Multichannel equalization applied to speech dereverberation

Rajan Sobhana Rashobh

2015

Rajan Sobhana Rashobh. (2015). Multichannel equalization applied to speech dereverberation. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/62174>

<https://doi.org/10.32657/10356/62174>

Multichannel Equalization Applied to Speech Dereverberation

RAJAN SOBHANA RASHOBH

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy

2015

Acknowledgments

Foremost, I would like to express my deepest appreciation and sincere gratitude to my supervisor Asst. Prof. Andy W. H. Khong who has given me an excellent opportunity to work with him; provided continual support, advice and guidance throughout my research. He has been a motivation for me with his patience, enthusiasm and knowledge. His guidance helped me in all the time of research and writing of this thesis.

I take this opportunity to thank Assoc. Prof. Gan Woon Seng, Assoc. Prof. Soon Ing Yann, and Assoc. Prof. Chng Eng Siong, the panel for my qualifying examination. I am extremely thankful to Ms. Di Liu for her inestimable help. I also thank in advance my examiners and thesis reviewers. I am grateful to all my friends, especially Dr. V. G. Reju and Mr. K. R. Arun, for their invaluable help, suggestions and discussions during tea breaks.

I thank my parents and other family members for their support and encouragement extended throughout my life. Special thanks to my wife, Deepa, for her love, patience, understanding and support which enabled me to complete my thesis. I want to also thank my little daughter Devna, who has been missing my pampering and care.

Last but not least, many thanks go to Mr. Stewart Chu, Workstation Resource Lab, and the lab staff at Computer Engineering lab and DSP lab for their kind assistance.

Abstract

Speech signals acquired by a distant microphone inside an enclosed space is often degraded by reverberation. Reverberation results from the multipath propagation of a sound wave from its source to receivers. Reverberation can cause a detrimental effect on the perceived quality as well as the intelligibility of the speech signals. This results in performance degradation of systems such as hand-free telephony, hearing aids, and automatic speech/speaker recognition systems. One of the popular approaches to mitigate the effects of reverberation is to achieve channel equalization via a two-stage process where acoustic impulse responses (AIRs) are first estimated using blind channel identification (BCI) techniques after which the received signals are filtered using inverse filters computed from the estimated AIRs. This thesis focuses on speech dereverberation employing BCI and inverse filtering.

A typical AIR is often non-minimum phase and its direct inversion will result in an unstable inverse filter. Multichannel equalization (MCEQ) algorithms developed for use with a microphone array are employed for the equalization of such non-minimum phase AIRs. Existing MCEQ algorithms achieve equalization in the time domain and in this thesis, a generalized framework that allows one to achieve equalization in different transform domains is proposed first. This is motivated from the fact that when equalization is performed on different domains, the inherent properties of the transforms can be exploited to achieve better equalization performance.

Noting that the computational complexity of the non-adaptive MCEQ algorithm is proportional to the AIR order, a set of adaptive time-domain MCEQ algorithms are proposed to achieve equalization of high-order AIRs with reduced complexity. These algorithms iteratively estimate the inverse filters by minimizing a cost function. To improve the convergence as well as equalization performance, the sparsity of the desired equalized response is taken into account in the cost function and update equation.

Although the time-domain adaptive algorithms reduce complexity, they suffer from slow convergence. To overcome this limitation, complexity reduction in the frequency domain is exploited. The proposed algorithm which achieves equalization in each frequency bin is derived from the proposed

generalized framework for MCEQ. It is shown that the proposed algorithm significantly reduces the complexity involved in MCEQ and exhibits higher robustness to channel estimation errors. To further reduce the processing time of the proposed frequency domain MCEQ algorithm, adaptive filtering techniques are introduced. To achieve convergence in a single step, an optimal step size is derived for the proposed adaptive algorithm.

