only indispensable to single-channel signal processing but also very important for elaborating multiple signal processing.

One of the most important issues in single-channel processing is the dereverberation principle that exploits the features of source signals. In Sect. 11.2, we first provide an overview of such principles that have been proposed for several existing technologies. Section 11.3 describes an important feature of speech signals, namely harmonicity, together with robust signal processing techniques with which to handle this feature. In Sect. 11.4, we provide a detailed explanation of a dereverberation principle based on the harmonicity of speech signals, and a dereverberation method based on this principle, which we call *Harmonicity based dEReverBeration*, (*HERB*). HERB is a recently proposed method, and is shown by simulation experiments to achieve high quality speech dereverberation even under long reverberation time conditions when a sufficiently large number of observed signals are available. The experimental simulation results are presented with HERB in Sect. 11.6. Several future directions related to HERB are discussed in Sect. 11.7, including the possibility of applying it to more realistic situations.

11.2 Overview of Existing Approaches

This section describes two different approaches that have been proposed for single-channel speech dereverberation.

11.2.1 Blind Inverse Filtering

When we can assume that the source signal is an independent and identically distributed (i.i.d.) sequence, the inverse filter, $w(n)$, that satisfies (11.4) can be obtained by estimating a filter that makes $y(n)$ in (11.2) an i.i.d. sequence. This principle often provides a basis for blind inverse filtering techniques. With speech signals, however, this principle needs to be appropriately modified because speech signals are not i.i.d. They have inherent features such as periodicity and a formant structure that makes a speech signal statistically dependent. Once $y(n)$ is made i.i.d., these important features of speech signals are excluded.

Several attempts have been made to compensate for this discrepancy. For example, Douglas et al. proposed a method, in which the error of the linear prediction (LP) is dealt with as signals that should be dereverberated [3], assuming the LP residue of a speech signal is an i.i.d. sequence. Furthermore, Gillespie et al. proposed another technique, referred to as correlation shaping [4]. They assume that the LP residue of a source speech signal is correlated only within a short time duration, and that the correlation between very different time indices is solely the result of the room impulse response. Therefore, the latter correlation can be decorrelated without losing any important features of the speech signals. They proposed a method for estimating the inverse filter that decorrelates only the long term correlation of the LP residue.

These methods were partially successful in dereverberating speech signals with relatively little reverberation, but it is still difficult to cope with more severe reverberation conditions where dereverberation is really required. We consider these methods to be limited by the impreciseness of their assumptions. For example, the LP residue of a speech signal is still not an i.i.d. sequence, and the LP coefficients, which mainly represent the formant structure of speech signals, are excluded from features that should be restored by the dereverberation.

11.2.2 Dereverberation Based on Speech Signal Features

Some researchers have also proposed dereverberation methodologies that utilize the properties of speech signals in a more direct manner [5], [6]. For example, Yegnanarayana et al. proposed a method that attenuates the signal peaks in the LP residue that are derived from the reverberation [5]. Their method categorizes the peaks into speech events and reverberation, based on their criterion referred to as signal-to-reverberation ratios. This method can improve the intelligibility of speech signals to a certain extent, but some artifacts occur in the resultant signals. We consider this limitation to be caused by the difficulty in strictly distinguishing the peaks derived from the signal and reverberation.

Unoki et al. proposed a method for recovering the temporal envelope of a signal from an observed reverberant signal by exploiting the characteristics of the modulation transfer functions (MTFs) [7] of source signals and a room impulse response [6]. They assumed that the temporal envelope of a room impulse response decays exponentially with time and that the carrier signals of the room impulse response and a speech signal can be modeled as mutually independent white noise functions. Then, the temporal envelope of the observed reverberant signal is equal to a convolution of those of the room impulse response and the speech signal. They proposed a method for estimating the reverberation time from the observed signal and decomposing two envelopes from the observed envelope based on the estimated reverberation time. However, there are still difficulties with their method and these must be overcome before it can be applied to actual speech signals because the method cannot recover carrier signals and they use assumptions that are not always satisfied by the properties of speech signals and reverberation.

11.3 Harmonicity of Speech Signals and Its Robust Estimation

In this section, we describe a basic property of speech signals, namely harmonicity and describe certain signal processing technologies for handling this property in a noisy environment, before introducing this feature as the key to speech dereverberation, in the next section.

Harmonicity has long been studied as a robust feature of speech signals in the real world. It is cited as a major clue in relation to a person's ability to extract desired speech from other sounds [8]. Many speech enhancement methods employ a harmonicity-based sound segregation scheme, and this has improved the performance of automatic speech recognition (ASR) [9], [10]. However, these methods have not succeeded in extracting the precise harmonic structure of speech signals in the presence of a long reverberation. This is because different fundamental frequencies (F_0) in different time regions are mixed into the reverberation, and thus the harmonic structure is severely degraded. Therefore, harmonicity had not been taken into account as a primary cue for enhancing or dereverberating reverberant speech signals until the recent development of harmonicity based speech dereverberation.

11.3.1 Model of Speech Harmonicity

A speech signal, $s(n)$, can be modeled by using the sum of a harmonic signal, $s_h(n)$, derived from a glottal vibration, and a non-harmonic signal, $s_n(n)$, such as fricatives and plosives as

$$s(n) = s_h(n) + s_n(n). \tag{11.5}$$

The most important properties with which to model a harmonic signal are voiced durations and their fundamental frequencies (F_0s). A voiced duration refers to the time during which a speaker's vocal cords vibrate to generate a harmonic signal, while fundamental frequency refers to the frequency of the fundamental component of a harmonic signal. Each harmonic component has a frequency that corresponds to F_0 or its multiples. Therefore, the harmonic signal, $s_h(n)$, within a time frame can be modeled by the sum of sinusoidal components whose frequencies coincide with the fundamental frequency (F_0) of the signal or its multiples.

Let l and $\dot{\theta}_l$ respectively be the index of a time frame and the F_0 of the harmonic signal at the time frame, and let $A_{k,l}$ and $\phi_{k,l}$ respectively be the amplitude and phase of its k-th harmonic component, then the waveform of $s_h(n)$ within the frame can be modeled as

$$s_h(n) = \sum_{k=1}^{N} A_{k,l} \cos\left(k\dot{\theta}_l \frac{n - n_l}{f_s} + \phi_{k,l}\right) \quad \text{for} \quad |n - n_l| < T/2, \tag{11.6}$$

where n_l is the time index of a sampled signal centered in the time frame, N is the number of harmonic components, T is the length of the time frame, and f_s is the sampling frequency of the signal.

Strictly speaking, there are certain errors in the above model since F_0 and amplitudes of a speech signal gradually change even within a time frame. These cause certain modeling errors in a harmonic signal, however, they are not very serious in many practical speech applications. In addition, we can introduce techniques that moderate these errors if necessary. We describe such techniques based on time warping analysis in Sect. 11.7.2.

11.3.2 Adaptive Harmonic Filtering

The harmonicity of speech signals plays an especially important role for speech enhancement in the presence of background noise. This is because we can extract a harmonic signal from other sounds by enhancing frequency components whose frequencies correspond to the F_0 of the signal or its multiples. An operation that realizes this enhancement is called harmonic filtering. Since the F_0 of a speech signal changes with time, the properties of the filter have to be adaptively modified frame by frame according to the F_0. Therefore, a harmonic filter is implemented by means of a time-varying filter, and this usually involves a procedure for detecting voiced durations and estimating their F_0s.

Various harmonic filters have been proposed for this purpose, for example, a widely used filter, referred to as a comb filter, is defined as $1 + z^{-\tau}$ where τ is the period of the signal to be enhanced [11]. In this section, we describe a harmonic filter based on a sinusoidal synthesis framework, which we use in the experiments described in Sect. 11.6. With this method, the F_0 of the signal is first estimated at each time frame, then the amplitudes and phases of individual harmonic components are estimated from the signal, and finally a harmonic sound is synthesized by adding sinusoids according to the estimated amplitudes and phases of the harmonic components. One advantageous feature of this filter is that it can precisely preserve the amplitude and phase of each harmonic component included in an observed signal when its frequency is estimated correctly.

The processing flow of this filter is summarized as follows:

1. F_0 of the observed signal, $x(n)$, is estimated at each time frame based on such F_0 estimation methods as those described in Sect. 11.3.3.
2. The amplitudes and phases of individual harmonic components are estimated at the time frame as follows:

$$X(l,m) = \sum_n g_1(n-n_l)x(n)\exp\left(-j2\pi\frac{m}{M}\frac{n-n_l}{f_s}\right), \tag{11.7}$$

$$\hat{A}_{k,l} = |X(l,[k\dot{\theta}_l])|, \tag{11.8}$$

$$\hat{\phi}_{k,l} = \angle X(l,[k\dot{\theta}_l]), \tag{11.9}$$

where $X(l,m)$ is a short time Fourier transform (STFT) of $x(n)$ at a time frame l, m is the index of a frequency bin, n_l is a time index of the sampled signal centered in the frame, M is the number of discrete Fourier transformation (DFT) points, $\hat{A}_{k,l}$ and $\hat{\phi}_{k,l}$ are respectively the estimated amplitude and phase of the k-th harmonic component, $\dot{\theta}_l$ is the F_0 estimate at the frame, $g_1(n)$ is a window function, and $[\cdot]$ is an operator that digitizes a continuous frequency into an index of the nearest frequency bin.

3. The output of the filter, $\hat{x}(n)$, is synthesized by adding sinusoids as (11.10) and by combining them over succeeding frames based on the overlap-add synthesis as (11.11):

$$\hat{x}_l(n) = \sum_k \hat{A}_{k,l} \cos(k\dot{\theta}_l(n - n_l)/f_s + \hat{\phi}_{k,l}), \tag{11.10}$$

$$\hat{x}(n) = \sum_m g_2(n - (n_l + m\triangle T))\hat{x}_{l+m}(n), \tag{11.11}$$

where $\hat{x}_l(n)$ is a synthesized harmonic sound corresponding to a time frame l, $\triangle T$ is the frame shift in the samples and $g_2(n)$ is a window function.

11.3.3 Robust F_0 Estimation and Voicing Detection

Robust and accurate F_0 estimation and voicing detection are important for harmonicity based speech enhancement because any errors in these values cause a deterioration in the enhanced speech quality. A number of useful estimation methods have already been reported for this purpose.

Here, we introduce a simple and effective F_0 estimation method that we used in our experiments in Sect. 11.6. This method is based on a modified version of a linear power spectrum, and we call it the ripple enhanced power spectrum (REPS) [13]. In a linear power spectrum, the dominant harmonic components in a speech signal are usually represented as sharp peaks compared with additive noise even when the signal-to-noise ratio (SNR) is as low as 0 dB. Based on this property, we can organize a robust F_0 estimation method in the presence of additive background noise. We confirmed that a better performance is generally obtained by modifying the linear power spectrum, $X(l,m)^2$, so as to enhance the spectral ripple corresponding to the glottal pulse in advance, than by directly using the spectrum.

The processing flow of the REPS based F_0 estimation is summarized as follows:

1. A REPS, $R(l,f)$, is obtained from a linear power spectrum $|X(l,m)|^2$ by using a method similar to cepstral liftering in the power spectrum (rather than in the log domain as with usual cepstral liftering), that is,
   (a) applying an inverse discrete Fourier transformation (IDFT) to $|X(l,m)|^2$,

(b) substituting zeros for the lower quefrency components, and
(c) applying DFT to the modified coefficients.

2. An F_0 decision measure, $H(l, f_n)$, referred to as harmonic dominance, is calculated for each discretized F_0 candidate, f_n $(n = 1, 2, \ldots)$, within an F_0 search range at each time frame l. This measure represents the power of harmonic components for an assumed F_0. As the measure becomes larger, it becomes more likely that the assumed F_0 will be correct. The measure is defined as follows:

$$H(l, f_n) = \sum_{k=1}^{[kf_n]<F_{\max}} \left\{ R(l, [kf_n]) - \bar{R}(l) \right\}, \tag{11.12}$$

where $F_{\max}$ is the maximum index of frequency bins that are taken into account for F_0 estimation. $\bar{R}(l)$ is a term unbiasing the REPS to reduce double/half errors in F_0. It is defined by

$$\bar{R}(l) = \frac{1}{F} \sum_{m=1}^{F} R(l, m), \tag{11.13}$$

Because $H(l, f_n)$ is defined as the sum of REPS corresponding to f_n and its multiples, its value is expected to be maximum value when f_n coincides with F_0. Therefore, we can determine F_0 as the f_n that maximizes $H(l, f_n)$.

3. Dynamic programming (DP) is employed to improve the robustness of the F_0 estimation. DP is expected to reduce the discontinuous F_0 transition errors by taking the F_0 transition cost into account. It tracks continuous peaks in $H(l, f_n)$ over succeeding time frames while minimizing the total F_0 transition costs. The F_0 candidates f_n on the track are determined as the resulting F_0 estimates.

It should be noted that the REPS based F_0 estimation method originally had a mechanism for further enhancing the precision of the F_0 estimates based on the instantaneous frequency [12], [13]. We skipped this procedure in our experiments because this improvement is not necessarily very effective for dereverberation.

Regarding voicing decision, we again introduce a method that uses harmonic dominance, $H(l, m)$. This method determines the voicing status based on the magnitude of harmonic components relative to the other components. Suppose the fundamental frequency, $\dot{\theta}_l$, is estimated for each time frame by assuming all the frames are voiced. Then, the relative magnitude of the harmonic components, $V(l)$, at a time frame l is defined as

$$V(l) = M_p \left\{ \frac{H(l, \dot{\theta}_l) - E\{H(l, f_n)\}}{\sigma\{H(l, f_n)\}} \right\}, \tag{11.14}$$

where $E_f\{\cdot\}$ and $\sigma_f\{\cdot\}$ are functions yielding the average and standard deviation of $H(l, f_n)$ over frequencies, respectively, and $M_p\{\cdot\}$ is a function that extracts a median value over p time frames. A frame is determined as voiced if $V(l)$ is larger than a fixed threshold value. Because $V(l)$ is a value normalized with the standard deviation, this threshold can be set independently of the signal level.

11.4 Harmonicity Based Dereverberation – HERB

In this section, we discuss how speech harmonicity can be utilized for blind speech dereverberation. We first describe the basic idea behind harmonicity based dereverberation, HERB, and then show that the dereverberation filter obtained by HERB is a good approximation of the inverse filter of a room transfer function. We also describe the implementation of a prototype system of HERB.

11.4.1 Basic Idea

Let $X(l,m)$ be a short time Fourier transform of an observed reverberant signal, $x(n)$, obtained using a sufficiently long time frame. $X(l,m)$ can be represented as the product of the source signal, $S(l,m)$, and the room transfer function, $H(m)$ as in (11.15). We assume the room transfer function, $H(m)$, is time invariant. This transfer function can be divided into two functions, $D(m)$ and $R(m)$ as in (11.16). The former transforms $S(l,m)$ to the direct signal $D(m)S(l,m)$ and the latter to the reverberation part $R(m)S(l,m)$:

$$\begin{aligned} X(l,m) &= H(m)S(l,m) && (11.15)\\ &= D(m)S(l,m) + R(m)S(l,m). && (11.16) \end{aligned}$$

In HERB, the direct signal, $X'(l,m) = D(m)S(l,m)$, is dealt with as the desired signal that should be obtained as a result of the dereverberation. $X'(l,m)$ can be obtained by subtracting $R(m)S(l,m)$ from (11.16), or by estimating the inverse filter $W(m)$ that satisfies (11.17) and multiplying it with $X(l,m)$ as in (11.18):

$$\begin{aligned} W(m) &= D(m)/H(m), && (11.17)\\ X'(l,m) &= W(m)X(l,m). && (11.18) \end{aligned}$$

To solve this problem, HERB estimates the inverse filter $W(m)$ principally according the following procedure.

1. The initial estimate of the direct signal of a harmonic signal included in $X(l,m)$, referred to as $\hat{X}(l,m)$, is determined using an adaptive harmonic filter.

2. Assuming that $\hat{X}(l,m)$ approximates the direct signal of $S(l,m)$, the initial estimate of the inverse filter $\hat{W}_0(l,m)$ is determined for each time frame as $W_0(l,m) = \hat{X}(l,m)/X(l,m)$.
3. In order to exclude the approximation errors included in the initial inverse filter estimate, $W_0(l,m)$, the estimate of the inverse filter, referred to as the dereverberation filter, is given by averaging the initial inverse filter estimate over different time frames as $W(m) = E_t\{W_0(l,m)\}$.

Let us consider a very simple case to explain the basic idea of the above procedure. Suppose that a signal $s(n)$ whose frequency sweeps from 100 to 7000 Hz occurs as a source signal, and is convolved with the room impulse response of a reverberant room. Figure 11.2 shows spectrograms of this source signal and its observed signal. In the spectrogram of the observed signal, the energy of the direct signal is concentrated in the area in which the signal first appears at each frequency. Therefore, we can approximately extract the direct signal by extracting a frequency component in this area. This can be done by tracking frequency $\dot{\theta}(l)$ of a dominant sinusoidal component in the reverberant signal at each short time frame, extracting its amplitude $\hat{A}(l)$ and phase $\hat{\phi}(l)$, and synthesizing it as

$$\hat{x}(n) = \sum_l g(n - n_l)\hat{A}(l)\cos(\dot{\theta}(l)n/f_s + \hat{\phi}(l)), \tag{11.19}$$

where $g(n)$ is a window function for overlap-add synthesis and n_l is the time index centered in a frame l. In this simple case, the accuracy of the direct signal estimation is rather high because no other signal appears at the same time as the direct signal, and thus the estimated inverse filter $W_0(m) = \hat{X}(m)/X(m)$, where $\hat{X}(m)$ and $X(m)$ are DFTs of whole signals of $\hat{x}(n)$ and $x(n)$, is expected to be very close to the desired inverse filter. Figure 11.3 shows the reverberation curves [14] of the room impulse response $h(n)$ of a reverberant room and the impulse response dereverberated by this simple method. This clearly demonstrates the effective reduction of the reverberation.

11.4.2 Model of Reverberant Speech Signal

We extend the dereverberation method discussed in the previous section to cope with speech signals. For this purpose, we first provide a further analysis of the source speech signal model. Equation (11.5) is rewritten with a frequency domain representation as

$$S(l,m) = S_h(l,m) + S_n(l,m). \tag{11.20}$$

The observed reverberant signal, $X(l,m)$, is then obtained by multiplying a room transfer function $H(l,m)$ by $S(l,m)$ as (11.21). This equation can be

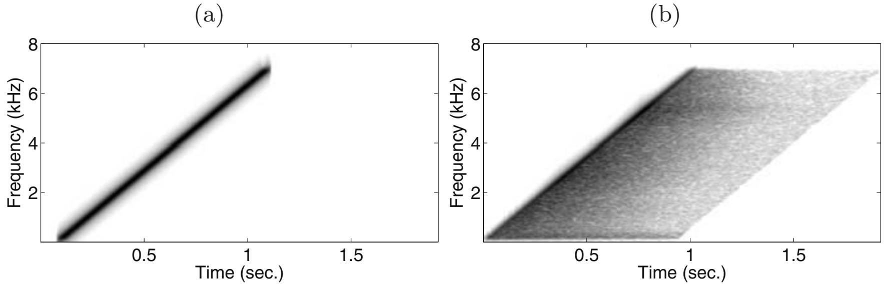


Fig. 11.2. Spectrograms of (a) a sweeping sinusoid and (b) its observed reverberant signal.

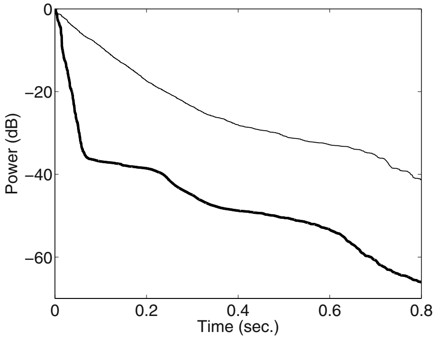


Fig. 11.3. Reverberation curves of original impulse response (thin line), and dereverberated impulse response (thick line).

again rewritten by dividing $H(m)$ into the two functions $D(m)$ and $R(m)$ as in (11.22),

$$X(l,m) = H(m)S(l,m), \tag{11.21}$$

$$X(l,m) = D(m)S(l,m) + R(m)S(l,m). \tag{11.22}$$

The observed signal, $X(l,m)$, is further rewritten as (11.23) using (11.20),

$$X(l,m) = D(m)S_h(l,m) + (R(m)S_h(l,m) + H(m)S_n(l,m)). \tag{11.23}$$

The first term on the right side of (11.23), $D(m)S_h(l,m)$, is the direct signal of the harmonic signal, and is as highly periodic as the harmonic signal in the source signal. By contrast, $R(m)S_h(l,m)$ in the second term on the right side is the reverberation part of the harmonic signal, and thus has degraded harmonicity. $H(m)S_n(l,m)$ is not harmonic because S_n is originally a non-harmonic part. Therefore, the second term on the right side represents the non-harmonic parts in the observed signal.

Of these components, $D(m)S_h(l,m)$ can be approximately extracted from $X(l,m)$ with an adaptive harmonic filter. This approximated direct signal

$\hat{X}(l,m)$ can be modeled as follows:

$$\hat{X}(l,m) = D(m)S_h(l,m) + (\hat{R}_h(l,m) + \hat{H}_n(l,m)), \tag{11.24}$$

where $\hat{R}_h(l,m)$ and $\hat{H}_n(l,m)$, respectively, are part of the reverberation of $S_h(l,m)$ and part of the direct signal and reverberation of $S_n(l,m)$, which unexpectedly remain in $\hat{X}(l,m)$ after the harmonic filtering. Here, we assume that all the estimation errors in $\hat{X}(l,m)$ are caused by $\hat{R}_h(l,m)$ and $\hat{H}_n(l,m)$ in (11.24). In general, it is difficult to estimate the F_0 of a reverberant speech signal precisely, therefore, $\hat{X}(l,m)$ also contains estimation errors caused by the F_0 estimation errors without any compensation. To reduce these kinds of errors, HERB has a mechanism that excludes them effectively through its dereverberation procedure. We describe this mechanism in detail in Sect. 11.5.

11.4.3 Dereverberation Filter

HERB estimates the inverse filter, hereafter called the dereverberation filter, as the time average of a filter that transforms observed reverberant signals, $x(n)$, into the output of an adaptive harmonic filter, $\hat{x}(n)$. Here, we assume that $\hat{x}(n)$ roughly approximates the direct harmonic components in the observed signals. The dereverberation filter, $W(\omega)$, is calculated using $\hat{X}(l,m)$ and $X(l,m)$, or STFTs of $x(n)$ and $\hat{x}(n)$, as

$$W_0(l,m) = \frac{\hat{X}(l,m)}{X(l,m)}, \tag{11.25}$$

$$W(m) = E_t\left\{W_0(l,m)\right\}, \tag{11.26}$$

where $E_t\{\cdot\}$ represents an average function that calculates the average value of $\hat{X}(l,m)/X(l,m)$ over time frames. With speech signals, this filter can be shown to provide a good approximation of the inverse filter of a room transfer function for speech signals. Next, we briefly interpret the property of this filter.

11.4.4 Interpretation of the Dereverberation Filter

By substituting $X(l,m)$ and $\hat{X}(l,m)$ in (11.25) with (11.21) and (11.24), the following equation can be derived:

$$W(m) \simeq \frac{D(m) + \hat{R}(m)}{H(m)} P\{|S_h(l,m)| > |S_n(l,m)|\}, \tag{11.27}$$

where

$$\hat{R}(m) = E_t\left\{\frac{\hat{R}_h(l,m)}{S_h(l,m)}\right\}_{|S_h(l,m)|>|S_n(l,m)|}, \tag{11.28}$$

where $P\{\cdot\}$ is a probability function, and $E_t\{\cdot\}_A$ represents an average function over time frames under a condition where A holds. Note that it is necessary to use the following assumptions to derive (11.27).

1. $\angle S_n(l,m)$ and a joint event composed of $S_h(l,m)$, $\hat{R}_h(l,m)$, and $|S_n(l,m)|$ are statistically independent,
2. $\angle S_h(l,m)$ and a joint event composed of $|S_h(l,m)|$, $\hat{H}_n(l,m)$, and $S_n(l,m)$ are statistically independent,
3. $\angle S_h(l,m)$ and $\angle S_n(l,m)$ are uniformly distributed within $[0,2\pi)$,
4. $|S_h(l,m)| \neq |S_n(l,m)|$.

Here, we omit the detailed derivation of (11.27).

Equation (11.27) means that $W(m)$ approximately coincides with the product of $(D(m)+\hat{R}(m))/H(m)$ and $P\{|S_h(l,m)| > |S_n(l,m)|\}$. The former, $(D(m)+\hat{R}(m))/H(m)$, strictly equals the inverse filter, $D(m)/H(m)$, when an adaptive harmonic filter can completely reduce $\hat{R}_h(l,m)$ in (11.24) without any errors. Although it is very difficult to reduce $\hat{R}_h(l,m)$ completely, a major part of $R(m)S_h(l,m)$ can be eliminated with an adaptive harmonic filter. In addition, $\hat{R}(m)$ is defined as an average filter that transforms $S_h(l,m)$ to $\hat{R}_h(l,m)$. Therefore, $\hat{R}(m)$ is expected to become a transformation that produces reduced reverberation. As a consequence, the signal obtained by multiplying the observed signal $X(l,m)$ by $(D(m)+\hat{R}(m))/H(m)$ is expected to be the sum of the direct signal and the reduced reverberation, that is, $((D(m)+\hat{R}(m))/H(m))X(l,m) = D(m)S(l,m) + \hat{R}(m)S(l,m)$. By contrast, $P\{|S_h(l,m)| > |S_n(l,m)|\}$ in (11.27) is the probability that the harmonic signal has a larger energy than the non-harmonic signal, and has a real value between 0.0 and 1.0. This term changes the gain of (11.26) but does not affect its dereverberation function.

The above analysis reveals the following properties of the dereverberation filter obtained by HERB.

- The dereverberation filter becomes an approximation of the inverse filter $D(m)/H(m)$ except that part of the reverberation remains and its gain changes. Therefore, this filter is expected to dereverberate both the harmonic and non-harmonic signals of speech signals.
- However, the filter gain becomes zero in the frequency region where harmonic components are not included in the training data used for estimating the dereverberation filter. This is because $P\{|S_h(l,m)| > |S_n(l,m)|\}$ becomes zero in such regions. These regions include frequency regions lower than the fundamental frequencies of speech signals or higher frequency regions where the harmonic structure becomes unclear.
- In addition, even in frequency regions in which harmonic components are included in the training data, the filter gain is assumed to decline as the frequency increases. This is because, with speech signals, the magnitude of the harmonic signal decreases relative to that of the non-harmonic signal as the frequency increases.

11.5 Implementation of a Prototype System

In this section, we describe a prototype system of HERB. We used it to examine the dereverberation effects of this approach. The prototype system consists mainly of the following subprocedures:

1. **F_0 estimation:** voiced durations and their F_0s are estimated from observed reverberant signals, $x(n)$.
2. **Harmonic filtering:** harmonic signals, $\hat{x}(n)$, included in $x(n)$, are estimated by means of waveforms based on adaptive harmonic filtering.
3. **Dereverberation filter estimation:** $x(n)$ and $\hat{x}(n)$ are divided into time frames and transformed into frequency-domain signals, as $X(l,m)$ and $\hat{X}(l,m)$, by short time discrete Fourier transformation. The dereverberation filter, $W(m)$, is estimated as the average of $\hat{X}(l,m)/X(l,m)$ over a number of time frames, and then transformed into a time domain filter, $w(n)$, by inverse discrete Fourier transformation.
4. **Dereverberation:** the dereverberated signals are obtained by convolving $x(n)$ with $w(n)$.

Figure 11.4 shows the complete processing flow of the prototype system. The dereverberation procedures are composed of three processing steps, with the aim of gradually improving the dereverberation performance in each step. All the subprocedures described above are employed in each step. They are summarized as follows:

STEP 1: F_0s, voiced durations, and harmonic components are all estimated directly from an observed reverberant signal, $x(n)$, therefore, the estimated values may contain many errors.

STEP 2: F_0s and voiced durations are estimated from signals dereverberated by the previous step, and harmonic components are estimated from observed reverberant signals, $x(n)$. Because the estimated F_0s and voiced durations are expected to improve, harmonic components estimated based on them are also expected to improve.

STEP 3: All the above values are estimated from signals dereverberated by the previous step. Because reverberant components, $\hat{R}_h$, inevitably included in (11.24) can further be reduced, more effective dereverberation is expected to be achieved.

In our preliminary experiments, the estimation of F_0 and voiced durations gradually improved when STEP 2 was repeated. By contrast, repeating STEP 3 did not always improve the quality of dereverberated signals. This is because estimation errors in the dereverberation filters accumulate in the dereverberated signals when the signals are multiplied by more than one dereverberation filter. In our experiments described in the next section, we employed all these steps without repeating any of them.

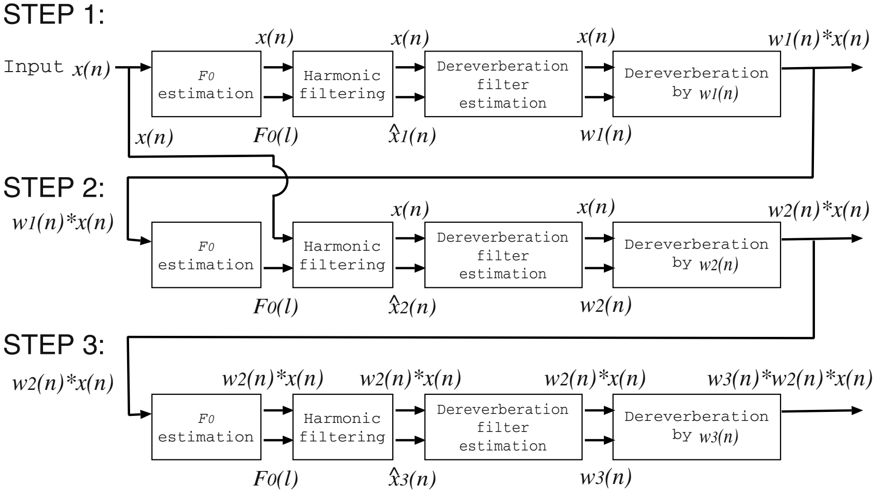


Fig. 11.4. Processing flow of the dereverberation by the prototype system.

11.5.1 Dereverberation Filter Calculation

In order to estimate the dereverberation filter precisely, we calculate the average function, $E_t\{\cdot\}$, in (11.26) which is weighted by the amplitude spectrum $|\hat{X}(l,m)|$ as

$$W(m) = \frac{\sum_m |\hat{X}(l,m)| W_0(l,m)}{\sum_m |\hat{X}(l,m)|}. \tag{11.29}$$

This is because we expect that values estimated at time frames that contain stronger harmonic components will be less affected by non-harmonic components, and thus more reliable.

Note that the estimated dereverberation filters result in delayed inverse filters when using our prototype systems. This means that the prototype system is applicable even to reverberant signals that include non-minimum phases [15].

11.5.2 Heuristics Improving Accuracy of F_0 Estimation and Voicing Decisions with Reverberation

Accurate F_0 estimation and voicing decisions are very important to achieve effective dereverberation in HERB. However, this is a difficult task especially for speech with long reverberation using some of the existing F_0 estimators [12], [13]. Although we employed a robust estimation method described in Sect. 11.3, it is not sufficiently robust, especially for speech signals with long reverberations. To cope with this problem, we introduced two types of preprocessing to the estimation: one uses a simple filter that reduces sound that continues at the same frequency [16], and the other uses the dereverberation filter itself. The effectiveness of these filters was confirmed in our preliminary experiments. The dereverberation filter based method is the most effective because the reverberation of the speech signals can be directly reduced by the filter. This mechanism is included in steps 2 and 3 of the dereverberation procedure, so more accurate F_0 estimation and voicing decisions can be achieved in steps 2 and 3 than in step 1.

11.6 Simulation Experiments

We evaluated the performance of the HERB prototype system using the dereverberation task described in Sect. 11.6.1 in terms of the energy decay curves of the impulse responses and speaker dependent automatic speech recognition (ASR).

11.6.1 Task: Dereverberation of Word Utterances

The task used in our experiments was the dereverberation of reverberant word utterances. We used 5240 Japanese word utterances provided by a male and a female speaker (MAU and FKM) included in the ATR database as source signals, $s(n)$. We used four impulse responses measured in a reverberant room whose reverberation times were about 0.1, 0.2, 0.5, and 1.0 sec. Reverberant signals, $x(n)$, were obtained by convolving $s(n)$ with the impulse responses.

To evaluate the fundamental performance of the prototype system, we assumed that the dereverberation filter was estimated using all male or all female word utterances. In addition, we assumed that each word utterance with reverberation was recorded separately, and that there was no time-overlap between utterances including their reverberation durations. When we estimated the dereverberation filter using (11.25) and (11.26), we calculated the STFT of $x(t)$ and $\hat{x}(t)$ with a time frame long enough to contain each whole word utterance with zero padding. The length of the dereverberation filter was 131,072 taps; that is, we used a 10.9 sec rectangular window for the X and $\hat{X}$ calculations. By contrast, we used a much shorter time frame, that is, a 42 msec Hanning window and a 1 msec window shift for the F_0 estimation and adaptive harmonic filtering in order to extract the time-varying features of the harmonic components. We used signals sampled at 12 kHz.

11.6.2 Energy Decay Curves of Impulse Responses

Figure 11.5 shows energy decay curves of room impulse responses and dereverberated impulse responses obtained by HERB while controlling the reverberation time. Each dereverberated impulse response was obtained by convolving a room impulse response with its dereverberation filter, and each decay curve was calculated using Schroeder's method [17].

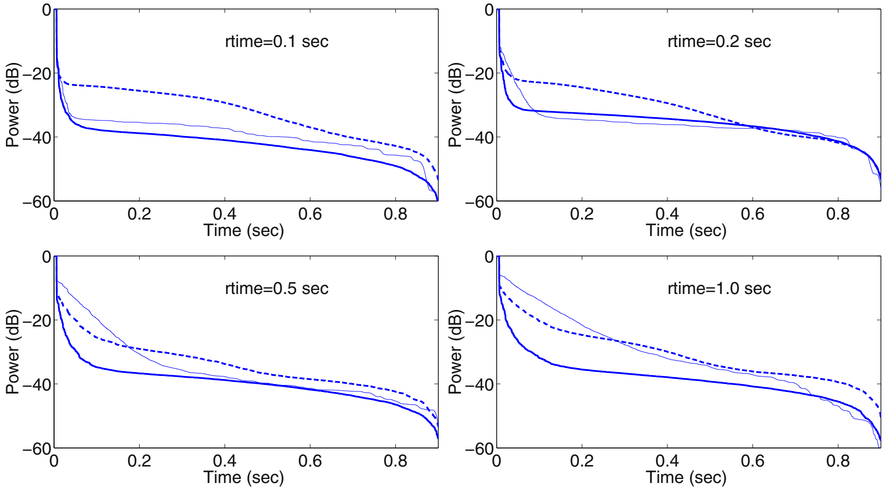


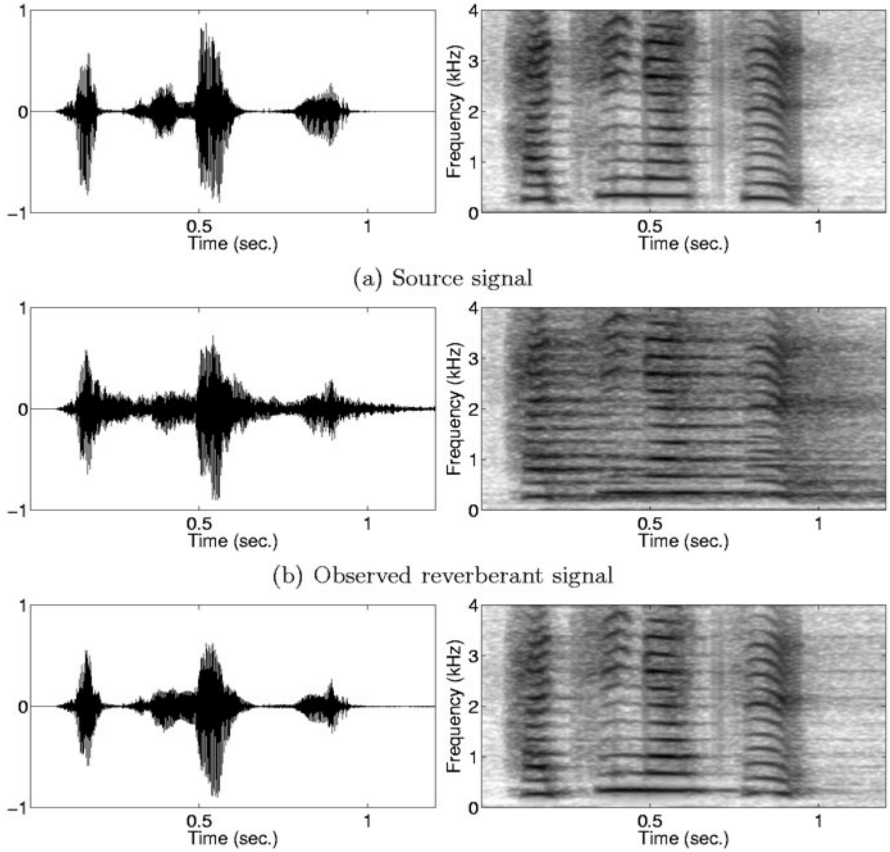
Fig. 11.5. Energy decay curves of the room impulse responses (thin solid line) and the dereverberated impulse responses when using female speech signals (thick solid line) and male speech signals (thick dashed line) as training data under different reverberation time (rtime) conditions.

The figure shows that the proposed method could effectively reduce the reverberation in the impulse responses for the female speaker when the reverberation time (rtime) was longer than 0.1 sec. For the male speaker, the reverberation effect in the lower time region was also effectively reduced. This means that strong reverberant components were reduced, and so the intelligibility of the target speech could be expected to be improved [14]. Although the reverberation effect in the higher time region for the male speaker was increased when rtime was 0.1 or 0.2 sec, the sound quality as a whole is expected to improve when rtime is 0.2 sec because the earlier reverberation that was much stronger than the later one was effectively reduced. This can be easily confirmed by listening to dereverberated signals, which are available on our web page [18].

Figure 11.6 shows waveforms and spectrograms of a source signal, an observed reverberant signal, and a signal dereverberated by the prototype system. The source signal was a Japanese word “Ha-Chi-Ga-Tsu” uttered by a female speaker. The reverberation time was 1.0 sec. It shows that HERB could effectively restore the time and frequency structure of the source signal.

11.6.3 Speaker Dependent Word Recognition Rate

We evaluated the speaker dependent word recognition rate (WRR) of speech signals dereverberated by the prototype system. For this purpose, we prepared four types of acoustic monophone model. The first two are models of



(a) Source signal

(b) Observed reverberant signal

(c) Dereverberated signal

Fig. 11.6. Waveforms (left panels) and spectrograms (right panels) of (a) a source signal, (b) an observed signal, and (c) a dereverberated signal, for the utterance "Ha-Chi-Ga-Tsu." (Reverberation time: 1.0 sec.)

clean speech signals. They are trained on the signals before and after being dereverberated by the prototype system, and are referred to as clean models. The remaining two are models of reverberant signals with a 0.1 sec reverberation time. Similarly, they are trained on the signals before and after dereverberation, and are referred to as short reverberation models. Acoustic models trained on the signals before and after dereverberation are used to recognize reverberant signals and dereverberated signals, respectively. 4740 words randomly selected from 5240 words were used as training data, and the remaining 500 words were used as testing data. The analysis conditions we adopted consists of 12 order MFCCs, 12 order delta MFCCs, three state HMMs, five mixture Gaussian distributions, 25 msec frame length, and 5 msec frame shift.

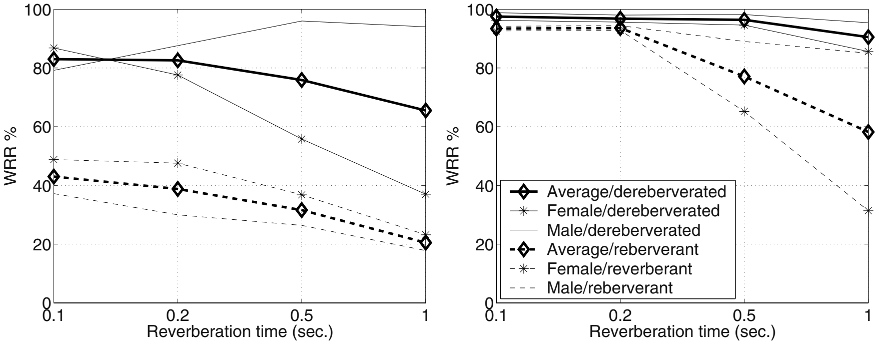


Fig. 11.7. Word recognition rates (WRRs) of reverberant and dereverberated signals when using the clean speech model (left panel) and the short reverberation model (right panel) under different reverberation time conditions.

The results are shown in Fig. 11.7. The left panel shows WRRs with the clean speech model. The average WRRs for the dereverberated signals (thick solid line) were much better than those for the reverberant signals (thick dotted line), but the WRRs were at most 55 %. By contrast, the right panel shows the WRRs with short reverberation models. The WRRs for dereverberated signals remained above 90 % even when the reverberation time was 1.0 sec while those of the reverberant signals degraded as the reverberation time exceeded 0.5 sec.

These results mean that speech signals dereverberated by HERB have similar spectral shapes independent of the reverberation time. In other words, HERB can successfully reduce the spectral variations in speech signals produced by reverberation without losing the speech features essential for ASR.

11.7 Future Directions

Although our simulation experiments showed the potential effectiveness of HERB, the prototype is still an immature system that cannot be applied to more realistic situations. For example,

- The prototype system requires a long observation time to achieve high quality speech dereverberation. In our experiments, a signal observed for more than one hour convolved with a constant impulse response is required for estimating a precise dereverberation filter.
- The speech model used in HERB does not precisely represent whole properties of speech signals, which leads to a bottleneck in the dereverberation performance. Non-harmonic components are disregarded simply as noise when estimating the dereverberation filter, and the F_0 of speech is assumed to be constant within a short time frame.

These problems have to be overcome if we are to apply HERB to a real problem.

In this section, we discuss extensions of HERB in order to consider its future directions.

11.7.1 Theoretical Extension of HERB

As a generalization of HERB, a quasi-periodicity based dereverberation principle has been proposed to provide a theoretical basis for discussing the limitation and extendibility of HERB [19]. With this principle, a harmonic sound, $s_h(n)$, in a speech signal is assumed to be a quasi-periodic signal that has the following features:

1. In each local time region around n_0 ($n_0 - \delta < n < n_0 + \delta$ for $^\forall n_0$), $s(n)$ is approximately a periodic signal whose period is $T(n_0)$.
2. Outside the region ($|n' - n_0| > \delta$), $s(n')$ is also a periodic signal within its neighboring time region, but it often has another period that is different from $T(n_0)$.

These features make the observed signal, $x(n)$ in (11.1), a non-periodic signal even within local time regions when the room impulse response, $h(n)$, contains non-zero values for $|m| > \delta$. This is because more than two periodic signals, $s(n)$ and $s(n-m)$, that have different periods, are added to $x(n)$ with weights of $h(0)$ and $h(m)$. Conversely, the goal of quasi-periodicity based dereverberation is to estimate $w(n)$ that makes $y(n)$ in (11.2) a periodic signal in each local time region. Once such a filter is obtained, $q(n) = \sum_{m=0}^{\infty} w(n-m)h(m)$ in (11.4) must have zero values for $|n| > \delta$, and thus, reverberant components longer than δ are eliminated from $y(n)$.

This principle suggested another dereverberation method based on a minimum mean squared error (MMSE) criterion [19]. This method, as described in Sect. 11.7.2, is expected to enable us to achieve a better estimation of the dereverberation filter with a small number of training data. This principle also explains why a reverberation with a short reverberation time remains after dereverberation by HERB. It suggests that a quasi-periodicity based dereverberation method cannot reduce the dereverberated impulse response, $q(n)$, in the time region for $|n| < \delta$. The results of the simulation experiments in Sect. 11.6 suggest that the δ value is about 0.1 sec.

11.7.2 Accuracy Improvement of Speech Model

The speech model presented in Sect. 11.3 contains two major problems as

1. It assumes the F_0 of a speech signal is constant within a short time frame.
2. It disregards non-harmonic components as a dereverberation key.

These problems constitute a bottleneck in the dereverberation performance of HERB, and so should be removed.

In order to model a harmonic signal precisely even when its F_0 changes within a short time frame, an extension of HERB is proposed, in which time warping analysis is incorporated into the adaptive harmonic filtering [20]. The time warping analysis expands and contracts the time axis of the signals to make their F_0s approximately constant. Once the F_0 of the signal is made constant, we can easily extract the precise features of the harmonic components based on harmonic filtering [21]. Simulation experiments showed that this extension improves the performance of HERB, especially for male speech signals. Consequently, it successfully reduced the energies of dereverberated impulse responses compared with the room impulse responses in all cases in almost all time regions for both genders. It was also shown that this extension improved the speaker dependent ASR so that it could achieve more than 90 % recognition rates using an acoustic model trained on clean speech signals, that is, the clean model, even with a 1.0 sec reverberation time.

It is a challenging problem to develop a speech model that can deal with non-harmonic components for dereverberation. We are investigating one approach to this problem, in which we assume that direct signals of harmonic components, $s_h(n)$ and non-harmonic components $s_n(n)$ are uncorrelated. This assumption can be formulated as

$$E_t\{(D(m)S_h(l,m))(D(m)S_n(l,m))\} = 0, \tag{11.30}$$

where $E_t\{\cdot\}$ is an expectation function over time frames. This can be rewritten using the desired dereverberation filter $W(l,m)$ as

$$E_t\{(D(m)S_h(l,m))(W(m)X(l,m) - D(m)S_h(l,m))\} = 0. \tag{11.31}$$

This implies that the solution to the above equation coincides with the filter $W(m)$ that minimizes the following cost function.

$$C(W(m)) = E_t\{(W(m)X(l,m) - D(m)S_h(l,m))^2\}. \tag{11.32}$$

In HERB, when the harmonic filter is applied to $X(l,m)$, the output, $\hat{X}(l,m)$, is assumed to become the initial approximation of the direct harmonic components. With this approximation, the above cost function can be further rewritten as

$$C(W(m)) = E_t\{(W(m)X(l,m) - \hat{X}(l,m))^2\}. \tag{11.33}$$

This cost function is referred to as the minimum mean squared error (MMSE) criterion. It is shown that dereverberation can be achieved by obtaining the dereverberation filter, $W(l,m)$, that minimizes the MMSE criterion [19]. In addition, our preliminary experiments showed that this method could achieve better dereverberation in terms of energy decay curves than the prototype system described in this chapter when the observed signal was short.

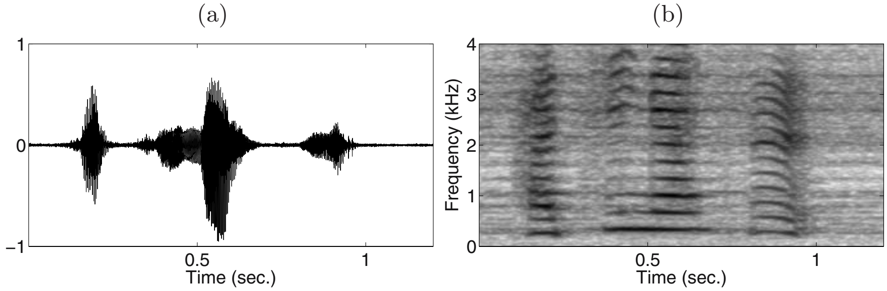


Fig. 11.8. (a) A waveform and (b) a spectrogram of a signal of the utterance "Ha-Chi-Ga-Tsu" dereverberated using a dereverberation filter trained on 100 word utterances. (Reverberation time: 1.0 sec.)

11.7.3 Reduction of Training Data Size

The reduction of the observed signal size required for achieving high quality speech dereverberation with HERB is one of the most important problems remaining to be solved. Figure 11.8 shows a waveform and a spectrogram of a dereverberated speech signal when the dereverberation filter was estimated using observed signals composed of 100 word utterances. It shows that the time and frequency structure of the speech signal can be effectively restored by the dereverberation. However, by looking into time and frequency regions where the energy of the signal is small in comparison with Fig. 11.6, we can confirm that stationary random noise remains after dereverberation. As the size of the observation decreases, the noise HERB generates by its dereverberation procedure becomes more intense. This problem must be overcome if we are to apply HERB to more realistic situations.

We think that one promising way to overcome this problem is to introduce a speech model that can deal with non-harmonic components as a key to dereverberation. The MMSE criterion described in Sect. 11.7.2 is one such model that may provide a better solution. Moreover certain noise reduction technologies may provide another way that can be expected to work well for improving the quality of dereverberation by reducing the random stationary noise generated by HERB.

11.8 Conclusions

This chapter described methodologies for single channel blind speech dereverberation. An important point as regards blind dereverberation is finding a way to exploit the features of source signals. We detailed a harmonicity based dereverberation approach known as HERB as a promising methodology. This approach estimates the inverse filter of a room impulse response by calculating the time average of a filter that transforms reverberant observed signals into their direct harmonic components, which are estimated using an adaptive harmonic filter. A prototype system of HERB was presented and shown to work effectively to achieve high quality dereverberation, provided a sufficient number of observed signals are available. The prototype system clearly restored the time and frequency structures, and improved the speaker dependent word recognition rates to more than 90 % even under a 1.0 sec reverberation time condition. Several future directions related to HERB were also presented with the aim of discussing how to extend it to cope with more realistic situations.