Finally, a frequency-domain adaptive BCI algorithm is proposed for the estimation of unknown channels. The proposed algorithm exploits the spatial diversity of a multichannel system and estimates the AIRs based on the cross-relation among the channels. To gain more insights into its performance, the misconvergence problem is analyzed and based on this analysis, a penalty term derived from a sparseness constraint is introduced to the cost function for noise robustness.

Contents

List of Figures	x
List of Tables	xiv
List of symbols	xv
Abbreviations	xix
1 Introduction	1
1.1 Motivation	1
1.2 Scope and outline of the thesis	3
1.3 Main contributions	4
2 Background of Speech Dereverberation	7
2.1 Brief introduction to reverberation	7
2.2 Review of existing dereverberation methods	10
2.2.1 Speech enhancement via spatial processing	10
2.2.2 Source model-based speech dereverberation	11
Speech enhancement using the dual excitation model	12
LP residual enhancement	12
Harmonic filtering	16
Spectral enhancement	17
Temporal envelope filtering	18
2.2.3 Dereverberation via homomorphic transformation	20
2.2.4 Dereverberation via blind channel identification and inverse filtering	21
2.3 Inverse filtering techniques	25

2.4	Speech dereverberation quality measures	29
2.4.1	Subjective measures	29
2.4.2	Objective measures	30
	Intrusive quality measures	30
	Intrusive perceptually-based measures	31
2.5	Conclusion	32
3	A Generalized Framework for Speech Dereverberation with Insights into the Multichannel Equalization Problem	33
3.1	Introduction	33
3.2	Review of multichannel equalization	34
3.3	The proposed generalized MCEQ framework	36
3.4	Proposed framework-based algorithms	41
3.4.1	MCEQ-T	41
3.4.2	Insights into the MCEQ-T algorithm	42
3.4.3	Equalization using optimal filter (MCEQ-T _{opt})	47
3.4.4	MCEQ-KL	50
3.5	Simulation results	54
3.5.1	Effect of L_g	55
3.5.2	Effect of L_h	55
3.5.3	Robustness to AIR estimation errors	56
3.5.4	Dereverberation	57
3.5.5	Comparison of time-domain algorithms with RMCLS	57
3.5.6	Equalization in noisy environment using MCEQ-KL with denoising filter	60
3.6	Conclusion	61

4	Subspace-based Adaptive Multichannel Equalization	63
4.1	Introduction	63
4.2	Review of A-MINT	64
4.3	The proposed TDA-MCEQ algorithm	65
4.3.1	TDA-MCEQ-SP using the sparseness constraint	67
4.4	Simulation results	68
4.4.1	Convergence performance	68
4.4.2	Equalization performance	71
4.5	Conclusion	73
5	Variable Step-size Acoustic Multichannel Equalization Algorithms Exploiting Sparseness Measure	77
5.1	Introduction	77
5.2	The proposed adaptive MCEQ (A-MCEQ) algorithms	78
5.2.1	Convergence analysis	80
5.2.2	Adaptation of step-size based on the sparseness measure	82
5.3	Simulation results	84
5.3.1	A-MCEQ and A-MCEQ-N	84
5.4	Conclusion	88
6	Multichannel Equalization in the Frequency Domain	91
6.1	Introduction	91
6.2	The proposed MCEQ-F algorithm	92

6.3	The proposed partitioned PMCEQ-F algorithm	98
6.4	Simulation results	102
6.4.1	Performance comparison	102
6.4.2	Effect of L	103
6.4.3	Robustness to AIR estimation errors	103
6.4.4	PMCEQ-F	105
6.4.5	Dereverberation using BCI and MCEQ	106
6.5	Conclusion	107
7	Adaptive Multichannel Equalization in the Frequency Domain	111
7.1	Introduction	111
7.2	The FDA-MCEQ and FDA-MCEQ_{fast} algorithms	112
7.2.1	FDA-MCEQ _{fast} and Wiener solution	116
7.3	Simulation results	118
7.4	Conclusion	121
8	Blind Channel Identification for Speech Dereverberation	123
8.1	Introduction	123
8.2	Problem formulation	124
8.2.1	Review of NMCFLMS	125
8.3	The proposed partitioned MCFLMS (PMCFLMS) algorithm	127
8.3.1	The PMCFLMS algorithm	130
8.3.2	Convergence analysis of PMCFLMS	133
8.4	Misconvergence of LMS-based BCI algorithms in the presence of noise	136

8.5	Noise robust PMCFLMS-SC algorithm	144
8.6	Simulation results	147
8.6.1	Convergence performance	147
	PMCFLMS	147
	PMCFLMS-SC	149
8.6.2	Estimation of time-varying system	150
8.6.3	Speech dereverberation	152
8.7	Conclusion	154
9	Conclusion and Recommendations	157
9.1	Conclusion	157
9.2	Recommendations for further research	159
A	Approximate diagonalization of filtering matrix by the Fourier matrix	161
A.1	diagonalization of filtering matrix	161
	References	166

List of Figures

2.1	Sound propagation in a reverberant room.	8
2.2	Spectrogram and waveform of (a) clean speech signal, and (b) reverberant speech signal. . .	9
2.3	A typical AIR.	10
2.4	General block diagram of LP enhancement method.	13
2.5	Block diagram of spectral subtraction technique.	18
2.6	Concept of temporal envelope filtering.	19
2.7	Blind channel identification and equalization system.	22
3.1	A single-input multiple-output (SIMO) model.	34
3.2	The generalized MCEQ framework.	40
3.3	Equalization using MCEQ-T.	42
3.4	(a) Variation of $e(\tau)$ with τ , and (b) variation of $\rho(l)$ with l , for different AIR sets.	44
3.5	Filter-norm ρ for MINT and MCEQ-T.	46
3.6	The frequency-domain magnitude response of equalized response $\hat{\mathbf{d}}$. (a) MINT. (b) SB-MCEQ. (c) MCEQ-T. (d) MCEQ-T _{opt} . (e) MCEQ-KL.	50
3.7	Equalization using MCEQ-KL.	53
3.8	SRR _{seg} performance of the proposed algorithms for different L_g , as compared to MINT. . .	55
3.9	SRR _{seg} of the proposed algorithms for different L_h , as compared to MINT and SB-MCEQ. .	56
3.10	Variation of SRR _{seg} with $\mathcal{M}_m$ for MINT, SB-MCEQ, MCEQ-KL, MCEQ-T, and MCEQ-T _{opt}	57
3.11	Speech dereverberation performance comparison: Waveform and spectrogram of (a) original source signal, (b) reverberant signal at channel 1, (c) signal recovered using MINT, (d) signal recovered using MCEQ-T, (e) signal recovered using MCEQ-T _{opt} and (f) signal recovered using MCEQ-KL.	58
3.12	Equalization performance measured using EDCs for MINT, RMCLS, MCEQ-T and MCEQ-T _{opt}	59
3.13	Comparison of SRR _{seg} against SNR for MINT, MCEQ-KL only, and MCEQ-KL with the Wiener filter.	60