References

1. A. Baba, A. Lee, H. Saruwatari, and K. Shikano, "Speech recognition by reverberation adapted acoustic model," in *Proc. of ASJ General Meeting*, 2002, pp. 27–28.
2. M. Miyoshi and Y. Kaneda, "Inverse filtering of room acoustics," *IEEE Trans. ASSP*, vol. ASSP-36(2), pp. 145–152, 1988.
3. S. C. Douglas and X. Sun, "Convolutive blind separation of speech mixtures using the natural gradient," *Speech Communication*, vol. 39, pp. 65–78, 2003.
4. B. W. Gillespie and L. E. Atlas, "Strategies for improving audible quality and speech recognition accuracy of reverberant speech," in *Proc. IEEE ICASSP*, 2003, vol. 1, pp. 676–679.
5. B. Yegnanarayana and P. S. Murthy, "Enhancement of reverberant speech using LP residual signal," *IEEE Trans. SAP*, vol. 8, no. 3, pp. 267–281, 2000.
6. M. Unoki, M. Furukawa, K. Sakata, and M. Akagi, "A method based on the MTF concept for dereverberating the power envelope from the reverberant signal," in *Proc. IEEE ICASSP*, 2003, vol. 1, pp. 840–843.
7. T. Houtgast and H. J. M. Steenken, "A review of the MTF concept in room acoustics and its use for estimating speech intelligibility in audioria," *J. Acoust. Soc. Am.*, vol. 77, pp. 1069–1077, Mar. 1985.
8. A. S. Bregman, *Auditory Scene Analysis – The Perceptual Organization of Sound*. MIT Press, 1990.
9. T. Nakatani, *Computational Auditory Scene Analysis Based on Residue-Driven Architecture and its Application to Mixed Speech Recognition.* Ph.D. thesis, Dept. of Applied Analysis & Complex Dynamical Systems, Kyoto Univ., Mar. 2002.
10. M. Weintraub, "A computational model for separating two simultaneous talkers," in *Proc. IEEE ICASSP,* 1986, vol. II, pp. 81–84.
11. A. Nehorai and B. Porat, "Adaptive comb filtering for harmonic signal enhancement," *IEEE Trans. ASSP*, vol. ASSP-34, Oct. 1986.
12. T. Nakatani and T. Irino, "Robust fundamental frequency estimation against background noise and spectral distortion," in *Proc. IEEE ICSLP*, 2002, vol. 3, pp. 1733–1736.
13. T. Nakatani and T. Irino, "Robust and accurate fundamental frequency estimation based on dominant harmonic components," *J. Acoust. Soc. Am.*, to appear.
14. B. Yegnanarayana and B. S. Ramakrishna, "Intelligibility of speech under nonexponential decay conditions," *J. Acoust. Soc. Am.*, vol. 58, pp. 853–857, Oct. 1975.

15. A. V. Oppenheim, and R. W. Schafer, *Discrete-Time Signal Processing (2nd edition)*. Chapter 11, Prentice Hall, 1999.
16. T. Nakatani and M. Miyoshi, "Blind dereverberation of single channel speech signal based on harmonic structure," in *Proc. IEEE ICASSP*, 2003, vol. 1, pp. 92–95.
17. M. R. Schroeder, "New method of measuring reverberation time," *J. Acoust. Soc. Am.*, vol. 37, pp. 409–412, 1965.
18. http://www.kecl.ntt.co.jp/icl/signal/nakatani/sound-demos/dm/derev-demos.html
19. T. Nakatani, M. Miyoshi, and K. Kinoshita, "One microphone blind dereverberation based on quasi-periodicity of speech signals," in *Advances in Neural Information Processing Systems 16 (NIPS 16)*, MIT Press, 2004.
20. T. Nakatani, K. Kinoshita, M. Miyoshi, and P. S. olfaghari, "Harmonicity based monaural speech dereverberation with time warping and F_0 adaptive window," in *Proc. ICSLP*, 2004, vol. 2, pp. 873–876.
21. T. Abe and M. Honda, "Sinusoidal modeling based on instantaneous frequency attractors," in *Proc. IEEE ICASSP*, 2003, vol. 6, pp. 133–136.

12 Separation and Dereverberation of Speech Signals with Multiple Microphones

Yiteng (Arden) Huang[1], Jacob Benesty[2], and Jingdong Chen[1]

[1] Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07974, USA
E-mail: {arden, jingdong}@researchbell-labs.com
[2] Université du Québec, INRS-EMT
Montréal, QC H5A 1K6, Canada
E-mail: benesty@inrs-emt.uquebec.ca

Abstract. Speech enhancement was not and should not be examined solely with the tool of time-frequency analysis. Approaching this problem from different perspectives or incorporating other knowledges helps to expand the number of options open to us when developing a speech enhancement system. Using multiple microphones at different locations makes it possible to develop more sophisticated source separation and dereverberation technologies for speech enhancement, which enable man-made systems to extract a speech signal of interest in a noisy environment with competing speech and/or noise sources. This phenomenon is referred to as the cocktail party effect demonstrated by human beings and many other creatures with few efforts. However, separating and dereverberating speech signals is a very difficult problem in reverberant environments and the state-of-the-art algorithms are still unsatisfactory. The challenge lies in the coexistence of spatial interference from competing sources and temporal echoes due to room reverberation in the observed microphone signals. Focusing only on optimizing the signal-to-interference ratio is inadequate for most speech processing systems where source separation and speech dereverberation are two fully-integrated problems. In this chapter, we study these two problems in a unified framework. We deduce that spatial interference and temporal reverberation can be separated and a SIMO system with the speech signal of interest as input is extracted from the MIMO system. Furthermore, this interference-free SIMO system is dereverberated using the MINT theorem. Such a two-stage procedure leads to a novel sequential source separation and speech dereverberation algorithm based on blind multichannel identification. Simulations with measurements obtained in the varechoic chamber at Bell Labs verified the proposed algorithm.

12.1 Introduction

Speech enhancement is essential for tremendous applications of speech processing and communications since we are living in a natural environment where noise or disturbance is perpetual and ubiquitous. Speech signals can seldom be recorded in pure form and in most cases they are immersed in acoustic ambient noise or reverberation.

In order to develop an effective approach to extracting a desired speech signal from their corrupt observations, we need to understand how distortions are introduced. From a statistical viewpoint, there are only two sources of distortion. One is uncorrelated or even independent noise or competing speech, and the other is correlated reverberation or echo. While many single-channel algorithms and techniques have had varying success in noise reduction as explained in previous chapters, speech enhancement in the sense of separation and dereverberation would be very difficult if not impossible to accomplish using only one microphone in distance. A listener has the ability of choosing to focus on a specific speaker in a room where several people are talking concurrently and where noise sources might meanwhile exist. This phenomenon is referred to as the *cocktail party effect* or *attentional selectivity* [1]. This effect is mainly attributed to the fact that we have two ears and our perception of speech is based on binaural hearing, which can be easily demonstrated by observing the difference in understanding between using both ears and with either ear covered when listening in a cocktail-party-like environment. This suggests the use of two or more microphones, i.e., microphone arrays, in the development of prospect speech separation and dereverberation algorithms and systems.

As a part of our daily experience, we know that distinguishing and even separating components of a mixture or collection depends on their distinctions. In a multispeaker environment, sound sources are different in location and statistics in addition to spectrum, which leads to two different categories of speech separation method using multiple microphones: beamforming and blind source separation (BSS).

Beamforming is a form of spatial filtering that enhances the signal from "look direction" and attenuates signals that propagate from directions other than the "look direction" [2]. Therefore a beamformer can not only separate multiple sound sources but also suppress reverberation for the speech source of interest. However, its performance is limited by a number of factors in practice. Beamforming relies on the knowledge of the speaker's position, which is seldom available. While the position of the speaker can be estimated by analysis of the microphone outputs, errors are inevitable particularly when the room is considerably reverberant [3]. Furthermore, current microphone array technologies including beamforming originated from array signal processing. But compared to classical sensor array processing with antenna arrays [4], the basic conditions are significantly different in acoustics: speech is a baseband signal and the localization and recording take place in the nearfield with respect to the microphone array.

Alternatively BSS methods tackle this problem by taking advantage of the difference in statistics among multiple sound sources under investigation [5]. BSS that is typically accomplished by independent component analysis (ICA) algorithms [6] assumes mutually independent sound sources. The mixing procedure is typically delineated with a multiple-input multiple-output

(MIMO) mathematical model, which is either memoryless or with memory, being referred to as instantaneous and convolutive mixtures, respectively. An ICA processes microphone signals with a de-mixing system whose outputs are estimates of the separated source signals satisfying the independent assumption. Existing ICA algorithms differ in the way the dependence of the separated source signals is defined, i.e., the employed criteria for minimization, which include second order statistics [7], higher (than two) order statistics [8], and information-theory-based measures [9]. BSS methods allow for near-field sources and reverberant acoustic environments. But in reverberant environments, they are either very complex (for time-domain approaches [10]) or have an inherent problem of so-called permutation inconsistency [11] (for frequency-domain algorithms [12]). Moreover, it is not true nevertheless that current BSS methods work for arbitrary source positions. When sources are at the positions such that the mixing matrix is singular, the de-mixing system (the inverse of the mixing matrix) does not exist and source separation cannot be attained. Finally, it should be noted that, in addition to the above drawbacks, independent but distorted source signals are valid solutions for BSS methods. Therefore deconvolution is usually needed to mitigate convolutive distortion and reconstruct original speech signals.

Speech dereverberation remains a challenging problem even after three decades of continuous research. While the number of employed microphone signals is a common way to classify current speech dereverberation methods, another insightful approach is based on whether the channel impulse responses need to be known or estimated. In the case of a single microphone with the corresponding acoustic channel impulse response not being able to access anyway, either cepstral-domain processing techniques were suggested to separate speech from reverberation [13], [14] or characteristics of speech (usually in statistical forms) could be exploited with the attempt to recover the energy envelope of the original speech [15]. But they achieved only moderate successes because of a very large variety of applications. If the acoustic channel impulse responses are known, speech dereverberation can be performed by inverting those impulse responses. It is well known that the impulse response of a single acoustic channel needs to a minimum-phase sequence for stable and causal *exact* inversion [16]. Otherwise the inverting filter would either be IIR (noncausal and with a long delay) for exact inversion or just produce an LS (least-squares) solution. However, using multiple microphones, we can carry out perfect speech dereverberation with causal FIR filters even for non-minimum-phase channels. The principle is widely known as the MINT (multichannel inverse) theorem [17].

In this chapter, we will investigate the problem of speech enhancement by separating the speech of interest from concurrent interference (speech and/or noise) sources and by mitigating distortion due to room reverberation from a novel perspective within a unified framework using multiple microphones. In a MIMO acoustic system, microphone outputs are convolutive mixtures containing both reverberant speech and competing interference. We will show that the reverberant speech and interference can be completely separated given the blindly estimated channel impulse responses from interference sources. It is assumed that the number of microphones would be greater than the number of speech and interference sources. Then by choosing different combinations of microphone outputs, we obtain a number of diversely distorted speech signals, which composes a single-input multiple-output (SIMO) system. For such a system, we can again blindly identify its channel impulse responses and then apply the MINT theorem to remove reverberation. Therefore, the speech enhancement algorithm that will be developed here is a two-step procedure dealing with interference and reverberation sequentially. As a result, we are able to mimic the cocktail party effect with man-made machines.

This chapter is organized as follows. Section 12.2 introduces the MIMO signal model, formulates the problem of speech separation and dereverberation, and explains all assumptions that will be made. In Section 12.3, we brief the technique of blindly identifying a SIMO system. Section 12.4 explains how to separate reverberant speech and interference. A SIMO system with the speech of interest as the input will be extracted from the MIMO system. Since the SIMO is free of interference, we can again blindly identify its impulse responses and perform exact dereverberation using the MINT theorem, which will be illustrated in Section 12.5. Section 12.6 evaluates the developed algorithm by simulations and Section 12.7 draws the conclusions.

12.2 Signal Model and Problem Formulation

We consider an acoustic environment where there are one speech source of interest, $M-1$ concurrent sound sources, and N microphones with $M < N$. The speech source and $M-1$ other sound sources are mutually independent. Those competing sound sources can be speech or noise, and are regarded as interference. Such a system is mathematically described by an $M \times N$ MIMO FIR model as shown in Fig. 12.1. Without loss of generality, we label the speech source of interest as the first. At the n-th microphone and at the k-th sample time, we have:

$$x_n(k) = \sum_{m=1}^{M} \mathbf{h}_{nm}^T \mathbf{s}_m(k, L_h) + b_n(k), \tag{12.1}$$
$$k = 1, 2, \cdots, K, \quad n = 1, 2, \cdots, N,$$

where $(\cdot)^T$ denotes the transpose of a matrix or a vector,

$$\mathbf{h}_{nm} = \left[\begin{array}{cccc} h_{nm,0} & h_{nm,1} & \cdots & h_{nm,L_h-1} \end{array}\right]^T,$$
$$n = 1, 2, \cdots, N, \quad m = 1, 2, \cdots, M,$$

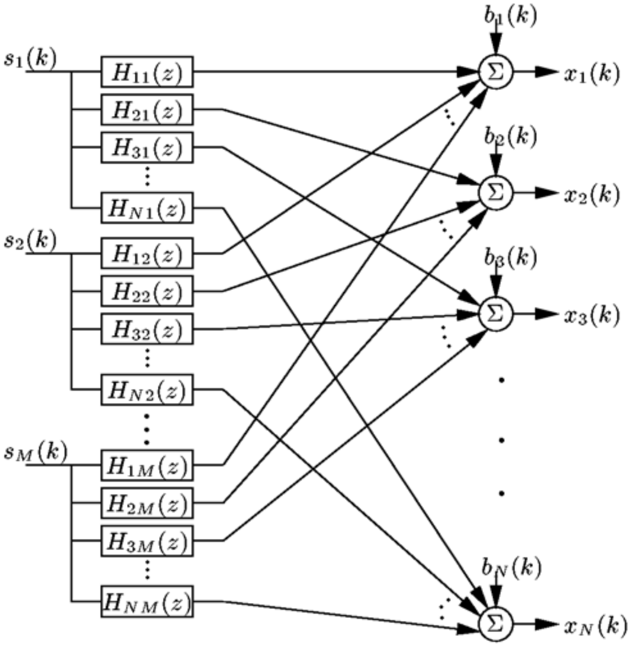


Fig. 12.1. Illustration of a MIMO FIR acoustic system having M sound sources and N microphones.

is the impulse response (of length L_h, $\forall m, n$) between source m and microphone n,

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

is a vector containing the last L_h samples of the m-th source signal s_m, and $b_n(k)$ is zero-mean additive white Gaussian noise (AWGN).

Using the z transform, the signal model of the MIMO system (12.1) is expressed as

$$X_n(z) = \sum_{m=1}^{M} H_{nm}(z) S_m(z) + B_n(z), \quad n = 1, 2, \cdots, N, \tag{12.2}$$

where

$$H_{nm}(z) = \sum_{l=0}^{L_h-1} h_{nm,l} z^{-l}. \tag{12.3}$$

In this system, we assume no *a priori* knowledge about the original speech signal $s_1(k)$, the interference signals $s_m(k)$ ($m = 2, \cdots, M$), or the channel impulse responses h_{nm}. All that we have are microphone outputs $x_n(k)$

($n = 1, 2, \cdots, N$). But the speech enhancement algorithm that will be developed needs to know the channel impulse responses from interference sources to each microphone h_{nm} ($m = 2, \cdots, M, \forall n$). Therefore we have to estimate them blindly. While blind MIMO identification is practically appealing, it still is a theoretically open problem and the current research in this area remains at the stage of feasibility investigation. This problem is already very difficult for communication systems with short channel impulse responses. Then for an acoustic system where the filter length of a channel impulse response in thousands of samples is not uncommon, blind MIMO identification seems formidable. Therefore we propose to decompose the problem into several subproblems in which SIMO systems are blindly identified. Presumably interference sources are motionless or move very slowly. Consequently their corresponding channel impulse responses change very slowly in time. It is further assumed that from time to time each interference source occupies at least one exclusive interval alone. Then during every single-talk interval, a SIMO system is blindly identified and its channel impulse responses are saved for later speech separation and dereverberation when all sources voice out simultaneously. Although these assumptions make the developed algorithm less flexible, they still are reasonable and stand in many practical scenarios. Apparently a sound source detection algorithm that distinguishes single and multiple talk is necessary and also interesting, but it is beyond the scope of this study.

In this chapter, we suppose that noise comes from one single point source or multiple point sources, and additive, dispersive noise is negligible, i.e., $b_n(k) = 0, \forall n, k$. Therefore, the blind SIMO identification system could yield accurate estimates of channel impulse responses and we can assure satisfactory performance for subsequent speech separation and dereverberation.

12.3 Blind Identification of a SIMO System

As assumed in the previous section, from time to time an interference source $s_m(k)$ ($m = 2, \cdots, M$) would alone occupy an exclusive interval, during which the MIMO system becomes a SIMO system and the corresponding channel impulse responses will be blindly estimated. In this section, we will briefly review the technique of blind SIMO identification and its adaptive implementations. In order to have a concise presentation and to keep consistent with the conventional notation used in the literature of blind SIMO identification, we omit the subscript indicating the source index m in and also *only* in this section, which we believe would cause no ambiguity if the reader could pay slightly more attention.

For a SIMO system, we have the following expression for microphone signals:

$$x_n(k) = h_n * s(k) + b_n(k), \quad n = 1, 2, \cdots, N, \tag{12.4}$$

where the symbol $*$ denotes the linear convolution operator and $b_n(k)$ can be neglected by assumption as explained in the previous section. In a vector/matrix form, such a signal model (12.4) becomes:

$$\mathbf{x}_n(k) = \mathbf{H}_n \cdot \mathbf{s}(k), \tag{12.5}$$

where

$$\begin{aligned}
\mathbf{x}_n(k) &= \left[\, x_n(k) \;\; x_n(k-1) \;\; \cdots \;\; x_n(k-L_h+1) \,\right]^T, \\
\mathbf{H}_n &= \begin{bmatrix} h_{n,0} & h_{n,1} & \cdots & h_{n,L_h-1} & 0 & \cdots & 0 \\ 0 & h_{n,0} & \cdots & h_{n,L_h-2} & h_{n,L_h-1} & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_{n,0} & h_{n,1} & \cdots & h_{n,L_h-1} \end{bmatrix}, \\
\mathbf{s}(k) &= \left[\, s(k) \;\; s(k-1) \;\; \cdots \;\; s(k-L_h+1) \;\; \cdots \;\; s(k-2L_h+2) \,\right]^T.
\end{aligned}$$

In order to ensure that the SIMO system can be blindly identified, the following two conditions (one on the channel diversity and the other on the input source signal) need to be met and are normally assumed in earlier studies as well as in this chapter [18]:

1. The polynomials formed from $\mathbf{h}_n = \left[h_{n,0}\; h_{n,1} \cdots h_{n,L_h-1}\right]^T$, $n = 1, 2, \cdots, N$, are co-prime, i.e., the channel transfer functions $H_n(z)$ do not share any common zeros;
2. The autocorrelation matrix $\mathbf{R}_{ss} = E\left\{\mathbf{s}(k)\mathbf{s}^T(k)\right\}$ of the source signal is of full rank (such that the SIMO system can be fully excited from a perspective of system identification), where $E\{\cdot\}$ denotes mathematical expectation.

The idea of blind SIMO identification using only second order statistics of the outputs was first proposed by Tong *et al.* [19] and now there are many different ways to explain the principle. We present here the one that we usually use in our research. It can be shown that the vector of channel impulse responses lies in the null space of a cross-correlation like matrix [20]:

$$\mathbf{R}_x \mathbf{h} = \mathbf{0}, \tag{12.6}$$

where

$$\begin{aligned}
\mathbf{R}_x &= \begin{bmatrix} \sum_{n\neq 1} \mathbf{R}_{x_n x_n} & -\mathbf{R}_{x_2 x_1} & \cdots & -\mathbf{R}_{x_N x_1} \\ -\mathbf{R}_{x_1 x_2} & \sum_{n\neq 2} \mathbf{R}_{x_n x_n} & \cdots & -\mathbf{R}_{x_N x_2} \\ \vdots & \vdots & \ddots & \vdots \\ -\mathbf{R}_{x_1 x_N} & -\mathbf{R}_{x_2 x_N} & \cdots & \sum_{n\neq N} \mathbf{R}_{x_n x_n} \end{bmatrix}, \\
\mathbf{R}_{x_i x_j} &= E\left\{\mathbf{x}_i(k)\mathbf{x}_j^T(k)\right\}, \quad i, j = 1, 2, \cdots, N, \\
\mathbf{h} &= \left[\, \mathbf{h}_1^T \;\; \mathbf{h}_2^T \;\; \cdots \;\; \mathbf{h}_N^T \,\right]^T.
\end{aligned}$$

If the SIMO system is blindly identifiable, Matrix $\mathbf{R}_x$ is rank deficient by 1 (in the absence of noise) and channel impulse responses can be uniquely determined from $\mathbf{R}_x$, which contains only the second-order statistics of the system outputs. When additive noise is present, $\mathbf{h}$ would be the eigenvector of $\mathbf{R}_x$ corresponding to its smallest eigenvalue.

To develop an adaptive implementation, a simple way is to take advantage of the cross relations among the outputs [21]. By following the fact that

$$x_i * h_j = s * h_i * h_j = x_j * h_i, \quad i, j = 1, 2, \cdots, N, \ i \neq j, \tag{12.7}$$

we have, in the absence of noise, the following cross relation at time k:

$$\mathbf{x}_i^T(k)\mathbf{h}_j = \mathbf{x}_j^T(k)\mathbf{h}_i, \quad i, j = 1, 2, \cdots, N, \ i \neq j. \tag{12.8}$$

When noise is present and/or the estimate of channel impulse responses is deviated from the true value, an *a priori* error signal is produced:

$$e_{ij}(k+1) = \mathbf{x}_i^T(k+1)\hat{\mathbf{h}}_j(k) - \mathbf{x}_j^T(k+1)\hat{\mathbf{h}}_i(k), \quad i, j = 1, 2, \cdots, N, \tag{12.9}$$

where $\hat{\mathbf{h}}_i(k)$ is the model filter for the i-th channel at time k. In order to avoid the trivial estimate of all zero elements, a unit-norm constraint is imposed on

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

leading to the normalized error signal

$$\epsilon_{ij}(k+1) = e_{ij}(k+1)/\|\hat{\mathbf{h}}(k)\|.$$

Accordingly, the cost function is formulated as:

$$J(k+1) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \epsilon_{ij}^2(k+1), \tag{12.10}$$

and the update equation of the multichannel LMS (MCLMS) algorithm is deduced as follows [21]:

$$\hat{\mathbf{h}}(k+1) = \hat{\mathbf{h}}(k) - \mu \nabla J(k+1), \tag{12.11}$$

where μ is a small positive step size,

$$\nabla J(k+1) = \frac{\partial J(k+1)}{\partial \hat{\mathbf{h}}(k)} = \frac{2\left[\tilde{\mathbf{R}}_x(k+1)\hat{\mathbf{h}}(k) - J(k+1)\hat{\mathbf{h}}(k)\right]}{\|\hat{\mathbf{h}}(k)\|^2}, \tag{12.12}$$

and

$$\tilde{\mathbf{R}}_x(k) = \begin{bmatrix} \sum_{n\neq 1} \tilde{\mathbf{R}}_{x_n x_n}(k) & -\tilde{\mathbf{R}}_{x_2 x_1}(k) & \cdots & -\tilde{\mathbf{R}}_{x_M x_1}(k) \\ -\tilde{\mathbf{R}}_{x_1 x_2}(k) & \sum_{n\neq 2} \tilde{\mathbf{R}}_{x_n x_n}(k) & \cdots & -\tilde{\mathbf{R}}_{x_N x_2}(k) \\ \vdots & \vdots & \ddots & \vdots \\ -\tilde{\mathbf{R}}_{x_1 x_N}(k) & -\tilde{\mathbf{R}}_{x_2 x_N}(k) & \cdots & \sum_{n\neq N} \tilde{\mathbf{R}}_{x_n x_n}(k) \end{bmatrix},$$

$$\tilde{\mathbf{R}}_{x_i x_j}(k) = \mathbf{x}_i(k)\mathbf{x}_j^T(k), \quad i, j = 1, 2, \cdots, N.$$

The idea of adaptive blind SIMO identification could be implemented in the frequency domain for computational efficiency and fast convergence [22]. The so-called unconstrained normalized multichannel frequency-domain LMS (UNMCFLMS) algorithm was shown to perform well with an acoustic system and will be employed in this chapter.

12.4 Separating Reverberant Speech and Concurrent Interference

In this section, we will explain how to extract reverberant speech from concurrent interfering sound sources. It is supposed that channel impulse responses corresponding to the interfering sound (speech or noise) sources have been blindly identified using the method developed in the previous section. The knowledge of these channel impulse responses is being used here to convert the $M \times N$ MIMO system into a SIMO system with the speech signal of interest as the sole input. The development begins with an example of the simplest 2×3 MIMO system and then extends to a general $M \times N$ case with $M < N$.

12.4.1 Example: Removing Interference Signals in a 2 × 3 MIMO Acoustic System

For a 2×3 MIMO acoustic system, interference signals can be cancelled by using two microphone outputs at a time. For instance, we can cancel the interference in $X_1(z)$ and $X_2(z)$ caused by $S_2(z)$ (note that Source 1 is supposed to be the speech source of interest) as follows:

$$\begin{aligned}
&X_1(z)H_{22}(z) - X_2(z)H_{12}(z) = \\
&\quad [H_{11}(z)H_{22}(z) - H_{21}(z)H_{12}(z)]\, S_1(z) + \\
&\qquad\qquad\qquad\qquad [H_{22}(z)B_1(z) - H_{12}(z)B_2(z)]\,, \quad (12.13)
\end{aligned}$$

where channel impulse responses $H_{12}(z)$ and $H_{22}(z)$ corresponding to Source 2 were blindly estimated ahead of time. Similarly, we can select different pair of microphone signals and obtain distinctive interference-free though distorted observations of $s_1(k)$. This procedure is visualized in Fig. 12.2 and will be described in a more systematic way in the following.

Let us consider the following equation:

$$\begin{aligned}
Y_p(z) &= H_{s_1,p1}(z)X_1(z) + H_{s_1,p2}(z)X_2(z) + H_{s_1,p3}(z)X_3(z) \\
&= \sum_{q=1}^{3} H_{s_1,pq}(z)X_q(z), \quad p = 1, 2, 3, \qquad (12.14)
\end{aligned}$$

where $H_{s_1,pp}(z) = 0$, $\forall p$. This means that (12.14) considers only two microphone signals for each p. The objective is to find the polynomials $H_{s_1,pq}(z)$,

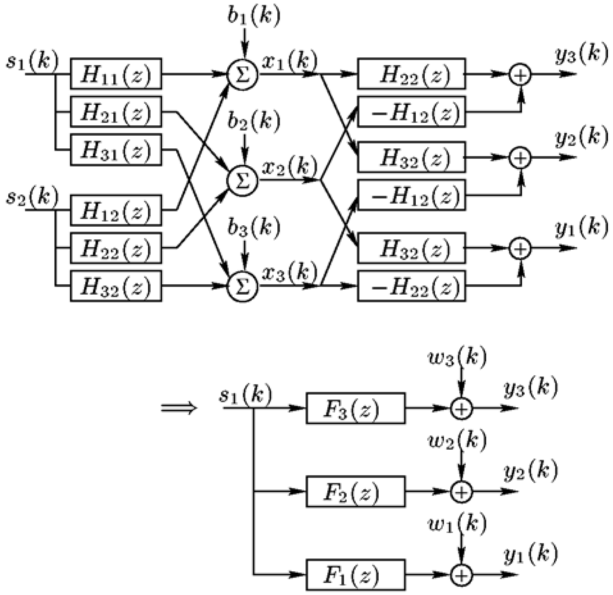


Fig. 12.2. Illustration of removing interference signals from a 2 × 3 MIMO acoustic system. Source 1 is the speech source of interest and Source 2 is an interference source as supposed.

$p, q = 1, 2, 3$, $p \neq q$, in such a way that:

$$Y_p(z) = F_p(z)S_1(z) + W_p(z), \quad p = 1, 2, 3, \tag{12.15}$$

which represents a SIMO system where s_1 is the source signal, $y_p(k), p = 1, 2, 3$, are the observed microphone signals, f_p are the corresponding acoustic channel impulse responses, and $w_p(k)$ is the noise at microphone p. If no dispersive noise is assumed, i.e., $b_n(k) = 0, \forall n, k$, then the noise component $w_p(k)$ is zero too.

Using (12.2) in (12.14), we deduce that:

$$\begin{aligned} Y_1(z) = {} & \left[H_{s_1,12}(z)H_{21}(z) + H_{s_1,13}(z)H_{31}(z)\right] S_1(z) + \\ & \left[H_{s_1,12}(z)H_{22}(z) + H_{s_1,13}(z)H_{32}(z)\right] S_2(z) + \\ & H_{s_1,12}(z)B_2(z) + H_{s_1,13}(z)B_3(z), \end{aligned} \tag{12.16}$$

$$\begin{aligned} Y_2(z) = {} & \left[H_{s_1,21}(z)H_{11}(z) + H_{s_1,23}(z)H_{31}(z)\right] S_1(z) + \\ & \left[H_{s_1,21}(z)H_{12}(z) + H_{s_1,23}(z)H_{32}(z)\right] S_2(z) + \\ & H_{s_1,21}(z)B_1(z) + H_{s_1,23}(z)B_3(z), \end{aligned} \tag{12.17}$$

$$\begin{aligned} Y_3(z) = {} & \left[H_{s_1,31}(z)H_{11}(z) + H_{s_1,32}(z)H_{21}(z)\right] S_1(z) + \\ & \left[H_{s_1,31}(z)H_{12}(z) + H_{s_1,32}(z)H_{22}(z)\right] S_2(z) + \\ & H_{s_1,31}(z)B_1(z) + H_{s_1,32}(z)B_2(z). \end{aligned} \tag{12.18}$$

As shown in Fig. 12.2, one possibility is to choose:

$$\begin{aligned}
&H_{s_1,12}(z) = H_{32}(z),\ \ H_{s_1,13}(z) = -H_{22}(z),\\
&H_{s_1,21}(z) = H_{32}(z),\ \ H_{s_1,23}(z) = -H_{12}(z),\\
&H_{s_1,31}(z) = H_{22}(z),\ \ H_{s_1,32}(z) = -H_{12}(z).
\end{aligned} \tag{12.19}$$

In this case, we find that:

$$\begin{aligned}
F_1(z) &= H_{32}(z)H_{21}(z) - H_{22}(z)H_{31}(z),\\
F_2(z) &= H_{32}(z)H_{11}(z) - H_{12}(z)H_{31}(z),\\
F_3(z) &= H_{22}(z)H_{11}(z) - H_{12}(z)H_{21}(z),
\end{aligned} \tag{12.20}$$

and

$$\begin{aligned}
W_1(z) &= H_{32}(z)B_2(z) - H_{22}(z)B_3(z),\\
W_2(z) &= H_{32}(z)B_1(z) - H_{12}(z)B_3(z),\\
W_3(z) &= H_{22}(z)B_1(z) - H_{12}(z)B_2(z).
\end{aligned} \tag{12.21}$$

Since $\deg\left[H_{nm}(z)\right] = L_h - 1$, where $\deg[\cdot]$ is the degree of a polynomial, therefore $\deg\left[F_p(z)\right] \leq 2L_h - 2$. We can see from (12.20) that polynomials $F_1(z)$, $F_2(z)$, and $F_3(z)$ share common zeros if $H_{12}(z)$, $H_{22}(z)$, and $H_{32}(z)$ [or if $H_{11}(z)$, $H_{21}(z)$, and $H_{31}(z)$] share common zeros.

Now suppose that $C_2(z) = \gcd\left[H_{12}(z), H_{22}(z), H_{32}(z)\right]$, where $\gcd[\cdot]$ denotes the greatest common divisor of the polynomials involved. We have:

$$H_{n2}(z) = C_2(z)H'_{n2}(z), \quad n = 1, 2, 3. \tag{12.22}$$

It is clear that the signal s_2 in (12.14) can be cancelled by using the polynomials $H'_{n2}(z)$ [instead of $H_{n2}(z)$ as given in (12.19)], so that the SIMO system represented by (12.15) will change to:

$$Y'_p(z) = F'_p(z)S_1(z) + W'_p(z), \ \ p = 1, 2, 3, \tag{12.23}$$

where

$$F'_p(z)C_2(z) = F_p(z), \quad W'_p(z)C_2(z) = W_p(z).$$

It should be pointed out that

$$\deg\left[F'_p(z)\right] \leq \deg\left[F_p(z)\right]$$

and that polynomials $F'_1(z)$, $F'_2(z)$, and $F'_3(z)$ share common zeros if and only if $H_{11}(z)$, $H_{21}(z)$, and $H_{31}(z)$ share common zeros.

12.4.2 Generalization

The approach to extracting reverberant speech from interference signals explained in the previous subsection on a simple example will be generalized here to an (M, N) MIMO acoustic system with $M < N$. We begin with writing (12.2) into a vector/matrix form

$$\vec{\mathbf{X}}(z) = \mathbf{H}(z)\vec{\mathbf{S}}(z) + \vec{\mathbf{B}}(z), \tag{12.24}$$

where

$$\begin{aligned}
\vec{\mathbf{X}}(z) &= \left[\, X_1(z) \;\; X_2(z) \;\; \cdots \;\; X_N(z) \,\right]^T, \\
\mathbf{H}(z) &= \begin{bmatrix} H_{11}(z) & H_{12}(z) & \cdots & H_{1M}(z) \\ H_{21}(z) & H_{22}(z) & \cdots & H_{2M}(z) \\ \vdots & \vdots & \ddots & \vdots \\ H_{N1}(z) & H_{N2}(z) & \cdots & H_{NM}(z) \end{bmatrix}, \\
\vec{\mathbf{S}}(z) &= \left[\, S_1(z) \;\; S_2(z) \;\; \cdots \;\; S_M(z) \,\right]^T, \\
\vec{\mathbf{B}}(z) &= \left[\, B_1(z) \;\; B_2(z) \;\; \cdots \;\; B_N(z) \,\right]^T.
\end{aligned}$$

If $C_m(z) = \gcd\left[H_{1m}(z), H_{2m}(z), \cdots, H_{Nm}(z)\right]$ $(m = 1, 2, \cdots, M)$, then $H_{nm}(z) = C_m(z)H'_{nm}(z)$ and the channel matrix $\mathbf{H}(z)$ can be rewritten as

$$\mathbf{H}(z) = \mathbf{H}'(z)\mathbf{C}(z), \tag{12.25}$$

where $\mathbf{H}'(z)$ is an $N \times M$ matrix containing the elements $H'_{nm}(z)$ and $\mathbf{C}(z)$ is an $M \times M$ diagonal matrix with $C_m(z)$ as its nonzero components.

Let us choose M from N microphone outputs and we have $P = C_N^M$ different ways of doing so. For the p-th $(p = 1, 2, \cdots, P)$ combination, we denote the index of the M selected microphone signals as p_m, $m = 1, 2, \cdots, M$, and get an $M \times M$ MIMO sub-system.

Consider the following equations:

$$Y_p(z) = \vec{\mathbf{H}}_{s_1,p}^T(z)\vec{\mathbf{X}}_p(z), \quad p = 1, 2, \cdots, P, \tag{12.26}$$

where

$$\begin{aligned}
\vec{\mathbf{H}}_{s_1,p}(z) &= \left[\, H_{s_1,p1}(z) \;\; H_{s_1,p2}(z) \;\; \cdots \;\; H_{s_1,pM}(z) \,\right]^T, \\
\vec{\mathbf{X}}_p(z) &= \left[\, X_{p_1}(z) \;\; X_{p_2}(z) \;\; \cdots \;\; X_{p_M}(z) \,\right]^T.
\end{aligned}$$

Let $\mathbf{H}_p(z)$ be the $M \times M$ matrix obtained from the system's channel matrix $\mathbf{H}(z)$ by keeping its rows corresponding to the M selected microphone signals. Then similar to (12.24), we have

$$\vec{\mathbf{X}}_p(z) = \mathbf{H}_p(z)\vec{\mathbf{S}}(z) + \vec{\mathbf{B}}_p(z), \tag{12.27}$$

where

$$\vec{\mathbf{B}}_p(z) = \left[\, B_{p_1}(z) \;\; B_{p_2}(z) \;\; \cdots \;\; B_{p_M}(z) \,\right]^T.$$

Substituting (12.27) into (12.26) yields

$$Y_p(z) = \vec{\mathbf{H}}_{s_1,p}^T(z)\mathbf{H}_p(z)\vec{\mathbf{S}}(z) + \vec{\mathbf{H}}_{s_1,p}^T(z)\vec{\mathbf{B}}_p(z). \tag{12.28}$$

In order to remove the interference from other competing speech or noise sources, the objective here is to find the vector $\vec{\mathbf{H}}_{s_1,p}(z)$ whose components are linear combinations of $H_{nm}(z)$ ($m = 2, 3, \cdots, M$, $n = 1, 2, \cdots N$) such that

$$\boldsymbol{\phi}_p^T(z) \triangleq \vec{\mathbf{H}}_{s_1,p}^T(z)\mathbf{H}_p(z) = \left[\, F_p(z) \;\; 0 \;\; \cdots \;\; 0 \,\right]. \tag{12.29}$$

Consequently, we have

$$Y_p(z) = F_p(z)S_1(z) + W_p(z), \tag{12.30}$$

where

$$W_p(z) = \vec{\mathbf{H}}_{s_1,p}^T(z)\vec{\mathbf{B}}_p(z).$$

If $\mathbf{C}_p(z)$ [obtained from $\mathbf{C}(z)$ in a similar way as $\mathbf{H}_p(z)$ is constructed] is not equal to the identity matrix, then $\mathbf{H}_p(z) = \mathbf{H}'_p(z)\mathbf{C}_p(z)$, where $\mathbf{H}'_p(z)$ has full column normal rank in acoustic environments as we assume in this chapter[1] (i.e. nrank $\left[\mathbf{H}'_p(z)\right] = M$, see [23] for a definition of normal rank), and the interference-free observations of $s_1(k)$ are determined as follows

$$Y'_p(z) = \vec{\mathbf{H}}'^{\,T}_{s_1,p}(z)\mathbf{H}'_p(z)\mathbf{C}_p(z)\mathbf{S}(z) + \vec{\mathbf{H}}'^{\,T}_{s_1,p}(z)\vec{\mathbf{B}}_p(z). \tag{12.31}$$

The filter vector $\vec{\mathbf{H}}'_{s_1,p}(z)$ is chosen in a way such that

$$Y'_p(z) = F'_p(z)S_1(z) + W'_p(z). \tag{12.32}$$

Obviously a good choice is to let the i-th element of $\vec{\mathbf{H}}'_{s_1,p}(z)$ be the $(i, 1)$-th cofactor[2] of $\mathbf{H}'_p(z)$. Consequently, the polynomial $F'_p(z)$ would be the determinant of $\mathbf{H}'_p(z)$. Note that the $(i, 1)$-th cofactor of $\mathbf{H}'_p(z)$ is only a linear combination of $H_{nm}(z)$ or $H'_{nm}(z)$ ($m = 2, 3, \cdots, M$, $n = 1, 2, \cdots N$). Therefore even though the channel impulse responses corresponding to the speech signal of interest $s_1(k)$ are not known or at least have not yet been blindly identified, we still are able to separate the reverberant speech from concurrent interference.

Since

$$\begin{aligned} F'_p(z) &= \vec{\mathbf{H}}'^{\,T}_{s,p}(z)\vec{\mathbf{H}}'_{p,:1}(z) \\ &= \sum_{q=1}^{M} H'_{s_1,pq}(z)H_{p,q1}(z), \end{aligned} \tag{12.33}$$

[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.

[2] The (i, j)-th cofactor c_{ij} of a matrix $\mathbf{A}$ is a signed version of $\mathbf{A}$'s minor d_{ij}:

$$c_{ij} \triangleq (-1)^{i+j} d_{ij},$$

where the minor d_{ij} is the determinant of a reduced matrix that is formed by omitting the i-th row and j-th column of the matrix A.

where $\vec{\mathbf{H}}_{p,:1}(z)$ is the first column vector of $\mathbf{H}_p(z)$ and $H'_{s_1,pq}(z)$ $(q = 1, 2, \cdots, M)$ are co-prime. It is clear that the polynomials $F'_p(z)$ $(p = 1, 2, \cdots, P)$ share common zeros if and only if the polynomials $H_{n1}(z)$ $(n = 1, 2, \cdots, N)$ share common zeros. Therefore, if the channels with respect to any one input are co-prime for the (M, N) MIMO acoustic system, we can remove interference from the reverberant speech of interest and obtain a SIMO system whose C_N^M channels are also co-prime.

Also, it can easily be check that $\deg\left[F_p(z)\right] \leq M(L_h - 1)$ (or $\deg\left[F'_p(z)\right] \leq M(L_h - 1)$). As a result, the length of the FIR filter f_p (or f'_p) would be

$$L_f \leq M(L_h - 1) + 1. \tag{12.34}$$

12.5 Speech Dereverberation

In the above, we showed that reverberant speech and competing interference could be separated given that the channel impulse responses corresponding to interference sources have been blindly identified. After this processing, we obtained a SIMO system with the speech signal of interest as the input. Although source separation has been achieved, the obtained multiple interference-free speech signals would sound possibly more reverberant due to the prolonged impulse response of the equivalent channels. In this section, we will illustrate how to perfectly remove those annoying reverberation and how to recover the original speech signal from the SIMO system. Here the assumption that the channel impulse responses $H_{nm}(z)$, $\forall m$ $(n = 1, 2, \cdots, N)$ are co-prime (i.e., the MIMO system is irreducible) needs to be employed to blindly identify the SIMO system first and then to perform speech dereverberation by using the MINT theorem. Therefore, the outputs of the SIMO system are given by (12.30).

12.5.1 Principle

For the SIMO system with respect to the source of interest s_1, we intend to apply the MINT theorem (also called the Bezout theorem in the mathematic literature). Let's consider the polynomials $G_p(z)$ $(p = 1,\ 2,\ \cdots, P)$ and the equation:

$$\begin{aligned}\widehat{S}_1(z) &= \sum_{p=1}^{P} G_p(z) Yp(z) \\ &= \left[\sum_{p=1}^{P} Fp(z) G_p(z)\right] S_1(z) + \sum_{p=1}^{P} G_p(z) Wp(z).\end{aligned} \tag{12.35}$$

The polynomials $G_p(z)$ should be found in such a way that $\widehat{S}_1(z) = S_1(z)$ in the absence of noise by using the Bezout theorem which is mathematically expressed as follows:

$$\begin{aligned}
&\gcd\left[F_1(z), F_2(z), \cdots, F_P(z)\right] = 1 \\
&\quad \Leftrightarrow \exists\; G_1(z), G_2(z), \cdots, G_P(z) \;:\; \sum_{p=1}^{P} F_p(z) G_p(z) = 1.
\end{aligned} \tag{12.36}$$

In other words, if the polynomials $F_p(z)$ $(p = 1, 2, \cdots, P)$ have no common zeros (which is equivalent to saying that the MIMO system is irreducible), it is possible to perfectly equalize (in the noiseless case) the SIMO system. The MINT theorem relieves the constraint on a single-channel acoustic system for perfect dereverberation that the channel impulse response must be a minimum-phase polynomial.

To find the dereverberation filters $G_p(z)$, we need to know the channel impulse responses $F_p(z)$. Since the MIMO system's channel impulse responses $H_{nm}(z)$, $\forall m$, $(n = 1, 2, \cdots, N)$ do not share common zeros as assumed in this chapter, the channel impulse responses f_p are co-prime as well such that they can be blindly identified again using the adaptive algorithms presented in Section 12.3. Starting from this point, we suppose that f_p's are known and we make no difference between f_p and its estimate.

Let's write the Bezout equation (12.36) in the time domain as follows:

$$\mathbf{F}_{\mathrm{c}}\mathbf{g} = \sum_{p=1}^{P} \mathbf{F}_{\mathrm{c},p}\mathbf{g}_p = \mathbf{e}_1, \tag{12.37}$$

where

$$\begin{aligned}
\mathbf{F}_{\mathrm{c}} &= \left[\; \mathbf{F}_{\mathrm{c},1} \;\; \mathbf{F}_{\mathrm{c},2} \;\; \cdots \;\; \mathbf{F}_{\mathrm{c},P} \;\right], \\
\mathbf{g} &= \left[\; \mathbf{g}_1^T \;\; \mathbf{g}_2^T \;\; \cdots \;\; \mathbf{g}_P^T \;\right]^T, \\
\mathbf{g}_p &= \left[\; g_{p,0} \;\; g_{p,1} \;\; \cdots \;\; g_{p,L_g-1} \;\right]^T, \\
& \quad p = 1, 2, \cdots, P,
\end{aligned}$$

L_g is the length of the FIR filter g_p,

$$\mathbf{F}_{\mathrm{c},p} = \begin{bmatrix}
f_{p,0} & 0 & \cdots & 0 \\
f_{p,1} & f_{p,0} & \cdots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
f_{p,L_f-1} & \cdots & \cdots & \vdots \\
0 & f_{p,L_f-1} & \ddots & \vdots \\
\vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & f_{p,L_f-1}
\end{bmatrix}$$

is an $(L_f + L_g - 1) \times L_g$ matrix, L_f is the length of the FIR filter f_p, and

$$\mathbf{e}_1 = \left[\; 1 \;\; 0 \;\; \cdots \;\; 0 \;\right]^T$$

is an $(L_f + L_g - 1) \times 1$ vector. In order to have a unique solution for (12.37), L_g must be chosen in such a way that $\mathbf{F}_\mathrm{c}$ is a square matrix. In this case, we have:

$$L_g = \frac{L_f - 1}{P - 1}. \tag{12.38}$$

Using (12.34), the length of the dereverberation filter is bounded by

$$L_g \leq \frac{M(L_h - 1)}{P - 1}. \tag{12.39}$$

12.5.2 The Least-Squares Implementation

It is now clear that by using the Bezout theorem the SIMO system with respect to the speech source of interest can be perfected dereverberated as long as their channel impulse responses share no common zeros. In addition, we derived what is the minimum length L_g of the dereverberation filters, as given in (12.39). Although finding the shortest dereverberation filters involves the lowest computational complexity and leads to the most cost effective implementation, the performance may not be the best due to noise in practice and error in the estimates of the channel impulse responses. Moreover, the smallest L_g may not be even possible since (12.38) does not guarantee an integer solution. Therefore, we choose a larger L_g than necessary in our implementation and solve (12.37) for $\mathbf{g}$ in the least squares sense:

$$\mathbf{g}_\mathrm{LS} = \mathbf{F}_\mathrm{c}^\dagger \mathbf{e}_1, \tag{12.40}$$

where

$$\mathbf{F}_\mathrm{c}^\dagger = \left(\mathbf{F}_\mathrm{c}^T \mathbf{F}_\mathrm{c}\right)^{-1} \mathbf{F}_\mathrm{c}^T$$

is the pseudo-inverse of the matrix $\mathbf{F}_\mathrm{c}$. If a decision delay d is taken into account, then the dereverberation filters turn out to be

$$\mathbf{g}_\mathrm{LS} = \mathbf{F}_\mathrm{c}^\dagger \mathbf{e}_d, \tag{12.41}$$

where

$$\mathbf{e}_d = \left[\begin{array}{ccc} \underbrace{0 \ \cdots \ 0}_{d} & 1 & \underbrace{0 \ \cdots \ 0}_{L_f + L_g - d - 2} \end{array}\right]^T.$$

Performing speech dereverberation based on the MINT theorem is sensitive to errors in the estimated channel impulse responses. In our research, we found that the performance of speech dereverberation would vary with the value of the decision delay d when a blind method has some difficulties to accurately identify the channels. Since this still is an open research problem, in our simulations, we either choose a fixed delay or search for the delay that produces the best speech dereverberation performance in the neighborhood of a pre-specified decision delay.

12.6 Simulations

In this section, we will evaluate the performance of the proposed blind source separation and speech dereverberation algorithm via simulations in realistic acoustic environments.

12.6.1 Performance Measures

Similar to what was adopted in our earlier study [22], we will use the normalized projection misalignment (NPM) to evaluate the performance of a BCI algorithm [24]. The NPM is defined as:

$$\mathrm{NPM} \triangleq 20 \log_{10} \left[\frac{\|\boldsymbol{\epsilon}\|}{\|\mathbf{h}\|} \right], \tag{12.42}$$

where

$$\boldsymbol{\epsilon} = \mathbf{h} - \frac{\mathbf{h}^T \hat{\mathbf{h}}}{\hat{\mathbf{h}}^T \hat{\mathbf{h}}} \hat{\mathbf{h}}$$

is the *projection misalignment* vector. By projecting $\mathbf{h}$ onto $\hat{\mathbf{h}}$ and defining a projection error, we take into account only the intrinsic misalignment of the channel estimate, disregarding an arbitrary gain factor.

To evaluate the performance of source separation and speech dereverberation, two measures, namely signal-to-interference ratio (SIR) and speech spectral distortion, are used in the simulations. For the SIR, we referred to the notion given in [25] but defined the measure in a different manner since their definition is applicable only for an $M \times M$ MIMO system. In this paper, our interest is in the more general $M \times N$ MIMO systems with $M < N$. Moreover, the M sources are equally important in [25] while here the first source is the speech source of interest and is more important than others.