4.1	Convergence performance for A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP with different step-size μ	70
4.2	Convergence performance of TDA-MCEQ-SP for different β	70
4.3	Convergence performance of A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP for recorded AIRs.	71
4.4	Sparseness performance comparisons for A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP with synthetic AIRs.	72
4.5	Sparseness performance of A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP with recorded AIRs.	72
4.6	Equalized response $\hat{\mathbf{d}}$ achieved using (a) A-MINT, (b) TDA-MCEQ and (c) TDA-MCEQ-SP with $\mu = 0.5$	75
4.7	Speech dereverberation performance comparison: Waveform and spectrogram of (a) Original source signal, (b) Reverberant signal at channel 1, (c) signal recovered using A-MINT, (d) signal recovered using TDA-MCEQ, (e) signal recovered using TDA-MCEQ-SP	76
5.1	The angle between $\mathbf{a}$ and $\mathbf{b}$	78
5.2	Target response $\mathbf{d}$ and estimated response $\hat{\mathbf{d}}(n)$ achieved using A-MCEQ without normalization for $\mu = 0.2$	81
5.3	Convergence performance comparison of A-MCEQ, A-MCEQ-N and A-MINT for various μ	85
5.4	Effect of μ on convergence performance of A-MCEQ and A-MCEQ-N.	85
5.5	Convergence performance of A-MINT with variable step-size.	86
5.6	Convergence performance of A-MCEQ with variable step-size.	86
5.7	Equalization performance with system mismatch $\mathcal{M}_m$	87
5.8	Equalized response $\hat{\mathbf{d}}$ obtained using (a) A-MINT and (b) A-MCEQ with variable step-size.	88
6.1	$\underline{\mathbf{H}}_m$ in the frequency domain for $L_h = 1600$ and $L_x = 2L_h$	94
6.2	Diagonality measure ξ_D of $\underline{\mathbf{H}}_m$ as a function of L	94
6.3	Equalization using MCEQ-F.	96
6.4	Computational complexity comparison in flops.	97

6.5	The frequency-domain magnitude response of equalized response $\hat{\mathbf{d}}$. (a) MINT. (b)SB-MCEQ. (c) MCEQ-T. (d) MCEQ-T _{opt} . (e) MCEQ-F	99
6.6	Equalization using PMCEQ-F.	101
6.7	Average processing time comparison. (a) SB-MCEQ. (b) MCEQ-F, configured to achieve the same SRR_{seg} performance as SB-MCEQ. (c) MCEQ-F with $L_x = 2L_h$ and $N = L_h$	102
6.8	SRR_{seg} for speech signal equalized with MCEQ-F, and SB-MCEQ.	102
6.9	Effect of L on SRR_{seg} of MCEQ-F.	104
6.10	Robustness to AIR mismatch: SRR_{seg} for speech equalized with (a) MCEQ-F with $L_x = 2L_h$ and $N = L_h$, (b) MCEQ-F with same SRR_{seg} as SB-MCEQ when $\hat{\mathbf{h}}_m = \mathbf{h}_m$, (c) MCEQ-T _{opt} , (d) SB-MCEQ, (e) MINT, and (f) original reverberant speech signal averaged across all channels.	105
6.11	Robustness to AIR order estimation inaccuracy: SRR_{seg} for speech equalized with (a) MCEQ-F with $L_x = 2L_h$ and $N = L_h$, (b) MCEQ-F with same SRR_{seg} as SB-MCEQ when $\hat{\mathbf{h}}_m = \mathbf{h}_m$, (c) MCEQ-T _{opt} , (d) SB-MCEQ, (e) MINT, and (f) original reverberant speech signal averaged across all channels.	105
6.12	Speech dereverberation performance comparison: Waveform and spectrogram of (a) Original source signal, (b) Reverberant signal at channel 1, (c) signal recovered using MINT, (d) signal recovered using SB-MCEQ, (e) signal recovered using MCEQ-T _{opt} and (f) signal recovered using MCEQ-F.	108
7.1	Equalization using FDA-MCEQ.	115
7.2	Convergence performance of FDA-MCEQ.	117
7.3	Convergence performance of FDA-MCEQ _{fast}	118
7.4	Average SRR_{seg} for SB-MCEQ, FDA-MCEQ, MCEQ-F and FDA-MCEQ _{fast}	119
7.5	Average processing time of (a) FDA-MCEQ, and MCEQ-F, (b) FDA-MCEQ _{fast} , and FDA-MCEQ, (c) FDA-MCEQ _{fast} and SB-MCEQ.	120
8.1	Convergence performance of NMCFLMS for $\text{SNR} = 50$ and 20 dB. At $\text{SNR} = 20$ dB the algorithm misconverges.	137