Since only the first speech source is what we are interested in extracting, the SIR would be defined in a way where a component contributed by $s_1(k)$ is treated as the signal and the rest as the interference. We first define the input SIR at microphone n as:

$$\mathrm{SIR}_n^{\mathrm{in}} \triangleq \frac{E\left\{[h_{n1} * s_1(k)]^2\right\}}{\sum_{i=2}^{M} E\left\{[h_{ni} * s_i(k)]^2\right\}}, \quad n = 1, 2, \cdots, N, \tag{12.43}$$

where $*$ denotes linear convolution. Then the overall average input SIR is given by:

$$\mathrm{SIR}^{\mathrm{in}} \triangleq \frac{1}{N} \sum_{n=1}^{N} \mathrm{SIR}_n^{\mathrm{in}}. \tag{12.44}$$

The output SIR is defined using the same principle but the expression will be more complicated. For a concise presentation, we denote ϕ_{p,s_i} ($p =$

$1, 2, \cdots, P$, $i = 1, 2, \cdots, M$) as the impulse response of the equivalent channel from the i-th source $s_i(k)$ to the output $y_p(k)$ for the p-th $M \times M$ separation subsystem. From (12.28) and (12.29), we know that ϕ_{p,s_i} corresponds to the i-th element of $\boldsymbol{\phi}_p(z)$ and $\phi_{p,s_1} = f_p$. Then the average output SIR for the p-th subsystem is:

$$\mathrm{SIR}_p^{\mathrm{out}} \triangleq \frac{E\left\{[f_p * s_1(k)]^2\right\}}{\sum_{i=2}^{M} E\left\{[\phi_{p,s_i} * s_i(k)]^2\right\}}, \quad p = 1, 2, \cdots, P. \tag{12.45}$$

Finally, the overall average output SIR is found as:

$$\mathrm{SIR}^{\mathrm{out}} \triangleq \frac{1}{P} \sum_{p=1}^{P} \mathrm{SIR}_p^{\mathrm{out}}. \tag{12.46}$$

To assess the quality of dereverberated speech signals, we employed the Itakura-Saito (IS) distortion measure [26], which is the ratio of the residual energies produced by the original speech when inverse filtered using the LP coefficients derived from the original and processed speech. Let $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_t'$ be the LP coefficient vectors of an original speech signal frame $\mathbf{s}_t$ and the corresponding processed speech signal frame $\mathbf{s}_t'$ under examination, respectively. Denote $\mathbf{R}_{tt}$ as the Toeplitz autocorrelation matrix of the original speech signal. Then the IS measure is given as:

$$d_{\mathrm{IS},t} = \frac{\boldsymbol{\alpha}_t'^{T} \mathbf{R}_{tt} \boldsymbol{\alpha}_t'^{T}}{\boldsymbol{\alpha}_t^{T} \mathbf{R}_{tt} \boldsymbol{\alpha}_t^{T}} - 1. \tag{12.47}$$

Such a measure is calculated on a frame-by-frame basis. For the whole sequence of two speech signals, the mean IS measure is obtained by averaging $d_{\mathrm{IS},t}$ over all frames. According to [27], the IS measure exhibits a high correlation (0.59) with subjective judgments, suggesting that the IS distance is a good objective measure of speech quality. It was reported in [28] that the difference in mean opinion score (MOS) between two processed speech signals would be less than 1.6 if their IS measure is less than 0.5 for various speech codecs. Many experiments in speech recognition show that if the IS measure is less than about 0.1, the two spectra that we compare are perceptually nearly identical.

In our simulations, IS measures are calculated at different points, after source separation and after speech dereverberation. After source separation, the IS measure is obtained by averaging the result with respect to each one of the P SIMO outputs $y_p(k)$ and is denoted by $d_{\mathrm{IS}}^{\mathrm{SS}}$. After speech dereverberation, the final IS measure is denoted by $d_{\mathrm{IS}}^{\mathrm{SD}}$.

12.6.2 Experimental Setup

The simulations were conducted with the impulse responses measured in the varechoic chamber at Bell Labs [29]. A diagram of the floor plan layout is

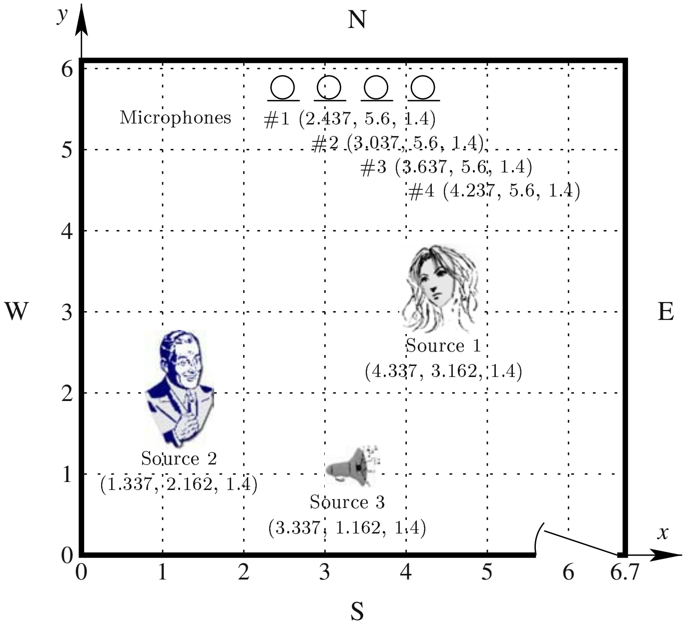


Fig. 12.3. Floor plan of the varechoic chamber at Bell Labs (coordinate values measured in meters).

shown in Fig. 12.3. 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 [30]. 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 [29], 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 in the 20-4000 Hz band. All four configurations were used in this paper for evaluating performance of the proposed algorithm.

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, four microphones and three speaker positions, which form a 3×4 MIMO system, were chosen and their locations are shown in Fig. 12.3. Signals were sampled at 8 kHz and the original impulse response measurements have 4096 samples. In the cases of 89% and 75% panels open, energy in reverberation decays quickly with arrival time and we cut impulse responses at $L_h = 256$. When 30% or none of planes are open, we set $L_h = 512$. Among the three sources, the first female speaker's speech is the target for extraction. The other two sources include one male speaker and one noise source. The two speech sources are equally loud in volume while the noise source is 5 dB weaker than the speech sources. For the noise source, we tried two different kinds of noise. One is car noise and the other is babbling noise recorded in the New York Stock Exchange (NYSE). The time sequence and spectrogram (30 Hz bandwidth) of these source signals for the first 1.5 seconds are shown in Fig. 12.4. From the spectrograms, we can tell that car noise is a low-pass signal while the bandwidth of babbling noise is much wider. From the perspective of system identification, babbling noise is more favorable than car noise as an exciting source signal. Silent periods were manually removed from the speech signals to make the BCI methods converge faster due to the reduced nonstationarity in the inputs and to make the average IS measures more meaningful with respect to speech only. This implies that in practice a voice activity detector needs to be used. After having source signals and channel impulse responses, we calculated microphone outputs by convolution.

As we expected, the performance of the proposed source separation and speech dereverberation algorithm would be greatly affected by the accuracy of the blindly estimated channel impulse responses. In the simulations, both adaptive (the UNMCFLMS algorithm) and batch (the SVD-based algorithm) implementations were investigated [22]. For the batch method, the empirical spatial covariance matrix was obtained over the first 1500 samples of the microphone captures. For source separation and speech dereverberation, speech signals of duration 10 seconds were utilized to assess the performance. The decision delay d in (12.41) was fixed as $3L_h/2 - 1$ in the cases of employing a batch method for BCI while its best value was searched in the neighborhood of $3L_h/2 - 1$ when an adaptive BCI algorithm was utilized.

12.6.3 Experimental Results

Table 12.1 summarizes the results of 16 experiments with different combination of room acoustics, BCI method, and type of noise. Figures 12.5 and 12.6 visualizes what was observed in the experiment with 89% of panels open, the UNMCFLMS algorithm employed for BCI, and car noise used as the third source. Figure 12.7 shows the results for the experiment with all panels closed, the batch method employed for BCI, and babbling noise in the NYSE used as the third source.

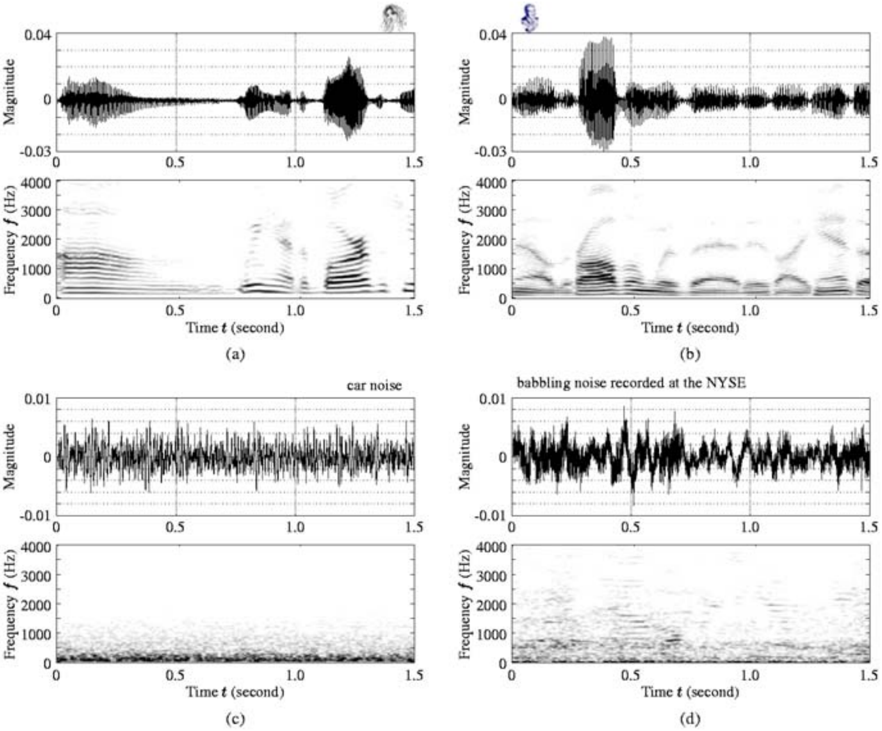


Fig. 12.4. Time sequence and spectrogram (30 Hz bandwidth) of the two speech source and two noise source signals used in the simulations for the first 1.5 seconds. (a) $s_1(k)$ (female speaker), (b) $s_2(k)$ (male speaker), (c) car noise, and (d) babbling noise recorded in the NYSE.

Let us first examine Table 12.1 and Fig. 12.5 for the accuracy of the channel impulse responses blindly estimated by the adaptive and batch BCI algorithms. It is clear that, given the same amount of microphone observations, the final projection misalignment error would be larger for the UNMCFLMS to identify a more reverberant SIMO system. Relatively, the batch method is more accurate and seems less dependent on L_h. After it collects microphone outputs for only 1500 samples (equivalently 0.1875 second), the batch BCI method can produce a reliable channel estimate with less than -60 dB NPM for SIMO systems with long channels of length $L_h = 512$. However, performing SVD of a $N \cdot L_h \times N \cdot L_h$ matrix in these simulations is too computationally intensive to be accomplished in real time by a commercial processor in the foreseeable near future. The reason why we carried out experiments with the batch BCI implementation and present here the results is to get an idea about what is the best possible performance of the proposed blind source separation

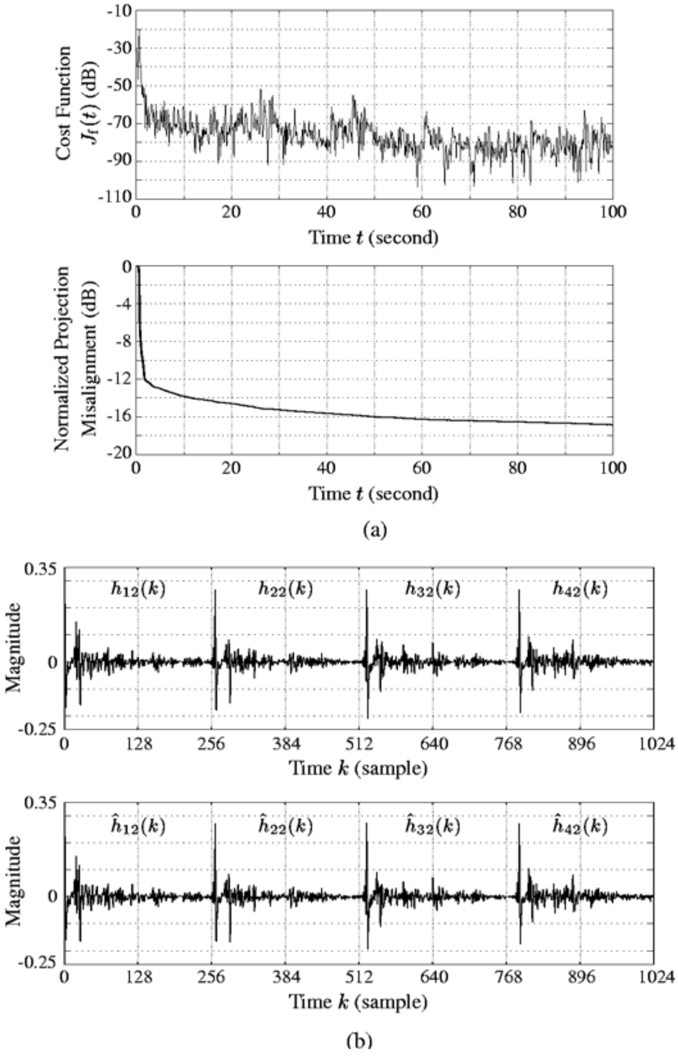


Fig. 12.5. Performance of the adaptive BCI (UNMCFLMS) algorithm with respect to the second source $s_2(k)$ in the varechoic chamber with 89% of panels open. (a) Running average (1000 samples) of the cost function and normalized projection misalignment, and (b) comparison of impulse responses between the actual channels and their estimates.

and speech dereverberation approach to speech enhancement with multiple microphones.

Figures 12.6 and 12.7 illustrate how the speech signal of interest is separated from other concurrent interference sources and how it is dereverberated.

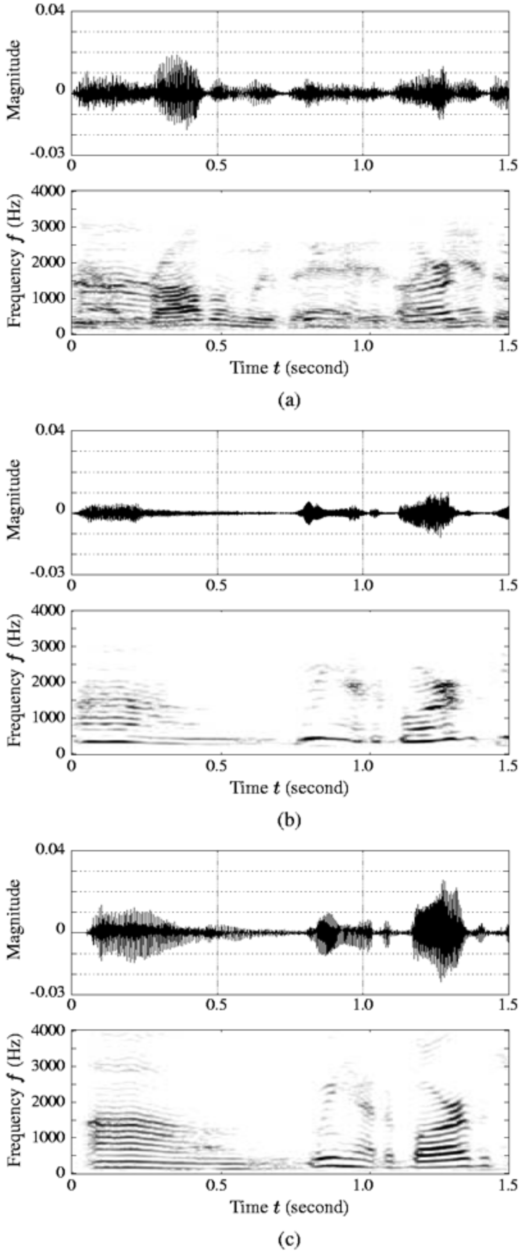
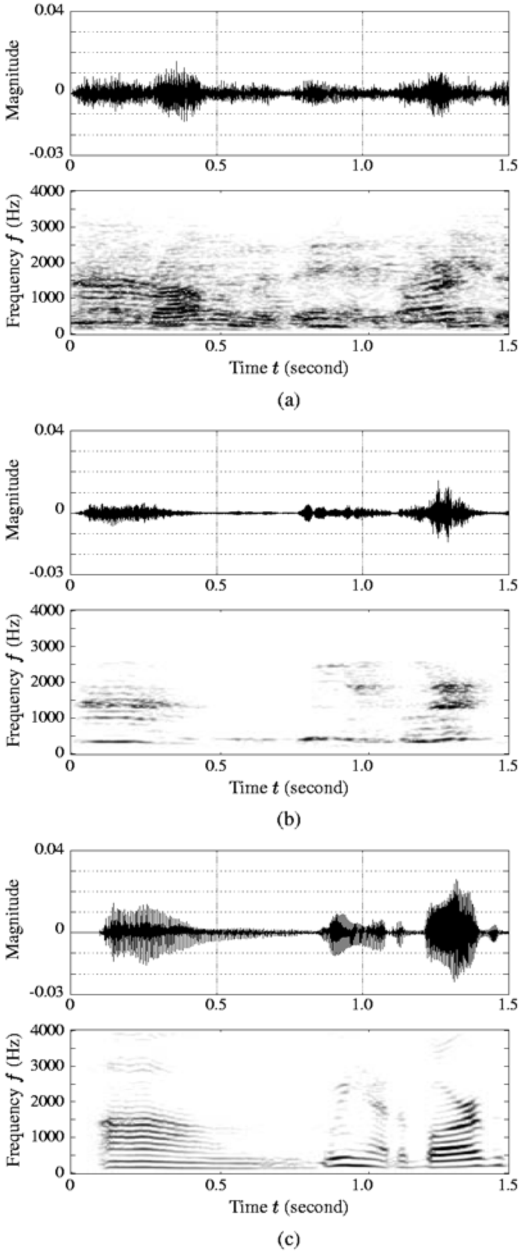


Fig. 12.6. Time sequence and spectrogram (30 Hz bandwidth) of (a) $x_1(k)$, (b) $y_1(k)$, and (c) $\hat{s}_1(k)$ for the experiment carried out in the varechoic chamber with 89% of panels open. In this experiment, $s_3(k)$ is car noise and the UNMCFLMS algorithm is used for BCI.



(a)

(b)

(c)

Fig. 12.7. Time sequence and spectrogram (30 Hz bandwidth) of (a) $x_1(k)$, (b) $y_1(k)$, and (c) $\hat{s}_1(k)$ for the experiment carried out in the varechoic chamber with all panels open. In this experiment, $s_3(k)$ is babbling noise recorded in the NYSE and the batch method is used for BCI.

Examining these figures together with the data in Table 12.1, we see that the output SIR's are very high (at least 41 dB) after source separation. Listening tests showed that these separated signals were certainly recognizable although they sounded more echoic as we expected. This can be justified by the spectrogram plots of $y_1(k)$ in Figs. 12.6(b) and 12.7(b). Apparently in periods of voiced speech on these narrow-band spectrograms, harmonics are vague, implying strong distortion which results in large IS measures (greater than at least 4.0). After dereverberation, the speech signal is satisfactorily recovered though delayed [clearly seen from time sequences of the recovered signal $\hat{s}_1(k)$ in these figures] with a relatively low IS measure. The accuracy of blindly identified channel impulse responses obviously has a great impact on the performance of the developed speech enhancement algorithm. But source separation and speech dereverberation are not equally affected by errors in the estimated channel impulse responses and the latter is more sensitive. When BCI is conducted with an adaptive algorithm, the NPM's are a lot lower than those obtained with a batch method. Although the final IS measures after speech dereverberation are significantly different particularly in more reverberant environments, their source separation performances in terms of output SIR are quite similar. Therefore it is our belief that using only SIR to evaluate a blind source separation (BSS) algorithm is inadequate if not misleading.

As explained before, the perceptual quality of distorted speech whose IS distance from its original signal is lower than 0.1 would not change with respect to either humans or an speech recognition system. The proposed algorithm incorporating the batch BCI method can surely deliver an enhanced speech signal reaching this level of voice quality. But the implementation based on an adaptive BCI algorithm can do so only when the room reverberation is low with 89% panels open. In reverberant environments, the adaptive BCI algorithm cannot produce highly accurate estimates of channel impulse responses such that the IS measures are still more than (though slightly) 0.1. As a matter of fact, it is imperative while challenging to develop accurate adaptive BCI algorithms for acoustic applications in reverberant environments. It is appealing that the recovered speech signal can attain high perceptual quality with an IS measure lower than 0.1. But in most applications of speech processing, this is an excessive and unnecessary if not practical requirement. What we observed in these simulations nevertheless show some promise of successful use of the proposed algorithm in prospect speech processing systems.

Table 12.1 Performance of the source separation and speech dereverberation algorithm based on the batch (SVD) and adaptive frequency-domain (AF) BCI implementations in the varechoic chamber at Bell Labs with different panel configurations.

		NPM (dB)			$\mathrm{SIR}^{\mathrm{in}}$	$\mathrm{SIR}^{\mathrm{out}}$		
Noise	BCI	SIMO_{s_1}	SIMO_{s_2}	SIMO_{s_3}	(dB)	(dB)	$d_{\mathrm{IS}}^{\mathrm{SS}}$	$d_{\mathrm{IS}}^{\mathrm{SD}}$
		89% panels open, $T_{60} = 240$ ms, $L_h = 256$						
Car	AF	-18.6361	-16.8234	-18.8090	1.3668	46.6966	4.4508	0.0449
	SVD	-84.6206	-110.3696	-152.6868	1.3668	47.6157	4.5653	0.0090
		75% panels open, $T_{60} = 310$ ms, $L_h = 256$						
	AF	-17.9231	-18.9300	-21.4186	2.3715	48.3984	5.3169	0.2389
	SVD	-109.6788	-100.6108	-187.8868	2.3715	48.8862	5.8647	0.0087
		30% panels open, $T_{60} = 380$ ms, $L_h = 512$						
	AF	-12.1323	-13.0353	-11.9475	1.3344	41.8391	5.8099	0.2609
	SVD	-67.0139	-106.2407	-167.2407	1.3344	43.5094	7.4319	0.0335
		Panels all closed, $T_{60} = 580$ ms, $L_h = 512$						
	AF	-12.5600	-13.5057	-14.3649	2.1065	44.1663	9.0386	0.2108
	SVD	-83.2605	-103.3190	-160.8024	2.1065	43.6628	11.1346	0.0198
NYSE	AF	-18.6361	-16.8234	-20.7545	0.9445	44.7547	4.4056	0.0668
	SVD	-84.6255	-110.3696	-176.5423	0.9445	45.2597	4.5653	0.0086
		75% panels open, $T_{60} = 310$ ms, $L_h = 256$						
	AF	-17.9231	-18.9300	-23.7211	1.8695	45.1628	5.4774	0.1920
	SVD	-100.3681	-114.6819	-184.1510	1.8694	44.9935	5.8647	0.0092
		30% panels open, $T_{60} = 380$ ms, $L_h = 512$						
	AF	-12.1323	-13.0353	-12.5460	0.8362	40.2743	5.6497	0.3215
	SVD	-79.5856	-93.7725	-174.1163	0.8362	41.4932	7.4319	0.0395
		Panels all closed, $T_{60} = 580$ ms, $L_h = 512$						
	AF	-12.5600	-13.5057	-16.8997	1.7245	42.2751	9.5378	0.1441
	SVD	-72.9542	-107.9821	-127.0545	1.7245	41.8808	11.1346	0.0192

NOTES: SIMO_{s_m} represents the SIMO system corresponding to source s_m.
T_{60} denotes 60-dB reverberation time in the 20-4000 Hz band.

12.7 Conclusions

Capturing a speech signal of interest among a number of competing sound sources in reverberant environments is difficult and a close-talking microphone is a common engineering solution to this problem. But in many speech communication systems, untethered voice access is demanded and speech enhancement in the sense of source separation and dereverberation must be performed. Existing blind source separation methods maximize solely the signal-to-interference ratio and possibly cause high distortion in their separated signals, which is neither pleasing to a listener nor can be used in following speech processing systems. We demonstrated in this chapter that spatial interference from competing sources and temporal echoes due to room reverberation can be perfectly separated and a SIMO system with the speech signal of interest as input is extracted from the MIMO system. The channel matrices of the interference-free SIMO system is irreducible given that the channels from the same source in the MIMO system share no common zeros. For such a SIMO system, the speech is then restored by using the MINT theorem. This derivation led to the proposal of a novel sequential source separation and speech dereverberation algorithm. We conducted experiments using real impulse responses measured in the varechoic chamber at Bell Labs. The results demonstrated the promise of the proposed algorithm.

References

1. 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.
2. B. D. Van Veen and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE ASSP Magazine*, vol. 5, pp. 4–24, Apr. 1988.
3. Y. Huang, J. Benesty, and G. W. Elko, "Source localization," in *Audio Signal Processing for Next-Generation Multimedia Communication Systems*, Y. Huang and J. Benesty, Eds., Boston, MA: Kluwer Academic, 2004.
4. B. Widrow, P. E. Mantley, L. J. Griffiths, and B. B. Goode, "Adaptive antenna systems," *Proc. of the IEEE*, vol. 55, pp. 2143–2159, Dec. 1967.
5. 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.
6. P. Comon, "Independent component analysis: a new concept?," *Signal Processing*, vol. 36, pp. 287–314, Apr. 1994.
7. L. Molgedey and H. G. Schuster, "Separation of a mixture of independent signals using time delayed correlations," *Phys. Rev. Lett.*, vol. 72, no. 23, pp. 3634–3637, June 1994.
8. J.-F. Cardoso, "Eigenstructure of the 4th-order cumulant tensor with application to the blind source separation problem," in *Proc. IEEE ICASSP*, 1989, pp. 2109–2112.
9. S. Amari, A. Cichocki, and H. H. Yang, "Blind signal separation and extraction: neural and information-theoretic approaches," in *Unsupervised Adaptive Filtering, Volume 1: Blind Source Separation*, S. Haykin, Ed., New York: John Wiley & Sons, 2000.
10. H. Wee and J. Principe, "A criterion for BSS based on simultaneous diagonalization of time correlation matrices," in *Proc. IEEE Workshop NNSP*, 1997, pp. 496–508.
11. M. Z. Ikram and D. R. Morgan, "Exploring permutation inconsistency in blind separation of speech signals in a reverberant environment," in *Proc. IEEE ICASSP*, 2000, vol. 2, pp. 1041–1044.

12. L. Parra and C. Spence, “Convolutive blind separation of non-stationary sources,” *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 320–327, May 2000.
13. D. Bees, M. Blostein, and P. Kabal, “Reverberant speech enhancement using cepstral processing,” in *Proc. IEEE ICASSP*, 1991, vol. 2, pp. 977–980.
14. S. Subramaniam, A. P. Petropulu, and C. Wendt, “Cepstrum-based deconvolution for speech dereverberation,” *IEEE Trans. Speech Audio Processing*, vol. 4, pp. 392–396, Sept. 1996.
15. T. Nakatani and M. Miyoshi, “Blind dereverberation of single channel speech based on harmonic structure,” in *Proc. IEEE ICASSP*, 2003, vol. I, pp. 92–95.
16. A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
17. M. Miyoshi and Y. Kaneda, “Inverse filtering of room acoustics,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 145–152, Feb. 1988.
18. G. Xu, H. Liu, L. Tong, and T. Kailath, “A least-squares approach to blind channel identification,” *IEEE Trans. Signal Processing*, vol. 43, pp. 2982–2993, Dec. 1995.
19. 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.
20. C. Avendano, J. Benesty, and D. R. Morgan, “A least squares component normalization approach to blind channel identification,” in *Proc. IEEE ICASSP*, 1999, vol. 4, pp. 1797–1800.
21. Y. Huang and J. Benesty, “Adaptive multi-channel least mean square and Newton algorithms for blind channel identification,” *Signal Processing*, vol. 82, pp. 1127–1138, Aug. 2002.
22. Y. Huang and J. Benesty, “A class of frequency-domain adaptive approaches to blind multi-channel identification,” *IEEE Trans. Signal Processing*, vol. 51, pp. 11–24, Jan. 2003.
23. P. P. Vaidyanathan, *Multirate Systems and Filter Bank*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
24. D. R. Morgan, J. Benesty, and M. M. Sondhi, “On the evaluation of estimated impulse responses,” *IEEE Signal Processing Lett.*, vol. 5, pp. 174–176, July 1998.
25. M. Z. Ikram and D. R. Morgan, “Exploring permutation inconsistency in blind separation of speech signals in a reverberant environment,” in *Proc. IEEE ICASSP*, 2000, vol. 2, pp. 1041–1044.
26. L. R. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
27. S. R. Quackenbush, T. P. Barnwell, M. A. Clements, *Objective Measures of Speech Quality*. Englewood Cliffs, NJ: Prentice-Hall, 1988.
28. G. Chen, S. N. Koh, I. Y. Soon, “Enhanced Itakura measure incorporating masking properties of human auditory system,” *Elsevier Science Signal Processing*, vol. 83, pp. 1445–1456, July 2003.
29. A. Härmä, “Acoustic measurement data from the varechoic chamber,” Technical Memorandum, Agere Systems, Nov. 2001.
30. 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.

13 Frequency-Domain Blind Source Separation

Hiroshi Sawada, Ryo Mukai, Shoko Araki, and Shoji Makino

NTT Communication Science Laboratories
Soraku-gun, Kyoto 619-0237, Japan
E-mail: {sawada, ryo, shoko, maki}@cslab.kecl.ntt.co.jp

Abstract. This chapter discusses the frequency-domain approach to the blind source separation (BSS) of convolutively mixed acoustic signals. In this approach, independent component analysis (ICA) is employed in each frequency bin to calculate the frequency responses of separation filters. Since convolutive mixtures in the time domain can be approximated as multiple instantaneous mixtures in the frequency domain, the advantage of this approach is that ICA is applied just for instantaneous mixtures, which is very simple. However, the permutation ambiguity of ICA solutions then becomes a problem. This chapter mainly deals with a method for solving the permutation problem. The method utilizes the source location information that can be estimated from the ICA solutions. We also discuss other important topics for frequency-domain BSS, such as complex-valued ICA, scaling alignment and spectral smoothing. To show the effectiveness of this frequency-domain approach, we report experimental results for separating up to four sources with a 4-element linear array, and also six sources with an 8-element planar array.

13.1 Introduction

Blind source separation (BSS) [1], [2] is a technique for estimating individual source components from their mixtures at sensors. The estimation is performed blindly, i.e. without possessing information about each source, such as its location and active time. Its potential audio signal applications include speech enhancement for speech recognition, teleconferences, and hearing aids. In such applications, signals are mixed in a convolutive manner with reverberations. This makes the BSS problem difficult. We need very long finite impulse response (FIR) filters (e.g. around a thousand taps for 8 kHz sampling) to separate acoustic signals mixed in such situations.

Independent component analysis (ICA) [3,4] is a major statistical tool for dealing with the BSS problem. If signals are mixed instantaneously, we can directly employ an instantaneous ICA algorithm to separate them. However, signals are mixed in a convolutive manner in the aforementioned applications. Therefore, we need to extend the ICA/BSS technique so that it is applicable to convolutive mixtures.

The first approach is time-domain BSS, where ICA is directly extended to the convolutive mixture model [5], [6], [7], [8], [9], [10]. It is theoretically sound and achieves good separation once an algorithm converges, since the algorithm correctly evaluates the independence of separated signals. However, an ICA algorithm for convolutive mixtures is not as simple as an ICA algorithm for instantaneous mixtures, and is computationally expensive for long FIR filters because it includes convolution operations.

The second approach is frequency-domain BSS, where complex-valued ICA for instantaneous mixtures is employed in each frequency bin [11–26]. The merit of this approach is that the ICA algorithm remains simple and can be performed separately at each frequency. Also, any complex-valued instantaneous ICA algorithm can be employed with this approach. The computational time for BSS can be reduced by employing a fast algorithm such as FastICA [27], [28], and/or by performing parallel computation for multiple frequency bins. However, the permutation ambiguity of the ICA solution becomes a serious problem. We need to align the permutation in each frequency bin so that a separated signal in the time domain contains frequency components from the same source. This problem is well known as the permutation problem of frequency-domain BSS [11–19,22–24], which is the main focus of this chapter.

Another problem relates to the circularity effect of discrete frequency representation. Frequency responses calculated in the frequency domain assume a periodic time-domain filter for their implementation. However, such a periodic filter is unrealistic, and we usually use its one-period realization for the separation filter. Therefore, the frequency responses should be smoothed so that the one-period realization does not rely on the circularity effect [16,26]. This chapter also discusses this problem.

The third approach uses both the time- and frequency-domains. In some time-domain BSS methods, convolutions in the time domain are speeded up by the overlap-save method in the frequency domain [9], [29]. There are also some methods [30], [31], [32] where filter coefficients are updated in the frequency domain but nonlinear functions for evaluating independence are applied in the time domain. The permutation problem does not occur in either case since the independence of separated signals is evaluated in the time domain. The circularity problem does not occur either with an appropriate constraint for filter coefficients [33] by such as rectangular windowing. However, the algorithm moves back and forth between the two domains at every iteration, spending non-negligible time on discrete Fourier transforms (DFTs) and inverse DFTs. Therefore, we consider that the permutation and circularity problems are inevitable if we hope to benefit from the merit of frequency-domain BSS mentioned above.

This chapter deals with the second approach, i.e. frequency-domain BSS. We begin by formulating the BSS problem for convolutive mixtures in Sect. 13.2. Section 13.3 provides an overview of frequency-domain BSS. We then present several important techniques that enable this approach to achieve the effective separation of many sources mixed in a reverberant environment. Section 13.4 discusses complex-valued ICA for instantaneous mix-

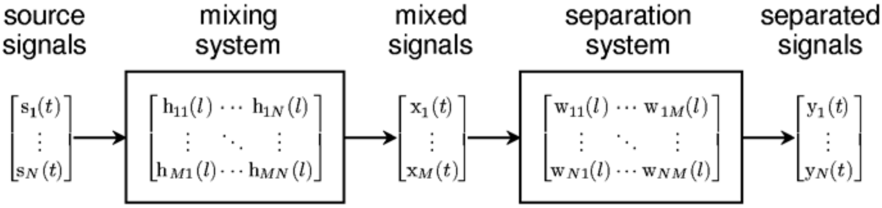


Fig. 13.1. BSS for convolutive mixtures.

tures. Section 13.6 presents a method for solving the permutation problem, which is the most important technique for frequency-domain BSS. To solve the permutation problem, information on source location is very useful. This can be estimated from ICA solutions as shown in Sect. 13.5. The key point with respect to source localization is that the estimation of the mixing system is easily obtained. This is because the ICA algorithm is just for instantaneous mixtures and therefore it is straightforward to calculate the (pseudo)-inverse of a separation matrix, which corresponds to the mixing system. This fact also makes it easy to solve the scaling ambiguity as shown in Sect. 13.7. Section 13.8 discusses a spectral smoothing technique designed to solve the circularity problem. Experimental results shown in Sect. 13.9 are very promising. Section 13.10 concludes this chapter.

13.2 BSS for Convolutive Mixtures

Figure 13.1 shows a block diagram of BSS. Suppose that N source signals $\mathrm{s}_i(t)$ are mixed and observed at M sensors

$$\mathrm{x}_j(t) = \sum_{i=1}^{N} \sum_{l} \mathrm{h}_{ji}(l)\, \mathrm{s}_i(t-l), \quad j = 1, \ldots, M, \tag{13.1}$$

where $\mathrm{h}_{ji}(l)$ represents the impulse response from source i to sensor j. We assume that the number of sources N is known or can be estimated in some way (e.g. by [34]), and the number of sensors M is more than or equal to N ($N \leq M$). The separation system typically consists of a set of FIR filters $\mathrm{w}_{ij}(l)$ of length L to produce N separated signals

$$\mathrm{y}_i(t) = \sum_{j=1}^{M} \sum_{l=0}^{L-1} \mathrm{w}_{ij}(l)\, \mathrm{x}_j(t-l), \quad i = 1, \ldots, N, \tag{13.2}$$

at the outputs. The separation filters $\mathrm{w}_{ij}(l)$ should be obtained blindly, i.e. without knowing $\mathrm{s}_i(t)$ or $\mathrm{h}_{ji}(l)$.

The ideal goal of BSS is to separate and deconvolve the mixtures $\mathrm{x}_j(t)$, and to have a delayed version of source $\mathrm{s}_i(t)$ at each output i. However, this is very difficult if $\mathrm{s}_i(t)$ is a colored signal, which is the case when separating natural sounds such as speech [8]. A practical alternative goal [7], [10] is to obtain the convolved version of a source $\mathrm{s}_i(t)$ measured at a sensor J_i

$$\mathrm{y}_i(t) = \sum_l \mathrm{h}_{J_i i}(l)\,\mathrm{s}_i(t - \frac{L}{2} - l), \tag{13.3}$$

where the sensor index J_i can be selected according to each output i. The way to attain this goal will be discussed in Sect. 13.7.

The performance of BSS is evaluated by a signal-to-interference ratio (SIR), which is the power ratio between the target component and the interference components. Let $\mathrm{u}_{ik}(l)$ be the impulse responses from source $\mathrm{s}_k(t)$ to separated signal $\mathrm{y}_i(t)$:

$$\mathrm{u}_{ik}(l) = \sum_{j=1}^{M}\sum_{\tau=0}^{L-1} \mathrm{w}_{ij}(\tau)\mathrm{h}_{jk}(l-\tau). \tag{13.4}$$

Then, the SIR of output i is calculated as

$$\mathrm{SIR}_i = 10\log_{10}\frac{\langle|\sum_l \mathrm{u}_{ii}(l)\mathrm{s}_i(t-l)|^2\rangle_t}{\langle|\sum_{k\neq i}\sum_l \mathrm{u}_{ik}(l)\mathrm{s}_k(t-l)|^2\rangle_t} \quad (\mathrm{dB}), \tag{13.5}$$

where $\langle\cdot\rangle_t$ denotes the averaging operator over time t.

13.3 Overview of Frequency-Domain Approach

Figure 13.2 shows the flow of frequency-domain BSS. Time-domain signals $\mathrm{x}_j(t)$ sampled at frequency f_s are converted into frequency-domain time-series signals $x_j(f,\tau)$ with an L-point short-time Fourier transform (STFT):

$$x_j(f,\tau) = \sum_{r=-\frac{L}{2}}^{\frac{L}{2}-1} \mathrm{x}_j(\tau + r)\,\mathrm{win}(r)\,e^{-\jmath 2\pi f r}, \tag{13.6}$$

where $f \in \{0, \frac{1}{L}f_s, \ldots, \frac{L-1}{L}f_s\}$ is a frequency, $\mathrm{win}(r)$ is a window that tapers smoothly to zero at each end, such as a Hanning window, $\frac{1}{2}(1+\cos\frac{2\pi r}{L})$, and τ is a new index representing time.

The remaining operations are performed in the frequency domain. The advantage is that the convolutive mixtures in (13.1) can be approximated as instantaneous mixtures in each frequency bin:

$$x_j(f,\tau) = \sum_{i=1}^{N} h_{ji}(f)s_i(f,\tau), \tag{13.7}$$

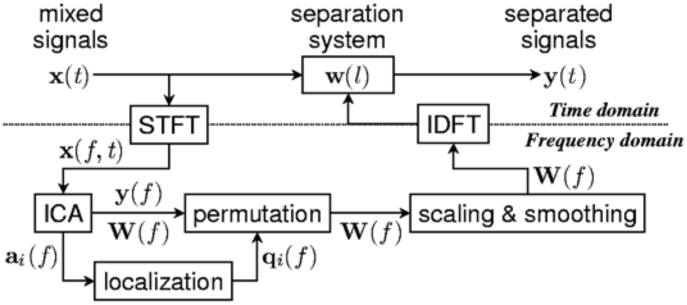


Fig. 13.2. Flow of frequency-domain BSS.

where $h_{ji}(f)$ is the frequency response from source i to sensor j, and $s_i(f,\tau)$ is a frequency-domain time-series signal of $\mathrm{s}_i(t)$ obtained by the same operation as (13.6). The vector notation of the mixing model (13.7) is

$$\mathbf{x}(f,\tau) = \sum_{i=1}^{N} \mathbf{h}_i(f) s_i(f,\tau), \tag{13.8}$$

where $\mathbf{x} = [x_1, \ldots, x_M]^T$ is a sensor sample vector and $\mathbf{h}_i = [h_{1i}, \ldots, h_{Mi}]^T$ is the vector of the frequency responses from source s_i to all M sensors.

To obtain the frequency responses $w_{ij}(f)$ of separation filters $\mathrm{w}_{ij}(l)$ in (13.2), complex-valued ICA

$$\mathbf{y}(f,t) = \mathbf{W}(f)\mathbf{x}(f,t) \tag{13.9}$$

is solved, where $\mathbf{y} = [y_1, \ldots, y_N]^T$ is a vector of separated signals and $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_N]^H$ is an $N \times M$ separation matrix whose elements are $w_{ij} = [\mathbf{W}]_{ij}$. The details of ICA algorithm will be discussed in Sect. 13.4.

Calculating the Moore-Penrose pseudoinverse $\mathbf{W}^+$ (which is reduced to the inverse $\mathbf{W}^{-1}$ if $N = M$) of $\mathbf{W}$

$$[\mathbf{a}_1, \cdots, \mathbf{a}_N] = \mathbf{W}^+, \ \mathbf{a}_i = [a_{1i}, \ldots, a_{Mi}]^T, \tag{13.10}$$

is very useful for source localization and scaling alignment as will be shown in Sects. 13.5 and 13.7, respectively. It should be noted that it is not difficult to make $\mathbf{W}$ invertible by using an appropriate ICA procedure (for an example see Sect. 13.4). By multiplying both sides of (13.9) by $\mathbf{W}^+$, the sensor sample vector $\mathbf{x}(\tau)$ is represented by a linear combination of basis vectors $\mathbf{a}_1, \ldots, \mathbf{a}_N$:

$$\mathbf{x}(f,\tau) = \sum_{i=1}^{N} \mathbf{a}_i(f) y_i(f,\tau). \tag{13.11}$$

It is well-known that an ICA solution (13.9) has permutation and scaling ambiguities: even if we permute the rows of $\mathbf{W}(f)$ or multiply a row by a constant, it is still an ICA solution. In matrix notation,

$$\mathbf{W}(f) \leftarrow \mathbf{\Lambda}(f)\mathbf{P}(f)\mathbf{W}(f) \tag{13.12}$$

is also an ICA solution for any permutation $\mathbf{P}(f)$ and diagonal $\mathbf{\Lambda}(f)$ matrix. Permutation alignment is to determine $\mathbf{P}(f)$ so that a time-domain separated signal contains frequency components from the same source. Section 13.6 presents a method for solving this problem. Scaling alignment is to determine $\mathbf{\Lambda}(f)$ so that a time-domain separated signal satisfies the goal (13.3), as will be discussed in Sect. 13.7.

Then, we perform spectral smoothing so that a time-domain separation filter tapers smoothly to zero at each end. This is typically achieved by multiplying the time-domain filter by a Hanning window, which is equivalent to smoothing the frequency-domain separation matrices as follows

$$\mathbf{W}(f) \leftarrow \frac{1}{4}\left[\mathbf{W}(f-\varDelta f) + 2\mathbf{W}(f) + \mathbf{W}(f+\varDelta f)\right],$$

where $\varDelta f = \frac{f_s}{L}$ is the difference from the adjacent frequency. However, this smoothing changes the ICA solution and causes an error. Section 13.8 discusses the error and how to minimize it.

Finally, separation filters $\mathrm{w}_{ij}(l)$ are obtained by applying inverse DFT to $w_{ij}(f) = [\mathbf{W}(f)]_{ij}$:

$$\mathrm{w}_{ij}(l) = \sum_{f \in \{0, \frac{1}{L}f_s, \ldots, \frac{L-1}{L}f_s\}} w_{ij}(f) e^{\jmath 2\pi f (l - \frac{L}{2})},$$

where $l = 0, \ldots, L-1$. The reason for using $e^{\jmath 2\pi f(l-\frac{L}{2})}$ instead of $e^{\jmath 2\pi f l}$ is to make the separation filter $\mathrm{w}_{ij}(l)$ causal. Then, the separated signals $\mathrm{y}_i(t)$ are produced by (13.2).

13.4 Complex-Valued ICA

This section discusses how to solve the ICA equation (13.9). One of the advantages of frequency-domain BSS is that we can employ any ICA algorithm for instantaneous mixtures, such as the information maximization approach (InfoMax) [35] combined with the natural gradient [36], FastICA [27], JADE [37], or an algorithm based on the non-stationarity of signals [38]. Here, we explain a procedure that has been shown to be efficient by the experiments described in Sect. 13.9. The procedure consists of the following three steps:

1. Dimension reduction and whitening by eigenvalue decomposition.
2. ICA by a unitary matrix (FastICA).
3. ICA by InfoMax combined with the natural gradient.

The first step performs a linear transformation

$$\mathbf{z}(\tau) = \mathbf{V}\mathbf{x}(\tau)$$

for M-dimensional sensor observations $\mathbf{x}(\tau)$ such that the dimension of $\mathbf{z}(\tau)$ is reduced (if necessary) to the number of sources N and $\mathbf{z}(\tau)$ is spatially whitened (sphered), i.e. $\langle \mathbf{z}(\tau)\mathbf{z}(\tau)^H \rangle_\tau = \mathbf{I}$, where $\mathbf{I}$ is the $N \times N$ identity matrix. The linear transformation $\mathbf{V}$ is typically obtained by eigenvalue decomposition. Let $\lambda_1 \geq \cdots \geq \lambda_M$ be sorted eigenvalues of the spatial correlation matrix $\mathbf{R} = \langle \mathbf{x}(\tau)\mathbf{x}(\tau)^H \rangle_\tau$, and $\mathbf{e}_1, \ldots, \mathbf{e}_M$ be their corresponding eigenvectors. Then, the linear transformation is

$$\mathbf{V} = \mathbf{D}^{-1/2}\mathbf{E}^H,$$

where $\mathbf{D} = \text{diag}(\lambda_1, \ldots, \lambda_N)$ is the diagonal matrix of the N largest eigenvalues, and $\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_N]$ is the matrix of their corresponding eigenvectors.

This step is practically important for the following two reasons. First, the outputs $\mathbf{y}(\tau)$ of ICA (13.9) adhere to the signal subspace that is identified by the N eigenvectors $\mathbf{e}_1, \ldots, \mathbf{e}_N$. This means that the following ICA algorithm does not pursue its solution in the noise subspace, which consequently stabilizes the algorithm and also has a noise/reverberation reduction effect [16]. A geometrical interpretation of the dimension reduction is given in [25]. Secondly, the whitening $\langle \mathbf{z}\mathbf{z}^H \rangle_\tau = \mathbf{I}$ is necessary for FastICA, and also gives an efficient convergence for InfoMax even if the step size is constant over all frequency bins.

The second step performs ICA in a constrained form:

$$\mathbf{y}(\tau) = \mathbf{B}\mathbf{z}(\tau),$$

where $\mathbf{B}$ is an $N \times N$ unitary matrix: $\mathbf{B}\mathbf{B}^H = \mathbf{I}$. This is performed by a complex-valued version of FastICA [27], [28]. It is very efficient as a fairly good solution can be obtained with only several iterations. The efficiency comes from the fact that $\mathbf{z}$ is whitened and $\mathbf{B}$ is unitary. However, there remains room for improving the solution by using another ICA algorithm. One of the reasons is that the output $\mathbf{y}$ of FastICA is whitened $\langle \mathbf{y}(f,\tau)\mathbf{y}(f,\tau)^H \rangle_\tau = \mathbf{I}$ and therefore uncorrelated, whereas original sources $s_1(f,\tau), \ldots, s_N(f,\tau)$ are not always completely uncorrelated with a limited number of samples.

The third step improves the ICA solution obtained so far

$$\mathbf{y}(\tau) = \mathbf{W}\mathbf{x}(\tau) = \mathbf{B}\mathbf{V}\mathbf{x}(\tau),$$

by employing another ICA algorithm that does not have the unitary constraint. Based on the use of InfoMax combined with the natural gradient, a separation matrix $\mathbf{W}$ is gradually improved by the learning rule:

$$\mathbf{W} \leftarrow \mathbf{W} + \mu \left[\mathbf{I} - \langle \mathbf{\Phi}(\mathbf{y}(\tau))\mathbf{y}(\tau)^H \rangle_\tau\right] \mathbf{W}, \tag{13.13}$$

where μ is a step-size parameter. $\mathbf{\Phi}(\mathbf{y}) = [\Phi(y_1), \ldots, \Phi(y_N)]^T$ is an element-wise nonlinear function defined by

$$\Phi(y_i) = -\frac{\partial}{\partial y_i} \log p(y_i), \tag{13.14}$$

where $p(y_i)$ is the probability density function (pdf) of a complex-valued signal $y_i = |y_i|\, e^{\jmath \cdot \arg(y_i)}$. Since y_i is a frequency-domain signal whose phase can be shifted arbitrarily by shifting the STFT window position (13.6), a feasible assumption is that the pdf is independent of the phase $p(y_i) = \alpha \cdot p(|y_i|)$, where α is a constant. This assumption reduces (13.14) to

$$\Phi(y_i) = \varphi(|y_i|)\, e^{\jmath \cdot \arg(y_i)}, \quad \varphi(|y_i|) = -\frac{\partial}{\partial |y_i|} \log p(|y_i|). \tag{13.15}$$

If we assume the Laplacian distribution $p(|y_i|) = \frac{1}{2} e^{-|y_i|}$, which is typical for speech modeling, we have $\varphi(|y_i|) = 1$ and therefore a simple nonlinear function:

$$\Phi(y_i) = e^{\jmath \cdot \arg(y_i)}.$$

A nonlinear function of the form (13.15) has a better convergence property [20] than one where the nonlinearity is applied separately to the real and imaginary parts of a complex-valued signal y_i.

13.5 Source Localization

This section presents a source localization method by analyzing the ICA solution (13.9) or equivalently (13.11). The information about source locations can be used to solve the permutation problem, as described in the next section. Many source localization methods have been proposed. A widely used method is MUSIC (MUltiple SIgnal Classification) [39], which employs subspace analysis with second order statistics. The ICA-based method, on the other hand, employs higher order statistics (or multiple second order statistics based on non-stationarity). In this sense, the ICA-based method has certain advantages over the subspace based method [40].

The source localization technique that employs ICA is a by-product of research on frequency-domain BSS. Direction of arrival (DOA) estimation methods [17], [18], [19] have been proposed that are based on beamforming theory [41]. They calculate directivity patterns from the separation matrix $\mathbf{W}$, and then search the null directions, which correspond to the directions of sources [21]. However, it is simpler and more effective to estimate the directions directly from the basis vectors $\mathbf{a}_i$, which are given by the pseudoinverse of $\mathbf{W}$. The source localization method [22], [23], [24], [40] presented in this section is based on this idea. Such an idea was taken for granted in research on blind identification [42], [43], where the mixing system is estimated directly.

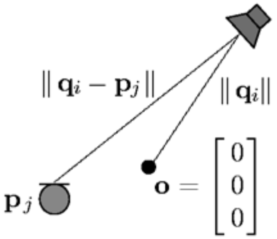

$\mathbf{q}_i$
$\| \mathbf{q}_i - \mathbf{p}_j \|$
$\| \mathbf{q}_i \|$
$\mathbf{p}_j$
$\mathbf{o} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$

Fig. 13.3. Nearfield (direct-path) model.

13.5.1 Basic Theory for Nearfield Model

Let us assume a mixing model that is suitable for source localization. Although the mixing model (13.1) in the time domain is a multi-path mixing model, we approximate the frequency response $h_{ji}(f)$ in (13.7) with a nearfield (direct-path) model (Fig. 13.3):

$$h_{ji}(f) \approx \frac{1}{||\mathbf{q}_i - \mathbf{p}_j||} e^{\jmath 2\pi f c^{-1}(||\mathbf{q}_i - \mathbf{p}_j|| - ||\mathbf{q}_i||)}, \tag{13.16}$$

where $\mathbf{p}_j$ and $\mathbf{q}_i$ are 3-dimensional vectors representing the locations of sensor j and source i, respectively, and c is the propagation velocity of the signals. We assume that the amplitude is attenuated based on the distance $||\mathbf{q}_i - \mathbf{p}_j||$. We also assume that the phase depends on the difference between the distances $||\mathbf{q}_i - \mathbf{p}_j|| - ||\mathbf{q}_i||$ from the source to the sensor and to the origin $\mathbf{o} = [0, 0, 0]^T$. This makes the phase zero at the origin. If the phase $2\pi f c^{-1}(||\mathbf{q}_i - \mathbf{p}_j|| - ||\mathbf{q}_i||)$ is outside the range $(-\pi, \pi)$, this model suffers from spatial aliasing. Therefore, this model is feasible as long as the condition

$$f < \left| \frac{c}{2 \cdot (\, ||\mathbf{q}_i - \mathbf{p}_j|| - ||\mathbf{q}_i||\,)} \right|$$

is satisfied.

The ICA-based source localization discussed in this section estimates the location $\mathbf{q}_i$ of source i from information about sensor locations $\mathbf{p}_j$ and the separation matrix $\mathbf{W}(f)$ obtained by ICA (13.9). Let us assume here that the decomposition (13.11) of observations $\mathbf{x}(f, \tau)$ has been obtained in each frequency bin by the pseudoinverse of $\mathbf{W}(f)$. By comparing (13.8) and (13.11), we observe the following fact. If the ICA algorithm works well and the outputs $y_1, \ldots, y_N$ are the estimation of the sources $s_1, \ldots, s_N$, then the basis vectors $\mathbf{a}_1, \ldots, \mathbf{a}_N$ are also estimations of the mixing vectors $\mathbf{h}_1, \ldots, \mathbf{h}_N$ up to the permutation and scaling ambiguity.

Following the model (13.16), the ratio between two elements $a_{ji}, a_{j'i}$ of the same basis vector $\mathbf{a}_i$ provides the key equation for source localization:

$$\frac{a_{ji}}{a_{j'i}} = \frac{\alpha_i h_{ji}}{\alpha_i h_{j'i}} = \frac{||\mathbf{q}_i - \mathbf{p}_{j'}||}{||\mathbf{q}_i - \mathbf{p}_j||} e^{\jmath 2\pi f c^{-1}(||\mathbf{q}_i - \mathbf{p}_j|| - ||\mathbf{q}_i - \mathbf{p}_{j'}||)}, \tag{13.17}$$

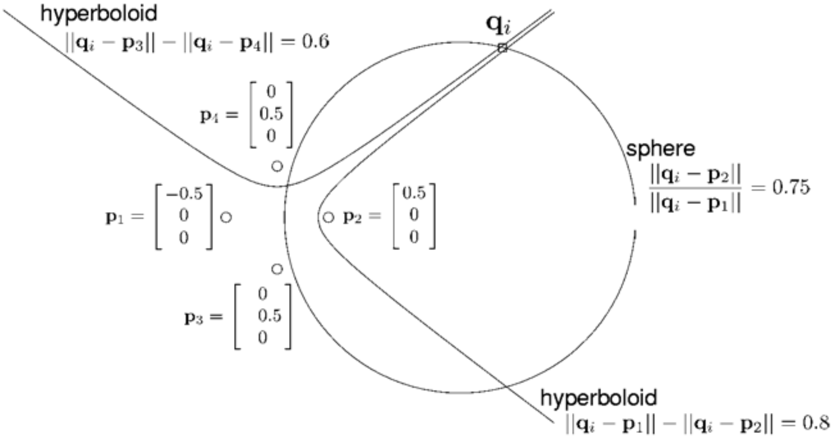

hyperboloid
$\|\mathbf{q}_i - \mathbf{p}_3\| - \|\mathbf{q}_i - \mathbf{p}_4\| = 0.6$
$\mathbf{q}_i$
$\mathbf{p}_4 = \begin{bmatrix} 0 \\ 0.5 \\ 0 \end{bmatrix}$
sphere
$\frac{\|\mathbf{q}_i - \mathbf{p}_2\|}{\|\mathbf{q}_i - \mathbf{p}_1\|} = 0.75$
$\mathbf{p}_1 = \begin{bmatrix} -0.5 \\ 0 \\ 0 \end{bmatrix}$
$\mathbf{p}_2 = \begin{bmatrix} 0.5 \\ 0 \\ 0 \end{bmatrix}$
$\mathbf{p}_3 = \begin{bmatrix} 0 \\ 0.5 \\ 0 \end{bmatrix}$
hyperboloid
$\|\mathbf{q}_i - \mathbf{p}_1\| - \|\mathbf{q}_i - \mathbf{p}_2\| = 0.8$

Fig. 13.4. Source localization by the intersection of two hyperboloids and a sphere.

where the scaling ambiguity α_i is canceled out by calculating the ratio. The permutation ambiguity still remains. However, if we estimate the location $\mathbf{q}_i$ for all $i = 1, \ldots, N$, the set of all estimated locations does not depend on the permutation.

The set of vectors $\mathbf{q}_i$ in the argument of (13.17)

$$||\mathbf{q}_i - \mathbf{p}_j|| - ||\mathbf{q}_i - \mathbf{p}_{j'}|| = \frac{\arg(a_{ji}/a_{j'i})}{2\pi f c^{-1}}, \tag{13.18}$$

defines a surface where the difference between the distances from $\mathbf{p}_j$ and $\mathbf{p}_{j'}$ is constant. The surface is one sheet of a two-sheeted hyperboloid. Alternatively, the set of vectors $\mathbf{q}_i$ in the modulus of (13.17)

$$\frac{||\mathbf{q}_i - \mathbf{p}_{j'}||}{||\mathbf{q}_i - \mathbf{p}_j||} = \left|\frac{a_{ji}}{a_{j'i}}\right|. \tag{13.19}$$

defines a sphere where the ratio of the distances from $\mathbf{p}_j$ and $\mathbf{p}_{j'}$ is constant. Therefore, with these two equations (13.18) and (13.19), we can estimate the possible location $\mathbf{q}_i$ of source s_i. Such hyperboloid and sphere are defined by a pair of sensors j and j'. If we select another pair of sensors, a different hyperboloid and sphere are obtained. In this way, the location $\mathbf{q}_i$ is estimated as the intersection of several hyperboloids and spheres. An example is shown in Fig. 13.4.

13.5.2 DOA Estimation with Farfield Model

Although it is useful to estimate a 3-dimensional location, calculating the intersections of hyperboloids and spheres is computationally demanding. There

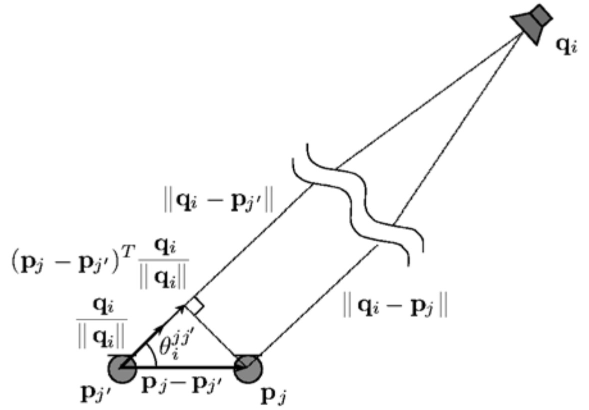

$\mathbf{q}_i$
$||\mathbf{q}_i - \mathbf{p}_{j'}||$
$(\mathbf{p}_j - \mathbf{p}_{j'})^T \frac{\mathbf{q}_i}{||\mathbf{q}_i||}$
$\frac{\mathbf{q}_i}{||\mathbf{q}_i||}$
$\theta_i^{jj'}$
$||\mathbf{q}_i - \mathbf{p}_j||$
$\mathbf{p}_{j'}$
$\mathbf{p}_j - \mathbf{p}_{j'}$
$\mathbf{p}_j$

Fig. 13.5. Farfield model.

are many cases where it is sufficient to estimate just the direction of arrival (DOA) of source s_i. If we assume the source location $\mathbf{q}_i$ is far from sensors $\mathbf{p}_j$ and $\mathbf{p}_{j'}$, (13.18) can be approximated as farfield model (Fig. 13.5)

$$(\mathbf{p}_j - \mathbf{p}_{j'})^T \frac{\mathbf{q}_i}{||\mathbf{q}_i||} = \frac{\arg(a_{ji}/a_{j'i})}{2\pi f c^{-1}}, \tag{13.20}$$

and the cosine of the angle $\theta_i^{jj'}$ between two vectors $\mathbf{q}_i$ and $\mathbf{p}_j - \mathbf{p}_{j'}$ can be calculated as

$$\cos\theta_i^{jj'} = \frac{(\mathbf{p}_j - \mathbf{p}_{j'})^T \mathbf{q}_i}{||\mathbf{p}_j - \mathbf{p}_{j'}|| \cdot ||\mathbf{q}_i||} = \frac{\arg(a_{ji}/a_{j'i})}{2\pi f c^{-1} ||\mathbf{p}_j - \mathbf{p}_{j'}||}. \tag{13.21}$$

The set of vectors $\mathbf{q}_i$ that satisfy (13.20) represents a cone [23], which is the asymptotic surface of the corresponding hyperboloid (13.18). To estimate the DOA of a source, the intersections of several cones should be obtained. Let us assume that we select u cones whose corresponding sensor pairs are $(j_1, j_1'), \ldots, (j_u, j_u')$. The set of equations (13.20) for u sensor pairs is represented as

$$\mathbf{D} \frac{\mathbf{q}_i}{||\mathbf{q}_i||} = \frac{\mathbf{r}_i}{2\pi f c^{-1}}, \tag{13.22}$$

where

$$\mathbf{D} = [\mathbf{p}_{j_1} - \mathbf{p}_{j_1'}, \ldots, \mathbf{p}_{j_u} - \mathbf{p}_{j_u'}]^T,$$
$$\mathbf{r}_i = [\arg(a_{j_1 i}/a_{j_1' i}), \ldots, \arg(a_{j_u i}/a_{j_u' i})]^T.$$

In practical situations, there is no exact solution for (13.22) because the u conditions do not coincide exactly. Therefore, we typically solve it in the

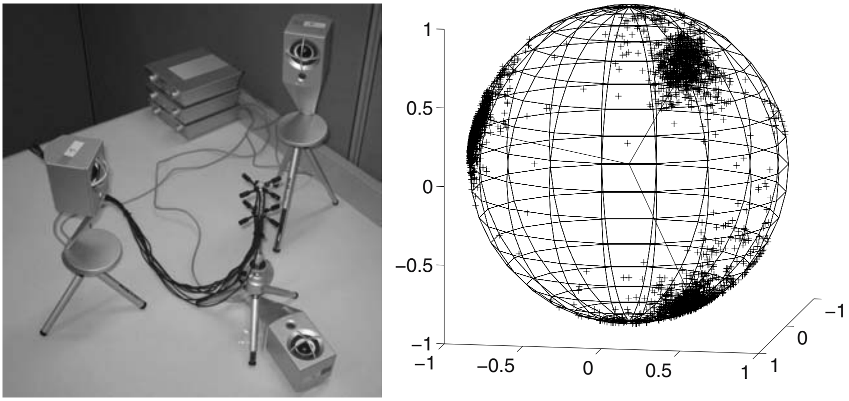


Fig. 13.6. 3-dimensional arrangement of eight microphones and three loudspeakers (left) and DOA estimation results for this case (right).

least-square sense by using the Moore-Penrose pseudoinverse [24]:

$$\frac{\mathbf{q}_i}{||\mathbf{q}_i||} = \frac{\mathbf{D}^{+}\mathbf{r}_i}{2\pi f c^{-1}}. \tag{13.23}$$

If rank($\mathbf{D}$) ≥ 3, the set of vectors $\mathbf{q}_i$ that satisfy (13.23) represents a line in 3-dimensional space, which represents the DOA of a source i.

The left photo in Fig. 13.6 shows a case where eight microphones and three loudspeakers are arranged 3-dimensionally, and the right plot shows the DOA estimation results for this case. Each point shows a location vector $\mathbf{q}_i(f)$ that is normalized to unit norm $\mathbf{q}_i(f) \leftarrow \frac{\mathbf{q}_i(f)}{||\mathbf{q}_i(f)||}$. The estimations are obtained for all frequencies f and all output indexes i. As shown in the plot, they form clusters, each of which corresponds to the location of each source.

If the sensor and source locations are limited to a 2-dimensional plane, the dimensionality of location vectors, such as $\mathbf{p}_i$ and $\mathbf{q}_i$, can be reduced to two. In this case, rank($\mathbf{D}$) ≥ 2 is sufficient to have a solution in (13.23). Moreover, the DOA of source i can be represented simply by the angle θ_i that satisfies

$$\mathbf{q}_i = [\cos(\theta_i), \sin(\theta_i)]^T, \quad -180^\circ < \theta_i \leq 180^\circ. \tag{13.24}$$

Figure 13.14 shows a case where the sensor and source locations are limited to 2-dimensions. The DOA estimations in this case are shown in Figs. 13.15 and 13.16.

If the sensors are arranged linearly and the potential source location is in a 2-dimensional half-plane, which is one side of the sensor arrangement line, the angle $\theta_i^{jj'}$ ($0^\circ \leq \theta_i^{jj'} \leq 180^\circ$) by (13.21) provides sufficient information on the source location. For example, Fig. 13.12 shows DOA estimation results for such a case whose condition is shown in Fig. 13.11.

13.6 Permutation Alignment

This section discusses how to solve the permutation problem. Various methods have already been proposed. With reference to the ICA equation (13.9) and also the decomposition (13.11) of observations $\mathbf{x}(f, \tau)$, we classify these methods into four categories based on the following strategies:

1. Applying an operation to the separation matrix $\mathbf{W}(f)$,
2. Utilizing the information on the separation matrix $\mathbf{W}(f)$ itself,
3. Utilizing the information on the basis vectors $\mathbf{a}_1(f), \ldots, \mathbf{a}_N(f)$,
4. Utilizing the information on the separated signals $y_1(f, \tau), \ldots, y_N(f, \tau)$.

The operation of the first strategy basically involves making the separation matrices smooth in the frequency domain. This has been realized by reducing the filter length by rectangular windowing in the time domain [11], [12], [13], [9], or averaging the separation matrices with adjacent frequencies [11]. However, this operation makes the separation matrix $\mathbf{W}(f)$ different from the ICA solution (13.9), and this may have a detrimental effect on the separation performance. A possible way to solve this problem is to interleave the ICA update, e.g. (13.13), and this operation until convergence. In this sense, this strategy is related to the third approach for BSS discussed in the Introduction.

The second category includes the beamforming approach [17], [18], [19], where the directivity patterns formed by the separation matrix are analyzed to identify the DOA of each source. The third category includes an approach that utilizes the results of source localization with the basis vectors [22], [23], [24], [43]. The theory and operation for source localization were discussed in Sect. 13.5. These two approaches of the second and the third categories utilize basically the same information because the separation matrix $\mathbf{W}(f)$ and the basis vectors $\mathbf{a}_1(f), \ldots, \mathbf{a}_N(f)$ are directly connected by the pseudoinverse operation (13.10). However, the information used in the third category is easier to handle since it directly represents the mixing system (13.8). The last category contains an approach that employs the inter-frequency correlations of output signal envelopes [15], [14]. This is particularly effective for a non-stationary signal such as speech.

In the next two subsections, we explain the approaches of the third and the fourth categories, respectively. Since these two approaches have different complementary characteristics, integrating them is a good way to pursue a better solution to the permutation problem [22]. Subsection 13.6.3 presents a method that effectively integrates the two approaches to solve the permutation problem in a better way. In the following subsections, let Π_f be a permutation corresponding to the inverse $\mathbf{P}^{-1}(f)$ of the permutation matrix of (13.12). The permutation problem can be formulated to obtain Π_f for every frequency f, which is a mapping from source index k to output index i:

$$i = \Pi_f(k).$$

13.6.1 Localization Approach

The basic idea of this approach is to estimate the locations of sources and cluster them to decide the permutation. ICA-based source localization (Sect. 13.5) estimates the location $\mathbf{q}_i(f)$ of a source that corresponds to the i-th basis vector $\mathbf{a}_i(f)$ for each frequency f. Let the following function *localize* estimate the location in this way:

$$\mathbf{q}_i(f) = \mathit{localize}(f, \mathbf{a}_i(f)) \,.$$

If just the DOA estimation is adequate, the location vector $\mathbf{q}_i(f)$ should be normalized to the unit norm $\mathbf{q}_i(f) \leftarrow \frac{\mathbf{q}_i(f)}{||\mathbf{q}_i(f)||}$. If the locations of sensors and sources are limited to a 2-dimensional plane, we might obtain $\theta_i(f)$ that satisfies (13.24) as a DOA estimation.

Then, we employ a clustering algorithm to find N clusters $C_1, \ldots, C_N$ formed by estimated locations $\mathbf{q}_i(f)$ or $\theta_i(f)$. Each C_k corresponds to the location of source k. Let the following function *clustering* perform clustering for all the estimated locations $\mathbf{q}_i(f)$ and return the centroid $\mathbf{c}_k$ and the variance σ_k^2 of each cluster C_k:

$$[\mathbf{c}_1, \sigma_1, \ldots, \mathbf{c}_N, \sigma_N] = \mathit{clustering}({}^{\forall} f,\, \mathbf{q}_1(f), \ldots, \mathbf{q}_N(f)),$$

$$\mathbf{c}_k = \sum_{\mathbf{q} \in C_k} \frac{\mathbf{q}}{|C_k|}, \;\; \sigma_k^2 = \sum_{\mathbf{q} \in C_k} \frac{||\mathbf{c}_k - \mathbf{q}||^2}{|C_k|},$$

where $|C_k|$ is the number of vectors in the cluster. The optimization criterion for clustering is to minimize the total sum $\sum_{k=1}^{N} \sigma_k^2$ of the variances. The optimization is efficiently performed with the k-means clustering algorithm [44]. Once we have N clusters, permutations for all frequencies f can be decided by

$$\Pi_f = \mathrm{argmin}_{\Pi} \sum_{k=1}^{N} ||\mathbf{c}_k - \mathbf{q}_{\Pi(k)}(f)||^2. \tag{13.25}$$

The advantage of this source localization approach is that it is very simple to decide the permutation Π_f for each frequency once the centroids of N clusters are obtained. However, the downside of this approach is that estimated locations or DOAs are not accurate for some frequencies and therefore neither are the permutations Π_f. Such situations typically happen at low frequencies, where the phase difference caused by the sensor spacing is very small, as shown in Fig. 13.12.

13.6.2 Correlation Approach

This subsection presents an approach to permutation alignment based on the inter-frequency correlation of separated signals. The correlation should be

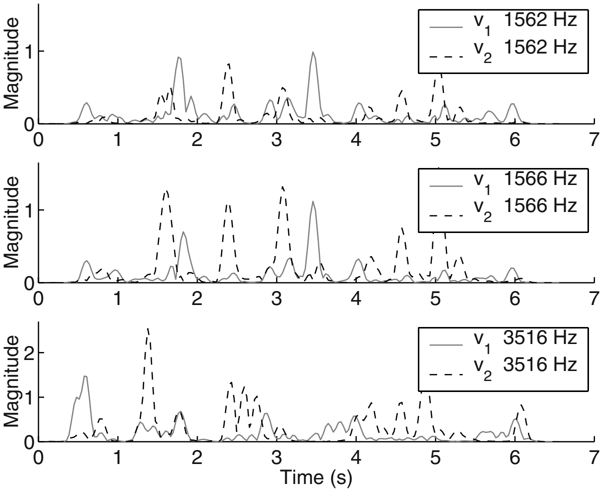


Fig. 13.7. Envelopes of two output signals at different frequencies.

calculated for the amplitude $|y_i(f,\tau)|$ or (log-scaled) power $|y_i(f,\tau)|^2$ of separated signals. The correlation of raw complex-valued signals $y_i(f,\tau)$ would be very low because of the STFT property. Here, we use the amplitude (so-called envelope),

$$v_i^f(\tau) = |y_i(f,\tau)|,$$

of a separated signal $y_i(f,t)$. The correlation of two sequences $x(\tau)$ and $y(\tau)$ is usually calculated by the correlation coefficient

$$\mathrm{cor}(x,y) = (\mu_{x\cdot y} - \mu_x \cdot \mu_y)/(\sigma_x \cdot \sigma_y),$$

where μ_x is the mean and σ_x is the standard deviation of x. Based on this definition, $\mathrm{cor}(x,x) = 1$, and $\mathrm{cor}(x,y) = 0$ if x and y are uncorrelated.

Envelopes have high correlations at neighboring frequencies if separated signals correspond to the same source signal. Figure 13.7 shows an example. Two envelopes v_1^{1562} and v_1^{1566}, as well as v_2^{1562} and v_2^{1566}, are highly correlated. Thus, calculating such correlations helps us to align permutations.

A simple criterion for deciding Π_f is to maximize the sum of the correlations between neighboring frequencies within distance δ:

$$\Pi_f = \mathrm{argmax}_{\Pi} \sum_{|g-f|\le\delta} \sum_{i=1}^{N} \mathrm{cor}(v^f_{\Pi(i)}, v^g_{\Pi_g(i)}), \tag{13.26}$$

where Π_g is the permutation at frequency g. This criterion is based on local information and has a drawback in that mistakes in a narrow range of frequencies may lead to the complete misalignment of the frequencies beyond the range.

To avoid this problem, the method in [15] does not limit the frequency range in which correlations are calculated. It decides permutations one by one based on the criterion:

$$\Pi_f = \mathrm{argmax}_\Pi \sum_{i=1}^{N} \mathrm{cor}(\, v^f_{\Pi(i)}, \sum_{g \in \mathcal{F}} v^g_{\Pi_g(i)} \,),$$

where $\mathcal{F}$ is a set of frequencies in which the permutation is decided. This method assumes high correlations of envelopes even between frequencies that are not close neighbors. This assumption is not satisfied for all pairs of frequencies, as v_i^{1566} and v_i^{3516} in Fig. 13.7 do not have a high correlation. Therefore, this method still has a drawback in that permutations may be misaligned at many frequencies.

If a source signal has a harmonic structure, as in the case of speech, there are strong correlations between the envelopes of a fundamental frequency f and its harmonics $2f$, $3f$, ... Therefore, maximizing the correlation among harmonics is another idea for permutation alignment [22]:

$$\Pi_f = \mathrm{argmax}_\Pi \sum_{g \in \mathcal{H}(f)} \sum_{i=1}^{N} \mathrm{cor}(v^f_{\Pi(i)}, v^g_{\Pi_g(i)}), \tag{13.27}$$

where $\mathcal{H}(f)$ provides a set of harmonic frequencies of f. The permutation accuracy improves if we take the signal's harmonic structure into consideration. However, maximizing (13.26) and (13.27) simultaneously is not very straightforward and is computationally expensive.

13.6.3 Integrated Method

This subsection presents a method that integrates the two approaches discussed in the last two subsections. The intention behind this integration is to solve the permutation problem robustly and precisely. Let us review the characteristics of the two approaches.

- **Robustness**: The localization approach is robust since a misalignment at one frequency does not affect other frequencies. The correlation approach is not robust since a misalignment at one frequency affects the results of other frequencies, and may cause consecutive misalignments.
- **Preciseness**: The localization approach is not precise since the evaluation is based on an approximation (13.16) of the mixing system. The correlation approach is precise as long as signals are well separated by ICA since the measurement is based on the separated signals themselves.

To benefit from both advantages, namely the robustness of the localization approach and the preciseness of the correlation approach, the integrated method first decides permutations with the localization approach and then refines the solution with the correlation approach. An implementation of the integrated method consists of the following four steps [22]:

1. Decide the permutations by the localization approach (13.25) at certain frequencies where the confidence of source localization is sufficiently high,
2. Decide the permutations based on neighboring correlations (13.26) as long as the criterion gives a clear-cut decision,
3. Decide the permutations at certain frequencies where the correlation among harmonics (13.27) is sufficiently high,
4. Decide the permutations for the remaining frequencies based on neighboring correlations (13.26).

Figure 13.8 shows the pseudo-code. A set $\mathcal{F}$ contains frequencies where the permutation has been decided. The key to the first step is that we fix a permutation only if the confidence of source localization is sufficiently high. We assume that the confidence is high if the squared distance between an estimated location and its corresponding centroid is smaller than the variance, i.e. $||\mathbf{c}_k - \mathbf{q}_{\Pi(k)}(f)||^2 < \sigma_k^2$. In the second step, permutations are decided one by one for the frequency f where the sum of the correlations with fixed frequencies $g \in \mathcal{F}$ within distance $|g - f| \leq \delta$ is the maximum. This is repeated until the maximum correlation sum is larger than a threshold th_{cor}. In the third step, the permutations are decided for frequencies f where the sum of the correlations among harmonics is larger than a threshold th_{ha}. The last step decides the permutations for the remaining frequencies with the same criterion as the second step.

Let us discuss the advantages of the integrated method. The main advantage is that it does not cause a large misalignment as long as the permutations fixed by the localization approach are correct. Moreover, the correlation part (steps 2, 3 and 4) compensates for the lack of preciseness of the localization approach. The correlation part consists of three steps for two reasons. First, the harmonics part (step 3) works well if most of the other permutations are fixed. Secondly, the method becomes more robust by quitting the step 2 if there is no clear-cut decision. With this structure, we can avoid fixing the permutations for consecutive frequencies without high confidence. As shown in the experimental results (Sect. 13.9), this integrated method is effective at separating many sources.

13.7 Scaling Alignment

The scaling ambiguity $\mathbf{\Lambda}(f)$ in (13.12) is easily solved by calculating the (pseudo)-inverse of a separation matrix $\mathbf{W}(f)$ [15,7]. The frequency-domain counterpart of the BSS goal (13.3) is

$$y_i(f, \tau) = h_{J_i i}(f) s_i(f, \tau), \qquad (13.28)$$

where J_i can be selected according to each output i but should be the same for all frequencies f. Let us assume that the ICA and the permutation problem have been solved. Then the $\mathbf{a}_i$ term in (13.11) is close to the $\mathbf{h}_i$ term in (13.8):

$$\mathbf{h}_i(f) s_i(f, \tau) \approx \mathbf{a}_i(f) y_i(f, \tau). \qquad (13.29)$$

```
$\mathcal{F} = \emptyset$    /* the set of fixed frequencies */
/* 1. Fix permutations by the localization approach */
for $({}^\forall f$ and ${}^\forall i)$ {
  $\mathbf{q}_i(f) = \mathit{localize}(f, \mathbf{a}_i(f))$ /* source localization */
}
$[\mathbf{c}_1, \sigma_1, \ldots, \mathbf{c}_N, \sigma_N] = \mathit{clustering}({}^\forall f,\ \mathbf{q}_1(f), \ldots, \mathbf{q}_N(f))$
for $({}^\forall f)$ {
  $\Pi_f = \mathrm{argmin}_\Pi \sum_{k=1}^{N} ||\mathbf{c}_k - \mathbf{q}_{\Pi(k)}(f)||^2$
  if $({}^\forall k,\ ||\mathbf{c}_k - \mathbf{q}_{\Pi(k)}(f)||^2 < \sigma_k^2)$ {
    $\mathcal{F} = \mathcal{F} \cup \{f\}$
  }
}
/* 2. Fix permutations by neighboring correlations */
while $({}^\exists f \notin \mathcal{F})$ {
  for $({}^\forall f \notin \mathcal{F})$ {
    $R_f = \max_\Pi \sum_{|g-f|\leq\delta, g\in\mathcal{F}} \sum_{i=1}^{N} \mathrm{cor}(v_{\Pi(i)}^f, v_{\Pi_g(i)}^g)$
    $\Pi_f = \mathrm{argmax}_\Pi \sum_{|g-f|\leq\delta, g\in\mathcal{F}} \sum_{i=1}^{N} \mathrm{cor}(v_{\Pi(i)}^f, v_{\Pi_g(i)}^g)$
  }
  if $(\max_f R_f > th_{cor})$ {
    $\psi = \mathrm{argmax}_f\ R_f$
    $\mathcal{F} = \mathcal{F} \cup \{\psi\}$
  } else {
    break
  }
}
/* 3. Fix permutations by harmonic structure */
for $({}^\forall f \notin \mathcal{F})$ {
  $R_f = \max_\Pi \sum_{g\in\mathcal{H}(f)\cap\mathcal{F}} \sum_{i=1}^{N} \mathrm{cor}(v_{\Pi(i)}^f, v_{\Pi_g(i)}^g)$
  if $(R_f \geq th_{ha})$ {
    $\Pi_f = \mathrm{argmax}_\Pi \sum_{g\in\mathcal{H}(f)\cap\mathcal{F}} \sum_{i=1}^{N} \mathrm{cor}(v_{\Pi(i)}^f, v_{\Pi_g(i)}^g)$
    $\mathcal{F} = \mathcal{F} \cup \{f\}$
  }
}
/* 4. Fix permutations again by neighboring correlations */
while $({}^\exists f \notin \mathcal{F})$ {
  for $({}^\forall f \notin \mathcal{F})$ {
    $R_f = \max_\Pi \sum_{|g-f|\leq\delta, g\in\mathcal{F}} \sum_{i=1}^{N} \mathrm{cor}(v_{\Pi(i)}^f, v_{\Pi_g(i)}^g)$
    $\Pi_f = \mathrm{argmax}_\Pi \sum_{|g-f|\leq\delta, g\in\mathcal{F}} \sum_{i=1}^{N} \mathrm{cor}(v_{\Pi(i)}^f, v_{\Pi_g(i)}^g)$
  }
  $\psi = \mathrm{argmax}_f\ R_f$
  $\mathcal{F} = \mathcal{F} \cup \{\psi\}$
}
```

Fig. 13.8. Pseudo-code for an implementation of the integrated method.

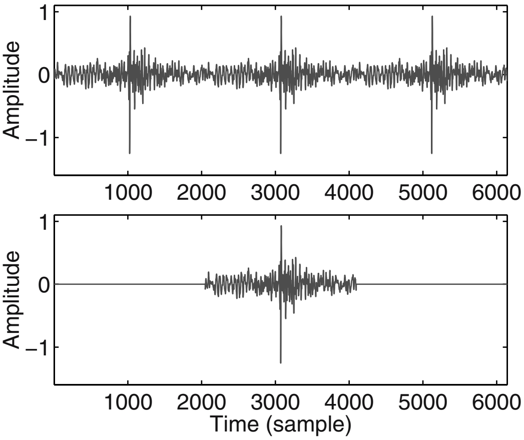


Fig. 13.9. Periodic time-domain filter represented by frequency responses sampled at $L = 2048$ points (above) and its one-period realization (below).

By substituting (13.28) into (13.29), we have a condition for scaling alignment

$$\mathbf{h}_i(f) \approx \mathbf{a}_i(f) h_{J_i i}(f) \Leftrightarrow a_{J_i i}(f) \approx 1.$$

This condition, i.e. $a_{J_i i}(f) = 1$, is attained by

$$\mathbf{W}(f) \leftarrow \mathbf{\Lambda}(f)\mathbf{W}(f), \;\; \mathbf{\Lambda}(f) = \mathrm{diag}(a_{J_1 1}(f), \ldots, a_{J_N N}(f)),$$

where $a_{ji}(f) = [\mathbf{W}^+(f)]_{ji}$ is an element of the pseudoinverse of $\mathbf{W}(f)$.

13.8 Spectral Smoothing

The frequency-domain BSS described in this chapter is influenced by the circularity of discrete frequency representation. The circularity refers to the fact that frequency responses sampled at L points with an interval f_s/L (f_s: sampling frequency) represent a periodic time-domain signal whose period is L/f_s. Figure 13.9 shows two time-domain filters. The upper part of the figure shows a periodic infinite-length filter represented by frequency responses $w_{ij}(f) = [\mathbf{W}(f)]_{ij}$ calculated by ICA at L points. Since this filter is unrealistic, we usually use its one-period realization shown in the lower part.

However, such one-period filters may cause a problem. Figure 13.10 shows impulse responses from a source $\mathrm{s}_k(t)$ to an output $\mathrm{y}_i(t)$ defined by (13.4). Those on the left $\mathrm{u}_{11}(l)$ correspond to the extraction of a target signal, and those on the right $\mathrm{u}_{14}(l)$ correspond to the suppression of an interference signal. The upper responses are obtained with infinite-length filters, and the lower ones with one-period filters. We see that the one-period filters create spikes, which distort the target signal and degrade the separation performance.

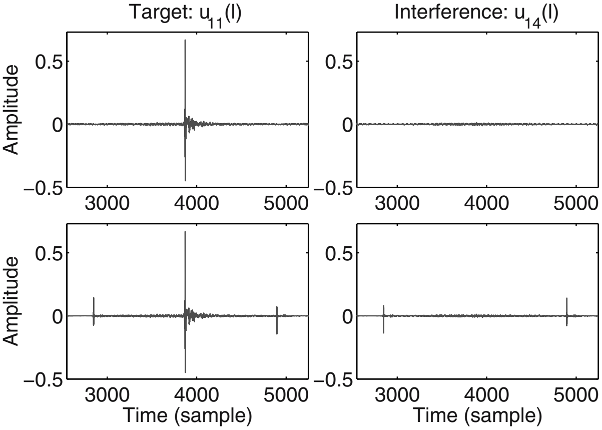


Fig. 13.10. Impulse responses $\mathrm{u}_{ik}(l)$ obtained with periodic filters (above) and with their one-period realization (below).

13.8.1 Windowing

To solve this problem, we need to control the frequency responses $w_{ij}(f)$ so that the corresponding time-domain filter $\mathrm{w}_{ij}(l)$ does not rely on the circularity effect whereby adjacent periods work together to perform some filtering. The most widely used approach is spectral smoothing, which is realized by multiplying a window $\mathrm{g}(l)$ that tapers smoothly to zero at each end, such as a Hanning window $\mathrm{g}(l) = \frac{1}{2}(1+\cos\frac{2\pi l}{L})$. This makes the resulting time-domain filter $\mathrm{w}_{ij}(l)\cdot\mathrm{g}(l)$ fit length L and have small amplitude around the ends [16]. As a result, the frequency responses $w_{ij}(f)$ are smoothed as

$$\tilde{w}_{ij}(f) = \sum_{\phi=0}^{f_s-\Delta f} g(\phi) w_{ij}(f-\phi),$$

where $g(f)$ is the frequency response of $\mathrm{g}(l)$ and $\Delta f = \frac{f_s}{L}$. If a Hanning window is used, the frequency responses are smoothed as

$$\tilde{w}_{ij}(f) = \frac{1}{4}\left[\, w_{ij}(f-\Delta f) + 2w_{ij}(f) + w_{ij}(f+\Delta f)\,\right], \tag{13.30}$$

since the frequency responses $g(f)$ of the Hanning window are $g(0) = \frac{1}{2}$, $g(\Delta f) = g(f_s-\Delta f) = \frac{1}{4}$, and zero for the other frequency bins.

The windowing successfully eliminates the spikes. However, it changes the frequency response from $w_{ij}(f)$ to $\tilde{w}_{ij}(f)$ and causes an error. Let us evaluate the error for each row $\mathbf{w}_i(f) = [w_{i1}(f), \ldots, w_{iM}(f)]^T$ of the ICA solution $\mathbf{W}(f)$. The error is

$$\mathbf{e}_i(f) = \min_{\alpha_i}[\tilde{\mathbf{w}}_i(f) - \alpha_i \mathbf{w}_i(f)] = \tilde{\mathbf{w}}_i(f) - \frac{\tilde{\mathbf{w}}_i(f)^H \mathbf{w}_i(f)}{||\mathbf{w}_i(f)||^2}\mathbf{w}_i(f), \tag{13.31}$$

where $\tilde{\mathbf{w}}_i(f) = [\tilde{w}_{i1}(f), \ldots, \tilde{w}_{iM}(f)]^T$ and α_i is a complex-valued scalar representing the scaling ambiguity of the ICA solution. The minimization $\min_{\alpha_i}$ is based on least-squares, and can be represented by the projection of $\tilde{\mathbf{w}}_i$ to $\mathbf{w}_i$. We can evaluate the error for the Hanning window case by substituting (13.30) for $\tilde{\mathbf{w}}$ of (13.31):

$$\mathbf{e}_i(f) = \frac{1}{4}\,[\,\mathbf{e}_i^-(f) + \mathbf{e}_i^+(f)\,], \tag{13.32}$$

where

$$\mathbf{e}_i^-(f) = \mathbf{w}_i(f-\Delta f) - \frac{\mathbf{w}_i(f-\Delta f)^H \mathbf{w}_i(f)}{||\mathbf{w}_i(f)||^2}\mathbf{w}_i(f), \tag{13.33}$$

$$\mathbf{e}_i^+(f) = \mathbf{w}_i(f+\Delta f) - \frac{\mathbf{w}_i(f+\Delta f)^H \mathbf{w}_i(f)}{||\mathbf{w}_i(f)||^2}\mathbf{w}_i(f). \tag{13.34}$$

This $\mathbf{e}_i^-$ (or $\mathbf{e}_i^+$) represents the difference between two vectors $\mathbf{w}_i(f)$ and $\mathbf{w}_i(f-\Delta f)$ (or $\mathbf{w}_i(f+\Delta f)$). Since these differences are usually not very large, the error $\mathbf{e}_i$ does not seriously affect the separation if we use a Hanning window for spectral smoothing.

13.8.2 Minimizing Error by Adjusting Scaling Ambiguity

Even if the error caused by the windowing is not very large, the separation performance is improved by minimizing the error [26]. The minimization is performed by adjusting the scaling ambiguity of the ICA solution before the windowing. Let $d_i(f)$ be a complex-valued scalar for the scaling adjustment:

$$\mathbf{w}_i(f) \leftarrow d_i(f)\mathbf{w}_i(f). \tag{13.35}$$

We want to find $d_i(f)$ such that the error (13.31) is minimized. The scalar $d_i(f)$ should be close to 1 to avoid any great change in the predetermined scaling. Thus, an appropriate total cost to be minimized is

$$\mathcal{J} = \sum_f J_i(f), \quad J_i(f) = \frac{||\mathbf{e}_i(f)||^2}{||\mathbf{w}_i(f)||^2} + \beta|d_i(f) - 1|^2,$$

and β is a parameter indicating the importance of maintaining the predetermined scaling. With the Hanning window, the error after the scaling adjustment is easily calculated by substituting (13.35) for (13.32)

$$\mathbf{e}_i(f) = \frac{1}{4}\,[\,d_i(f-\Delta f)\mathbf{e}_i^-(f) + d_i(f+\Delta f)\mathbf{e}_i^+(f)\,], \tag{13.36}$$

where $\mathbf{e}_i^-$ and $\mathbf{e}_i^+$ are defined in (13.33) and (13.34), respectively.

The minimization of the total cost can be performed iteratively by

$$d_i(f) = d_i(f) - \mu\frac{\partial \mathcal{J}}{\partial d_i(f)}, \tag{13.37}$$

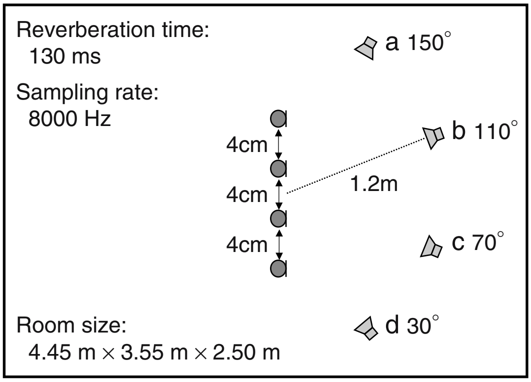


Fig. 13.11. Experimental conditions with linear array.

Table 13.1 Separation performance with linear array.

#sources / position	2 / a c		3 / a b d		4 / a b c d	
Spectral smoothing	no	yes	no	yes	no	yes
Average SIR at microphones (dB)	0.1		-2.9		-4.6	
Average SIR (dB)	20.1	22.3	14.7	17.0	9.3	11.5
Execution time (s)	5.2	5.2	8.0	8.1	12.3	12.4

with a small step-size μ. With the Hanning window, the gradient is

$$\frac{\partial \mathcal{J}}{\partial d_i(f)} = \frac{\partial J_i(f-\Delta f)}{\partial d_i(f)} + \frac{\partial J_i(f+\Delta f)}{\partial d_i(f)} + \frac{\partial J_i(f)}{\partial d_i(f)} \tag{13.38}$$

$$= \frac{\mathbf{e}_i(f-\Delta f)^H \mathbf{e}_i^+(f-\Delta f) + \mathbf{e}_i(f+\Delta f)^H \mathbf{e}_i^-(f+\Delta f)}{8 \cdot ||\mathbf{w}_i(f)||^2} + 2\beta(d_i(f)-1).$$

With equations from (13.36) to (13.38), we can optimize the scalar $d_i(f)$ for the scaling adjustment, and minimize the error caused by spectral smoothing (13.30) with the Hanning window.

13.9 Experimental Results

13.9.1 Linear Array

We performed experiments to separate speech signals in an environment whose conditions are summarized in Fig. 13.11. Our experiments involved two, three and four sources whose locations are indicated in Table 13.1. The sensors were arranged linearly, and the number of sensors used was the same as the number of sources. We used filters of length $L = 2048$ because this

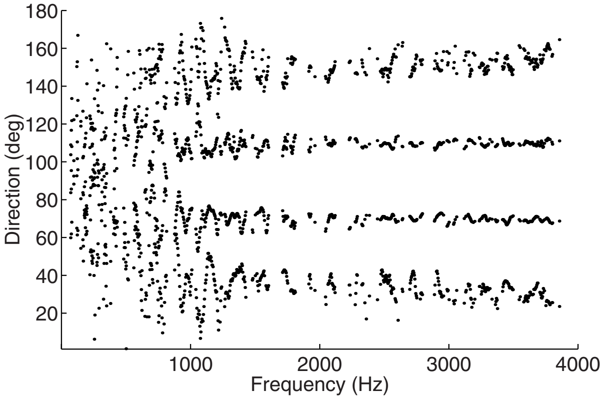


Fig. 13.12. DOA estimations by (13.21) with four sources.

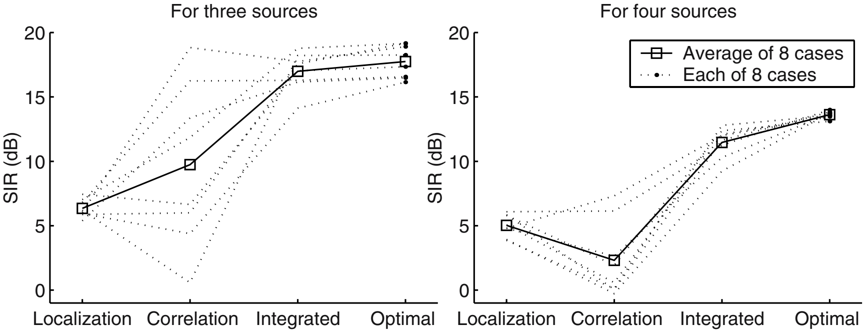


Fig. 13.13. Comparison of different methods for solving the permutation problem.

length provided the best performance under the conditions. The BSS program was coded in Matlab and run on Athlon XP 3200+.

The results shown in Table 13.1 are the average SIRs of eight combinations of 7-second speeches. We see that the spectral smoothing discussed in Sect. 13.8 improves the average SIR with every setup. The short execution time, as shown in Table 13.1, enables the BSS system to perform in real-time if the number of source signals is not very large.

Figure 13.13 shows SIRs for three and four sources with the different methods for solving the permutation problem discussed in Sect. 13.6: "Localization" is the localization approach alone, "Correlation" is the correlation approach (13.26) alone, "Integrated" is the integrated method, and "Optimal" is the optimal solution obtained by utilizing the $\mathrm{s}_i(t)$ and $\mathrm{h}_{ji}(l)$ information. The performance of "Localization" was stable but insufficient. The performance of "Correlation" was unstable and very poor in the four-source cases.

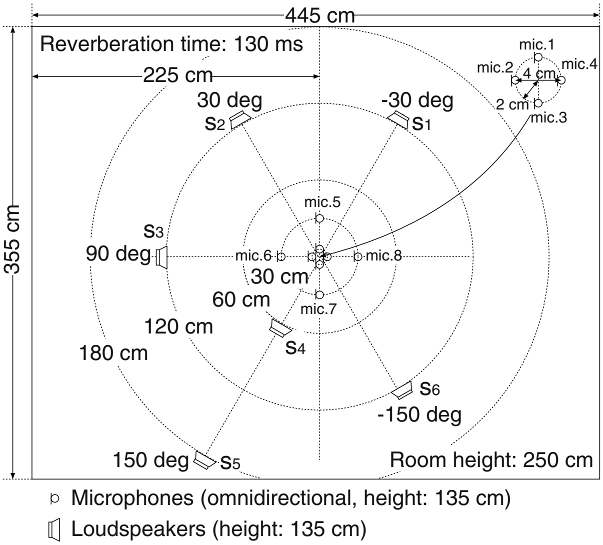


Fig. 13.14. Experimental conditions for planar array case.

The integrated method "Integrated" performed very well and was close to "Optimal."

13.9.2 Planar Array

Then, we carried out experiments on separating six sources with a planar array of eight microphones. The room layout and other experimental conditions are shown in Fig. 13.14. All six sources were 6-second speech signals, and two came from the same direction. The filter length was again $L = 2048$ for an 8 kHz sampling rate.

Let us explain the method for solving the permutation problem in this situation. First, the source directions were estimated with small spacing microphone pairs (1-3, 2-4, 1-2 and 2-3 shown in the right-top corner of Fig. 13.14). This was performed based on (13.20), (13.22) and (13.23). Figure 13.15 shows a histogram of the estimated DOAs. There are five clusters in this histogram, and one cluster is twice the size of the others. This implies that two sources came from the same direction (about 150°). We solved the permutation problem for the other four sources by using this DOA information as shown on the left hand side of Fig. 13.16.

Then, to distinguish between the two sources that came from the same direction, the spheres of these sources were estimated with large spacing microphone pairs (7-5, 7-8, 6-5 and 6-8 shown in the center of Fig. 13.14). This was performed based on (13.19). The right hand side of Fig. 13.16 shows the radiuses of the spheres estimated with microphone pair 7-5. Although the

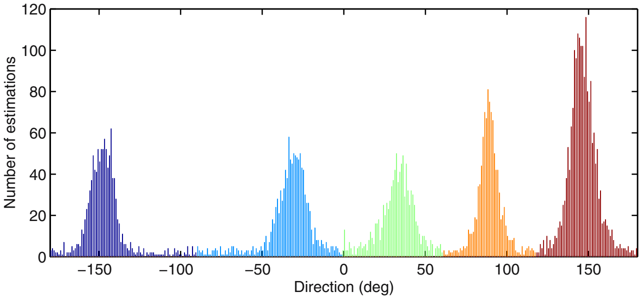


Fig. 13.15. Histogram of DOAs estimated with small spacing microphone pairs.