8.2	Convergence performance of PMCFLMS for SNR 40 and 5 dB.	143
8.3	$\underline{\underline{J}}^y(m)$ and $\underline{\underline{J}}^v(m)$ for (a) SNR = 40 and (b) SNR = 5 dB.	144
8.4	Plot of (a) $ \underline{h}_{1,l}^{100}(n, 1) $, and (b) $ \hat{\underline{h}}_{1,l}^{100}(n, 1) $	145
8.5	Sparseness measure comparison for SNR = 40, 10 and 5 dB.	145
8.6	Convergence performance of PMCFLMS as a function of β at SNR = 40 dB.	147
8.7	Convergence performance of PMCFLMS as a function of P at SNR = 40 dB.	148
8.8	Convergence performance of PMCFLMS using a single block of data.	148
8.9	Convergence performance of PMCFLMS and NMCFLMS at different SNRs.	149
8.10	Convergence performance of PMCFLMS-SC at different SNRs.	150
8.11	Convergence performance of RVSS-MCFLMS at different SNRs.	151
8.12	Sparseness measure comparison of PMCFLMS-SC for SNR = 40, 20, 10 and 0 dB.	151
8.13	The convergence performance of (a) PMCFLMS (SNR = 20 dB) and (b) PMCFLMS-SC (SNR = 10 dB) for time-varying AIRs.	152
8.14	(a) Actual AIRs. (b) AIRs estimated by PMCFLMS at SNR = 20 dB. (c) AIRs estimated by PMCFLMS-SC at SNR = 10 dB.	153
8.15	Equalized response. (a) PMCFLMS at SNR = 20 dB. (b) PMCFLMS-SC at SNR = 10 dB.	153

List of Tables

2.1	ITU-T recommended speech quality measurement scales [1].	29
3.1	PESQ scores.	59
4.1	The Proposed TDA-MCEQ Algorithm	66
6.1	Performance of PMCEQ-F with different L'_h	106
6.2	Dereverberation performance analysis.	107
8.1	Dereverberation performance analysis metrics.	154

List of symbols

$(.)^+$	Pseudoinverse operator
$*$	Linear convolution operator
C_T	Early-to-late reverberation ratio
$E(.)$	The expectation operator
F_0	Fundamental frequency of speech
$G(z)$	Inverse filter represented in the z-domain
$H(z)$	Acoustic impulse response represented in the z-domain
K	Total number of speech signal frames
L_B	Data block shift
L_c	Critical length for $\mathbf{g}_m$
L_d	Length of $\mathbf{d}$
L_g	Length of inverse filter
L_h	Length of acoustic impulse response
L_x	Length of speech data frame
M	Total number of channels
N	Frame shift
N_F	DFT length
N_c	Total number of critical bands in the Bark spectrum
P	Total number of acoustic impulse response frames
$\mathbf{\Lambda}$	Diagonal matrix of eigen values
$\mathcal{S}$	Step-size matrix
∇J	Gradient of the cost function J
χ	Normalized projection misalignment
ℓ	Microphone spacing

η	Normalized misalignment
$\mathbf{F}$	Inverse Fourier matrix
$\mathbf{G}$	Inverse filters in the time domain
$\mathbf{H}$	Concatenated filtering matrix in the time domain
$\mathbf{H}_m$	Filtering matrix constructed from $\mathbf{h}_m$
$\mathbf{I}$	Identity matrix
$\mathbf{Q}$	Matrix of eigenvectors $\mathbf{q}_l$
$\mathbf{R}$	Correlation matrix
$\mathbf{U}$	Unitary transform matrix
$\mathbf{X}(k)$	Data matrix
$\mathbf{d}$	Desired equalized response
$\mathbf{g}$	Concatenated inverse filter
$\mathbf{g}_m$	Inverse filter vector for the m th channel
$\mathbf{h}_m$	Acoustic impulse response vector corresponding to the m th channel
$\mathbf{h}_m(p)$	p -th frame of $\mathbf{h}_m$
$\mathbf{s}(k)$	k th frame of clean speech signal
$\mathbf{s}(n)$	Clean speech vector
$\mathbf{u}_l$	Unit-norm basis vector corresponding to the l th subspace
$\mathbf{v}_m(n)$	Additive noise vector of the m th channel
$\mathbf{x}_m(n)$	m th channel received data vector
$\mathbf{x}_m(k)$	k th frame of $\mathbf{x}_m(n)$
$\mathcal{M}_m$	System mismatch
sgn	Element-wise sign operator
μ	Step-size
ϕ	Spectral flatness measure
τ	Modeling delay

τ_{opt}	Optimal modeling delay
$\underline{J}$	Frequency-domain cost function
$\underline{x}(n)$	DFT of $x(n)$
$\hat{\mathbf{G}}$	Estimate of $\mathbf{G}$
$\hat{\mathbf{H}}$	Estimate of $\mathbf{H}$
$\hat{\mathbf{H}}_m$	Estimate of $\mathbf{H}_m$
$\hat{\mathbf{d}}$	Estimate of $\mathbf{d}$
$\hat{\mathbf{g}}$	Estimate of $\mathbf{g}$
$\hat{\mathbf{g}}_m$	Estimate of $\mathbf{g}_m$
$\hat{\mathbf{h}}_m$	Estimate of $\mathbf{h}_m$
$\hat{s}(n)$	Estimate of source signal $s(n)$
$\hat{s}_d(n)$	estimated direct-path signal
ξ	Sparsity measure
c	Velocity of sound
f_s	Sampling frequency
$h(n)$	Acoustic impulse response
$h(n_d)$	Direct-path response
n_c	Early reflection cutoff delay of acoustic impulse response in samples
n_d	Direct-path delay of acoustic impulse response in samples
r	The source-receiver distance
$r_x(\tau)$	Autocorrelation of $x(n)$
$s_d(n)$	Direct-path signal
$v_m(n)$	Additive noise at the m th channel
$x(n)$	Receiver output
$x_m(n)$	m th receiver output
J	Time-domain cost function

T_{60}	Reverberation time
τ_m	Propagation delay associated with the m th channel

Abbreviations

A-MINT	Adaptive multiple-input/output inverse theorem
AIR	Acoustic impulse response
AR	Autoregressive
ASR	Automatic speech recognition
BCI	Blind channel identification
BSD	Bark spectral distortion
CS	Channel shortening
DAM	Diagnostic acceptability measure
DE	Dual excitation
DSB	Delay-and-sum beamforming
EDC	Energy decay curve
ELR	Early-to-late reverberation ratio
flop	Floating-point operation
FSB	Filter-and-sum beamforming
GCC	Generalized cross-correlation
GCC-PHAT	Generalized cross-correlation with phase trans- form
GDE	Generalized dual excitation
GSC	Generalized sidelobe canceller

HERB	Harmonicity-based dereverberation
HOS	Higher-order statistics
IDFT	Inverse discrete Fourier transform
ITU	International telecommunication union
KLT	Karhunen-Loève transform
LCMV	Linear constrained minimum variance
LIME	Linear predictive multi-input equalization
LMS	Least-mean-square
LPC	Linear prediction coefficients
M-SBEE	Multi-microphone sub-band envelop estimation
MAP	Maximum a posteriori
MCEQ	Multichannel equalization
MCEQ-F	Frequency-domain MCEQ
MCEQ-KL	KL-domain MCEQ
MCEQ-T	Time-domain MCEQ
MCFLMS	Multichannel frequency-domain least-mean-square
MCLMS	Multichannel least-mean-square
MCLMS	Multichannel LMS
MCN	Multichannel Newton
MFA	Matched filter array
MIMO	Multiple input multiple output
MINT	Multiple-input/output inverse theorem
ML	Maximum Likelihood