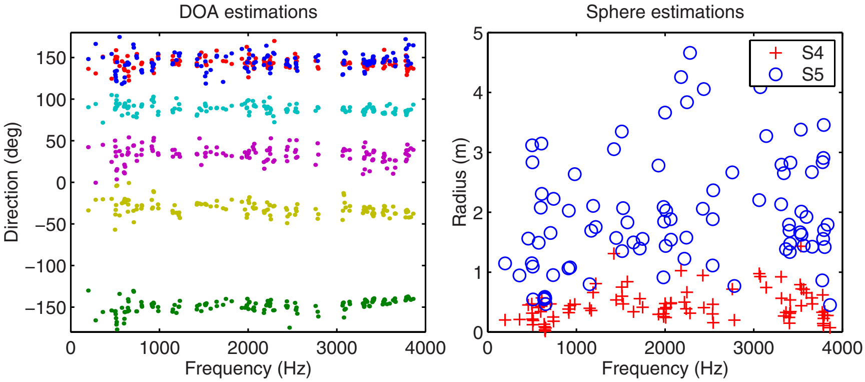


Fig. 13.16. Permutation solved by using estimated DOAs (left) and spheres (right).

radius estimations had large variances, it provided sufficient information to distinguish between the two sources. Consequently, the signal components of all frequencies were classified into six clusters. We determined the permutation only for frequency bins where the classification was reliable as discussed in Sect. 13.6.3.

To show the effectiveness of this method, we compared SIRs by three different methods for the permutation problem. Table 13.2 shows the result. The last row "DOA + Sphere + Correlation" shows the results obtained with this method. The two methods for comparison were "Correlation" where only the correlations (13.26) were maximized, and "DOA + Correlation" where only the DOA information was used for the source localization step in the integrated method (Fig. 13.8). To see how the SIRs were improved, we also measured the SIR of the mixture observed at microphone 1 ("SIR at microphone 1"). The effectiveness of the two integrated methods can again be recognized. If we compare the results of "DOA + Correlation" and "DOA + Sphere + Cor-

Table 13.2 Separation performance with planar array measured by SIR (dB).

	SIR_1	SIR_2	SIR_3	SIR_4	SIR_5	SIR_6	average
SIR at microphone 1	-8.3	-6.8	-7.8	-7.7	-6.7	-5.2	-7.1
Correlation	4.4	2.6	4.0	9.2	3.6	-2.0	3.7
DOA + Correlation	9.6	9.3	14.7	2.7	6.5	14.0	9.4
DOA + Sphere + Correlation	10.8	10.4	14.5	7.0	11.0	12.2	11.0

relation," the improvement is apparent for sources 4 and 5, which came from the same direction. This means that the sphere information was important in terms of distinguishing between sources coming from the same direction. The BSS program was again coded in Matlab and run on Athlon XP 3200+. The computational time for separating six speeches of 6 seconds was around one minute.

13.10 Conclusions

This chapter presented a comprehensive description of frequency-domain BSS, and also various techniques that enable frequency-domain BSS to be used for separating many speeches mixed in a real room environment. The permutation problem has been a major concern with the frequency-domain approach. However, with the methods described in Sect. 13.6, this problem can be solved even in a practical situation. Moreover, the locations of sources can be estimated by the method described in Sect. 13.5. This fact is unique to the frequency-domain approach, and cannot be seen in time-domain BSS. We have shown experimental results where the separation performance was fairly good and the computational cost was practical. These results show the effectiveness of the proposed frequency-domain BSS.

References

1. S. Haykin, Ed., *Unsupervised Adaptive Filtering (Volume I: Blind Source Separation)*. John Wiley & Sons, 2000.
2. A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing*. John Wiley & Sons, 2002.
3. A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. John Wiley & Sons, 2001.
4. T. W. Lee, *Independent Component Analysis - Theory and Applications*. Kluwer Academic Publishers, 1998.
5. S. Amari, S. Douglas, A. Cichocki, and H. Yang, "Multichannel blind deconvolution and equalization using the natural gradient," in *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications*, 1997, pp. 101–104.

6. M. Kawamoto, K. Matsuoka, and N. Ohnishi, "A method of blind separation for convolved non-stationary signals," *Neurocomputing*, vol. 22, pp. 157–171, 1998.
7. K. Matsuoka and S. Nakashima, "Minimal distortion principle for blind source separation," in *Proc. ICA*, 2001, pp. 722–727.
8. S. C. Douglas and X. Sun, "Convolutive blind separation of speech mixtures using the natural gradient," *Speech Communication*, vol. 39, pp. 65–78, 2003.
9. H. Buchner, R. Aichner, and W. Kellermann, "Blind source separation for convolutive mixtures: a unified treatment," in *Audio Signal Processing for Next-Generation Multimedia Communication Systems*, Y. Huang and J. Benesty, Eds., Kluwer Academic Publishers, 2004, pp. 255–293.
10. T. Takatani, T. Nishikawa, H. Saruwatari, and K. Shikano, "High-fidelity blind separation of acoustic signals using SIMO-model-based independent component analysis," *IEICE Trans. Fundamentals*, vol. E87-A, pp. 2063–2072, Aug. 2004.
11. P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, vol. 22, pp. 21–34, 1998.
12. L. Parra and C. Spence, "Convolutive blind separation of non-stationary sources," *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 320–327, May 2000.
13. L. Schobben and W. Sommen, "A frequency domain blind signal separation method based on decorrelation," *IEEE Trans. Signal Processing*, vol. 50, pp. 1855–1865, Aug. 2002.
14. J. Anemüller and B. Kollmeier, "Amplitude modulation decorrelation for convolutive blind source separation," in *Proc. ICA*, 2000, pp. 215–220.
15. 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.
16. F. Asano, S. Ikeda, M. Ogawa, H. Asoh, and N. Kitawaki, "Combined approach of array processing and independent component analysis for blind separation of acoustic signals," *IEEE Trans. Speech Audio Processing*, vol. 11, pp. 204–215, May 2003.
17. S. Kurita, H. Saruwatari, S. Kajita, K. Takeda, and F. Itakura, "Evaluation of blind signal separation method using directivity pattern under reverberant conditions," in *Proc. IEEE ICASSP*, 2000, pp. 3140–3143.
18. H. Saruwatari, S. Kurita, K. Takeda, F. Itakura, T. Nishikawa, and K. Shikano, "Blind source separation combining independent component analysis and beamforming," *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 11, pp. 1135–1146, 2003.
19. M. Z. Ikram and D. R. Morgan, "A beamforming approach to permutation alignment for multichannel frequency-domain blind speech separation," in *Proc. IEEE ICASSP*, 2002, pp. 881–884.
20. H. Sawada, R. Mukai, S. Araki, and S. Makino, "Polar coordinate based nonlinear function for frequency domain blind source separation," *IEICE Trans. Fundamentals*, vol. E86-A, pp. 590–596, Mar. 2003.
21. S. Araki, S. Makino, Y. Hinamoto, R. Mukai, T. Nishikawa, and H. Saruwatari, "Equivalence between frequency domain blind source separation and frequency domain adaptive beamforming for convolutive mixtures," *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 11, pp. 1157–1166, 2003.
22. H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Trans. Speech Audio Processing*, vol. 12, pp. 530–538, Sept. 2004.

23. R. Mukai, H. Sawada, S. Araki, and S. Makino, “Frequency domain blind source separation using small and large spacing sensor pairs,” in *Proc. ISCAS*, vol. V, 2004, pp. 1–4.
24. R. Mukai, H. Sawada, S. Araki, and S. Makino, “Frequency domain blind source separation for many speech signals,” in *Proc. ICA (LNCS 3195)*, 2004, pp. 461–469.
25. S. Winter, H. Sawada, and S. Makino, “Geometrical understanding of the PCA subspace method for overdetermined blind source separation,” in *Proc. IEEE ICASSP*, 2003, pp. 769–772.
26. H. Sawada, R. Mukai, S. de la Kethulle, S. Araki, and S. Makino, “Spectral smoothing for frequency-domain blind source separation,” in *Proc. IWAENC*, 2003, pp. 311–314.
27. A. Hyvärinen, “Fast and robust fixed-point algorithm for independent component analysis,” *IEEE Trans. Neural Networks*, vol. 10, pp. 626–634, 1999.
28. E. Bingham and A. Hyvärinen, “A fast fixed-point algorithm for independent component analysis of complex valued signals,” *International Journal of Neural Systems*, vol. 10, pp. 1–8, Feb. 2000.
29. M. Joho and P. Schniter, “Frequency domain realization of a multichannel blind deconvolution algorithm based on the natural gradient,” in *Proc. ICA*, 2003, pp. 543–548.
30. A. D. Back and A. C. Tsoi, “Blind deconvolution of signals using a complex recurrent network,” in *Proc. Neural Networks for Signal Processing*, 1994, pp. 565–574.
31. R. H. Lambert and A. J. Bell, “Blind separation of multiple speakers in a multipath environment,” in *Proc. IEEE ICASSP*, 1997, pp. 423–426.
32. T. W. Lee, A. J. Bell, and R. Orglmeister, “Blind source separation of real world signals,” in *Proc. ICNN*, 1997, pp. 2129–2135.
33. J. J. Shynk, “Frequency-domain and multirate adaptive filtering,” *IEEE Signal Processing Magazine*, vol. 9, pp. 14–37, Jan. 1992.
34. H. Sawada, S. Winter, R. Mukai, S. Araki, and S. Makino, “Estimating the number of sources for frequency-domain blind source separation,” in *Proc. ICA (LNCS 3195)*, 2004, pp. 610–617.
35. A. Bell and T. Sejnowski, “An information-maximization approach to blind separation and blind deconvolution,” *Neural Computation*, vol. 7, pp. 1129–1159, 1995.
36. S. Amari, “Natural gradient works efficiently in learning,” *Neural Computation*, vol. 10, pp. 251–276, 1998.
37. J.-F. Cardoso, “Blind beamforming for non-Gaussian signals,” *IEE Proceedings-F*, pp. 362–370, Dec. 1993.
38. K. Matsuoka, M. Ohya, and M. Kawamoto, “A neural net for blind separation of nonstationary signals,” *Neural Networks*, vol. 8, pp. 411–419, 1995.
39. R. O. Schmidt, “Multiple emitter location and signal parameter estimation,” *IEEE Trans. Antennas and Propagation*, vol. 34, pp. 276–280, Mar. 1986.
40. H. Sawada, R. Mukai, and S. Makino, “Direction of arrival estimation for multiple source signals using independent component analysis,” in *Proc. International Symposium on Signal Processing and its Applications*, 2003, pp. 411–414.
41. B. D. Van Veen and K. M. Buckley, “Beamforming: a versatile approach to spatial filtering,” *IEEE ASSP Magazine*, pp. 2–24, Apr. 1988.

42. J.-F. Cardoso, "Source separation using higher order moments," in *Proc. IEEE ICASSP*, vol. 4, 1989, pp. 2109–2112.
43. V. C. Soon, L. Tong, Y. F. Huang, and R. Liu, "A robust method for wideband signal separation," in *Proc. ISCAS*, vol. 1, 1993, pp. 703–706.
44. R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. Wiley Interscience, 2000.

14 Subband Based Blind Source Separation

Shoko Araki and Shoji Makino

NTT Communication Science Laboratories
Soraku-gun, Kyoto 619-0237, Japan
E-mail: {shoko, maki}@cslab.kecl.ntt.co.jp

Abstract. In this chapter, we address subband-based blind source separation (BSS) for convolutive mixtures of speech by reporting a large number of experimental results. The subband-based BSS approach offers a compromise between time-domain and frequency-domain techniques. The former is usually difficult and slow with many separation filter coefficients to estimate. With the latter it is difficult to estimate statistics when the adaptation data length is insufficient. With subband-based BSS, a sufficient number of samples for estimating statistics can be held in each subband by using a moderate number of subbands. Moreover, by using FIR filters in each subband, which are shorter than the filters used for time-domain BSS, we can handle long reverberation. In addition, subband-based BSS allows us to select the separation method suited to each subband. Using this advantage, we introduce efficient separation procedures that take both the frequency characteristics of the room reverberation and speech signals into consideration. In concrete terms, longer separation filters and an overlap-blockshift in BSS's batch adaptation in low frequency bands improve the separation performance. Consequently, frequency-dependent subband processing is successfully realized with subband-based BSS.

14.1 Introduction

Blind source separation (BSS) is an approach that estimates original source signals $\mathrm{s}_i(n)$ using only information on the mixed signals $\mathrm{x}_j(n)$ observed in each input channel. This technique can be applied for audio applications such as noise robust speech recognition, high-quality hands-free telecommunication, and hearing aid systems.

We consider the BSS of speech signals in a real environment, i.e., the BSS of convolutive mixtures of speech. In a real environment, signals are filtered by an acoustic room channel. To separate such complicated mixtures, we need to estimate the separation filters of several thousand taps. Several methods have been proposed for achieving the BSS of convolutive mixtures [1] and most of these utilize independent component analysis (ICA) [2], [3]. To solve the convolutive BSS problem, algorithms in the time and frequency domains have been proposed [4–12].

In time-domain BSS, ICA is directly applied to convolutive mixtures and separation FIR filters are directly estimated (e.g., [4–8]). Therefore, the independence of output signals can be evaluated directly. However, the convergence of time-domain BSS algorithms is generally not good. This is because the adaptation of such a long separation filter is very complex and there are many local minima [3]. The computational complexity is also a problem. Moreover, most time-domain BSS algorithms have another problem: the whitening effect, which means the signal's spectrum becomes flat. Because most time-domain BSS algorithms were designed for i.i.d. signals, these algorithms try to make output signals both spatially and temporally independent [8]. When we apply such time-domain BSS algorithms to mixtures of speech signals, the output speech signals are whitened and sound unnatural.

By contrast, in frequency-domain BSS, mixtures are converted into the frequency domain and ICA is applied to instantaneous mixtures at each frequency (e.g., [9–12]) as shown in the previous chapter. Although we can greatly reduce the computational complexity by using frequency-domain BSS, frequency-domain BSS algorithms have inherent issues. One is that the independence is evaluated at each frequency. In a real environment, an impulse response changes momentarily. Therefore it is preferable that we estimate separation filters using adaptation data that are as short as possible, especially when we use a batch algorithm. However, when we apply a longer frame that can cover realistic reverberation for speech mixtures of a few seconds, the number of samples in each frequency bin becomes small, and therefore, we cannot correctly estimate the statistics in each frequency bin [13]. This means that, in such a case, the independence is not evaluated correctly. This is our strongest reason for utilizing subband-domain BSS method. We also face permutation and scaling problems, which result in the estimated source signal being recovered with a different permutation and gain in different frequency bins. Recently, some solutions have been provided for these problems [12], [14–17] and some of these were introduced in the previous chapter.

Motivated by these facts, we introduce a BSS method that employs subband processing [18], [19]. Hereafter, we call this method subband BSS. With subband BSS, observed mixed signals are decomposed into the subband domain with a filterbank and then separated in each subband using a time-domain BSS algorithm. Then separated signals in each subband are synthesized to obtain fullband separated signals. With this method, we can choose a moderate number of subbands, therefore, we can maintain a sufficient number of samples in each subband. The subband system also allows us to estimate FIR filters as separation filters in each subband. Moreover, as the separation filter length in each subband is shorter than that for time-domain BSS, it is easier to estimate separation filters than with time-domain BSS. Therefore, we can obtain separation filters that are long enough to cover reverberation. That is, the subband BSS approach copes with both the frequency-domain approach's difficulty in estimating statistics and the time-domain technique's difficulty in adapting many parameters.

In addition, subband BSS mitigates the permutation problem and whitening effect. Because the permutation problem does not occur within each subband, there are few permutation problems in subband BSS. Moreover, because the whitening effect can be limited in each subband, subband BSS can mitigates the whitening effect. Of course, subband BSS reduces computational complexity [20], [21]. This is an additional merit of subband BSS.

Subband BSS offers another advantage in that it allows us to select the separation method suited to each subband. By using this advantage, we can employ an efficient separation procedure taking into consideration the frequency characteristics of room reverberation and speech signals [22], [23]. Generally speaking, an impulse response is usually longer in low frequency bands than in high frequency bands. This makes the separation in low frequency bands difficult. Moreover, because speech signals have high power in low frequency bands, the separation performance in low frequency bands dominates the speech separation performance. Therefore, it is very important to improve the separation performance in the low frequency bands for speech separation. In this chapter, we utilize longer separation filters and the overlap-blockshift technique in the low frequency bands.

The organization of this chapter is as follows. Section 14.2 describes the framework for the BSS of convolutive mixtures of speech. In Section 14.3, we explain the configuration of subband BSS and mention implementation issues. We confirm the validity of subband BSS in Section 14.4 by describing experiments undertaken with reverberant data. In Section 14.5, we show some ways to improve the low frequency subband performance in which the SIR is worse than at high frequencies. Here, we take into consideration the frequency characteristics of room reverberation and speech signals. The final section concludes this chapter.

14.2 BSS of Convolutive Mixtures

14.2.1 Model Description

In real environments, the observed microphone signals are affected by reverberation. Therefore, N_s signals recorded by N_m microphones are modeled as convolutive mixtures

$$\mathrm{x}_j(n) = \sum_{i=1}^{N_s} \sum_{l=1}^{P} \mathrm{h}_{ji}(l)\mathrm{s}_i(n-l+1), \;\; j = 1, \cdots, N_m, \tag{14.1}$$

where s_i is the source signal from a source i, x_j is the signal observed by a microphone j, and h_{ji} is the P-taps impulse response from source i to microphone j.

In order to obtain separated signals, we estimate the separation filters $\mathrm{w}_{ij}(n)$ of Q-taps, and obtain the separated signals

$$\mathrm{y}_i(n) = \sum_{j=1}^{N_m} \sum_{l=1}^{Q} \mathrm{w}_{ij}(l)\mathrm{x}_j(n-l+1), \;\; i = 1, \cdots, N_s. \tag{14.2}$$

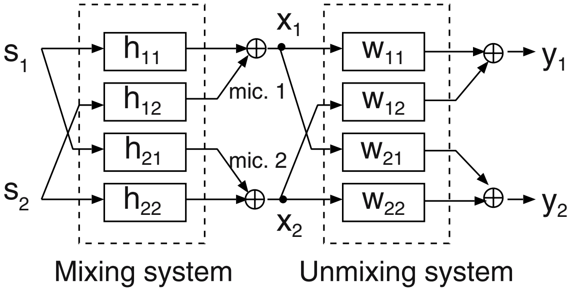


Fig. 14.1. BSS system configuration (when $N_s = N_m = 2$).

The separation filters are estimated so that the separated signals become mutually independent.

The BSS block diagram is shown in Fig. 14.1 for $N_s = N_m = 2$. In this chapter, we consider the case of $N_s = N_m = N_{sm}$.

14.2.2 Frequency-Domain BSS and Related Issue

Frequency-domain BSS. The frequency-domain approach to convolutive mixtures transforms the problem into an instantaneous BSS problem at each frequency [9–12]. Using T-point short-time Fourier transformation for (14.1), we obtain the approximate time-frequency representation of mixtures,

$$\mathbf{x}(f,m) = \mathbf{H}(f)\mathbf{s}(f,m), \quad m = 0, \cdots, L_m - 1, \tag{14.3}$$

where f denotes the frequency bin, m represents the time dependence of the short-time Fourier transformation (STFT), L_m is the number of data samples in each frequency bin, $\mathbf{s}(f,m) = [s_1(f,m), \cdots, s_{N_{sm}}(f,m)]^T$ is the source signal vector, and $\mathbf{x}(f,m) = [x_1(f,m), \cdots, x_{N_{sm}}(f,m)]^T$ is the observed signal vector. We assume that the $(N_{sm} \times N_{sm})$ mixing matrix $\mathbf{H}(f)$ is invertible and that its ji component $h_{ji}(f) \neq 0$. The STFT is usually executed by applying a window function of length T. In this chapter, we call this T the STFT frame size.

The separation process can be formulated in a frequency bin f:

$$\mathbf{y}(f,m) = \mathbf{W}(f)\mathbf{x}(f,m), \quad m = 0, \cdots, L_m - 1, \tag{14.4}$$

where $\mathbf{y}(f,m) = [y_1(f,m), \cdots, y_{N_{sm}}(f,m)]^T$ is the separated signal vector, and $\mathbf{W}(f)$ represents an $(N_{sm} \times N_{sm})$ separation matrix at frequency f. In this chapter, we assume that the STFT frame size T is equal to the separation filter length Q. The separation matrix $\mathbf{W}(f)$ is determined by ICA so that the outputs $y_i(f,m)$ become mutually independent. This calculation is carried out independently at each frequency.

Dilemma of Frequency-Domain BSS. In order to handle long reverberation, we need to estimate long separation filters $\mathrm{w}_{ij}(n)$ of Q-taps. If the filters are relatively short, we cannot reduce the reverberant components of interferences that are longer than the filters and this has a detrimental effect on the separation performance [24]. On the other hand, with a batch adaptation, it is desirable that separation filters can be estimated using adaptation data that are as short as possible. This is because an impulse response changes momentarily in a real environment. We therefore have to estimate long separation filters with short length of adaptation data.

However, we have reported in [13] that when we employ a long frame T with a frame shift of $T/2$ for several seconds of data in order to prepare a separation filter long enough to cover reverberation (note that we are assuming $T = Q$), the separation performance degrades. One reason for this is that it becomes difficult to maintain a sufficient number of data samples to estimate the statistics in each frequency. This makes the estimation of statistics difficult. In particular, the independence assumption between the source signals seems to collapse [13]. Therefore, we cannot obtain sufficient separation performance with a long frame with frequency-domain BSS for short adaptive data.

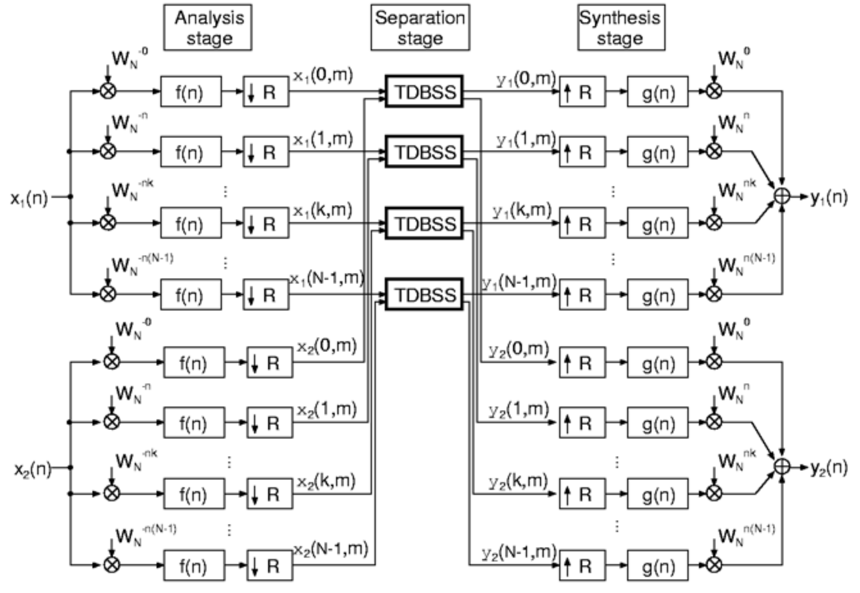
14.3 Subband Based BSS

Subband BSS discussed in this section provides a solution to the dilemma of frequency-domain BSS described in the previous section. With this method, we can choose a moderate number of subbands, and therefore maintain a sufficient number of samples in each subband. Subband BSS also allows us to estimate short FIR filters as separation filters in each subband, due to the down-sampling procedure at the subband analysis stage. Therefore, we should be able to obtain a separation filter long enough to cover reverberation. Moreover, as the separation filter length in each subband is shorter than that for time-domain BSS, it is easier to estimate separation filters than in time-domain BSS. That is, the subband BSS approach offers a compromise between a time-domain technique, which is usually difficult and slow with many parameters to estimate, and a frequency domain technique, which has difficulty estimating statistics.

14.3.1 Configuration of Subband BSS

Basic Configuration of Subband BSS. The subband BSS system is composed of three parts: a subband analysis stage, a separation stage, and a subband synthesis stage (Fig. 14.2) [18], [19].

First, in the subband analysis stage, input signals $\mathrm{x}_j(n)$ are divided into N subband signals $x_j(k, m)$, $k = 0 \cdots, N-1$, where k is the subband index, m is the time index, and N is the number of subbands. A polyphase



N: # of subband, R: down-sampling rate, $W_N = e^{j2\pi/N}$
f(n): prototype filter for analysis, g(n): prototype filter for synthesis

Fig. 14.2. Basic system configuration of subband BSS. TDBSS denotes time-domain BSS. A 2 × 2 case is depicted.

filterbank [25], including a cosine modulated filterbank [20] and a discrete Fourier transform (DFT) filterbank [21,26], is widely used as the subband analysis/synthesis system, because of its low computational complexity. A polyphase filterbank analyzer (synthesizer) basically consists of a modulator (demodulator), a prototype filter with a low pass characteristic, and a decimator (interpolator). The cosine modulated filterbank realizes a perfect reconstruction filterbank with real valued coefficients. The DFT filterbank can be effectively realized by using FFT, however, the subband analyzed signals $x_j(k,m)$ become complex number sequences. Since the outputs of a prototype filter are band-limited in each subband, we can employ decimation at the down-sampling rate R. However, as it is impossible to make an ideal low pass filter as a prototype filter, the adjacent bands overlap each other, i.e., aliasing occurs. Therefore, we should use a down-sampling rate of $R < N$ in order to reduce the aliasing distortion [27], which degrades the separation performance [20].

Then, time-domain BSS is executed on $x_j(k,m)$ and the separated signals $y_j(k,m)$ are obtained in each subband in the separation stage. If we utilize DFT filterbanks, we have to use a complex version of the time-domain BSS algorithm [21]. In each subband, we estimate FIR filters as separation filters so as to cover the reverberation. Since we employ down-sampling, short FIR

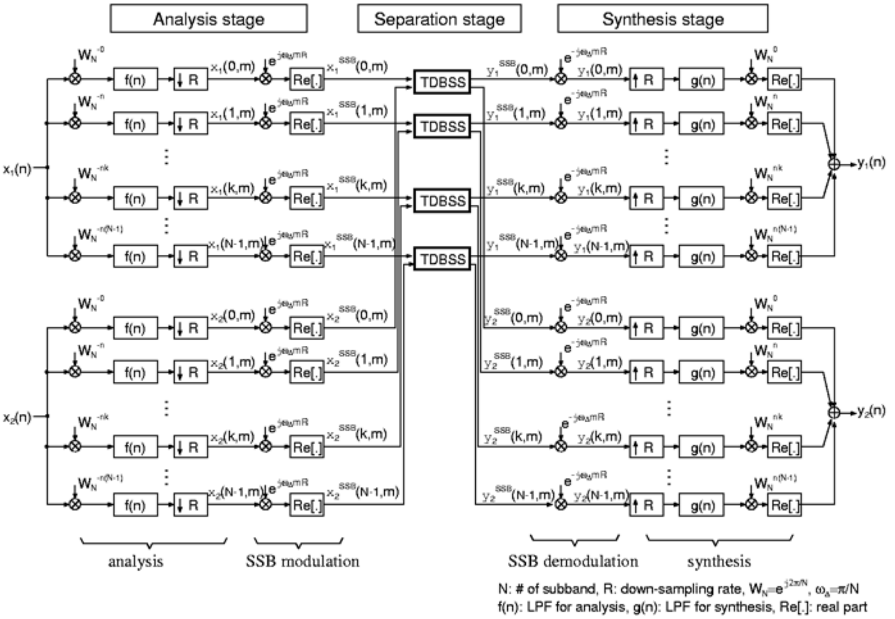


Fig. 14.3. Block diagram of subband BSS with an SSB filterbank. TDBSS denotes time-domain BSS. LPF denotes low pass filter. A 2 × 2 case is depicted.

filters of length Q/R are sufficient to separate the subband signals in each subband.

Finally, in the subband synthesis stage, separated signals $\mathrm{y}_i(n)$ are obtained by synthesizing each separated signal $y_i(k, m)$.

Subband BSS with SSB Filterbank. In this chapter, we utilize a polyphase filterbank [25] with single sideband (SSB) modulation [28], which is widely used in the echo canceller area [29,30]. A block diagram of subband BSS with an SSB filterbank is shown in Fig. 14.3. Since this also has the form of a generalized discrete Fourier transform (GDFT) filterbank [28], the filterbank can be realized effectively by FFT. Furthermore, in order to make the analyzed signals real-valued, SSB modulation is adopted in the analysis stage (Fig. 14.3). Moreover, to avoid the aliasing problem, the SSB-modulated subband signals are not critically sampled, but oversampled, i.e., $R < N$. Here, we employ two-times oversampling $R = \frac{N}{4}$. The low-pass filter used here in the analysis filterbank as a prototype filter is $f(n) = \mathrm{sinc}(\frac{n\pi}{N/2})$ of length $6N$. By using SSB modulation, we obtain SSB modulated real-valued signals $x_j^{\mathrm{SSB}}(k, m)$ in each subband.

Thanks to the SSB modulation, in the separation stage, we can apply the time-domain BSS algorithm to $x_j^{\mathrm{SSB}}(k, m)$ without expanding it into a complex-valued version. A detailed explanation of the time-domain BSS algorithm we employed is provided in the following subsection.

After obtaining the separated signals $y_i^{\mathrm{SSB}}(k,m)$ in each subband, we execute the SSB demodulation and synthesize them to obtain output signals $\mathrm{y}_i(n)$ in the time domain. The low-pass (prototype) filter used in the synthesis filterbank is $g(n) = \mathrm{sinc}(\frac{n\pi}{R/2})$ of length $6R$.

14.3.2 Time-Domain BSS Implementation for a Separation Stage

Thanks to the SSB modulation, we can use any real-valued time-domain BSS algorithm for subband BSS, including a higher order statistics based algorithm [4,5] and a second order statistics based algorithm [6,31]. A generic framework is discussed in [7]. Here, we describe the algorithm we used in the experiments reported in this chapter. In addition, this section describes how we can design the initial values of the separation filters for each subband.

Time-Domain BSS Algorithm. Here, we employ an algorithm based on time-delayed decorrelation for non-stationary signals [31]. Relying on the non-stationarity and non-whiteness of the source signals, this algorithm minimizes the cross-correlation of output signals for some time lags for all analysis blocks, simultaneously. It is verified that this algorithm works for convolutive mixtures of speech signals [32].

We estimate FIR filters as the separation filters $w_{ij}^k(m)$ in each subband k. We write them in a matrix form $\mathbf{W}^k(m)$ where its ij component is $w_{ij}^k(m)$ for convenience. The adaptation rule of the i-th iteration is

$$\mathbf{W}_{i+1}^k(m) = \mathbf{W}_i^k(m) + \frac{\alpha}{BS}\sum_{b=0}^{BS-1}\{(\mathrm{diag}\mathbf{R}_y^b(0))^{-1}(\mathrm{diag}\mathbf{R}_y^b(m)) \\ -(\mathrm{diag}\mathbf{R}_y^b(0))^{-1}\mathbf{R}_y^b(m)\} * \mathbf{W}_i^k(m), \qquad (14.5)$$

where $\mathbf{R}_y^b(\tau)$ represents the covariance matrix of outputs $\mathbf{y}(m) \equiv [y_1^{\mathrm{SSB}}(k,m), \cdots, y_{N_{sm}}^{\mathrm{SSB}}(k,m)]^T$ in the b-th (b=0, $\cdots$, $B-1$) analysis block with time delay τ, [i.e., $\mathbf{R}_y^b(\tau) = \frac{1}{L}\sum_{t=1}^{L}\mathbf{y}(b\frac{L}{S}+t)\mathbf{y}^T(b\frac{L}{S}+t-\tau)$], α denotes a step-size parameter, $*$ denotes a convolution operator, L is the block length and S is the blockshift rate. Note that the algorithm we used here is a *batch* algorithm, i.e., the algorithm runs by using all the data on each iteration.

Initial Value Design of Separation Filters. A suitable initialization of the separation filters helps the convergence of time-domain BSS and mitigates the permutation problem in subband BSS. We can use constraint null beamformers, which makes spatial nulls towards given directions, as the initial value of the separation filters [32]. This is based on the fact that the

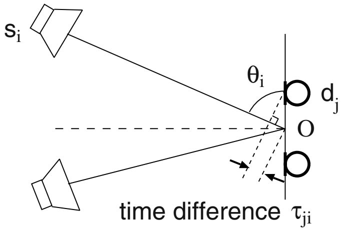


Fig. 14.4. Setup of null beamformer.

BSS solution behaves as adaptive beamformers, which form nulls in the jammer directions [33]. Based on this fact, we design null beamformers towards possible sound directions and utilize them as our initial values for the BSS adaptation. Here, we give an example.

Here, we assume a linear microphone array with a known microphone spacing. First, we assume that the mixing matrix $\mathbf{H}(f)$ represents only the time difference of direct sound arrival τ_{ji} with respect to the midpoint between the microphones (Fig. 14.4). This $\mathbf{H}(f)$ is written in the frequency domain as follows:

$$\mathbf{H}(f) = \begin{bmatrix} \exp(\jmath 2\pi f \tau_{11}) & \cdots & \exp(\jmath 2\pi f \tau_{1N_{sm}}) \\ \vdots & \ddots & \vdots \\ \exp(\jmath 2\pi f \tau_{N_{sm}1}) & \cdots & \exp(\jmath 2\pi f \tau_{N_{sm}N_{sm}}) \end{bmatrix}, \tag{14.6}$$

where $\tau_{ji} = \frac{d_j}{c}\cos\theta_i$, d_j is the position of the j-th microphone, θ_i is the direction of the i-th source as an initial value, and c is the speed of sound. Note that these d_j values need not be precise because this $\mathbf{H}(f)$ is used only for the initialization of BSS. It should be also noted that the precise directions of sources, which are not given in a blind scenario, are not required for the initialization. That is the θ_i values can be very rough approximations, e.g., ±60° for the 2×2 case (i.e., left position or right position, for example).

Then we calculate the inverse of $\mathbf{H}(f)$ at each frequency, $\mathbf{W}(f) = \mathbf{H}^{-1}(f)$ and convert the elements $w_{ij}(f)$ of this $\mathbf{W}(f)$ into the time domain, $\mathrm{w}_{ij}(n) = \mathrm{IFFT}(w_{ij}(f))$. We can use this $\mathrm{w}_{ij}(n)$ as the initial value for time-domain BSS. Then, by applying subband analysis on these $\mathrm{w}_{ij}(n)$, we obtain the initial values of the separation filters in each subband $\mathbf{W}_0^k(m)$ for (14.5).

14.3.3 Solving the Permutation and Scaling Problems

Scaling and permutation problems occur in subband BSS in a way similar to that found with frequency-domain BSS, i.e., the estimated source signal components are recovered with a different order and gain in the different frequencies. Thanks to the initial value mentioned in the previous subsection, we can mitigate the permutation problem, however, it sometimes still occurs.

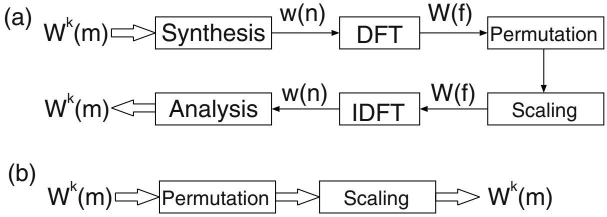


Fig. 14.5. Flows to solve the permutation and scaling problems (a) in the frequency domain and (b) in the subband domain.

In order to solve the permutation problem, we can also employ an adaptive-beamformer-like characteristic of the BSS solution [34]. We can solve the problem by reordering the row of estimated separation filters $\mathbf{W}^k(m)$ so that the null of the directivity pattern obtained by $\mathbf{W}^k(m)$ is sorted and forms a null toward almost the same direction in all subbands. This procedure is easily realized by looking at the directivity pattern of $\mathbf{W}(f)$ in the frequency domain [Fig. 14.5 (a)] [34]. We can also solve the permutation problem by sorting the row of the estimated separation filters $\mathbf{W}^k(m)$ so that the cross-correlation of separated signals $y_i^{\text{SSB}}(k,m)$ in adjacent subbands is maximized [12], [14]. With the correlation method, we can solve the problem in the subband domain [Fig. 14.5 (b)].

For the scaling problem, we can also use the directivity pattern calculated with the separation filters [35], that is, we normalize the row of the estimated separation filters $\mathbf{W}^k(m)$ so that the gains and phases of the target directions become 0 dB and 0, respectively. It can be performed by transforming $\mathbf{W}^k(m)$ into the frequency domain [Fig. 14.5 (a)] [34]. The minimal distortion principle [17] or the projection back method [10] can also be employed for $\mathbf{W}(f)$ to solve the scaling problem [32], e.g., $\mathbf{W}(f) \leftarrow \text{diag}[\mathbf{W}^{-1}(f)]\mathbf{W}(f)$. We can also solve the problem naively by normalizing the separation filters $\mathbf{W}^k(m)$ so that each component $w_{ij}^k(m)$ has the same power as the corresponding component of null beamformers $\mathbf{W}_{\text{NBF}}^k(m)$, which have nulls in the jammer directions. This can be executed in the subband domain [Fig. 14.5 (b)].

We can combine some solutions mentioned above. Here is one of the solutions to the permutation and scaling problems which we employed:

i) Synthesize $\mathbf{W}^k(m)$ to obtain $\mathbf{W}(n)$ in the time-domain, then obtain $\mathbf{W}(f)$ using a discrete Fourier transform (DFT).
ii) Estimate signal directions θ_i $(i = 1, \cdots, N_{sm})$ from the directivity gain pattern of $\mathbf{W}(f)$ [35]. When $N_{sm} \geq 3$, it is recommended that signal directions be estimated analytically from $\mathbf{W}(f)$ [36].
iii) Solve the permutation problem by reordering the $\mathbf{W}(f)$ row so that the θ_i values are sorted.

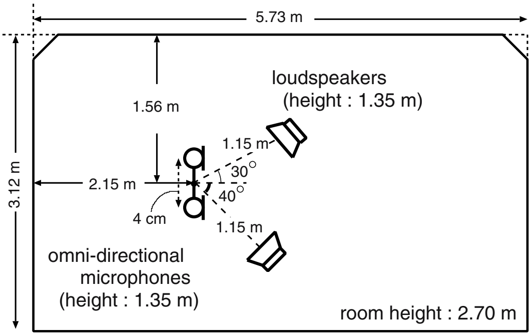


Fig. 14.6. Layout of room used in experiments. T_{R} =300 ms.

iv) Make null beamformers by using (14.6) with the estimated θ_i in step ii), and by calculating $\mathbf{W}(f) = \mathbf{H}^{-1}(f)$. We call this null beamformer $\mathbf{W}_{\mathrm{NBF}}(f)$ and use it to solve the scaling problem.

v) Calculate the inverse DFT of $\mathbf{W}_{\mathrm{NBF}}(f)$ and perform subband analysis to obtain $\mathbf{W}^k_{\mathrm{NBF}}(m)$.

vi) Rescale $\mathbf{W}^k(m)$ so that $||w^k_{ij}(m)|| = ||w^k_{\mathrm{NBF}ij}(m)||$, where $||x(m)||$ means $\sum_m^{Q_k} x^2(m)$ and Q_k is the separation filter length in the k-th subband.

14.4 Basic Experiments for Subband BSS

14.4.1 Experimental Setup

In order to confirm the performance of subband BSS, we undertook separation experiments using speech data convolved with impulse responses measured in a real environment for a 2 × 2 case. The impulse responses were measured in the room shown in Fig. 14.6. The reverberation time T_{R} was 300 ms. Since the sampling rate was 8 kHz, 300 ms corresponds to 2400 taps. As the original speech, we used two sentences spoken by two male and two female speakers. Investigations were carried out for six combinations of speakers. Each mixed speech signal was about eight seconds long. We used the first three seconds of the mixed data for learning, and we separated the entire eight second data.

To evaluate the performance, we used the signal-to-interference ratio (SIR), defined as

$$\mathrm{SIR}_i = \mathrm{SIR}_{\mathrm{O}i} - \mathrm{SIR}_{\mathrm{I}i}, \tag{14.7}$$

$$\mathrm{SIR}_{\mathrm{O}i} = 10 \log \frac{\sum_n \mathrm{y}^2_{i\mathrm{s}_i}(n)}{\sum_n (\sum_{j \neq i} \mathrm{y}_{i\mathrm{s}_j}(n))^2},$$

$$\mathrm{SIR}_{\mathrm{I}i} = 10 \log \frac{\sum_n \mathrm{x}^2_{k\mathrm{s}_i}(n)}{\sum_n (\sum_{j \neq i} \mathrm{x}_{k\mathrm{s}_j}(n))^2},$$

where $\mathrm{y}_{i\mathrm{s}_j}$ is the output of the whole system at y_i when only s_j is active, and $\mathrm{x}_{k\mathrm{s}_i} = \mathrm{h}_{ki} * \mathrm{s}_i$ ($*$ is a convolution operator, $k = i$ in our experiments). SIR is the ratio of a target-originated signal to jammer-originated signals.

14.4.2 Subband System

For subband analysis and synthesis, we used a polyphase filterbank [25] with single sideband (SSB) modulation/demodulation [28], which we mentioned in Section 14.3.1. Here, the number of subbands N was 64 and the down-sampling rate R was 16 ($R = \frac{N}{4}$). We decided this number of subbands N so that the down-sampling rate of subband BSS corresponded to that of conventional frequency-domain BSS (see Section 14.4.3) of frame size $T = 32$ with the half frame-shift.

For the time-domain algorithm used in subband BSS, we estimated separation filters $w_{ij}^k(m)$ of 64 and 128-taps in each subband. The step-size for adaptation α was 0.02 and the number of blocks B was fixed at 20 for three seconds of speech. We adopted θ_i=±60° as the initial values of the separation filters (see Section 14.3.2).

14.4.3 Conventional Frequency-Domain BSS

The frequency-domain BSS iteration algorithm was a natural gradient based algorithm

$$\varDelta\mathbf{W}_i(f) = \eta\left[\mathrm{diag}\left(\langle\varPhi(\mathbf{y})\mathbf{y}^{\mathrm{H}}\rangle\right) - \langle\varPhi(\mathbf{y})\mathbf{y}^{\mathrm{H}}\rangle\right]\mathbf{W}_i(f),$$

where $\mathbf{y}$=$\mathbf{y}(f,m)$, superscript H denotes a conjugate transpose and $\langle x(m)\rangle$ denotes the time average with respect to time m: $\frac{1}{L_m}\sum_{m=0}^{L_m-1} x(m)$. Subscript i is used to express the value of the i-th step in the iterations, η is a step-size parameter, and $\varPhi(\cdot)$ is a nonlinear function. As the nonlinear function $\varPhi(\cdot)$, we used $\varPhi(\mathbf{y}) = \tanh(g \cdot \mathrm{abs}(\mathbf{y}))e^{j\arg(\mathbf{y})}$ [37], where g is a parameter to control the nonlinearity and we utilized $g = 100$. As the initial value of the separation matrix, we utilized $\mathbf{W}(f) = \mathbf{H}^{-1}(f)$ with θ_i=±60° (see Section 14.3.2).

We fixed the frame shift at half the STFT frame size T, so that the number of samples in the time-frequency domain were the same. To solve the scaling and permutation problems, we also used the beamforming approach [34]: first, from the directivity pattern obtained by $\mathbf{W}(f)$ we estimated the source directions and reordered the row of $\mathbf{W}(f)$ so that the directivity pattern formed a null toward the same direction in all frequencies, then we normalized the row of $\mathbf{W}(f)$ so that the gains of the target directions became 0 dB.

It should be noted that we used the time-average of $\mathbf{y}(f,m)$ of three seconds for adaptation, i.e., we used a *batch* algorithm. It should also be noted that if we fix the data length and frame shift at half the frame size, the number of samples L_m of sequences $\mathbf{y}(f,m)$ in each frequency depends on the frame size T: roughly speaking, $L_m \propto$ (data length)$/T$.

Here we utilized the frequency-domain algorithm based on higher order statistics (HOS) despite the fact that we are using a time-domain algorithm relied on second order statistics (SOS). The performance of time-domain BSS based on SOS has already been compared with that based on HOS [38], and it was shown that the performance is not significantly different when we use an adaptive-beamformer-like initial value. It has also been shown [39] that the decorrelation-based algorithm and the fourth order moment-based algorithm perform identically for speech. Therefore, we consider that we will see the same tendency as that shown by our results if we compare time- subband- and frequency-domain BSS using HOS/SOS only.

14.4.4 Conventional Fullband Time-Domain BSS

We also examined fullband time-domain BSS. The algorithm was the same as that used in subband BSS, i.e., (14.5). In this case, the output signal vector $\mathbf{y}(n)$ consisted of the signals in the time domain $[\mathrm{y}_1(n), \cdots, \mathrm{y}_{N_{sm}}(n)]^T$. We used values of $\alpha = 0.002$ and $B = 20$. To obtain the initial condition of the separation filters, we also utilized $\mathbf{W}(f) = \mathbf{H}^{-1}(f)$ with θ_i=±60° and converted it into the time domain (see Section 14.3.2).

In fullband time-domain BSS, the output speech signals are distorted and whitened (see [40] and Section 14.4.6). We evaluated the SIR values after compensating for this whitening effect [32].

14.4.5 Results

Subband System Evaluation. To evaluate the subband analysis-synthesis system, we measured the signal-to-distortion ratio (SDR), which is defined as

$$\mathrm{SDR} = 10 \log \frac{\sum_n^{L_\delta} \mathrm{b}^2(n-D)}{\sum_n^{L_\delta} \{\mathrm{b}(n-D) - \mathrm{a}(n)\}^2} \text{ [dB]}, \tag{14.8}$$

where the system input $\mathrm{b}(n) = \delta(n - \frac{L_\delta}{2})$, L_δ is the length of the delta function, D is the delay caused by low-pass filters (LPF) in the analysis and synthesis stages, and $\mathrm{a}(n)$ is the output (impulse response) of the subband analysis-synthesis system. The SDR was 59.2 dB. This distortion caused by subband analysis and synthesis can be ignored because the separation performance SIR is at most 15 dB (see Fig. 14.7), and thus masks this distortion.

Separation Performance of Subband BSS. In order to confirm the superiority of subband BSS, we compared the separation performance of subband BSS with that of frequency-domain BSS and time-domain BSS.

Figure 14.7 shows the separation result SIR and the value of the average correlation coefficient between source signals $\mathrm{CC}(N) = \frac{1}{N}\sum_{k=1}^{N} |r_k|$, where

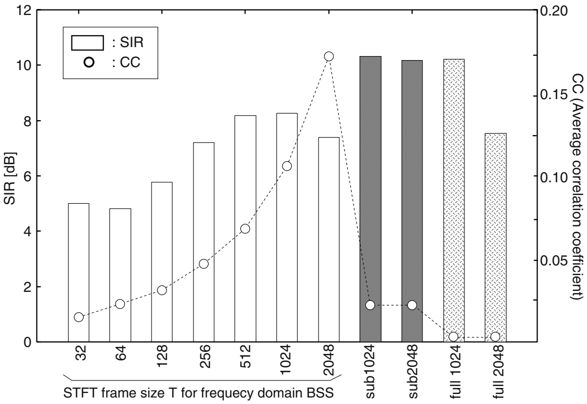


Fig. 14.7. Separation performance of frequency-domain BSS (white bars), subband BSS (black bars) and fullband time-domain BSS (gray bars). "CC": average correlation coefficient. Adaptation data length=3 s and separated data length=8 s. $T_{\mathrm{R}} = 300$ ms.

N is the number of subbands for subband BSS or number of frequencies for frequency-domain BSS and r_k is the correlation coefficient between source signals of a k-th frequency/subband.

For frequency-domain BSS, the parameter was the STFT frame size T. In Fig. 14.7, T is shown by the horizontal axis. For subband BSS, we used separation filters $w_{ij}^{k}(m)$ of 64 and 128-taps in each subband; this corresponds to 1024 and 2048-taps in a fullband, respectively. In Fig. 14.7, they are shown as "sub1024" and "sub2048", respectively. Our $N = 64$ subbands with decimation $R = 16$ corresponds to $T = 32$ in frequency-domain BSS with regard to down-sampling rate. The number of learning data samples in the time-frequency domain was the same for subband and frequency-domain BSS.

With frequency-domain BSS, although we should use long frame to handle the reverberation, CC becomes large and the independent assumption seems to collapse as frame size T becomes large. This is because the number of samples in each frequency becomes small. Therefore, the performance degraded when we used separation filters of 2048-taps (i.e., frame size $T = 2048$). Please note that the adaptation data length was three seconds and the half frame-shift was utilized.

With fullband time-domain BSS ("full1024" and "full2048" in Fig. 14.7), on the other hand, the CC was very small and we obtained a good result when the separation filter length was 1024. However, when we employed a separation filter length of 2048, it became difficult to estimate the separation filters and the performance degraded. The performance for various separation filter lengths with fullband time-domain BSS can be seen in [32].