MOS	Mean opinion score
MTF	Modulation transfer function
MVDR	Minimum variance distortionless response
NMCFLMS	Normalized multichannel frequency-domain least-mean-square
NPM	Normalized projection misalignment
P-MINT	Partial multiple-input/output inverse theorem
PESQ	Perceptual evaluation of speech quality
PMCFLMS	Partitioned multichannel frequency-domain LMS
PMCFLMS	Partitioned multichannel frequency-domain LMS with sparseness constraint
RMCLS	Relaxed multichannel least-squares
SB-MCEQ	Subband MCEQ
SFM	Spectral flatness measure
SOS	Second-order statistics
SRR_{seg}	Segmental signal-to-reverberation ratio
STPT	Short-term power spectrum
TDA-MCEQ	Adaptive time-domain MCEQ
TSML	Two-step maximum likelihood

Chapter 1

Introduction

1.1 Motivation

Within an enclosed space such as an office room, a speech signal received by a distant microphone is often degraded by the acoustic channel [2]. The received signals are typically degraded by (a) reverberation which results from the multi-path propagation of sound signal from the source to the microphone, and (b) additive noise and (possibly) other interferences. This thesis involves the development of algorithms to achieve speech dereverberation.

In an anechoic room, an acoustic wave reaches the microphone via a direct path. On the contrary, in a typical room with reflective surfaces, e.g., walls, floor and furniture, some of the wave energy is absorbed (depending on the absorption coefficient of the surfaces) and the remaining energy gets reflected. In such an environment, the received signal comprises of a sum of reflected waves from multiple propagation paths and the direct-path wave. Differences in phase and amplitude for acoustic waves propagating via different paths cause time-spreading effect. This time-spreading effect results in temporal and spectral smearing which, in turn, masks lower energy sounds and distorts both the envelope and fine structure of a speech signal. As a result, the speech signal becomes less intelligible and distorted not only for human ears but also for hearing-aids and automatic speech recognition (ASR) systems [3].

In order to enhance the quality of the received speech signal, degradation due to the channel effects need to be mitigated by applying dereverberation algorithms. Many dereverberation algorithms have been proposed so far. A comprehensive review can be found in [4]. These algorithms can be broadly dichotomized into single- or multichannel techniques. One of the well-known techniques to achieve speech dereverberation is to estimate the unknown acoustic impulse responses (AIRs) blindly and filter the received signals using inverse (equalization) filters computed from the estimated AIRs. This tech-

nique can, in theory, achieve perfect dereverberation if the equalization filters can be perfectly estimated. Although this appears to be a simple problem, it is not for many reasons. Firstly, AIRs are generally unknown and perfect estimation of AIRs is still a challenging problem. Secondly, even if AIRs are perfectly estimated, one cannot achieve stable inverse filters by simply inverting them. This is because, an AIR is, in general, non-minimum phase [4] and its direct inversion results in an unstable filter with poles outside the unit circle [5]. Both single- and multichannel algorithms have been proposed to estimate stable inverse filters from non-minimum phase AIRs. Nevertheless, multichannel techniques are gaining popularity since they can, in theory, achieve perfect equalization.

Most of the existing multichannel equalization (MCEQ) algorithms operate in the time domain and are computationally expensive. In this regard, achieving equalization in different transform domains has not yet been examined and this motivated the formulation of a generalized framework for MCEQ in this thesis. Using this framework, equalization algorithms can be derived in different transform domains and one can gain more insights into the equalization problem. Existing MCEQ algorithms are sensitive to the modeling delay associated with the desired equalized response [6]. Based on the proposed framework, time-domain algorithms which are robust to channel estimation error have been proposed. In addition, equalization in a noisy environment is a challenging problem. It has been shown that better denoising can be achieved in the Karhunen-Loève transform (KLT) domain [7]. This led to the development of a KLT-based MCEQ algorithm. This is achieved by applying the proposed MCEQ framework in the KLT domain and it will be shown that the proposed algorithm can achieve joint denoising and equalization.