By contrast, we achieved better separation performance with subband BSS even when we estimated separation filters of 2048-taps. Moreover, with subband BSS, we were able to confirm that the CC value was sufficiently small. From the CC values, we can say that the independence assumption held well in subband BSS. Another possible reason for the superior performance of subband BSS is that the permutation problem does not arise in the subbands. This point is discussed in the next subsection.

14.4.6 Discussion

Using subband BSS, we can maintain the number of samples in each subband and obtain better separation performance. Using one second of speech as adaptation data, we still obtained acceptable separation performance: SIR = 7.47 dB for $T_{\mathrm{R}} = 300$ ms. If the adaptive data length is sufficiently long, the same performance would be obtained by time-domain BSS, frequency-domain BSS, and subband BSS. Our experimental results showed that subband BSS works effectively when the adaptation data length is short.

Moreover, using subband BSS, we obtained separated signals with less whitening effect than when using fullband time-domain BSS. When we use the usual time-domain BSS algorithm, the output signal spectrum is flattened [40]. This is because we remove the time dependence of the speech signals. These whitened speech signals sound unnatural. In contrast, because this whitening effect is limited in each subband, it can be diminished by subband BSS. Figure 14.8 shows an example of separated speech obtained with time-domain BSS and subband BSS. The separated signal is whitened using time-domain BSS, while the shape of the spectrum holds well using subband BSS.

Furthermore, although we did not face the permutation problem due to the initialization with null beamformers, this problem occurs in frequency-domain BSS and subband BSS in general; the spectral components of sources are recovered in a different order at different frequencies/subbands. This makes the time domain reconstruction of separated signals difficult. However, this problem is less serious in subband BSS than in frequency-domain BSS. This is because the permutation problem does not occur in each subband as the separation procedure is executed in each subband. Therefore, we face a smaller number of permutation problems than with frequency-domain BSS. In particular, subband BSS encounters very few permutation problems in low frequency bands, where it is difficult to solve the problems with frequency-domain BSS [15]. Moreover, we can use a wider band signal than frequency-domain BSS to solve the permutation problem in between subbands. Therefore, we can use more information on separated signals and separation filters, and can solve the problem more easily than in frequency-domain BSS.

Finally, we discuss the computational cost. Because the calculation of convolution and correlation in the time domain (14.5) is expensive, we calculate them in the frequency domain. As discussed in [20], [21], we can reduce

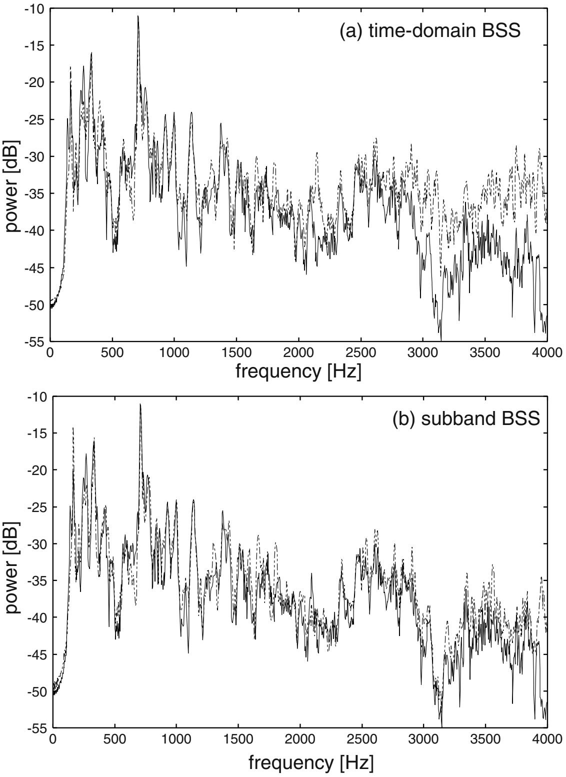


Fig. 14.8. Example spectra of a separated signal with (a) time-domain BSS and (b) subband BSS (broken lines). The solid lines show the spectrum of the original speech.

the computational cost by using subband processing. When we consider the decimation R, the computational cost for N subbands per time is reduced to about $(N/2+1)/(R \times R)$ times that of fullband time-domain BSS. As $R = N/4$ in our case, we can reduce the computational cost by about $2/R$.

14.5 Frequency-Appropriate Processing for Further Improvement

Subband BSS allows us to use different separation methods to estimate the separation filter for different subbands. By exploiting this advantage, in this section, we concentrate on low frequency bands for speech separation.

With speech separation, the SIR is generally worse in low frequency bands as shown in Fig. 14.9, which plots the SIR values of separated signals for each subband. One reason for the poor performance at low frequencies is that the impulse response is usually longer (see Fig. 14.10) and therefore it is harder

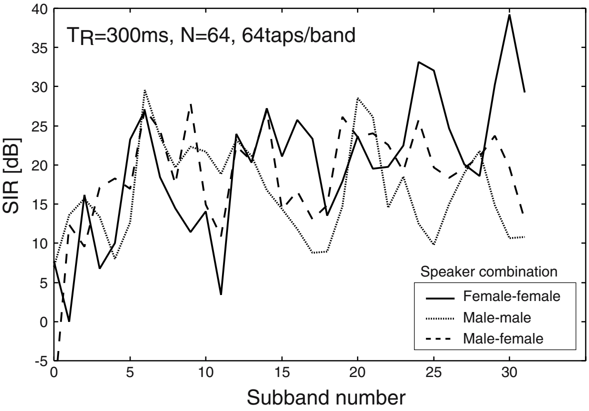


Fig. 14.9. SIR of separated signals in each subband. We can see that the SIR is poor in low frequency bands for every speaker combination.

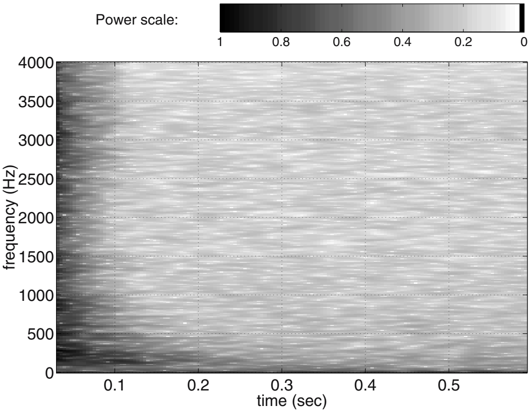


Fig. 14.10. Spectrogram example of a room acoustic impulse response. Black indicates high power and white indicates low power. We can see that the reverberation is longer at low frequencies than at high frequencies.

to separate signals in low frequency bands than in high frequency bands. Moreover, since speech signals have high power in low frequency bands, the performance in these bands dominates the overall speech signal separation performance. Therefore, it is important for speech separation to improve the separation performance in low frequency bands to obtain better overall separation performance.

14.5.1 Longer Separation Filters in Low Frequency Bands

One possible way to improve the SIR in low frequency bands is to estimate longer separation filters in these bands in order to cover the long reverber-

Table 14.1 Separation performance of subband BSS. (A)-(F) the overlap-blockshift was executed only for low frequency bands 0-5, and (G) and (H) the overlap-blockshift was executed for *all* subbands. $N = 64$.

	# of taps		S I R [dB]		
	band 0-5	band 6-32	no-overlap	overlap (x2)	overlap (x4)
(A)	32	32	6.0		
(B)	64	32	9.9	9.8	
(C)	128	32	9.5	10.1	10.4
(D)	64	64	10.3	10.8	10.7
(E)	128	64	10.5	11.4	12.2
(F)	128	128	10.1	11.0	11.7
(G)	64	64	10.3	10.7	10.7
(H)	128	128	10.1	11.2	12.2

ation. If the length of the separation filters is insufficient, we cannot reduce reverberant components of interferences that are longer than the filters and we obtain poor SIR [24].

We therefore employ longer separation filters for low frequency bands (bands 0-5). Figure 14.10 shows that the reverberation is long below about 600 Hz. Therefore, we used long filters for these frequency bands. The column labelled "no-overlap" in Table 14.1 shows the separation performance for each separation filter length condition.

In Table 14.1 (A)-(C), we used a 32-tap separation filter for high frequency bands, and we changed the filter length for low frequency bands (bands 0-5). We can see that a 32-tap long separation filter cannot achieve good performance [see Table 14.1 (A)]. This is conceivable that it cannot cover the reverberation in low frequency bands. When we used long separation filters only in low frequency bands [Table 14.1 (B)], the separation performance was greatly improved. However, when we used 128-taps in low frequency bands, the separation performance degraded [see Table 14.1 (C)]. Figure 14.11 shows the SIR for cases (A) - (C). We can see that the performance of (C) is worse than (B). This is attributed to the fact that the number of samples in each subband is too small to allow us to estimate a 128-tap separation filter precisely. The proposal in the next section (Section 14.5.2) will overcome this problem.

14.5.2 Overlap-Blockshift in Low Frequency Bands

Another possible way to improve the SIR in low frequency bands is to utilize a fine overlap-blockshift in the time-domain BSS stage. Using the fine overlap-blockshift, we can increase outwardly the number of samples in each

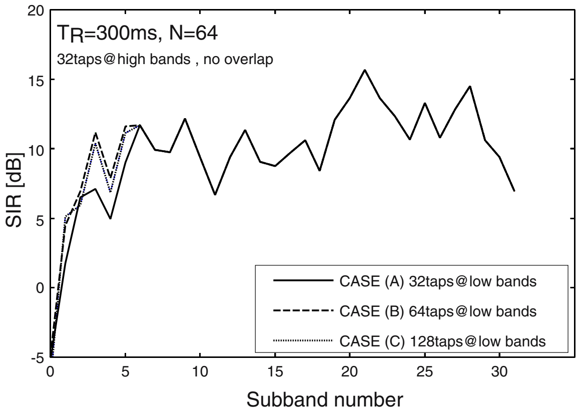


Fig. 14.11. Effect of filter length for low frequency bands.

subband, and can estimate the separation filters more precisely. Since our time-domain BSS algorithm (14.5) divides signals into B blocks to utilize the non-stationarity of signals, we can divide signals into blocks with an overlap, as long as the non-stationarity is expressed among blocks. It should be noted that this overlap-blockshift is executed in the separation stage, i.e., after the decimation for subband analysis.

In Table 14.1 [(B)-(F)], the columns show the SIR obtained by the overlap-blockshift only for low frequency bands (bands 0-5). "Overlap (×2)" and "overlap (×4)" means that the blockshift rate $S = 2$ and 4 in (14.5), respectively. Table 14.1 [(B)-(F)] show that when we used the overlap-blockshift only for low frequency bands, we obtained better separation performance. With a fourfold overlap-blockshift for (E), we were able to estimate the separation filters of 128-taps in low frequency bands, and we obtained the best separation performance (underlined in Table 14.1). Figure 14.12 shows the effect of the fine overlap-blockshift in low frequency bands.

14.5.3 Discussion

Even when we used 128-taps for all the frequency bands [(F) in Table 14.1], the performance was no better than when we used 128-taps only for the low frequency bands [(E) in Table 14.1]. Figure 14.13 shows the SIR in each subband for (E) and (F). We can see that the use of the long separation filters is not so effective in the high frequency bands. Sometimes, short filters achieve better separation performance than long filters in the high frequency bands. We can say that the employment of long separation filters only in low frequency bands is enough for the separation.

Furthermore, when the overlap-blockshift was used in all subbands [see (G) and (H) in Table 14.1], the increase in SIR was very small compared with the SIR for (D) and (F) in Table 14.1. Figure 14.14 shows the improvement

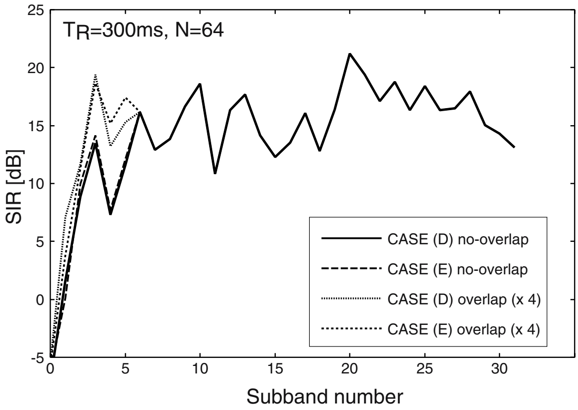


Fig. 14.12. Effect of overlap-blockshift only in low frequency bands.

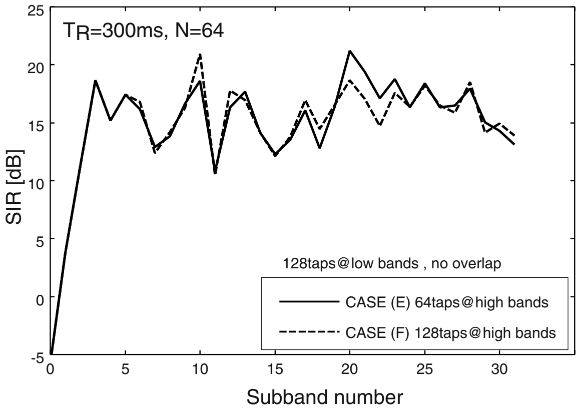


Fig. 14.13. Example of SIR in each subband when we use a long filter in all frequency bands.

in separation performance provided by the overlap-blockshift. The overlap-blockshift is also effective in high frequency bands. However, the contribution of the improvement to SIR in the high frequency bands is not significant for the whole performance [see (F) and (H) in Table 14.1]. This is because the original power of the high frequency components of the speech signal is smaller than that of the low frequency components. Therefore, we can conclude that the use of a fine overlap-blockshift only in low frequencies is sufficient to obtain improved performance.

By using long separation filters and the fine overlap-blockshift technique only in low frequency bands, we can efficiently separate convolutive mixtures of speech. Such frequency-dependent processing is impossible with time-domain BSS and intricate with frequency-domain BSS. Moreover, we can save the computation cost without degrading the separation performance by lim-

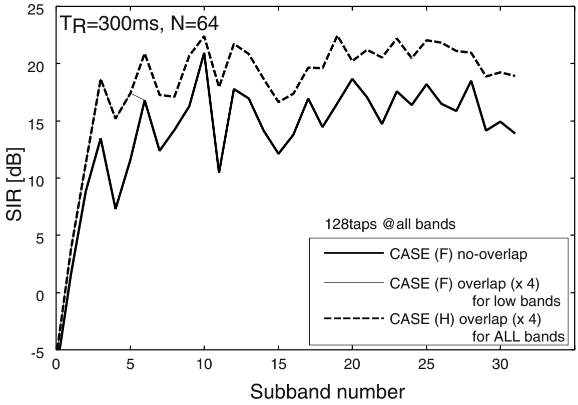


Fig. 14.14. Example of SIR in each subband obtained with the overlap-blockshift in all subbands.

iting the use of long separation filters and the fine overlap-blockshift only to low frequency bands.

There could be other ways to improve the separation performance. For instance, we may be able to use different microphone pairs with appropriate spacing for each subband. From a beamforming point of view, the resolution of a spatial cancellation is related to the frequency. If the microphone spacing is greater than half the wavelength, spatial aliasing occurs. This tends to happen at high frequencies. On the other hand, if the spacing is too small, the phase and amplitude difference between observations at low frequency becomes too small and therefore, it becomes difficult to achieve good performance. That is, the small phase difference between the observations at the microphones is also a reason for the poor performance in low frequency bands. A low frequency generally prefers a long spacing and a high frequency likes a short spacing [41]. In this chapter, we considered the case of N_m microphones whose number and spacing are fixed and ignored the multiple spacing microphone case. However, if we could configure the microphone spacing according to frequency, we would obtain better performance.

14.6 Conclusions

In this chapter, subband processing was applied to BSS for convolutive mixtures of speech. The subband-based BSS approach offers a compromise between the time-domain technique, which is usually difficult and slow with many separation filter coefficients to estimate, and a frequency domain technique, which has difficulty estimating statistics when the adaptation data length is insufficient. Our proposed subband BSS can maintain a sufficient number of samples to estimate the statistics in each subband and estimate a separation filter long enough to cover the reverberation. We confirmed the effectiveness of subband BSS experimentally.

Furthermore, making good use of subband processing, i.e., employing an appropriate separation method for each frequency band, we showed that we can improve the separation performance with long separation filters and the overlap-blockshift technique only in low frequency bands. Subband BSS is a powerful separation tool when the source signals s_i or the impulse response of the system h_{ji} have different characteristics in different frequency bands.

Acknowledgments

We wish to thank Dr. H. Saruwatari for his valuable discussions and for providing the impulse responses used in the experiments. We also thank Dr. Y. Haneda, Mr. A. Nakagawa, and Mr. S. Sakauchi for their help with the SSB filterbank, and Mr. R. Aichner and Mr. T. Nishikawa for detailed discussions on time-domain blind source separation.

References

1. S. Haykin, *Unsupervised Adaptive Filtering.* John Wiley & Sons, 2000.
2. A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis.* John Wiley & Sons, 2001.
3. T. W. Lee, *Independent Component Analysis – Theory and Applications.* Kluwer Academic Publishers, 1998.
4. S. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang, “Multichannel blind deconvolution and equalization using the natural gradient,” in *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications*, 1997, pp. 101–104.
5. K. Torkkola, “Blind separation of delayed and convolved sources,” in *Unsupervised Adaptive Filtering*, S. Haykin, Ed., vol. 1, pp. 321–375, John Wiley & Sons, 2000.
6. M. Kawamoto, K. Matsuoka, and N. Ohnishi, “A method of blind separation for convolved non-stationary signals,” *Neurocomputing*, vol. 22, pp. 157–171, Nov. 1998.
7. H. Buchner, R. Aichner, and W. Kellermann, “Blind source separation for convolutive mixtures: a unified treatment,” in *Audio Signal Processing for Next-Generation Multimedia Communication Systems*, Y. Huang and J. Benesty, Eds., pp. 255–293, Kluwer Academic Publishers, 2004.
8. S. C. Douglas, “Blind separation of acoustic signals,” in *Microphone Arrays: Techniques and Applications*, M. Brandstein and D. B. Ward, Eds., pp. 355–380, Springer-Verlag, 2001.
9. P. Smaragdis, “Blind separation of convolved mixtures in the frequency domain,” *Neurocomputing*, vol. 22, pp. 21–34, Nov. 1998.
10. S. Ikeda and N. Murata, “A method of ICA in time-frequency domain,” in *Proc. ICA*, 1999, pp. 365–370.

11. M. Z. Ikram and D. R. Morgan, "Exploring permutation inconsistency in blind separation of speech signals in a reverberant environment," in *Proc. IEEE ICASSP*, 2000, pp. 1041–1044.
12. J. Anemüller and B. Kollmeier, "Amplitude modulation decorrelation for convolutive blind source separation," in *Proc. ICA*, 2000, pp. 215–220.
13. S. Araki, R. Mukai, S. Makino, T. Nishikawa, and H. Saruwatari, "The fundamental limitation of frequency domain blind source separation for convolutive mixtures of speech," *IEEE Trans. Speech Audio Processing*, vol. 11, pp. 109–116, Mar. 2003.
14. 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.
15. H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust approach to the permutation problem of frequency-domain blind source separation," in *Proc. IEEE ICASSP*, 2003, pp. 381–384.
16. K. Rahbar and J. P. Reilly, "A new fast-converging method for BSS of speech signals in acoustic environments," in *Proc. IEEE WASPAA*, 2003, pp. 21–24.
17. K. Matsuoka and S. Nakashima, "Minimal distortion principle for blind source separation," in *Proc. ICA*, 2001, pp. 722–727.
18. S. Araki, S. Makino, R. Aichner, T. Nishikawa, and H. Saruwatari, "Blind source separation for convolutive mixtures of speech using subband processing," in *Proc. SMMSP (International Workshop on Spectral Methods and Multirate Signal Processing)*, 2002, pp. 195–202.
19. S. Araki, S. Makino, R. Aichner, T. Nishikawa, and H. Saruwatari, "Subband based blind source separation for convolutive mixtures of speech," in *Proc. IEEE ICASSP*, 2003, pp. 509–512.
20. J. Huang, K.-C. Yen, and Y. Zhao, "Subband-based adaptive decorrelation filtering for co-channel speech separation," *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 402–406, July 2000.
21. F. Duplessis-Beaulieu and B. Champagne, "Fast convolutive blind speech separation via subband adaptation," in *Proc. IEEE ICASSP*, 2003, pp. 513–516.
22. S. Araki, S. Makino, R. Aichner, T. Nishikawa, and H. Saruwatari, "Subband based blind source separation with appropriate processing for each frequency band," in *Proc. ICA*, 2003, pp. 499–504.
23. S. Araki, S. Makino, R. Aichner, T. Nishikawa, and H. Saruwatari, "Subband-based blind separation for convolutive mixtures of speech," *IEEE Trans. Speech Audio Processing*, submitted.
24. R. Mukai, S. Araki, H. Sawada, and S. Makino, "Evaluation of separation and dereverberation performance in frequency domain blind source separation," *Acoustical Science and Technology*, vol. 25, pp. 119–126, Mar. 2004.
25. M. R. Portnoff, "Implementation of the digital phase vocoder using the fast Fourier transform," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 24, pp. 243–248, June 1976.
26. N. Grbic, X.-J. Tao, S. E. Nordholm, and I. Claesson, "Blind signal separation using overcomplete subband representation," *IEEE Trans. Speech Audio Processing*, vol. 9, pp. 524–533, July 2001.
27. S. L. Gay and R. J. Mammone, "Fast converging subband acoustic echo cancellation using RAP on the WE DSP16A," in *Proc. IEEE ICASSP*, 1990, pp. 1141–1144.

28. R. Crochiere and L. Rabiner, *Multirate Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1983.
29. P. L. Chu, "Weaver SSB subband acoustic echo canceller," in *Proc. IWAENC*, 1993, pp. 173–176.
30. S. Makino, J. Noebauer, Y. Haneda, and A. Nakagawa, "SSB subband echo canceller using low-order projection algorithm," in *Proc. IEEE ICASSP*, 1996, pp. 945–948.
31. T. Nishikawa, H. Saruwatari, and K. Shikano, "Blind source separation of acoustic signals based on multistage ICA combining frequency-domain ICA and time-domain ICA," *IEICE Trans. Fundamentals*, vol. E86-A, pp. 846–858, Apr. 2003.
32. R. Aichner, S. Araki, S. Makino, T. Nishikawa, and H. Saruwatari, "Time domain blind source separation of non-stationary convolved signals by utilizing geometric beamforming," in *IEEE International Workshop on Neural Networks for Signal Processing*, 2002, pp. 445–454.
33. S. Araki, S. Makino, Y. Hinamoto, R. Mukai, T. Nishikawa, and H. Saruwatari, "Equivalence between frequency domain blind source separation and frequency domain adaptive beamforming for convolutive mixtures," *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 11, pp. 1157–1166, 2003.
34. S. Kurita, H. Saruwatari, S. Kajita, K. Takeda, and F. Itakura, "Evaluation of blind signal separation method using directivity pattern under reverberant conditions," in *Proc. IEEE ICASSP*, 2000, pp. 3140–3143.
35. H. Saruwatari, S. Kurita, and K. Takeda, "Blind source separation combining frequency-domain ICA and beamforming," in *Proc. IEEE ICASSP*, 2001, pp. 2733–2736.
36. H. Sawada, R. Mukai, and S. Makino, "Direction of arrival estimation for multiple source signals using independent component analysis," in *Seventh International Symposium on Signal Processing and its Applications*, 2003, vol. 2, pp. 411–414.
37. H. Sawada, R. Mukai, S. Araki, and S. Makino, "Polar coordinate based nonlinear function for frequency domain blind source separation," in *Proc. IEEE ICASSP*, 2002, pp. 1001–1004.
38. T. Nishikawa, H. Saruwatari, K. Shikano, S. Araki, and S. Makino, "Multistage ICA for blind source separation of real acoustic convolutive mixture," in *Proc. ICA*, 2003, pp. 523–528.
39. S. Van Gerven, D. Van Compernolle, L. Nguyen Thi, and C. Jutten, "Blind separation of sources: A comparative study of a 2nd and a 4th order solution," in *Signal Processing VII, Theories and Applications*, M. J. J. Holt, C. F. N. Cowan, P. M. Grant, and W. A. Sandham, Eds., Elsevier, pp. 1153–1156, 1994.
40. X. Sun and S. Douglas, "A natural gradient convolutive blind source separation algorithm for speech mixtures," in *Proc. ICA*, 2001, pp. 59–64.
41. H. Sawada, S. Araki, R. Mukai, and S. Makino, "Blind source separation with different sensor spacing and filter length for each frequency range," in *IEEE International Workshop on Neural Networks for Signal Processing*, 2002, pp. 465–474.

15 Real-Time Blind Source Separation for Moving Speech Signals

Ryo Mukai, Hiroshi Sawada, Shoko Araki, and Shoji Makino

NTT Communication Science Laboratories
Soraku-gun, Kyoto 619-0237, Japan
E-mail: {ryo, sawada, shoko, maki}@cslab.kecl.ntt.co.jp

Abstract. In this chapter, we present a method for the real-time blind source separation (BSS) of moving speech signals in a room. The method employs frequency-domain independent component analysis (ICA) using a blockwise batch algorithm in the first stage, and the separated signals are refined by postprocessing using crosstalk component estimation and non-stationary crosstalk cancellation in the second stage. The blockwise batch algorithm achieves better performance than an online algorithm when sources are stationary, and the postprocessing compensates for performance degradation caused by source movement. Experimental results using speech signals recorded in a real room show that our method realizes robust real-time separation for moving sources.

15.1 Introduction

Blind source separation (BSS) is a technique for estimating original source signals using only observed mixtures. Independent component analysis (ICA) [1–5] is one of the main statistical methods used for BSS. The BSS of audio signals has a wide range of applications including noise robust speech recognition, hands-free telecommunication systems and high-quality hearing aids. In most realistic applications, the source location may change. Accordingly a large amount of research has been undertaken on BSS for moving source signals [6–12].

A mixing system is time-varying when source signals move. A naive approach for tracking a time-varying system is an online algorithm that updates the separation system sample by sample. Indeed the online algorithm can track a time-varying system, however, its performance is generally worse than a batch algorithm, which can employ a number of samples, when the system is stationary. Although we are dealing with moving sources, we do not want to degrade the performance for fixed sources.

In this chapter, we describe a robust real-time BSS method [6] that employs frequency-domain ICA with a blockwise batch algorithm in the first stage, and the postprocessing of crosstalk component estimation and non-stationary crosstalk cancellation in the second stage. When we adopt blockwise frequency-domain ICA, we need to solve a permutation problem for

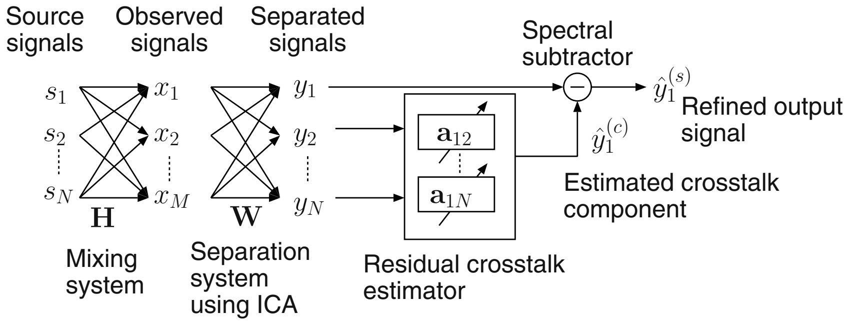


Fig. 15.1. Block diagram of our system for one output channel in one frequency bin. The separation system using ICA is described in Sect. 15.2, and the residual crosstalk estimator is described in Sect. 15.3.

every block, and this is a time-consuming process especially when the block length is short. We use an algorithm based on an analytical calculation of the source directions to solve the permutation problem quickly [13]. Another problem inherent to batch algorithms is an input-output delay. To reduce this delay, we use a technique for computing the output signal without waiting for the calculation of the separation system to be completed. These techniques are useful for realizing low-delay real-time BSS.

The blockwise batch algorithm achieves better separation performance than an online algorithm for fixed source signals, but the performance deteriorates for moving sources. As we pointed out in [14], the ICA solution works like an adaptive beamformer, which forms a spatial null towards an interference signal. This characteristic means that BSS using ICA is fragile as regards a moving interference signal but robust with respect to a moving target signal. Utilizing this nature, we can estimate residual crosstalk components even when the interference signal moves. To compensate for the degradation that occurs when the interference signal moves, we employ postprocessing in the second stage.

Figure 15.1 shows a block diagram of our method for one output channel in one frequency bin. In contrast to the original spectral subtraction [15], which assumes stationary noise and periods with no target signal when estimating the noise spectrum, our method requires neither assumption because we use BSS in the first stage.

Our postprocessing method is also effective when source signals do not move, and it can improve the separation performance for fixed sources [16,17]. Generally, the performance of BSS using only ICA is insufficient for most applications in a real-world environment. Accordingly, BSS with postprocessing has been attracting the interest of many researchers, and various methods have recently been proposed [16–20].

This chapter is organized as follows. In the next section, we summarize the algorithm of frequency-domain BSS for convolutive mixtures and formulate a blockwise batch algorithm. In Sect. 15.3, we describe an algorithm to estimate and subtract residual crosstalk components from the separated signals. Section 15.4 presents experimental results using speech signals recorded in a room and shows the effectiveness of the method in realizing robust real-time separation. Section 15.5 sums up this chapter.

15.2 ICA Based BSS of Convolutive Mixtures

In this section, we briefly review the BSS algorithm that uses frequency-domain ICA and formulate a blockwise batch algorithm including an online algorithm as a special case. We also describe a fast algorithm for solving permutation problems, which is necessary for real-time processing.

15.2.1 Frequency-Domain ICA

When N source signals are $s_1(t), ..., s_N(t)$, the signals observed by M microphones are $x_1(t), ..., x_M(t)$, and the separated signals are $y_1(t), ..., y_N(t)$, the BSS model can be described by the following equations:

$$x_j(t) = \sum_{k=1}^{N} (h_{jk} * s_k)(t), \tag{15.1}$$

$$y_i(t) = \sum_{j=1}^{M} (w_{ij} * x_j)(t), \tag{15.2}$$

where h_{jk} are the impulse response from source i to microphone j, w_{ij} are the separation filters, and $*$ denotes the convolution operator.

Using a short-time Fourier transform (STFT) for (15.1), the model is approximated as:

$$\mathbf{x}(f, n) = \mathbf{H}(f)\mathbf{s}(f, n), \tag{15.3}$$

where f is the frequency and n represents the frame index. The separation process can be formulated in each frequency bin as:

$$\mathbf{y}(f, n) = \mathbf{W}(f)\mathbf{x}(f, n), \tag{15.4}$$

where $\mathbf{s}(f, n) = [s_1(f, n), ..., s_N(f, n)]^T$ is the source signal in frequency bin f, $\mathbf{x}(f, n) = [x_1(f, n), ..., x_M(f, n)]^T$ denotes the observed signals, $\mathbf{y}(f, n) = [y_1(f, n), ..., y_N(f, n)]^T$ is the estimated source signal, and $\mathbf{W}(f)$ represents the separation matrix.

A convolutive mixture in the time domain can be approximated as multiple instantaneous mixtures in the frequency domain. Therefore, we can apply an ordinary (instantaneous) ICA algorithm in the frequency domain to solve a BSS problem. $\mathbf{W}(f)$ is determined so that the elements of $\mathbf{y}(f, n)$ become mutually independent for each f.

To calculate the separation matrix $\mathbf{W}$, we use an optimization algorithm based on the minimization of the mutual information of $\mathbf{y}$. The optimal $\mathbf{W}$ is obtained by the following iterative equation using the natural gradient approach [21]:

$$\mathbf{W}^{(l+1)} = \mathbf{W}^{(l)} + \mu[\mathbf{I} - \langle \mathbf{\Phi}(\mathbf{y})\mathbf{y}^H \rangle]\mathbf{W}^{(l)}, \tag{15.5}$$

where l is an index for the iteration, $\mathbf{I}$ is the identity matrix, μ is a step size parameter, $\langle \cdot \rangle$ denotes the averaging operator, and $\mathbf{\Phi}(\mathbf{y}) = [\varPhi(y_1), ..., \varPhi(y_N)]^T$ is an elementwise nonlinear function. Because the signals have complex values in the frequency domain, we use a polar coordinate based nonlinear function, which is effective for fast convergence, especially when the number of input data samples is small [22]. We adopted the following function for the experiments in Sect. 15.4:

$$\varPhi(y_i) = \tanh(g \cdot |y_i|) e^{\jmath \arg(y_i)}, \tag{15.6}$$

where g is a gain parameter that controls the nonlinearity. The complex-valued ICA is discussed in detail in Chapter 13.

15.2.2 Permutation and Scaling Problems

Once we have completed the ICA for all frequencies, we need to solve the permutation and scaling problems. Since we are handling signals with complex values, the scaling factors are also complex values.

We use a direction of arrival (DOA) based method to solve the permutation problem. The permutation problem is solved so that the DOAs of the separated signals are aligned. The DOA of the i-th separated signal $\theta_i(f)$ can be calculated analytically as:

$$\theta_i(f) = \arccos \frac{\arg([\mathbf{W}(f)^{-1}]_{ji}/[\mathbf{W}(f)^{-1}]_{j'i})}{2\pi f c^{-1} |d_j - d_{j'}|}, \tag{15.7}$$

where $[\cdot]_{ji}$ denotes ji-th element of the matrix, c is the speed of sound, and d_j represents a location of microphone j. This method does not require the directivity pattern to be scanned, thus we can solve the permutation problem quickly. The derivation of (15.7) is described in [13], and the more general case is discussed in Chapter 13.

With regard to the scaling problem there is a simple and reasonable method called minimal distortion principle (MDP) proposed by Matsuoka [23]. The scaling alignment according to the MDP in the frequency domain is given by the following operation:

$$\mathbf{W}(f) \leftarrow \mathrm{diag}[\mathbf{W}(f)^{-1}]\mathbf{W}(f). \tag{15.8}$$

By using this solution, the output signal $y_i(t)$ becomes an estimation of the convolved version of source $s_i(t)$ measured at microphone i. See Chapter 13 for more detail on the permutation and scaling problems.

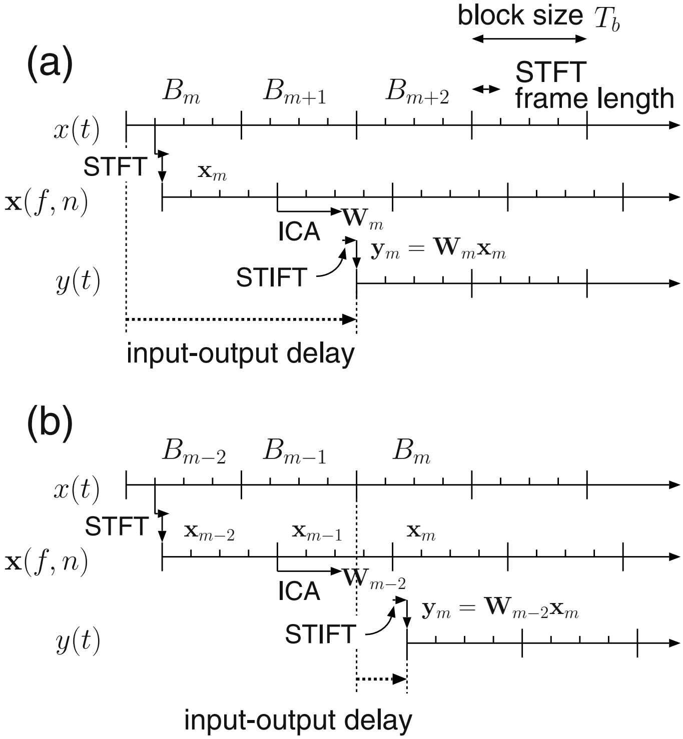


Fig. 15.2. Input-output delay of (a) BSS using ordinary blockwise batch algorithm, and (b) BSS without waiting for calculation of $\mathbf{W}_m$.

15.2.3 Low Delay Blockwise Batch Algorithm

In order to track the time-varying mixing system, we update the separation matrix for each time block $B_m = \{t : (m-1)T_b \le t < mT_b\}$, where T_b is the block size, and m represents the block index ($m \ge 1$).

Koutras et al. have proposed a similar method in the time domain [8]. When T_b equals the STFT frame length, this procedure can be considered an online algorithm in the frequency domain.

We use the separation matrix of the previous block as the initial iteration value for a new block, i.e. $\mathbf{W}^{(0)}_{m+1}(f) = \mathbf{W}^{(N_I)}_m(f)$, where N_I is the number of iterations for (15.5).

The batch algorithm has an inherent delay, because the calculation of $\mathbf{W}$ needs to wait for the arrival of a data block. Moreover, the calculation itself also takes time [Fig. 15.2(a)]. However, when the calculation is completed within T_b and we use $\mathbf{W}_{m-2}$ for separation of the signals in B_m, we can avoid the delay for waiting and calculation [Fig. 15.2(b)]. This technique can reduce the input-output delay and is suitable for low-delay real-time applications.

It seems that this method fails when a source signal moves, but it is actually robust for the moving target signal, which is shown in Sect. 15.4.3. Unfortunately, this method suffer performance deterioration when an interference signal moves. To cope with this problem, we employ a postprocessing method using crosstalk component estimation and non-stationary crosstalk cancellation which reduces the performance deterioration.

15.3 Residual Crosstalk Cancellation

In this section, we mention the postprocessing method which improves the separation performance of BSS. First, we examine the nature of separated signals obtained by the frequency-domain ICA described in the previous section. We then describe an algorithm to estimate and subtract residual crosstalk components in these signals.

15.3.1 Straight and Crosstalk Components of BSS

When we concatenate the mixing system (15.1) and the separation system (15.2), we have

$$y_i(t) = \sum_{j=1}^{M} \sum_{k=1}^{N} (w_{ij} * h_{jk} * s_k)(t). \tag{15.9}$$

We define the concatenation of the paths from source k to output i as:

$$g_{ik}(t) = \sum_{j=1}^{M} (w_{ij} * h_{jk})(t), \tag{15.10}$$

then each of the separated signals $y_i(t)$ can be described as follows:

$$y_i(t) = \sum_{k=1}^{N} (g_{ik} * s_k)(t). \tag{15.11}$$

In the same way as (15.3) and (15.4), this can be approximated in the frequency domain:

$$\mathbf{G}(f) = \mathbf{W}(f)\mathbf{H}(f), \tag{15.12}$$

and separated signals y_i can be described as follows:

$$y_i(f,n) = \sum_{k=1}^{N} G_{ik}(f) s_k(f,n). \tag{15.13}$$

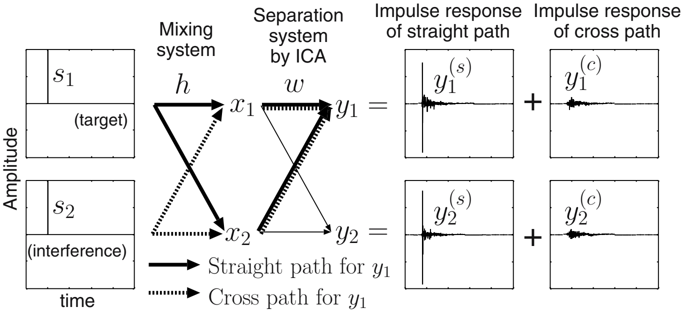


Fig. 15.3. Impulse responses of straight path and cross path.

We decompose y_i into the sum of straight component $y_i^{(s)}$ derived from target signal s_i and crosstalk component $y_i^{(c)}$ derived from interference signals $s_k (k \neq i)$. Then, we have

$$y_i(f,n) = y_i^{(s)}(f,n) + y_i^{(c)}(f,n), \tag{15.14}$$

$$y_i^{(s)}(f,n) = G_{ii}(f) s_i(f,n), \tag{15.15}$$

$$y_i^{(c)}(f,n) = \sum_{k \neq i} G_{ik}(f) s_k(f,n). \tag{15.16}$$

We denote estimation of $y_i^{(s)}$ and $y_i^{(c)}$ as $\hat{y}_i^{(s)}$ and $\hat{y}_i^{(c)}$, respectively. Our goal is to estimate the spectrum of $y_i^{(c)}$ using only $y_1, ..., y_N$ and obtain $\hat{y}_i^{(s)}$ by subtracting $\hat{y}_i^{(c)}$ from y_i.

In our previous research [24], we measured the impulse responses of the straight and cross paths of a BSS system. As a result, we found that the direct sound of an interference can be almost completely removed by BSS, and also that residual crosstalk components are derived from the reverberation. (See $y_1^{(c)}$ and $y_2^{(c)}$ in Fig. 15.3.) We utilize these characteristics of separated signals to estimate the crosstalk components.

15.3.2 Model of Residual Crosstalk Component Estimation

Figure 15.4 shows an example of a narrow band power spectrum of straight and crosstalk components in separated signals obtained by a two-input two-output BSS system. The crosstalk component $y_1^{(c)}$ is in y_1 and the straight component $y_2^{(s)}$ is in y_2. Both components are derived from source signal s_2; $y_1^{(c)}$ is derived from the reverberation of s_2 and $y_2^{(s)}$ is mainly derived from the direct sound of s_2. Accordingly, for the narrow band signal in each frequency bin, the crosstalk component $y_1^{(c)}$ can be approximated by the filtered version of $y_2^{(s)}$.

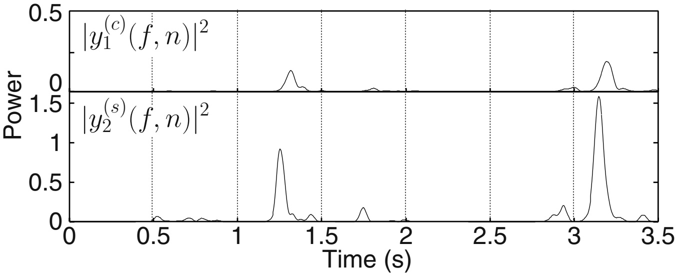


Fig. 15.4. Example of narrow band power spectrum of straight and crosstalk components ($f = 320$ Hz).

We extend this approximation to multiple signals by introducing filters $\mathbf{a}_{ij}(f,n) = [a_{ij0}(f,n), ..., a_{ijL-1}(f,n)]^T$ for each frequency bin f and combination of channels i and j ($i \neq j$), where L is the length of filters.

Furthermore, we use y_j as an approximation of $y_j^{(s)}$, because $y_j^{(s)}$ is actually unknown. Therefore, the model for estimating residual crosstalk components is formulated as follows:

$$|y_i^{(c)}(f,n)|^\beta \approx \sum_{j\neq i}\sum_{l=0}^{L-1} a_{ijl}(f,n)|y_j^{(s)}(f,n-l)|^\beta \tag{15.17}$$

$$\approx \sum_{j\neq i}\sum_{l=0}^{L-1} a_{ijl}(f,n)|y_j(f,n-l)|^\beta, \tag{15.18}$$

where the exponent $\beta = 1$ for the magnitude spectrum and $\beta = 2$ for the power spectrum.

15.3.3 Adaptive Algorithm and Spectrum Estimation

Figure 15.5 shows a block diagram of our method for one output channel. We estimate filters $\mathbf{a}_{ij}$ described in the previous section by using an adaptive algorithm based on the normalized LMS (NLMS) algorithm [25].

For each i, the filters $\hat{\mathbf{a}}_{ij}(f,n)$ are adapted so that the sum of the output signals becomes $|y_i^{(c)}(f,n)|^\beta$ for input signals $|y_j^{(s)}(f,n)|^\beta$ ($1 \leq j \leq N$, $j \neq i$). If $|y_i^{(c)}(f,n)|$ and $|y_j^{(s)}(f,n)|$ are available, the update equation according to the NLMS algorithm is described as follows:

$$\hat{\mathbf{a}}_{ij}(f,n+1) = \hat{\mathbf{a}}_{ij}(f,n) + \frac{\eta}{\delta + ||\mathbf{u}_j(f,n)||^2}\mathbf{u}_j(f,n)e_{ij}(f,n), \tag{15.19}$$

where

$$\mathbf{u}_j(f,n) = [|y_j^{(s)}(f,n)|^\beta, |y_j^{(s)}(f,n-1)|^\beta, ..., |y_j^{(s)}(f,n-L+1)|^\beta]^T \tag{15.20}$$

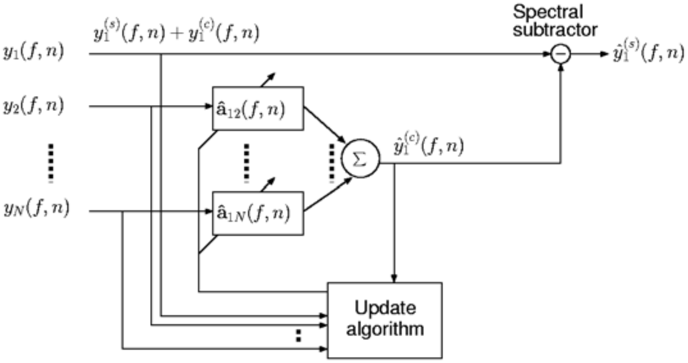


Fig. 15.5. Adaptive filters and spectral subtractor to estimate $y_1^{(s)}$. (Residual crosstalk estimator and spectral subtractor in Fig. 15.1.)

is an input vector and

$$e_{ij}(f,n) = |y_i^{(c)}(f,n)|^{\beta} - \sum_{j \neq i} \hat{\mathbf{a}}_{ij}^T(f,n)\mathbf{u}_j(f,n) \tag{15.21}$$

is an estimation error. Here, η is a step size parameter and δ is a positive constant to avoid numerical unstability when $||\mathbf{u}_j||$ is very small.

Unfortunately, $|y_i^{(c)}(f,n)|$ and $|y_j^{(s)}(f,n)|$ are unknown, so they are substituted by $|y_i(f,n)|$ and $|y_j(f,n)|$, respectively. We assume that $|y_i^{(s)}(f,n)|$ can be approximated by $|y_i(f,n)|$ when $|y_i(f,n)|$ is large and $|y_i^{(c)}(f,n)|$ can be approximated by $|y_i(f,n)|$ when $|y_i(f,n)|$ is small. This assumption is based on the sparse characteristics of narrow band signals, i.e. $y_i^{(s)}$ and $y_j^{(s)}$ seldom have large power simultaneously, especially when the source signals are speech signals. A detailed analysis of overlapping frequency components of speech signals can be found in [26] and [27].

Since not all $|y_i^{(c)}(f,n)|$ and $|y_i^{(s)}(f,n)|$ can be approximated by $|y_i(f,n)|$, only a subset of the filters is updated at each iteration. To formulate a selective update algorithm, we introduce sets of channel index numbers,

$$\mathcal{I}_S(f,n) = \{i : |\, |y_i(f,n)| - |y_i^{(s)}(f,n)|\, | < \epsilon_s\},$$
$$\mathcal{I}_C(f,n) = \{i : |\, |y_i(f,n)| - |y_i^{(c)}(f,n)|\, | < \epsilon_c\},$$

where ϵ_s and ϵ_c are small parameters that determine tolerance. This means that $|y_i^{(s)}(f,n)|$ can be approximated by $|y_i(f,n)|$ for $i \in \mathcal{I}_S(f,n)$ and $|y_i^{(c)}(f,n)|$ can be approximated by $|y_i(f,n)|$ for $i \in \mathcal{I}_C(f,n)$.

One example implementation for determining $\mathcal{I}_S(f,n)$ $\mathcal{I}_C(f,n)$ is

$$\mathcal{I}_S(f,n) = \{i : i = \underset{i}{\operatorname{argmax}}\, |y_i(f,n)|\},$$
$$\mathcal{I}_C(f,n) = \overline{\mathcal{I}_S(f,n)}.$$

Another example is

$$\mathcal{I}_S(f,n) = \{i : |y_i(f,n)| > \mathit{threshold}\},$$
$$\mathcal{I}_C(f,n) = \overline{\mathcal{I}_S(f,n)}.$$

We substitute $y_j^{(s)}$ in (15.20) with y_j for $j \in \mathcal{I}_S(f,n)$, and $y_i^{(c)}$ in (15.21) with y_i for $i \in \mathcal{I}_C(f,n)$. The filters $\hat{\mathbf{a}}_{ij}$ are updated for $i \in \mathcal{I}_C(f,n)$ and $j \in \mathcal{I}_S(f,n)$ by using (15.19). Therefore, the update procedure is given by

$$\hat{\mathbf{a}}_{ij}(f,n+1) = \begin{cases} \hat{\mathbf{a}}_{ij}(f,n) + \dfrac{\eta}{\delta + ||\mathbf{u}_j(f,n)||^2}\mathbf{u}_j(f,n)e_{ij}(f,n) \\ \qquad\qquad (\text{if } i \in \mathcal{I}_C(f,n), \text{ and } j \in \mathcal{I}_S(f,n)) \\ \hat{\mathbf{a}}_{ij}(f,n) \quad (\text{otherwise}) \end{cases}, \tag{15.22}$$

where

$$\mathbf{u}_j(f,n) = [|y_j(f,n)|^\beta, |y_j(f,n-1)|^\beta, ..., |y_j(f,n-L+1)|^\beta]^T \tag{15.23}$$

is an input vector and

$$e_{ij}(f,n) = |y_i(f,n)|^\beta - \sum_{j \neq i} \hat{\mathbf{a}}_{ij}^T(f,n)\mathbf{u}_j(f,n) \tag{15.24}$$

is an estimation error.

We apply the estimated filters to the model (15.18), and obtain an estimation of the power of residual crosstalk components:

$$|\hat{y}_i^{(c)}(f,n)|^\beta = \sum_{j \neq i} \hat{\mathbf{a}}_{ij}^T(f,n)\mathbf{u}_j(f,n). \tag{15.25}$$

Finally, we obtain an estimation of the straight component as $\hat{y}_i^{(s)}$ by the following spectral subtraction procedure:

$$\hat{y}_i^{(s)}(f,n) = \begin{cases} (|y_i(f,n)|^\beta - |\hat{y}_i^{(c)}(f,n)|^\beta)^{1/\beta}\dfrac{y_i(f,n)}{|y_i(f,n)|} \\ \qquad\qquad (\text{if } |y_i(f,n)| > |\hat{y}_i^{(c)}(f,n)|) \\ 0 \qquad\qquad (\text{otherwise}) \end{cases}. \tag{15.26}$$

15.4 Experiments and Discussions

15.4.1 Experimental Conditions

To examine the effectiveness of our method, we carried out experiments using speech signals recorded in a room. The reverberation time of the room was 130 ms. We used two omni-directional microphones with an inter-element spacing of 4 cm. The layout of the room is shown in Fig 15.6. The target

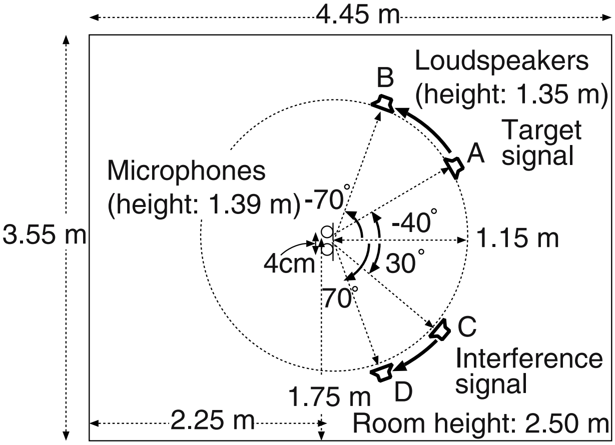


Fig. 15.6. Layout of room used in experiments. $T_R = 130$ ms.

source signal was first located at A, and then moved to B at a speed of 30 deg/s. The interference signal was located at C and moved to D at a speed of 40 deg/s.

The step size parameter μ in (15.5) affects the separation performance of BSS when the block size changes. We carried out preliminary experiments and chose μ to optimize the performance for each block size. Other conditions are summarized in Table 15.1. The frame shift and the filter length L in the postprocessing part were decided so that the filter could cover the reverberation.

To update filters $\hat{\mathbf{a}}_{ij}(f,n)$, we used the following simple selective update policy:

```
if $|y_1(f,n)| > |y_2(f,n)|$
  then $\mathcal{I}_S(f,n) = \{1\},\ \mathcal{I}_C(f,n) = \{2\}$
  else $\mathcal{I}_S(f,n) = \{2\},\ \mathcal{I}_C(f,n) = \{1\}$ .
```

We assumed the straight component $y_1^{(s)}$ as a signal, and the difference between the output signal and the straight component as interference. We defined the output signal-to-interference ratio (SIR_O) in the time domain as follows:

$$\mathrm{SIR}_O \equiv 10 \log \frac{\sum_t |y_1^{(s)}(t)|^2}{\sum_t |y_1(t) - y_1^{(s)}(t)|^2} \text{ (dB)}. \tag{15.27}$$

Similarly, the input SIR (SIR_I) is defined as,

$$\mathrm{SIR}_I \equiv 10 \log \frac{\sum_t \sum_{i=1}^{2} |(h_{i1} * s_1)(t)|^2}{\sum_t \sum_{i=1}^{2} |(h_{i2} * s_2)(t)|^2} \text{ (dB)}. \tag{15.28}$$

Table 15.1 Experimental conditions.

Common	Sampling rate = 8 kHz Window = hanning Reverberation time T_R=130 ms
ICA part	Frame length T_{ICA} = 1024 point (128 ms) Frame shift = 256 point (32ms) $g = 100.0$ μ = optimized for block size T_b Number of iterations $N_I = 100$
Post processing part	Frame length T_{SS} = 1024 point (128 ms) Frame shift = 64 point (8 ms) Filter length $L = 16$ $\beta = 2$ $\delta = 0.01$ $\eta = 0.1$

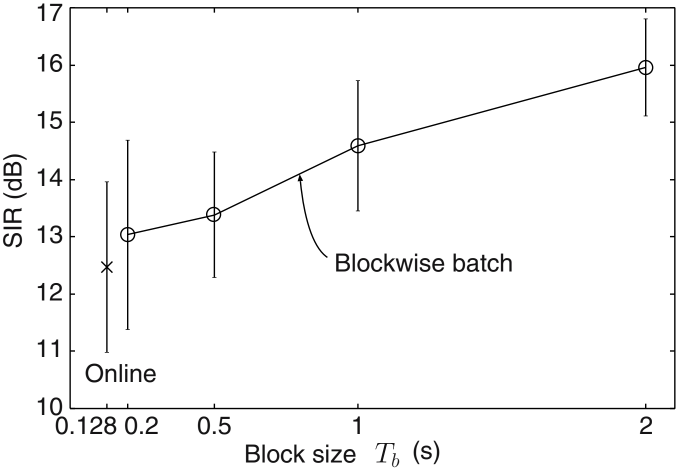


Fig. 15.7. Average and standard deviation of SIR for fixed sources.

We use SIR = $\mathrm{SIR}_O - \mathrm{SIR}_I$ as a performance measure. This measurement is consistent with the performance evaluation of BSS in which the crosstalk component is assumed as interference. We measured SIRs with 30 combinations of source signals using three male and three female speakers, and averaged them.

15.4.2 Performance for Fixed Sources

Although we are dealing with moving sources, we do not want the performance for fixed sources to deteriorate. First, we measured the BSS performance using ICA without postprocessing. Figure 15.7 shows the average and

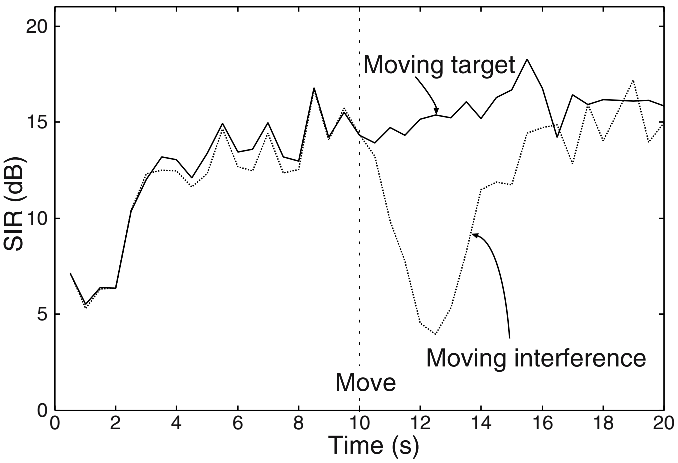


Fig. 15.8. SIR of blockwise batch algorithm without postprocessing. Target and interference signals moved at 10 s ($T_b = 1.0$ s).

standard deviation of SIR for fixed sources (the target is at A and the interference at C in Fig. 15.6). This indicates that the blockwise batch algorithm outperforms the online algorithm (in which μ is tuned to optimize the performance), when we use the update equation (15.5). In addition, the deviation of the batch algorithm is smaller than that of the online algorithm. This is why we adopt the blockwise batch algorithm in the first stage. We used T_b = 1.0 s in the following experiments.

15.4.3 Moving Target and Moving Interference

Before considering the result obtained with the postprocessing method, we investigate the BSS performance for moving sources using the blockwise batch algorithm. Figure 15.8 shows the SIR for a moving target (solid line) and for a moving interference (dotted line). We can see that the SIR is not degraded even when the target moves. By contrast, interference movement causes a decline in the SIR.

This can be explained by the directivity pattern of the separation system obtained by ICA. The solution of frequency domain BSS works in the same way as an adaptive beamformer, which forms a spatial null towards an interference signal (Fig. 15.9). Because of this characteristic, BSS using ICA is robust as regards a moving target signal but fragile with respect to a moving interference signal. The relationship between the separation filter obtained ICA and the adaptive beamformer is detailed in [14].

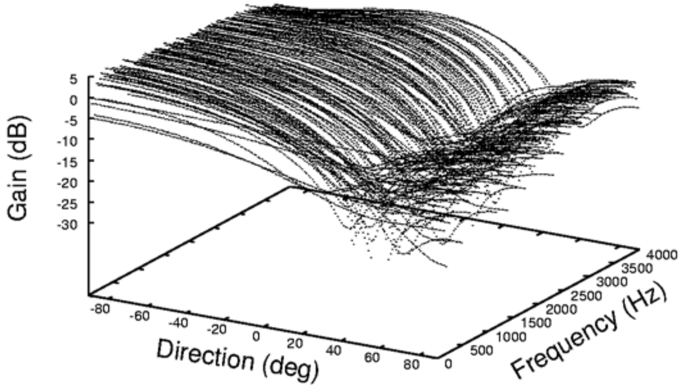


Fig. 15.9. Directivity pattern of separation system obtained by frequency-domain ICA.

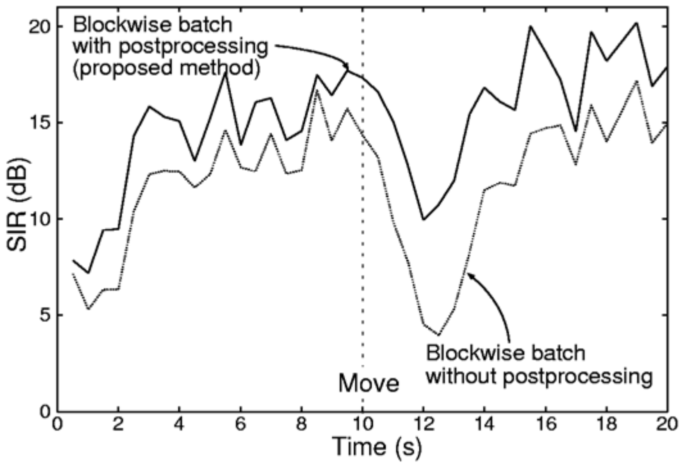


Fig. 15.10. Effect of postprocessing. Interference signal moved from C to D at 10 s ($T_b = 1.0$ s).

15.4.4 Performance of Blockwise Batch Algorithm with Postprocessing

The most important factor when estimating the crosstalk component $y_1^{(c)}$ using (15.22) and (15.25) is the separated signal y_2. We can estimate y_2 robustly even when s_2 moves, because s_2 is a target signal for y_2. Therefore, postprocessing works robustly even when the interference signal s_2 moves.

Figure 15.10 shows the SIR of blockwise batch algorithm with postprocessing when the interference signal moves (solid line). We can see that the SIR is improved by the postprocessing, and the drop of the SIR when the interference moves is reduced. This result shows that our postprocessing method can compensate the fragility of the blockwise batch algorithm when

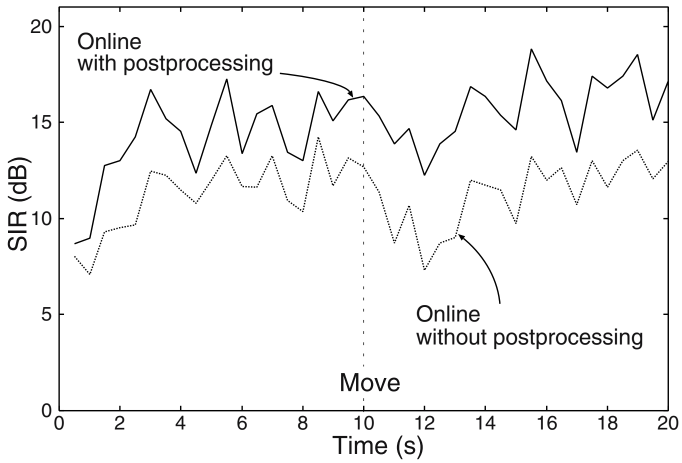


Fig. 15.11. Performance of online algorithm with and without postprocessing. Interference signal moved from C to D at 10 s ($T_b = 1.0$ s).

an interference signal moves. Although crosstalk components still remaining in the postprocessed output signal sometimes make a musical noise, the power is much smaller than ordinary spectral subtraction.

15.4.5 Performance of Online Algorithm

Figure 15.11 shows the SIR of online algorithm with and without postprocessing. The online algorithm is more stable than blockwise algorithm, however the performance is worse when the sources are stationary, as we described in Sect. 15.4.2. The postprocessing is also effective for this case, thus we may choose the algorithm in the first stage according to requirements of the application.

15.5 Conclusions

We presented a robust real-time BSS method for moving source signals. The combination of the blockwise batch and the postprocessing realizes a robust low-delay real-time BSS. We can solve a permutation problem quickly by using analytical calculation of source directions, and this technique is useful for solving convolutive BSS problems in realtime. Postprocessing using crosstalk component estimation and non-stationary crosstalk cancellation improves the separation performance and reduces the performance deterioration when an interference signal moves. Experimental results using speech signals recorded in a room showed the effectiveness of our method. Some sound examples can be found on our web site [28].

References

1. A. J. Bell and T. J. Sejnowski, “An information-maximization approach to blind separation and blind deconvolution,” *Neural Computation*, vol. 7, pp. 1129–1159, 1995.
2. S. Haykin, ed., *Unsupervised Adaptive Filtering.* John Wiley & Sons, 2000.
3. T. W. Lee, *Independent Component Analysis.* Kluwer Academic Publishers, 1998.
4. A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing.* John Wiley& Sons, 2002.
5. A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis.* John Wiley & Sons, 2001.
6. R. Mukai, H. Sawada, S. Araki, and S. Makino, “Blind Source Separation for Moving Speech Signals using Blockwise ICA and Residual Crosstalk Subtraction,” *IEICE Trans. Fundamentals*, vol. E87-A, pp. 1941–1948, 2004.
7. J. Anemüller and T. Gramss, “On-line blind separation of moving sound sources,” in *Proc. ICA*, 1999, pp. 331–334.
8. A. Koutras, E. Dermatas, and G. Kokkinakis, “Blind speech separation of moving speakers in real reverberant environment,” in *Proc. IEEE ICASSP*, 2000, pp. 1133–1136.
9. I. Kopriva, Z. Devcic, and H. Szu, “An adaptive short-time frequency domain algorithm for blind separation of non-stationary convolved mixtures,” in *Proc. IJCNN*, 2001, pp. 424–429.
10. K. E. Hild II, D. Erdogmus, and J. C. Principe, “Blind source separation of time-varying, instantaneous mixtures using an on-line algorithm,” in *Proc. IEEE ICASSP*, 2002, pp. 993–996.
11. R. Aichner, H. Buchner, F. Yan, and W. Kellermann, “Real-time convolutive blind source separation based on a broadband approach,” in *Proc. ICA (Lecture Notes in Computer Science 3195)*, Springer-Verlag, 2004, pp. 840–848.
12. D. W. E. Schobben, *Real-Time Adaptive Concepts in Acoustics.* Kluwer Academic Publishers, 2001.
13. H. Sawada, R. Mukai, and S. Makino, “Direction of arrival estimation for multiple source signals using independent component analysis,” in *Proc. ISSPA*, vol. 2, 2003, pp. 411–414.
14. S. Araki, S. Makino, Y. Hinamoto, R. Mukai, T. Nishikawa, and H. Saruwatari, “Equivalence between frequency domain blind source separation and frequency domain adaptive beamforming for convolutive mixtures,” *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 11, pp. 1157–1166, 2003.
15. S. F. Boll, “Suppression of acoustic noise in speech using spectral subtraction,” *IEEE Trans. Acoust., Speech, and Signal Processing*, vol. ASSP-27, pp. 113–120, Apr. 1979.
16. R. Mukai, S. Araki, H. Sawada and S. Makino, “Removal of residual cross-talk components in blind source separation using time-delayed spectral subtraction,” in *Proc. IEEE ICASSP*, 2002, pp. 1789–1792.
17. R. Mukai, S. Araki, H. Sawada, and S. Makino, “Removal of residual crosstalk components in blind source separation using LMS filters,” in *Proc. NNSP*, 2002, pp. 435–444.
18. S. Y. Low, S. Nordholm, and R. Togneri, “Convolutive blind signal separation with post-processing,” *IEEE Trans. Speech Audio Processing*, vol. 12, pp. 539–548, Sept. 2004.

19. D. Kolossa, and R. Orglmeister, “Nonlinear postprocessing for blind speech separation,” in *Proc. ICA (Lecture Notes in Computer Science 3195)*, Springer-Verlag, 2004, pp. 832–839.
20. C. Choi, G. Jang, Y. Lee, and S. R. Kim, “Adaptive cross-channel interference cancellation on blind source separation outputs,” in *Proc. ICA (Lecture Notes in Computer Science 3195)*, Springer-Verlag, 2004, pp. 857–864.
21. S. Amari, A. Cichocki, and H. H. Yang, “A new learning algorithm for blind signal separation,” *Advances in Neural Information Processing Systems 8*, pp. 757–763, The MIT Press, 1996.
22. H. Sawada, R. Mukai, S. Araki, and S. Makino, “Polar coordinate based nonlinear function for frequency-domain blind source separation,” *IEICE Trans. Fundamentals*, vol. E86-A, pp. 590–596, Mar. 2003.
23. K. Matsuoka and S. Nakashima, “Minimal distortion principle for blind source separation,” in *Proc. ICA*, 2001, pp. 722–727.
24. R. Mukai, S. Araki, and S. Makino, “Separation and dereverberation performance of frequency domain blind source separation,” in *Proc. ICA*, 2001, pp. 230–235.
25. S. Haykin, *Adaptive Filter Theory.* Fourth Edition. Prentice Hall, 2002.
26. M. Aoki, M. Okamoto, S. Aoki, H. Matsui, T. Sakurai, and Y. Kaneda, “Sound source segregation based on estimating incident angle of each frequency component of input signals acquired by multiple microphones,” *Acoust. Sci. & Tech.*, vol. 22, pp. 149–157, Feb. 2001.
27. Ö. Yılmaz and S. Rickard, “Blind separation of speech mixtures via time-frequency masking,” *IEEE Trans. Signal Processing*, vol. 52, pp. 1830–1847, 2004.
28. http://www.kecl.ntt.co.jp/icl/signal/mukai/demo/ieice2004/

16 Separation of Speech by Computational Auditory Scene Analysis

Guy J. Brown[1] and DeLiang Wang[2]

[1] University of Sheffield, Department of Computer Science
Sheffield S1 4DP, United Kingdom
E-mail: g.brown@dcs.shef.ac.uk

[2] The Ohio State University, Department of Computer Science & Engineering
and Center for Cognitive Science
Columbus, OH 43210, USA
E-mail: dwang@cse.ohio-state.edu

Abstract. The term *auditory scene analysis (ASA)* refers to the ability of human listeners to form perceptual representations of the constituent sources in an acoustic mixture, as in the well-known ‘cocktail party’ effect. Accordingly, *computational auditory scene analysis (CASA)* is the field of study which attempts to replicate ASA in machines. Some CASA systems are closely modelled on the known stages of auditory processing, whereas others adopt a more functional approach. However, all are broadly based on the principles underlying the perception and organization of sound by human listeners, and in this respect they differ from ICA and other approaches to sound separation. In this chapter, we review the principles underlying ASA and show how they can be implemented in CASA systems. We also consider the link between CASA and automatic speech recognition, and draw distinctions between the CASA and ICA approaches.

16.1 Introduction

Imagine a recording of a busy party, in which you can hear voices, music and other environmental sounds. How might a computational system process this recording in order to segregate the voice of a particular speaker from the other sources? Independent component analysis (ICA) offers one solution to this problem. However, it is not a solution that has much in common with that adopted by the best-performing sound separation system that we know of – the human auditory system. Perhaps the key to building a sound separator that rivals human performance is to model human perceptual processing?

This argument provides the motivation for the field of *computational auditory scene analysis* (CASA), which aims to build sound separation systems that adhere to the known principles of human hearing. In this chapter, we review the state-of-the-art in CASA, and consider its similarities and differences with the ICA approach. We also consider the relationship between CASA and techniques for robust automatic speech recognition in noisy environments, and comment on the challenges facing this growing field of study.

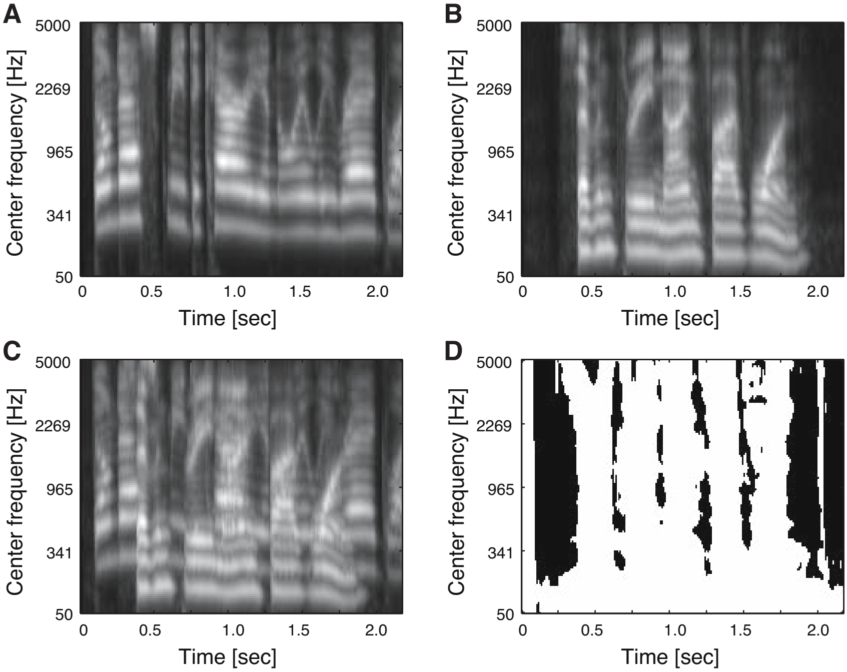


Fig. 16.1. (**A**) Auditory spectrogram for the utterance "don't ask me to carry an oily rag" spoken by a female. (**B**) Auditory spectrogram for the utterance "seven three nine five one" spoken by a male. (**C**) Auditory spectrogram for a mixture of the male and female utterances. Light pixels correspond to regions of high energy, and dark pixels correspond to regions of low energy. (**D**) Ideal binary mask for the male utterance, obtained by using the criterion given in (16.1). White pixels indicate reliable regions, and black pixels indicate unreliable regions.

16.2 Auditory Scene Analysis

In naturalistic listening situations, several sound sources are usually active at the same time, and the pressure variations in air that they generate combine to form a mixture at the ears of the listener. A common example of this is the situation in which the voices of two talkers overlap, as illustrated in Figure 16.1C. The figure shows the simulated auditory nerve response to a mixture of a male and female voice, obtained from a computational model of auditory processing. How can this complex acoustic mixture be parsed in order to retrieve a description of one (or both) of the constituent sources?

Bregman [5] was the first to present a coherent answer to this question (see also [17] for a more recent review). He contends that listeners perform an *auditory scene analysis* (ASA), which can be conceptualized as a two-stage process. In the first stage, the acoustic mixture is decomposed into elements. An element may be regarded as an atomic part of the auditory scene, which describes a significant acoustic event. Subsequently, a grouping process combines elements that are likely to have arisen from the same acoustic source, forming a perceptual structure called a *stream.* For example, consider the voice of a speaker; in Bregman's terms, the vocal tract of the speaker is the acoustic source, whereas the mental representation of the speaker's voice is the corresponding stream.

Grouping processes may be *data-driven* (primitive), or *schema-driven* (knowledge-based). In the former case, it is thought that listeners exploit heuristics similar to those proposed by the Gestalt psychologists for describing the ways in which elements of an image combine to form a coherent object. In schema-driven grouping, listeners apply learned knowledge of sound sources (such as speech and music) in a top-down manner. Examples of speech-related schemas include prosodic, semantic and pragmatic knowledge.

Consideration of Fig. 16.1C suggests that a number of primitive grouping cues could be applied to segregate the mixture. First, consider cues which might act upon acoustic components that overlap in time (so-called *simultaneous organization*). At about 0.5 sec., the male speech begins and this gives rise to an abrupt onset in acoustic energy across all frequency channels. Hence, a principle of 'common onset' might allow the voices of the two speakers to be segregated in this region – frequency regions that exhibit an abrupt increase in energy at the same time are probably dominated by the same acoustic source. Similarly, a principle of 'common offset' could be applied to segregate the two speakers in the region close to 2 sec., when the male speech ceases. Another powerful grouping cue is harmonicity. In the figure, horizontal bands of energy are visible which correspond to harmonics of the same fundamental frequency (F0). In principle, these harmonics can be sorted into two sets, such that those related to the same F0 are grouped together.

Secondly, consider grouping cues which could act upon nonoverlapping acoustic components (so-called *sequential organization*). In Fig. 16.1C, the male and female speakers occupy different average pitch ranges; hence the continuity of their F0s might be exploited in order to group successive utterances from the same speaker. Similarly, concentrations of energy in the time-frequency plane tend to change smoothly, such as those due to formant transitions. Again, such continuity can be exploited in order to separate one voice from the other. Some cues to sequential organization are not illustrated by the figure. For example, listeners tend to group sounds that have a similar timbre, and which originate from the same location in space.

16.3 Computational Auditory Scene Analysis

The structure of a typical data-driven CASA system is closely related to Bregman's conceptual model, as shown in Fig. 16.2. In the first stage, the input mixture is processed in order to derive acoustic features. Subsequent grouping processes may operate directly on these features, or more usually

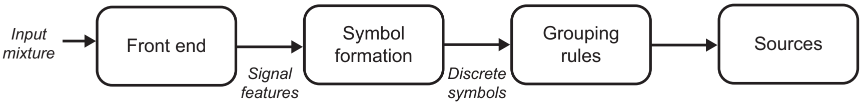


Fig. 16.2. Flow of processing in a typical data-driven CASA system, such as that of Brown and Cooke [7].

they will be used to derive an intermediate representation prior to grouping. In many systems, significant components in the time-frequency plane are encoded as discrete symbols. Grouping rules are then applied, in order to identify components that are likely to have arisen from the same source. The grouping heuristics may be encoded explicitly in a rule-based system, or may be implicitly encoded in a signal processing algorithm or neural network.

Once representations of individual sources are obtained, the auditory representation can usually be inverted in order to recover a time-domain waveform for the segregated source. This allows the separated signal to be evaluated using listening tests, or by performance metrics that involve a comparison between the original and reconstructed signals. Alternatively, an evaluation may be performed on the auditory representation directly.

An important notion in many CASA systems is the *time-frequency mask*. Given a description of the acoustic input in the time-frequency plane, a specific source may be recovered by applying a weighting to each time-frequency bin, such that regions dominated by the desired source receive a high weight and those dominated by other sources receive a low weight. The mask values may be binary or real-valued. Weintraub [67] was the first to use this approach in a CASA system, and it has since been adopted by several other workers [6], [7], [66], [54], [13]. The use of binary masks is motivated by the phenomenon of masking in human hearing, in which a weaker signal is masked by a stronger one within the same critical band (see Moore [41] for a review). It has also been noted that the reconstruction of a masked signal may be interpreted as a highly nonstationary Wiener filter [54].

What is the upper limit on the performance of a system that uses binary masks? Cooke *et al.* [13] have adapted a conventional speech recognizer so that reliable and unreliable (or missing) acoustic features are treated differently during decoding, and report excellent recognition performance using so-called *a priori* masks. Assuming that the clean speech and noise signals are available prior to mixing, the *a priori* mask is formed by selecting time-frequency regions in which the mixture energy lies within 3 dB of the energy in the clean speech. From the perspective of speech separation, Wang and colleagues [27,52,29] have subsequently proposed the *ideal binary mask* as a computational goal of CASA. Considering the auditory representation of a speech signal $s(t, f)$ and noise signal $n(t, f)$, where t and f index time and frequency respectively, the ideal binary mask $m(t, f)$ is given by

$$m(t, f) = \begin{cases} 1 \text{ if } s(t, f) > n(t, f) \\ 0 \text{ otherwise} \end{cases} . \tag{16.1}$$

A similar approach has been advocated by Jourjine *et al.* [31], who note that different speech utterances tend to be orthogonal in a high-resolution time-frequency representation, and can therefore be separated by binary masking. A number of workers have demonstrated that speech reconstructed from ideal binary masks is highly intelligible, even when extracted from a mixture of two or three concurrent speakers [54], [52]. Speech intelligibility tests using both speech and babble noise interference also show that ideal binary masking can lead to substantial intelligibility improvements for human listeners [52]. An extensive discussion on ideal binary masks can be found in [65].

In the following sections, we first review the feature extraction stage of CASA, and then focus on monaural (one-microphone) and binaural (two-microphone) approaches. We also consider the issue of cue integration, and review a number of different computational frameworks that allow multiple grouping cues to be brought to bear on an acoustic signal.

16.3.1 Peripheral Auditory Processing and Feature Extraction

The first stage of a CASA system is usually a time-frequency analysis that mimics the frequency selectivity of the human ear. Typically, the input signal is passed through a bank of bandpass filters, each of which simulates the frequency response associated with a particular position on the basilar membrane. The 'gammatone' filter is often used, which is an approximation to the physiologically-recorded impulse responses of auditory nerve fibres [50], [11]. The parameters of the gammatone filterbank (i.e., the filter order, bandwidth and frequency spacing) are usually chosen to provide a match to psychophysical data. Neuromechanical transduction in the cochlea may be approximated by half-wave rectifying and compressing the output of each filter; alternatively a detailed simulation of inner hair cell function can be employed [26]. We note, however, that not all CASA systems use an auditory-motivated time-frequency analysis. The short-term Fourier transform and discrete wavelet transform are also sometimes employed [42], [56], [38], [43].

Examples of auditory spectrograms generated using a gammatone filterbank are shown in Fig. 16.1. Note that a nonlinear frequency scale is used, and that the bandwidth of each filter varies proportionately to its center frequency. In low frequency regions, filter bandwidths are narrow and hence individual harmonics of a complex sound (such as speech) are resolved. In high-frequency regions, the bandwidths are broader and several components interact within the same filter.

Most CASA systems further process the peripheral time-frequency representation in order to extract features that are useful for grouping. The motivation here is to explicitly encode properties which are implicit in the acoustic signal. Typical is the ‘synchrony strand’ representation of Cooke [11], which is a symbolic encoding of significant features in the time-frequency plane. Cooke demonstrates that the grouping stage of CASA (e.g., identifying harmonically related components) is facilitated by using a representation in which continuity in time-frequency is made explicit. A further example is the system of Brown [7], which forms representations of onset and offset events, periodicity and frequency transitions. Similar rich ‘mid level’ representations of the acoustic signal have been proposed by other workers [21], [66].

16.3.2 Monaural Approaches

Although binaural cues contribute substantially to ASA, human listeners are able to segregate sounds when listening with a single ear, or when listening diotically to a single-channel recording. Perceptually, one of the most potent cues for monaural sound segregation is fundamental frequency (F0); specifically, listeners are able to exploit a difference in F0 in order to segregate the harmonics of one sound from those of interfering sounds. Accordingly, much of the work on monaural CASA has focussed on the problem of identifying the multiple F0s present in an acoustic mixture (so-called ‘multipitch analysis’), and using them to separate the constituent sounds. Perhaps the earliest example is the system for separating two concurrent speakers described by Parsons [49]. In his approach, the harmonics of a target voice are selected by peak picking in the spectral domain, and the voice of each speaker is tracked using pitch continuity.

An important class of algorithms for F0 estimation is based on a temporal model of pitch perception proposed by Licklider [36]. The first computational implementation of Licklider’s theory was described by Weintraub [67], who referred to it as an ‘auto-coincidence’ representation; subsequently, Slaney and Lyon [57] introduced the term *correlogram*. The correlogram is computed in the time domain by performing an autocorrelation at the output of each channel of a cochlear filter analysis,

$$A(t,f,\tau) = \sum_{n=0}^{N-1} h(t-n,f)h(t-n-\tau,f)w(n). \tag{16.2}$$

Here, $h(t,f)$ represents the cochlear filter response for channel f at time frame t, τ is the autocorrelation delay (lag), and w is a window function of length N samples (typically a Hanning, exponential, or rectangular window is used). Alternatively, the autocorrelation may be performed in the frequency domain by means of the discrete Fourier transform (DFT) and its inverse transform (IDFT), i.e.

$$\mathrm{IDFT}(|\mathrm{DFT}(h)|^k), \tag{16.3}$$

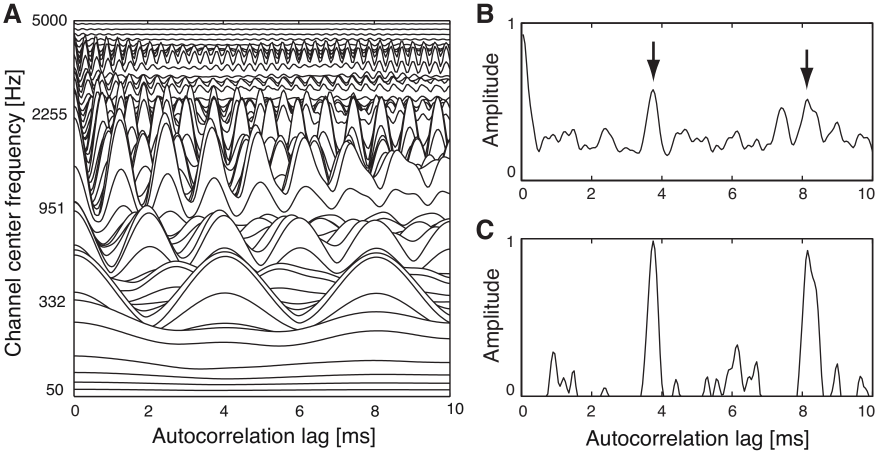


Fig. 16.3. A. Correlogram for time frame 100 of the mixture of two speakers shown in Fig. 16.1. B. Summary autocorrelation function (SACF). The pitch periods of the two speakers are marked with arrows. The male voice has a period of 8.1 ms (corresponding to a F0 of 123 Hz) and the female voice has a period of 3.8 ms (corresponding to a F0 of 263 Hz). C. Enhanced SACF, in which one iteration of processing has been used to remove sub-octave multiples of the significant peaks.

where h is a windowed section of the cochlear filter response. The introduction of a parameter k allows for a 'generalised autocorrelation' [62]. For conventional autocorrelation $k = 2$, but smaller values of k are advantageous because this leads to sharper peaks in the resulting function ([62] suggest a value of $k = 0.67$).

The correlogram is an effective means for F0 estimation because it detects the periodicities present in the output of the cochlear filterbank. For example, consider a voice with a F0 of 125 Hz. A channel responding to the fundamental component of the voice has a period of 8 ms, and hence a peak occurs in the corresponding autocorrelation function at a lag of 8 ms. Similarly, a channel responding to the second harmonic (250 Hz) has an autocorrelation peak at 4 ms, but because of the periodic nature of the autocorrelation function, peaks also occur at 8 ms, 12 ms, 16 ms and so on. In high-frequency regions, cochlear filters are wider and a number of harmonics interact within the same filter, causing amplitude modulation (AM). These interacting components 'beat' at a rate corresponding to the fundamental period, and also cause a peak in the autocorrelation function at the corresponding lag. Hence, for a periodic sound a 'spine' occurs in the correlogram which is centered at the fundamental period (8 ms in our example); for an example, see Fig. 16.3A. A convenient means of emphasizing this F0-related structure is to sum the channels of the correlogram over frequency,

$$S(t,\tau) = \sum_{f=1}^{M} A(t,f,\tau). \tag{16.4}$$

The resulting *summary autocorrelation function (SACF)* $S(t,\tau)$ exhibits a peak at the period of each F0, and can be used as the basis for multipitch analysis (Fig. 16.3B). For example, Tolonen and Karjalainen [62] describe a computationally efficient multipitch model based on the SACF. Computational savings are made by splitting the input signal into two bands (below and above 1 kHz) rather than performing a multi-band frequency analysis. A generalized autocorrelation is then computed for the low-frequency band and for the envelope of the high frequency band, and added to give an SACF. Further processing is then performed to enhance the representation of different F0s. Specifically, the SACF is half-wave rectified and then expanded in time by a factor of two, subtracted from the original SACF and half-wave rectified again. This removes peaks that occur at sub-octave multiples, and also removes the high-amplitude portion of the SACF close to zero delay (Fig. 16.3C). The operation may be repeated for time expansions of a factor of 3, 4, 5 and so on, in order to remove higher-order multiples of significant pitch peaks. In [32], the authors show how pitch tracks from this system can be used to separate harmonic sounds (two vowels) by applying a comb-notch filter, which removes the harmonics of the pitch track to which it is tuned. Ottaviani and Rocchesso [46] also describe a speech separation system based on F0 tracking using the enhanced SACF. They resynthesize a separated speech signal from a highly zero-padded Fourier spectrum, which is selectively weighted to emphasize harmonics of the detected pitch.

One of the more sophisticated algorithms for tracking the pitch of multiple speakers is reported by Wu *et al.* [69]. Their approach consists of four stages, shown schematically in Fig. 16.4. In the first stage, the digitized input signal is filtered by a bank of gammatone filters, in order to simulate cochlear filtering. Further stages of processing treat low-frequency channels (which have a center frequency below 800 Hz) and high-frequency channels differently. In low-frequency channels the correlogram is computed directly from the filter outputs, whereas in high-frequency channels the envelope in each channel is autocorrelated.

In the second stage of the system, 'clean' correlogram channels (i.e., those that are likely to contain reliable information about the periodicity of a single sound source, and are relatively uncorrupted by noise) are identified. The third stage of the system estimates the pitch periods present in each individual time frame using a statistical approach. Specifically, the difference between the true pitch period and the time lag of the closest correlogram peaks in each channel is employed as a means of quantifying the support for a particular pitch period hypothesis.

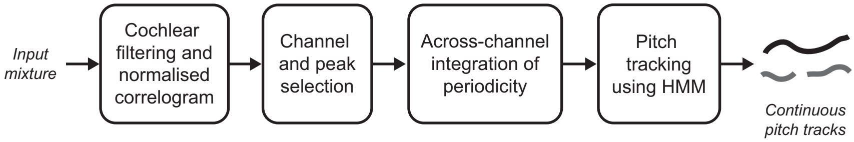

Input mixture
Cochlear filtering and normalised correlogram
Channel and peak selection
Across-channel integration of periodicity
Pitch tracking using HMM
Continuous pitch tracks

Fig. 16.4. Schematic diagram of the Wu, Wang, and Brown [69] system for tracking multiple fundamental frequencies in an acoustic mixture.

Periodicity information is then integrated across channels in order to derive the conditional probability of observing a set of pitch peaks $P(\Phi|x)$ given a pitch state x. Since zero, one or two pitches may be present, the pitch state is regarded as a pair $x = (y, Y)$ where $y \in R^Y$ is the pitch period and $Y \in \{0, 1, 2\}$ is the space index. Channel conditional probabilities are combined into frame conditional probabilities by assuming the mutual independence of the responses in all channels.

In the final stage of Wu *et al.*'s system, pitch periods are tracked across time using a hidden Markov model (HMM). Hidden nodes in the HMM represent the possible pitch states in each time frame, and observation nodes represent the set of selected peaks in each time frame. Transition probabilities between time frames are estimated from a small corpus of speech signals. Transition probabilities between state spaces of zero, one and two pitches are also estimated from the same corpus of speech signals, with the assumption that a single speaker is present for half of the time and two speakers are present for the remaining time. The optimal state sequence is found by the Viterbi algorithm, and may consist of zero, one or two pitch states. Figure 16.5 shows an example of the F0 tracks derived by Wu *et al.*'s system for a mixture of two speakers.

Wu *et al.*'s system suffers from a number of limitations. In principle, the algorithm could be modified to track more than two simultaneous speakers by considering more than three pitch spaces, but it remains to be seen how well such an approach would work in practice. Although it is robust to the presence of an interfering speaker (and can track its F0), Khurshid and Denham [35] find that Wu *et al.*'s system is less robust in the presence of background noise. They also find that although Wu's algorithm tracks the F0 of the dominant speaker in a mixture very accurately, its estimate of the nondominant F0 can be poor.

Khurshid and Denham suggest an alternative approach that rectifies some of these problems, which is a based on the analysis of the output of a bank of damped harmonic oscillators, which model the frequency analysis performed by the cochlea. Analysis of the fine time structure (consecutive zero crossings and amplitude peaks) of each oscillator output is performed in order to determine the driving frequency. An algorithm is then used to hypothesize the F0 (or multiple F0s) that are present in order to explain the observed frequency components. This is achieved by identifying salient spectral peaks

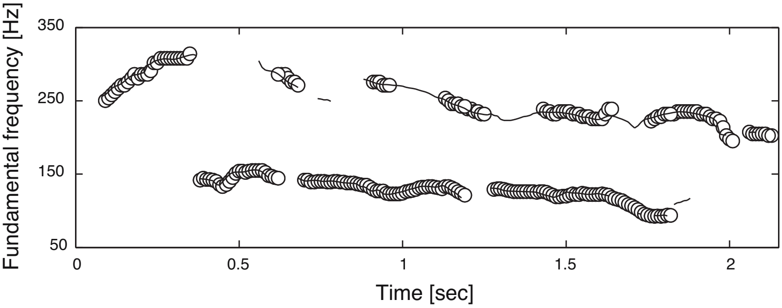


Fig. 16.5. Pitch tracks for the mixture of two speech signals shown in Fig. 16.1, obtained using the algorithm of Wu, Wang, and Brown [69]. Solid lines show the ground-truth pitch tracks for each speaker, open circles show the estimated pitch periods at each time frame.

(similar to the place groups described by Cooke [13]) and then assessing the support for every subharmonic of the peak that falls within the normal range of voice pitch. Such a frequency 'remapping' leads to noise robustness, and may be regarded as a simple model of simultaneous masking in the auditory nerve. Simple continuity constraints are used to track two F0s over time. Khurshid and Denham performed a comparison and reported that, although Wu *et al.*'s system is able to more accurately track the dominant pitch, their own system tracks the nondominant F0 more reliably and is also more robust to noise.

The correlogram, as described in (16.2), is based on an autocorrelation operation such that a large response occurs at the period of a F0. An alternative approach, advocated by de Cheveigné [15], is to perform *cancellation* rather than autocorrelation. In his approach, a time-domain comb filter of the form

$$q(t) = \delta(t) - \delta(t - \tau) \tag{16.5}$$

is applied to the acoustic signal (or to the output from each channel of a cochlear filter bank), where $\delta(t)$ is the delta function, t is time and τ is the lag parameter. The filter has zeros at frequencies $f = 1/\tau$ and all its multiples, and hence its response to a signal with a period of τ is zero. F0 analysis is therefore performed by applying the filter for different values of τ and searching for the minimum response. de Cheveigné and Kawahara [16] further suggest that this approach can be extended to the problem of multipitch estimation by cascading N filters, each of which is tuned to cancel a particular period. Hence, to perform multipitch estimation it is simply necessary to search the N-dimensional space of lag parameters until the minimum is found. The authors evaluated their algorithm on a corpus consisting of mixtures of two or three harmonic complexes, with impressive results. However, their joint cancellation technique has certain limitations. It is computationally expensive (although amenable to parallelism), and cancellation of one source may partially cancel another source if their periods are related by an integer multiple.

As noted previously, the correlogram deals with two cues to the F0 of a sound in a unified way; resolved harmonics in low-frequency regions and AM (‘beating’) in high-frequency regions. However, this unified treatment leads to poor segregation in the high-frequency range because AM alters autocorrelation structure and makes it difficult to group high-frequency components [29]. Some CASA systems process resolved and unresolved harmonic regions using different mechanisms (e.g., [11]). Hu and Wang [29] describe a recent system in which AM is extracted in high frequency regions and used to segregate unresolved harmonics, whereas conventional techniques are used to segregate resolved harmonics. AM detection is based on an ‘envelope correlogram’, which is of the form given in (16.2) except that the autocorrelation is performed on the envelope of each filter response rather than the fine structure. Correlations between adjacent channels are then computed in order to identify significant acoustic components. An initial segmentation based on F0 is then performed, which is similar to that described by [66]. Further processing is used to refine the F0 track for the dominant source, based on temporal smoothness and a periodicity constraint. Time-frequency units are then labelled according to whether they are dominated by the target speech signal or not, using heuristics that are based on the conventional correlogram in low-frequency regions and the envelope correlogram in high frequency regions. Finally, segments are generated based on cross-channel envelope correlation and temporal continuity, and these are grouped with low-frequency segments that share a common F0. The authors show that their system performs consistently better than that of Wang and Brown [66] across 10 noise conditions. In all but one noise condition it also outperforms a conventional spectral subtraction scheme for speech enhancement.

Explicit representations of AM have been proposed as an alternative to the correlogram. For instance, Berthommier and Meyer [2] describe a system for separating sounds on the basis of their F0s using the modulation spectrum (see also [34]). Each channel of a gammatone filterbank is half-wave rectified and bandpass filtered to remove the DC component and frequencies above the pitch range. The magnitude of a DFT is then computed to give a two-dimensional representation of tonotopic frequency against AM frequency. A harmonic sieve is then applied to perform F0 analysis and grouping according to common F0. In a subsequent paper [3], the authors extend their system by introducing two further stages of processing. The first of these addresses a problem caused by the distributive nature of the DFT, namely that evidence for a particular F0 is distributed across various harmonic frequencies along the modulation frequency axis of the map. The author’s solution is to compute a pooled map, in which evidence for each F0 is integrated. The resulting representation is better suited to grouping and pitch analysis, since a single peak occurs in the pooled map for each period source. The second stage of processing is an identification map, which estimates the correlation between stored spectral prototypes and each spectral slice along the modulation frequency axis. This allows classification of vowel spectra without the need for an explicit F0 detection stage. It is an open question whether a similar mechanism could be used to segregate continuous speech, rather than isolated vowels; the computational cost may be prohibitive.

Other principles of auditory organization, such as spectral smoothness, may also be used to improve F0 detection and tracking. Klapuri [33] describes a multipitch estimation technique which exploits a spectral smoothness principle. His system uses an iterative approach to multipitch estimation, in which a predominant F0 is found, and then the corresponding harmonic spectrum is estimated and linearly subtracted from the mixture. This process is then repeated for the residual. However, this approach has a tendency to make errors when constituent F0s in the mixture are harmonically related, because cancellation of one F0 may inadvertently remove a frequency component that is shared by another source. The solution proposed in [33] is to smooth the spectrum before subtraction; partials containing energy from more than one source extend above the smoothed envelope, so that they are preserved in the residual when the smoothed envelope is subtracted from the mixture. Klapuri shows that application of the spectral smoothness constraint reduces the error rate for pitch analysis of four-pitch mixtures by about half.

Finally, although most monaural CASA systems have used F0-based cues, there have been some attempts to exploit other cues such as frequency modulation [40] and common onset [18], [6], [7], [28]. For example, Denbigh and Zhao [18] describe a system which selects the harmonics for a target voice in a manner that is similar to the approach described by Parsons [49]. Additionally, their system compares adjacent spectra in order to determine whether the onset of a new voice has occurred. This allows their pitch tracking algorithm to extract weak voiced sounds, and increases the accuracy of pitch tracking when two voices are active. Common onset is currently a somewhat under-utilized cue in CASA systems, although a recent study has been described in which it is employed to segregate stop consonants [28].

16.3.3 Binaural Approaches

The principal cues that human listeners use to determine the location of a sound source are those that involve a comparison between the two ears. A sound source located to one side of the head generates sound pressure waves that arrive at the nearer ear slightly before the farther ear; hence there is an interaural time difference (ITD) which provides a cue to source location. Similarly, the sound intensity will be greater in the nearer ear, causing an interaural intensity difference (IID). The IID is usually expressed in decibels, in which case it is termed the interaural level difference (ILD). The relative efficacy of ITD and ILD cues depends on frequency. At low frequencies, sounds diffract around the head and hence there is no appreciable ILD below about 500 Hz. At high frequencies, ITD does not provide a reliable cue for the location of tonal sounds because of phase ambiguities. However, the envelope of complex sounds can be compared at the two ears in high frequency regions; this cue is referred to as the interaural envelope difference (IED).

In addition to binaural comparisons, direction-dependent filtering by the head, torso and pinnae provide cues to source location. These provide some ability to localize sounds monaurally, and are particularly important for discrimination of elevation and for resolving front-back confusions. Such cues are seldom used explicitly by CASA systems and are not considered here; however, their use in CASA systems remains an interesting area for future research. Preliminary work on a sound localization system that exploits pinna cues is reported in [25].

Computational systems for binaural signal separation have been strongly influenced by two key ideas in the psychophysical literature. The first is Durlach's [20] equalization-cancellation (EC) model of binaural noise suppression, which is a two-stage scheme. In the first stage, equalization is applied to make the noise components identical in each of two binaural channels. This is followed by a cancellation stage, in which the noise is removed by subtracting one channel from the other. Many two-microphone approaches to noise cancellation may be regarded as variants of the EC scheme (e.g., [61], [38], [56]).

The second key idea motivating binaural signal separation systems is the cross-correlation model of ITD processing proposed by Jeffress [30]. In this scheme, neural firing patterns arising from the same critical band of each ear travel along a dual delay-line system, and coincide at a delay corresponding to the ITD. Computationally, the Jeffress model may be expressed as a cross-correlation of the form

$$C(t, f, \tau) = \sum_{n=0}^{N-1} h_L(t-n, f) h_R(t-n-\tau, f) w(n), \tag{16.6}$$

where $h_L(t, f)$ and $h_R(t, f)$ represent the simulated auditory nerve response in the left and right ears respectively for time frame t and frequency channel f, and $w(n)$ is a window of size N samples. The resulting cross-correlogram $C(t, f, \tau)$ is closely related to the correlogram given in (16.2); both are three-dimensional representations in which frequency, time and lag are represented on orthogonal axes. Figure 16.6A shows a cross-correlogram for a mixture of a male and female speaker, originating from azimuths of -15 degrees and $+10$ degrees respectively. As with the correlogram, it is convenient to sum the cross-correlation functions in each frequency band to give a summary cross-correlation function (SCCF), in which large peaks occur at the ITD of each source (Fig. 16.6B). The figure also shows the ILD for this mixture,

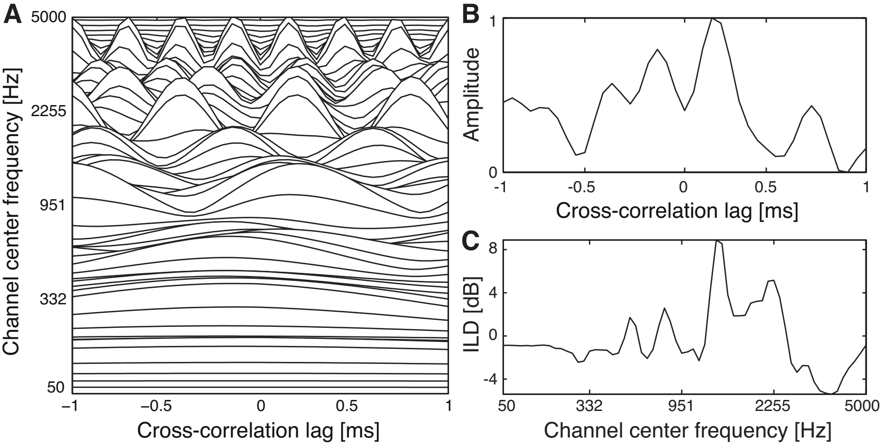


Fig. 16.6. A. Cross-correlogram for time frame 100 of the mixture of speakers shown in Fig. 16.1, for which the male speaker has been spatialised at an azimuth of -15 degrees and the female speaker at an azimuth of +10 degrees. B. Summary cross-correlogram. C. Interaural level difference (ILD) in each frequency channel.

computed using

$$ILD(t, f) = 10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} h_R(t+n, f)^2}{\sum_{n=0}^{N-1} h_L(t+n, f)^2} \right). \tag{16.7}$$

Note that, as expected, the ILD is negligible in low frequency channels. However, in the mid-frequency region channels tend to be dominated by the female speaker and exhibit a large positive ILD. Above 2.5 kHz, the male speaker is dominant and a substantial negative ILD is observed.

Early attempts to exploit spatial cues in a system for speech segregation include the work of Lyon [39] and the binaural 'cocktail party processor' described by Bodden [4]. Bodden's system is based on a cross-correlation mechanism for localizing the target and interfering sources, and uses a time-variant Wiener filter to enhance the target source. Effectively this filter applies a window function to the azimuth axis of the cross-correlogram, such that energy from a speaker at a target azimuth is retained, and the remainder is cancelled. Bodden reports good performance for mixtures of two or three speakers in anechoic conditions.

Bodden's system uses a modification of the Jeffress scheme in which contralateral inhibition is employed to sharpen the cross-correlogation pattern. Numerous other approaches have been described for improving the accuracy of location estimates from cross-correlation processing, such as the 'stencil' filter proposed by Liu *et al.* [37]. The SCCF shown in Fig. 16.6B is a direct way of estimating the ITD of a sound source from the cross-correlogram, but it assumes that the ITD is independent of frequency. This assumption does not hold if the binaural recordings are obtained from a dummy head, because diffraction around the head introduces a weak frequency dependence to the ITD. The 'stencil' approach described by Liu *et al.* is a more sophisticated way of determining source location, which is based on pattern-matching the peaks in the cross-correlogram. At the ITD of a sound source, the pattern of peaks in the cross-correlogram exhibits a structure in which curved traces fan out from a central vertical line (two such structures are visible in Fig. 16.6A, which is a cross-correlogram for a two-source mixture). Accordingly, Liu *et al.* derive a SCCF by integrating activity in the cross-correlogram over a template ('stencil') for each location, which matches the expected pattern of peaks. They report enhanced localization of sources in the azimuthal plane using this method.

A related approach is described by Palomäki *et al.* [47], who form a 'skeleton' cross-correlogram by identifying local peaks and replacing each with a narrower Gaussian. This avoids the problem of very wide peaks, which occur in low-frequency channels and bias the location estimates in the SCCF (see Fig. 16.6A). Furthermore, in the process of forming the skeleton cross-correlogram, peak positions are mapped from ITD to an azimuth axis using frequency-dependent look-up tables. Again, this overcomes the problems associated with the frequency dependence of ITD. Palomäki *et al.*'s system also includes a mechanism for reducing the effect of echoes on localization estimates. This consists of a delayed inhibition circuit, which ensures that location cues at the onset of a sound source have more weight that those that arrive later. In this respect, it may be regarded as a simple model of the precedence effect (for a review, see [41]). The authors report that use of the inhibition mechanism improves the robustness of source localization in mildly reverberant environments.

Roman *et al.* [52] describe a binaural speech separation algorithm which is based on location estimates from skeleton cross-correlograms. They observe that, within a narrow frequency band, modifying the relative strength of a target and interfering source leads to systematic changes in the observed ITD and ILD. For a given location, the deviation of the observed ITD and ILD from ideal values can therefore be used to determine the relative strength of the target and interferer, and in turn this can be used to estimate the ideal binary mask [see eq. (16.1)]. Specifically, a supervised learning method is used for different spatial configurations and frequency bands based on an ITD-ILD feature space. Given an observation x in the ITD-ILD feature space, two hypotheses are tested for each channel; whether the target is dominant (H_1) and whether the interferer is dominant (H_2). Based on estimates of the bivariate densities $p(x|H_1)$ and $p(x|H_2)$, classification is performed using a maximum *a posteriori* (MAP) decision rule, i.e. $p(H_1)p(x|H_1) > p(H_2)p(x|H_2)$. Roman *et al.*'s system includes a resynthesis pathway, in which the target speech signal is reconstructed only from those time-frequency regions selected in the binary mask. They report a performance in anechoic environments which is very close to that obtained using the ideal binary mask, as determined using three different evaluation criteria (signal-to-noise ratio (SNR), automatic speech recognition accuracy and listening tests).

A limitation of binaural systems is that they generally perform well when two sources are present, but their performance degrades in the presence of multiple interferers. Liu *et al.* [38] describe a multi-band mechanism which allows this limitation to be overcome to some extent. Their binaural cancellation scheme is based on a subtraction of the two input signals. Essentially, their system generates a nulling pattern for each point on the lag-axis of the cross-correlogram, such that the null occurs at the direction of the noise source and unity gain is maintained in the direction of the target sound. An innovative aspect of their approach is that the null in each frequency band can be steered independently, so that at each time instant it cancels the noise source that emits the most energy in that band. This allows their system to cancel multiple noise sources, provided that their locations are known; this information is provided by the author's system for sound localization, discussed above. Liu *et al.* show that when four talkers are present in an anechoic environment, their system is able to cancel each of the three interfering speakers by 3-11 dB whilst causing little degradation to the target speaker. Similar results were obtained for a six-talker scenario. However, in a moderately reverberant room the total noise cancellation fell by about 2 dB; this raises doubts as to whether the system is sufficiently robust for use in real-world acoustic environments.

A number of workers have developed systems that combine binaural processing with other grouping cues (usually those related to periodicity). An early example is the system proposed by Kollmeier and Koch [34]. They describe a speech enhancement algorithm which works in the domain of the modulation spectrum, i.e. a two-dimensional representation of AM frequency vs. center frequency. Energy from each sound source tends to form a cluster in the modulation spectrum, allowing sources with different modulation characteristics to be separated from one another. Binaural information (ITD and ILD cues) is used to suppress clusters that do not arise from a desired spatial location.

Related speech separation systems which use F0 and binaural cues are described by Denbigh and colleagues [18], [56]. In the latter approach, the cross-correlation between two microphones is continuously monitored in order to determine the azimuth of the most intense sound in each time frame. If the most intense sound lies close to the median plane, it is assumed to be speech and an initial estimate is made of the speech spectrum. An estimate of the total interference (noise and reverberation) is also obtained by cancelling the dominant target signal. Subsequent processing stages refine the estimate of the speech spectrum, by subtracting energy from it that is likely to be contributed by the interference. Finally, F0 analysis is performed on the extracted target signal, and cross-referenced against F0 tracks from the left and right microphones. A continuity constraint is then applied to ensure that the F0 of the estimated target speech varies smoothly. The target speech signal is reconstructed using the overlap-add technique and passed to an automatic speech recogniser. The authors report a large gain in ASR accuracy for an isolated word recognition task in the presence of a speech masker and mild reverberation; accuracy increased from 30% to 95% after processing by the system, for a SNR of 12 dB.

Okuno *et al.* [45] also describe a system which combines binaural and F0 cues, and they assess its ability to segregate mixtures of two spatially separated speakers. Harmonic fragments are found in each of the left and right input channels, and then a direction is computed for pairs of fragments using ITD and ILD cues. The authors evaluate their system on a speech recognition task, but focus on the ability of the system to recognize *both* utterances rather than a single target utterance. They find that ASR error rates are substantially reduced by using their system, compared to performance on the unprocessed speech mixtures. They also report that binaural cues play an important role in this result; the ASR accuracy of a system which only used harmonic fragments was about half that of a system which used both harmonic and binaural cues.

16.3.4 Frameworks for Cue Integration

So far, we have focused on the cues that are pertinent to CASA, but a key issue remains – how can cues be combined in order to find organization within an acoustic signal, and hence retrieve a description of a target sound source from a mixture?

The earliest approaches to CASA were motivated by classical artificial intelligence techniques, in that they emphasized representation and search. For example, Cooke's system [11] employs a synchrony strand representation, in which the acoustic scene is encoded as a collection of symbols that extend through time and frequency. A search algorithm is employed to identify groups of strands that are likely to have arisen from the same source. This is mainly achieved on the basis of harmonicity; search proceeds from a 'seed' strand, and other strands that are harmonically related to the seed strand are added to its group. In a second grouping stage a pitch contour is derived for each group, using the frequency of resolved harmonics in low-frequency regions, and using AM frequency in high-frequency regions. Groups that share a common pitch contour are then combined. In addition, a subsumption stage removes groups whose strands are contained in a larger grouping. Brown [6], [7] describes a similar approach, but substantially expands the palette of acoustic representations by including time-frequency 'maps' of onset activity, offset activity, frequency transition and periodicity. These are combined to form a symbolic representation of the acoustic scene, which is searched in a similar manner to Cooke's.

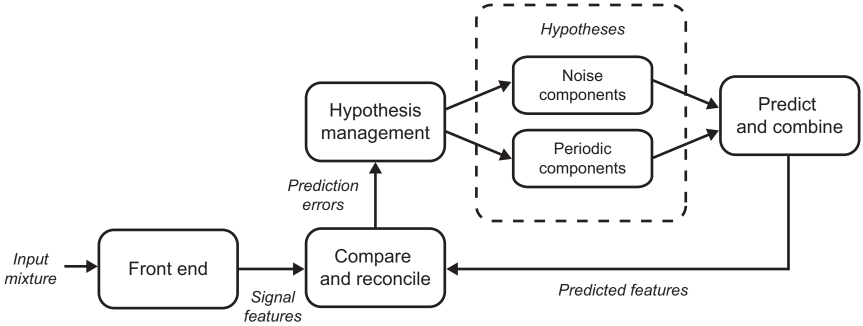


Fig. 16.7. Flow of processing in the prediction-driven architecture of Ellis [21]. Redrawn from [23].

Neither of the systems described above constitute a generic architecture for cue integration. Rather, Cooke's system combines groups of strands using a single derived property (pitch contour) and Brown's system performs cue integration during the formation of time-frequency objects. Hence, it is not clear how other cues (such as those relating to spatial location) could be included in these systems. Also, both are essentially data-driven architectures, as shown in Fig. 16.2. In general, solution of the CASA problem requires the application of top-down knowledge as well as bottom-up processing. Finally, both systems run in 'batch' mode; they process the acoustic signal in its entirety in order to derive an intermediate representation, which is then searched. Clearly, an architecture that allows real-time processing is necessary for most applications (such as hearing prostheses and automatic speech recognition).

A more generic architecture for cue integration is the blackboard, as advocated by Cooke *et al.* [12] and Godsmark and Brown [24]. In this scheme, grouping principles such as harmonicity are cast as knowledge sources ('experts') that communicate through a globally accessible data structure (the blackboard). Experts indicate when they are able to perform an action, and place their results back on the blackboard. For example, a harmonicity expert might be initiated because harmonically related partials are available on the blackboard, and would compute a pitch contour from them. In turn, another expert might combine groups that have the same pitch contour. Centralized control is provided by a scheduler, which determines the order in which experts perform their actions. Blackboard architectures are well suited to CASA because they were developed to deal with problems that have a large solution space, involve noisy and unreliable data, and require many semi-independent sources of knowledge to form a solution.

Godsmark and Brown's system [24] is specialized for musical signals, rather than speech, but is interesting because it suggests a mechanism for resolving competition between grouping principles. Such competition might arise if, for example, two acoustic components were sufficiently distant in frequency to be regarded as separate, but sufficiently close in time to be regarded as grouped. They adopt a 'wait and see' approach to this problem; many possible organizations are maintained within a sliding time window. Within the sliding window, alternate organizations of synchrony strands are scored by grouping experts. An organization is only imposed on a section of the acoustic signal after the window has passed over it, thus allowing contextual information to influence the organization of strands into groups. The authors also show how top-down and bottom-up processing can be combined in a multi-layered blackboard architecture. For example, predictions about anticipated events, based on a previously observed temporal pattern, can be used to influence the formation and grouping of synchrony strands.

A similar emphasis on the role of top-down processing is found in the study by Ellis [21], who describes a *prediction-driven* architecture for CASA. By way of contrast with the Cooke and Brown systems, in which the flow of information is linear and data-driven, Ellis's approach involves a feedback loop so that predictions derived from a 'world model' can be compared against the input (see Fig. 16.7). The front-end processing of Ellis' system forms two representations, a time-frequency energy envelope and correlogram. These representations are reconciled with predictions based on world-model hypotheses by a comparison block. The world model itself consists of a hierarchy of increasingly specific sound source descriptions, the lowest level of which is couched in terms of three sound elements; wefts (which represent pitched sounds), noise clouds and transient clicks. A reconciliation engine, which is based on a blackboard system, updates the world model according to differences detected between the observed and predicted signals.

Okuno *et al.* [44] describe a residue-driven architecture for CASA which is closely related to Ellis' approach, in that it compares the acoustic input against predictions from a world model. However, Ellis' system makes this comparison at the level of intermediate acoustic representations (such as the smoothed spectral envelope). In contrast, the residue-driven architecture reconstructs a time-domain waveform for the modelled signal components, and subtracts this from the acoustic input to leave a residue which is then further analyzed. Okuno *et al.* implement the residue-driven approach within a multi-agent system, in which three kinds of agent (event-detectors, tracer-generators and tracers) initiate and track harmonic fragments. The multi-agent framework is similar in concept to the blackboard – 'experts' and 'agents' are roughly equivalent – except that agents communicate directly rather than through a global data structure.

Some recent approaches to cue integration in CASA have been motivated by the development of powerful algorithms in the machine learning community, rather than classical artificial intelligence techniques. For example, Nix *et al.* [43] describe a statistical approach to CASA which is based on a state-space approach. Specifically, they consider the problem of separating three speakers using two microphones. The problem is formulated as a Markov state-space of the form

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{v}_{k-1}), \tag{16.8}$$

$$\mathbf{z}_k = \mathbf{g}_k(\mathbf{x}_k, \mathbf{n}_k). \tag{16.9}$$

Here, $\mathbf{x}_k$ represents the azimuth, elevation and short-time magnitude spectrum of each speaker at time k (which are unknown), and $\mathbf{z}_k$ is the power spectral density observed at the two microphones. The function $\mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{v}_{k-1})$ corresponds to the probability density function $p(\mathbf{x}_k|\mathbf{x}_{k-1})$, i.e. the probability that one pair of directions and magnitude spectra at time k succeed another pair of directions and magnitude spectra at time $k-1$. The function $\mathbf{g}_k(\mathbf{x}_k, \mathbf{n}_k)$ corresponds to $p(\mathbf{z}_k|\mathbf{x}_k)$, which is the probability that an observation $\mathbf{z}_k$ is made when the state of the system is $\mathbf{x}_k$. The random variables $\mathbf{v}_k$ and $\mathbf{n}_k$ are termed the 'process noise' and 'observation noise' respectively, and have known statistics. The task is to estimate $\mathbf{x}_k$ from the values of $\mathbf{z}_i$, $1 \leq i \leq k$, in an optimal manner. This estimation task is performed by a sequential Monte Carlo method (also known as the 'particle filter' or 'condensation' algorithm).

The performance of the system reported in [43] is somewhat disappointing; although the system reliably tracks the direction and short-time magnitude spectrum of a single source, it is unable to estimate the spectra of two concurrent voices with any accuracy. Additionally, the computational requirement of the algorithm is high and training is time consuming; the authors report that it took several weeks to estimate $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ from a large database of recorded speech.

Finally, we note that the problem of cue integration in CASA is closely related to the *binding problem*. This term refers to the fact that information about a single sensory event is distributed across many areas of the brain – how is this information bound together to form a coherent whole? One possibility is that the grouping of neural responses is performed by *oscillatory correlation*. In this scheme, neurons that represent features of the same sensory event have synchronized responses, and are desynchronized from neurons that represent different events. Wang and colleagues [64], [66] have used the principle of oscillatory correlation to build a neurobiologically-motivated architecture for CASA. In the first stage of their scheme, the correlogram is employed to detect the periodicities present in local time-frequency regions. Subsequently, processing is performed by a two-layer oscillator network which mirrors the two conceptual stages of ASA. In the first (segmentation) layer, periodicity information is used to derive segments, each of which encodes a significant feature in the time-frequency plane. Oscillators belonging to the same segment are synchronized by local connections. In the second (grouping) layer, links are formed between segments that have compatible periodicity information along their length (i.e., those that are likely to belong to the same F0). As a result, groups of segments form in the second layer which correspond to sources that have been separated by their F0.

Frameworks for CASA based on neural oscillators have two attractive features. Firstly, they are based on a parallel and distributed architecture which is suitable for implementation in hardware. Secondly, because the oscillator in each time-frequency region may be regarded as 'on' or 'off' at any particular time instant, the output of an oscillator array may be interpreted as a binary time-frequency mask. This makes them eminently suitable as a front-end to ASR systems that employ missing feature techniques (see below).

16.4 Integrating CASA with Speech Recognition

Conventional ASR systems are constructed on the assumption that the input to them will be speech. In practice this is usually not the case, because speech is uttered in acoustic environments in which other sound sources may be present. As a result, the performance of conventional ASR system declines sharply in the presence of noise.

ASR is a pattern recognition problem in which observed acoustic features $\mathbf{X}$ must be assigned to some class of speech sound. This is achieved by selecting the word sequence W which maximizes the posterior probability $P(W|\mathbf{X})$, which can be expressed using Bayes theorem as

$$\hat{W} = \underset{W}{\text{argmax}} \frac{P(\mathbf{X}|W)P(W)}{P(\mathbf{X})}, \tag{16.10}$$

where $P(W)$ is the language model and the likelihood $P(\mathbf{X}|W)$ is the acoustic model. One approach to improving the noise-robustness of ASR is to enhance the speech in the acoustic mixture, so that the observed features resemble the acoustic model as closely as possible. This provides the most straightforward approach to integrating CASA and ASR; the CASA system segregates the speech from the acoustic mixture, and then a 'clean' signal is resynthesized and passed to the recognizer. A number of CASA systems have included such a resynthesis pathway by inverting a time-frequency representation (e.g., see [67], [11], [6]). Other representations can also be inverted. For example, Slaney *et al.* [58] describe an approach for inverting the correlogram, which allows sounds to be segregated according to their F0s and then reconstructed.

An advantage of the resynthesis approach is that it allows the use of unmodified ASR systems; this is preferable, because the front-end processing used by CASA systems does not usually provide acoustic features that are suitable for training a conventional ASR system. However, the approach has met with limited success. For example, the system described by Weintraub paired CASA with a speaker-independent continuous-digit-recognition system, and attempted to recognize utterances simultaneously spoken by a male and female speaker. A modest improvement in recognition accuracy was obtained for the (dominant) male voice, but performance for the female speaker actually fell as a result of CASA processing.

A further criticism of the resynthesis approach is that it embodies a very weak link between the CASA and ASR systems; given the important role of schema-driven grouping, one would expect that a tighter integration of CASA and speech models would be beneficial. Ellis [23] has addressed this issue by integrating a speech recognizer into his prediction-driven CASA architecture. When presented with an acoustic mixture, his system attempts to interpret it as speech. Following decoding by the ASR system, an estimate of the speech component of the mixture is used to determine the characteristics of the remaining (nonspeech) signal features. In turn, the estimate of the nonspeech components can be used to re-estimate the speech. Hence, an iterative cycle of estimation and re-estimation develops, which finally converges on an explanation of the acoustic mixture in terms of the speech and nonspeech components present.

Techniques such as correlogram inversion [58] attempt to reconstruct areas of the speech spectrum that have been obliterated by an interfering noise. An alternative approach is to identify those time-frequency regions that are missing or considered unreliable, and treat them differently during the decoding stage of ASR. Specifically, Cooke *et al.* [13] have proposed a *missing feature* approach to ASR which links closely with CASA. In their approach, the observed acoustic features $\mathbf{X}$ are partitioned into two sets, X_r and X_u, which correspond to reliable and unreliable features respectively. Using a modified continuous-density HMM (CDHMM), the authors show that it is possible to impute the values of the missing features X_r. Alternatively, the maximum a posteriori estimate of the speech class can be found as given in (16.10), by replacing the likelihood $P(\mathbf{X}|W)$ with the marginal distribution $P(X_r|W)$. Furthermore, a 'bounded marginalization' approach may be used in which the values of the missing features are constrained to lie within a certain range. For example, the value of a spectral feature must lie between zero and the observed spectral energy.

In practice, a missing feature recognizer is provided with a set of acoustic features and a time-frequency mask, which is typically obtained from a CASA system. The mask may be binary (in which case each time-frequency region is regarded as either reliable or unreliable) or real-valued. In the latter case, each mask value may be interpreted as the probability that the corresponding time-frequency region is reliable.

A number of workers have described ASR systems that combine a missing feature speech recognizer with CASA-based mask estimation. For example, Roman *et al.* [52] employ binaural cues to estimate the ideal binary mask for a target speaker which is spatially separated from an interfering sound source. A large improvement in recognition accuracy was obtained compared to a conventional ASR system, for a connected digit recognition task. Similar results were obtained by Palomäki *et al.* [47], also using a binaural model and missing feature ASR system. Their system computes a binary mask by examining the cross-correlation functions in each time-frequency region. Their system has been evaluated in the presence of moderate reverberation and obtains substantial ASR improvements.

Neural oscillator frameworks for CASA represent an ideal front-end for missing feature ASR systems, because the activity of oscillators arranged in a time-frequency grid can be directly interpreted as a mask. Brown *et al.* [8] describe an oscillator-based CASA system which segregates speech from interfering noise using F0 information (derived from a correlogram). Additionally, unpitched interference is removed by noise estimation and cancellation; oscillators are deactivated (and hence give rise to a mask value of zero) if they correspond to acoustic components that lie below the noise floor. The authors report good performance on a digit recognition task at low SNRs, but note that unvoiced regions of speech are not represented in the oscillator array; as a result, the performance of their system falls below that of a conventional ASR system at high SNRs.

We also note that mask estimation can be achieved in a purely top-down manner. Roweis [54] describes a technique for estimating binary masks using an unsupervised learning method. Specifically, speaker-dependent HMMs are trained on the speech of isolated talkers, and then combined into a factorial HMM (FHMM). The latter consists of two Markov chains which evolve independently. Given an mixture of two utterances, the underlying state sequence in the FHMM is inferred and the output predictions for each Markov chain are computed. A binary mask is then determined by comparing the relative values of these output predictions.

The missing feature approach achieves a tighter integration between CASA and ASR, but still embodies a unidirectional flow of information from the front-end to the recognizer. However, Barker *et al.* [1] report a further development of the missing feature technique which accommodates data-driven and schema-driven processing within a common framework; the so-called *multisource decoder*. In this approach, it is assumed that the observed features $\mathbf{Y}$ represent a mixture of speech and interfering sound sources. The goal is therefore to find the word sequence W and segregation mask S which jointly maximize the posterior probability,

$$\hat{W}, \hat{S} = \underset{W,S}{\operatorname{argmax}}\, P(W, S|\mathbf{Y}). \tag{16.11}$$

Barker *et al.* show that $P(W, S|\mathbf{Y})$ can be written in terms of the speech features $\mathbf{X}$ (which are now considered to be unobserved) by integrating over their possible values, giving

$$P(W, S|\mathbf{Y}) = P(W) \left(\int P(\mathbf{X}|W) \frac{P(\mathbf{X}|S, \mathbf{Y})}{P(\mathbf{X})} d\mathbf{X} \right) P(S|\mathbf{Y}). \tag{16.12}$$

As before, $P(W)$ and $P(\mathbf{X}|W)$ in (16.12) represent the language model and acoustic model respectively. However, two new terms are introduced. $P(S|\mathbf{Y})$ is a segregation model, which describes the probability of a particular mask

S given the observed features $\mathbf{Y}$, but independent of the word hypothesis W. Such information can be obtained from a data-driven CASA system. The remaining term $P(\mathbf{X}|S, \mathbf{Y})/P(\mathbf{X})$ is a likelihood weighting factor. Most importantly, the maximization in (16.11) occurs over both W and S so that both schema-driven and data-driven information are incorporated in the search.

Barker *et al.* derive an efficient search technique for evaluating (16.12) within a CDHMM system, and test the decoder on a noise-corrupted connected digit task. Segregation masks were obtained using a simple spectral subtraction approach. A reduction in word error rate of about 25% was obtained, relative to a conventional ASR system. The authors predict that further performance gains can be achieved by using CASA processing to estimate the masks.

Yet another way of integrating CASA and speech recognition is to use speech schemas triggered by recognition to restore speech which has been masked by noise. Specifically, Srinivasan and Wang propose a schema-based model for phonemic restoration [60]. Their model estimates reliable time-frequency regions and feeds them to a missing feature recognizer. Successful recognition activates a word template, which is then dynamically time warped to the noisy word so as to restore the speech frames corresponding to the noisy portion of the word. Unlike earlier data-driven efforts, their model can restore both voiced and unvoiced phonemes with a high degree of naturalness.

16.5 CASA Compared to ICA

CASA and ICA differ somewhat in their approaches to speech separation; here, we consider some of the differences and also comment on the possibility of harmonizing the two approaches.

Broadly, CASA and ICA differ in a number of respects. For example, CASA emphasizes the role of intermediate signal representations such as the correlogram, whereas ICA usually operates directly on the sampled acoustic signal. Likewise, CASA algorithms exploit continuity in time and frequency, whereas ICA does not. The performance profile of ICA also differs substantially from that of human listeners. For instance, ICA typically aims to segregate every source signal from a mixture, whereas human listeners perform figure/ground segregation. Similarly, CASA systems – which are motivated by human performance – often aim to separate a target speaker from the acoustic background rather than completely demix the input (e.g., [66], [40], [8]).

A direct comparison of CASA and ICA was reported by van der Kouwe *et al.* [63]. They compared the performance of Wang and Brown's CASA system [66] with two schemes for ICA, one of which was the fourth-order JADE method [9]. The algorithms were evaluated on Cooke's [11] corpus of speech and noise mixtures, and performance was expressed in terms of the gain in SNR obtained. It was found that the CASA and ICA algorithms performed well under very different conditions. In general, CASA techniques require that the acoustic mixture exhibits well-defined regions in the time-frequency plane which correspond to one or more sound sources. Hence, the performance of the CASA system was best in conditions in which the interferer was tonal or locally narrowband. The JADE algorithm did not perform as well in these conditions, presumably because the narrowband interferers yielded poor higher-order statistics. On the other hand, the CASA system performed poorly in conditions where there was substantial spectral overlap between the speech and interferer. Again the situation for JADE was the opposite; it performed particularly well with broadband interferers (such as speech and random noise), which contain rich higher order joint statistics.

It should be noted that comparison of CASA and ICA is frustrated by the lack of a suitable corpus. The speech and noise mixtures employed by van der Kouwe *et al.* were not ideal, because the mixing process was constant and linear, the mixing matrix was far from singular, there were two mixtures and two sources, and source signals were perfectly temporally aligned in both mixtures. Such conditions meet all of the requirements for ICA (except for statistical independence of the sources), but are not representative of mixtures recorded in real acoustic environments. On the other hand, the corpus was designed to present a challenging test for CASA systems [11], which do not have such requirements. Clearly, further comparison of CASA and ICA techniques would be facilitated by the availability of a corpus that was designed for evaluating both approaches.

Although CASA and ICA differ in their approaches, there are some similarities between them. For example, de Cheveigné [14] notes the similarity between frequency-domain ICA and equalization-cancellation models of binaural signal detection. Also, there are possibilities for combining the two approaches [59]. Yilmaz and Rickard [71] describe an approach for separating speech mixtures via the blind estimation of time-frequency masks, which is closely related to the system of Roman *et al.* [52]. Such an approach could be integrated with CASA systems that use a similar time-frequency representation (e.g., [7], [66], [29]). Another example of the combination of ICA and CASA technique is provided by Rutkowski *et al.* [55], who describe a system in which ICA is applied to each frequency channel of a correlogram. The extracted signals in each channel that have a periodic structure are used to reconstruct a time-domain waveform using correlogram inversion [58], whereas the remaining noisy signals are discarded. The authors report good performance for the separation of two sources recorded in a reverberant room, which exceeds the performance expected using CASA or ICA alone.

16.6 Challenges for CASA

In this penultimate section, we briefly review some of the challenges that remain for CASA, and make suggestions for further work.

Evaluation is an important issue for CASA that requires further thought. Research in ASR has undoubtedly benefitted from the adoption of standard metrics and evaluation tasks for comparing performance, such as those introduced by the US National Institute of Standards and Technology (NIST). The situation in CASA is very different; workers rarely compare their work on the same corpus and use a variety of performance metrics. The latter include comparisons of intermediate auditory representations [11], various metrics related to SNR [7], [66] and ASR performance using conventional or 'missing feature' speech recognizers [67], [52], [47]. Ellis [22] argues that the CASA research community should standardize on an evaluation domain that is relevant to a real-world problem (such as acoustic analysis of multi-party meetings), and that the performance of CASA systems should be judged against human performance on the same task.

On a related point, CASA is informed and motivated by the psychophysical literature on ASA (and to a lesser extent, the physiological literature). However, if CASA systems are 'models' of human function in a true sense, then they should be able to generate hypotheses that can be tested by further psychophysical experimentation. In fact, there is currently little evidence of such synergy occurring. A notable exception is the work of Cooke [10], who has proposed a 'glimpsing' model of human speech perception based on insights gained from his missing feature approach to ASR.

Most work in CASA assumes that sound sources remain in fixed positions for the duration of an input signal. This is not representative of real-world environments, and dealing with moving sound sources remains a challenging research issue. Early work on this problem is reported by Roman and Wang [51], who describe a binaural model based on the same principles as the multi-pitch tracking algorithm of Wu *et al.* [69]. Following auditory filtering and cross-correlation, an HMM is used to form continuous location tracks and estimate the number of sound sources present. Their approach performs well; for example, it is able to simultaneously track two sound sources whose trajectories cross in space.

Another interesting area for further study is the combination of CASA algorithms with established signal processing techniques. For example, Drake *et al.* [19] describes a two-stage algorithm for CASA-enhanced beamforming (CASA-EB). In the first stage of her system, the output from each channel of an auditory model is processed by a beamformer and mapped to a three dimensional space with dimensions of frequency, time and arrival angle. In the second stage, acoustic components are grouped according to F0 and estimated location. Drake *et al.* demonstrate that in most conditions, the performance of CASA-EB is superior to that of monaural CASA or beamforming alone.

The role of attention in CASA has largely been overlooked in computational studies, and merits further work. A model of auditory attention might allow a single source to be tracked in a changing acoustic environment, or allow the most salient source to be extracted from a mixture. Preliminary work in this regard has been reported by Wrigley and Brown [70]. In their model, a network of neural oscillators performs stream segregation using a principle of oscillatory correlation. A weighting is given to specific frequency regions using a Gaussian-shaped function, which determines the connection weights between oscillators and an attentional unit. When the activity of the attentional unit is synchronized with a group of oscillators, the corresponding acoustic features are held to be in the attentional foreground. The authors have demonstrated the ability of their model to explain psychophysical findings related to the perception of tonal stimuli, but the model remains to be tested with complex stimuli such as speech.

Another challenge for CASA is the monaural separation of unvoiced speech; this issue has received much less attention than the problem of segregating voiced speech using pitch cues. Recently, Hu and Wang [28] have described a system for separating stop consonants from background interference. Stops generally consist of a weak closure and a subsequent burst, which is usually unvoiced and cannot therefore be separated from interference on the basis of pitch. Instead, Hu and Wang's system identifies stops by detecting onsets in the average rate response at the output of each channel of an auditory filterbank. If a significant onset has occurred, it is classified by a Bayesian decision rule on the basis of its spectral shape, intensity and decay time in order to determine whether it corresponds to a stop consonant (as opposed to another impulsive non-speech sound). The authors evaluate their algorithm on a task involving the detection of stop consonants mixed with several types of environmental noise. The system performs respectably when the SNR is high (20 dB or above) but detection performance falls rapidly at lower SNRs. However, the number of confusions (interfering signals which are erroneously identified as stops) remains relatively low, even at 0 dB SNR.

Finally, relatively few workers have evaluated their CASA systems in reverberant conditions, and still fewer have included mechanisms that deal specifically with reverberated input. Exceptions include the binaural model of Palomäki *et al.* [47], which employs a simple model of the 'precedence effect' to remove echoes due to room reverberation. Evaluation of sound separation systems in reverberant conditions has also been undertaken by Kollmeier and Koch [34], Shamsoddini and Denbigh [56] and Roman and Wang [53]. Dealing with reverberation in single-microphone recordings is a particularly challenging issue. Recently, Wu and Wang [68] have proposed an approach to this problem which is based on a two-stage algorithm. In the first stage, a pitch-based metric is used to estimate the reverberation time; this is based on the author's previous multipitch tracking algorithm [69]. In the second stage, an enhancement method estimates and subtracts the acoustic energy due to echoes. An novel approach to the problem is also proposed by Palomäki *et al.* [48]. They describe a system for recognition of reverberated speech in which a time-frequency 'reverberation mask' is estimated for use with a missing feature ASR system. Elements that are selected in the mask correspond to time-frequency regions that are relatively uncontaminated by reverberation. Because their system is based on a mask representation, it would be relatively straightforward to combine it with other CASA algorithms.

16.7 Conclusions

In summary, CASA aims to replicate the perceptual processes by which human listeners segregate simultaneous sounds. It is a growing research area, which is attracting the interest of workers in the machine learning community as well as those in the signal processing and computational modelling communities. Much progress has been made in CASA-based speech separation in the last few years.

Given the topic of this volume, we conclude with some further comments on the differences between CASA and ICA. Firstly, we note that speech segregation need not require the resynthesis of a high-quality speech signal, as is assumed in most ICA studies. If the goal is robust ASR, then only a time-frequency mask and its corresponding acoustic features are needed. Secondly, we have indicated that there are some prospects for marrying the CASA and ICA approaches. For example, ICA can be used to find the independent components in each channel of an auditory filterbank, or in mid-level auditory representations such as the correlogram.

CASA is motivated by an account of auditory perception; indeed, the term 'model' is frequently used to describe CASA systems. We believe that adherence to the general principles of auditory processing is likely to give rise to CASA systems that make fewer assumptions than those based on ICA, and we are hopeful that this will translate into superior performance in real-world acoustic environments.

Acknowledgments

GJB was supported by EPSRC grant GR/R47400/01. DLW was supported in part by grants from the AFOSR, AFRL/IF and NSF.

References

1. J. P. Barker, M. P. Cooke, and D. P. W. Ellis, "Decoding speech in the presence of other sources," *Speech Communication*, 2004, in press.
2. F. Berthommier and G. F. Meyer, "Source separation by a functional model of amplitude demodulation," in *Proc. EUROSPEECH*, 1995, vol. 4, pp. 135–138.
3. F. Berthommier and G. F. Meyer, "Improving amplitude modulation maps for F0-dependent segregation of harmonic sounds," in *Proc. EUROSPEECH*, 1997, vol. 5, pp. 2483–2486.
4. M. Bodden, "Modelling human sound-source localization and the cocktail party effect," *Acta Acustica*, vol. 1, pp. 43–55, 1993.

5. A. S. Bregman, *Auditory Scene Analysis.* MIT Press, Cambridge MA, 1990.
6. G. J. Brown, *Computational Auditory Scene Analysis: A Representational Approach.* Ph.D. Thesis, University of Sheffield, 1992.
7. G. J. Brown and M. P. Cooke, "Computational auditory scene analysis," *Computer Speech and Language*, vol. 8, pp. 297–336, 1994.
8. G. J. Brown, D. L. Wang, and J. Barker, "A neural oscillator sound separator for missing data speech recognition," in *Proc. IJCNN*, 2001, vol. 4, pp. 2907–2912.
9. J. F. Cardoso, "High-order contrasts for independent component analysis," *Neural Computation*, vol. 11, pp. 157–192, 1999.
10. M. P. Cooke, "Making sense of everyday speech: a glimpsing account," in *Speech Separation by Humans and Machines*, edited by P. Divenyi, Springer, New York, 2004.
11. M. P. Cooke, *Modelling Auditory Processing and Organization.* Cambridge University Press, Cambridge, UK, 1993.
12. M. P. Cooke, G. J. Brown, M. D. Crawford, and P. Green, "Computational auditory scene analysis: listening to several things at once," *Endeavour*, vol. 17, no. 4, pp. 186–190, 1993.
13. M. P. Cooke, P. Green, L. Josifovski, and A. Vizinho, "Robust automatic speech recognition with missing and unreliable acoustic data," *Speech Communication*, vol. 34, pp. 267–285, 2001.
14. A. de Cheveigné, "The cancellation principle in acoustic scene analysis," in *Speech Separation by Humans and Machines*, edited by P. Divenyi, Springer, New York, 2004.
15. A. de Cheveigné, "Cancellation model of pitch perception," *J. Acoust. Soc. Am.*, vol. 103, pp. 1261–1271, 1998.
16. A. de Cheveigné and H. Kawahara, "Multiple period estimation and pitch perception model," *Speech Communication*, vol. 27, pp. 175–185, 1999.
17. C. J. Darwin and R. P. Carlyon, "Auditory grouping," in *Hearing*, edited by B. C. J. Moore, Academic Press, 1995.
18. P. N. Denbigh and J. Zhao, "Pitch extraction and separation of overlapping speech," *Speech Communication*, vol. 11, pp. 119–125, 1992.
19. L. A. Drake, A. Katsaggelos, J. C. Rutledge, and J. Zhang, "Sound source separation via computational auditory scene analysis-enhanced beamforming," in *Proc. of the IEEE Sensor Array and Multichannel Signal Processing Workshop*, 2002.
20. N. I. Durlach, "Note on the equalization and cancellation theory of binaural masking level differences," *J. Acoust. Soc. Am.*, vol. 32, no. 8, pp. 1075–1076, 1960.
21. D. P. W. Ellis, *Prediction-Driven Computational Auditory Scene Analysis.* Ph.D. Thesis, Department of Electrical Engineering and Computer Science, M.I.T, 1996.
22. D. P. W. Ellis, "Evaluating speech separation systems", in *Speech Separation by Humans and Machines*, edited by P. Divenyi, Springer, New York, 2004.
23. D. P. W. Ellis, "Using knowledge to organize sound: the prediction-driven approach to computational auditory scene analysis, and its application to speech/nonspeech mixtures," *Speech Communication*, vol. 27, pp. 281–298, 1998.
24. D. J. Godsmark and G. J. Brown, "A blackboard architecture for computational auditory scene analysis," *Speech Communication*, vol. 27, pp. 351–366, 1999.

25. J. G. Harris, C. J. Pu, and J. C. Principe, “A monaural cue sound localizer,” *Analog Integrated Circuits and Signal Processing*, vol. 23, pp. 163–172, 2000.
26. M. J. Hewitt and R. Meddis, “An evaluation of eight computer models of mammalian inner hair-cell function,” *J. Acoust. Soc. Am.*, vol. 90, no. 2, pp. 904–917, 1991.
27. G. Hu and D. L. Wang, “Speech segregation based on pitch tracking and amplitude modulation,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 1991, pp. 79–82.
28. G. Hu and D. L. Wang, “Separation of stop consonants,” in *Proc. IEEE ICASSP*, 2003, vol. 2, pp. 749–752.
29. G. Hu and D. L. Wang, “Monaural speech segregation based on pitch tracking and amplitude modulation,” *IEEE Trans. Neural Networks*, vol. 15, no. 5, pp. 1135–1150, 2004.
30. L. A. Jeffress, “A place theory of sound localization,” *Journal of Comparative and Physiological Psychology*, vol. 41, pp. 35–39, 1948.
31. A. Jourjine, S. Rickard, and O. Yilmaz, “Blind separation of disjoint orthogonal signals: demixing N sources from 2 mixtures,” in *Proc. IEEE ICASSP*, 2000, pp. 2985–2988.
32. M. Karjalainen and T. Tolonen, “Multi-pitch and periodicity analysis model for sound separation and auditory scene analysis,” in *Proc. IEEE ICASSP*, 1999, vol. 2, pp. 929–932.
33. A. P. Klapuri, “Multiple fundamental frequency estimation based on harmonicity and spectral smoothness,” *IEEE Trans. Speech Audio Processing*, vol. 11, no. 6, pp. 804–816, 2003.
34. B. Kollmeier and R. Koch, “Speech enhancement based on physiological and psychoacoustical models of modulation perception and binaural interaction,” *J. Acoust. Soc. Am.*, vol. 95, pp. 1593–1602, 1994.
35. A. Khurshid and S. L. Denham, “A temporal analysis based pitch estimation system for noisy speech with a comparative study of performance of recent systems,” *IEEE Trans. Neural Networks*, vol. 15, no. 5, pp. 1112–1124, 2004.
36. J. C. R. Licklider, “A duplex theory of pitch perception,” *Experimentia*, vol. 7, pp. 128–133, 1951.
37. C. Liu, B. C. Wheeler, W. D. O’Brien, R. C. Bilger, C. R. Lansing, and A. S. Feng, “Localization of multiple sound sources with two microphones,” *J. Acoust. Soc. Am.*, vol. 108 (4), pp. 1888–1905, 2000.
38. C. Liu, B. C. Wheeler, W. D. O’Brien, C. R. Lansing, R. C. Bilger, D. L. Jones, and A. S. Feng, “A two-microphone dual delay-line approach for extraction of a speech sound in the presence of multiple interferers,” *J. Acoust. Soc. Am.*, vol. 110, no. 6, pp. 3218–3231, 2001.
39. R. F. Lyon, “A computational model of binaural localization and separation,” in *Proc. IEEE ICASSP*, 1983, pp. 1148–1151.
40. D. Mellinger, *Event Formation and Separation in Musical Sound*. Ph.D. Thesis, Stanford University, 1991.
41. B. C. J. Moore, *An Introduction to the Psychology of Hearing* (5th edition). Academic Press, 2003.
42. T. Nakatani and H. G. Okuno, “Harmonic sound stream segregation using localisation and its application to speech stream segregation,” *Speech Communication*, vol. 27, pp. 209–222, 1999.

43. J. Nix, M. Kleinschmidt, and V. Hohmann, “Computational auditory scene analysis by using statistics of high-dimensional speech dynamics and sound source direction,” in *Proc. EUROSPEECH*, 2003, pp. 1441–1444.
44. H. G. Okuno, T. Nakatani, and T. Kawabata, “A new speech enhancement: speech stream segregation,” in *International Conference on Spoken Language Processing*, 1996, vol. 4, pp. 2356–2359.
45. H. G. Okuno, T. Nakatani, and T. Kawabata, “Listening to two simultaneous speeches,” *Speech Communication*, vol. 27, pp. 299–310, 1999.
46. L. Ottaviani and D. Rocchesso, “Separation of speech signal from complex auditory scenes,” in *Proc. of the Conference on Digital Audio Effects*, 2001.
47. K. J. Palomäki, G. J. Brown, and D. L. Wang, “A binaural processor for missing data speech recognition in the presence of noise and small-room reverberation,” *Speech Communication*, vol. 43, no. 4, pp. 273–398, 2004.
48. K. J. Palomäki, G. J. Brown, and J. P. Barker, “Techniques for handling convolutional distortion with ‘missing data’ automatic speech recognition,” *Speech Communication*, vol. 43, pp. 123–142, 2004.
49. T. W. Parsons, “Separation of speech from interfering speech by means of harmonic selection,” *J. Acoust. Soc. Am.*, vol. 60, no. 4, pp. 911–918, 1976.
50. R. D. Patterson, K. Robinson, J. Holdsworth, D. McKeown, C. Zhang, and M. Allerhand, “Complex sounds and auditory images,” in *Auditory Physiology and Perception*, edited by Y. Cazals, L. Demany, and K. Horner, Pergamon, Oxford, 1992.
51. N. Roman and D. L. Wang, “Binaural tracking of multiple moving sources,” in *Proc. IEEE ICASSP*, 2003, vol. 5, pp. 149–152.
52. N. Roman, D. L. Wang, and G. J. Brown, “Speech segregation based on sound localization,” *J. Acoust. Soc. Am.*, vol. 114, no. 4, pp. 2236–2252, 2003.
53. N. Roman and D. L. Wang, “Binaural sound segregation for multisource reverberant environments,” in *Proc. IEEE ICASSP*, 2004, vol. 2, pp. 373–376.
54. S. T. Roweis, “One microphone source separation,” *Neural Information Processing Systems*, vol. 13, pp. 793–799, 2000.
55. T. Rutkowski, A. Cichocki, and A. K. Barros, “Speech enhancement from interfering sounds using CASA techniques and blind source separation,” in *3rd International Conference on Independent Component Analysis and Blind Signal Separation*, 2001, San Diego, California, pp. 728–733.
56. A. Shamsoddini and P. N. Denbigh, “A sound segregation algorithm for reverberant conditions,” *Speech Communication*, vol. 33, pp. 179–196, 2001.
57. M. Slaney and R. F. Lyon, “A perceptual pitch detector,” in *Proc. IEEE ICASSP*, 1990, vol. 1, pp. 357–360.
58. M. Slaney, D. Naar, and R. F. Lyon, “Auditory model inversion for sound separation,” in *Proc. IEEE ICASSP*, 1994, pp. 77–80.
59. P. Smaragdis, *Redundancy Reduction for Computational Audition: A Unifying Approach.* Ph.D. Thesis, Program in Media Arts and Sciences, M.I.T., 1994.
60. S. Srinivasan and D. L. Wang, “A schema-based model for phonemic restoration,” *Speech Communication*, 2004, in press.
61. H. W. Strube, “Separation of several speakers recorded by two microphones (cocktail-party processing),” *Signal Processing*, vol. 3, no. 4, pp. 355–364, 1981.
62. T. Tolonen and M. Karjalainen, “A computationally efficient multi-pitch analysis model,” *IEEE Trans. Speech Audio Processing*, vol. 8, no. 6, pp. 708–716, 2000.

63. A. J. W. van der Kouwe, D. L. Wang, and G. J. Brown, “A comparison of auditory and blind separation techniques for speech segregation,” *IEEE Trans. Speech Audio Proc.*, vol. 9, no. 3, pp. 189–195, 2001.
64. D. L. Wang, “Primitive auditory segregation based on oscillatory correlation,” *Cognitive Science*, vol. 20, pp. 409–456, 1996.
65. D. L. Wang, “On ideal binary mask as the computational goal of auditory scene analysis,” in *Speech Separation by Humans and Machines*, edited by P. Divenyi, Springer, New York, 2004.
66. D. L. Wang and G. J. Brown, “Separation of speech from interfering sounds based on oscillatory correlation,” *IEEE Trans. Neural Networks*, vol. 10, no. 3, pp. 684–697, 1999.
67. M. Weintraub, *A Theory and Computational Model of Monaural Auditory Sound Separation.* Ph.D. Thesis, Standford University, 1985.
68. M. Wu and D. L. Wang, “A one-microphone algorithm for reverberant speech enhancement,” in *Proc. IEEE ICASSP*, 2003, vol. 1, pp. 844–847.
69. M. Wu, D. L. Wang, and G. J. Brown, “A multipitch tracking algorithm for noisy speech,” *IEEE Trans. Speech Audio Proc.*, vol. 11, no. 3, pp. 229–241, 2003.
70. S. N. Wrigley and G. J. Brown, “A computational model of auditory selective attention,” *IEEE Trans. Neural Networks*, vol. 15, no. 5, pp. 1151–1163, 2004.
71. O. Yilmaz and S. Rickard, “Blind separation of speech mixtures via time-frequency masking,” *IEEE Trans. Signal Processing*, vol. 52, no. 7, pp. 1830–1847, 2004.

Index