The computational complexity of existing non-adaptive time-domain MCEQ algorithms increases with the length of the AIRs. Adaptive algorithms has the potential of mitigating this high computational complexity and this motivated the application of adaptive techniques to the equalization problem. The proposed adaptive MCEQ algorithms exploit sparsity of target response to achieve better equalization with fast convergence.

Although time-domain adaptive algorithms may reduce computational complexity they may suffer from slow convergence. Motivated by this limitation, a frequency-domain MCEQ algorithm is proposed. It may be noted that insights obtained from the proposed generalized framework applied in the frequency domain reduces the computational complexity significantly. A modified version of the frequency-domain

MCEQ algorithm has also been proposed to reduce the complexity further. Adaptive filtering techniques are then combined with the proposed frequency-domain algorithm with the aim of achieving further reduction in processing time. The proposed frequency-domain adaptive algorithms achieve fast convergence.

While equalization of AIRs is important, these algorithms assume that the AIRs are well estimated. In many practical situations, most of the existing blind channel identification (BCI) algorithms are sensitive to noise and fail to properly estimate the AIRs [8], [9]. To achieve BCI in the presence of noise, a frequency-domain adaptive algorithm is also proposed in this thesis. This algorithm relies on the cross-relation between channels and to achieve robustness to noise, the proposed algorithm exploits AIR spectral dispersiveness and derives a sparsity based penalty function. The penalty function along with the cross-relation based cost function prevents the algorithm from misconvergence in the presence of noise.

1.2 Scope and outline of the thesis

This thesis focuses on two areas of blind speech dereverberation- blind multichannel estimation and equalization. A generalized framework for MCEQ is presented to achieve equalization in different transform domains. Based on this framework robust MCEQ algorithms are proposed. Adaptive MCEQ algorithms are introduced with the aim of achieving complexity reduction for the equalization of high-order AIRs. A fast equalization algorithm that achieves equalization in the frequency domain and its modified version are proposed for practical situations. An adaptive version of the proposed frequency-domain MCEQ algorithm, which can achieve fast equalization, has been derived.

To achieve good dereverberation reasonable estimates of AIRs is essential. This thesis also addresses the challenging problem of BCI in the presence of noise. A noise robust adaptive BCI algorithm is proposed which along with the proposed MCEQ algorithms makes speech dereverberation possible in a noisy environment.

This thesis is organized as follows: in Chapter 2 the reverberation problem is first described and some of the well-known techniques to achieve speech dereverberation are reviewed next. Following this, the multichannel equalization problem will be discussed. The chapter also presents a brief discussion on

some of the objective and subjective speech quality measures that are used in this thesis to assess the performance of the algorithms. In Chapter 3, a framework, which can be applied in different transform domains, is proposed to achieve MCEQ. Two illustrative examples, one in the time domain and the other in the KLT domain, are provided to demonstrate the application of the proposed framework. Chapter 4 introduces a subspace-based adaptive MCEQ technique while Chapter 5 introduces an adaptive MCEQ algorithm which exploits the sparsity of the target response. Chapter 6 presents a fast MCEQ algorithm that achieves equalization in the frequency domain. A modified version of the proposed algorithm is also presented to gain further insights into frequency-domain equalization. Chapter 7 presents an adaptive version of the algorithm presented in Chapter 6 which achieves further significant reduction in the complexity. Chapter 8 presents a noise robust algorithm for BCI. A theoretical analysis on the convergence performance of the proposed algorithm is presented where the effect of noise on the convergence performance of an adaptive BCI algorithm is shown. Conclusions and recommendations for future research are presented in Chapter 9.

1.3 Main contributions

- 1) A generalized framework that facilitates subband-based multichannel equalization in different transform domain is proposed. One can gain further insights into the MCEQ problem by applying it in different transform domains. A time-domain MCEQ algorithm is then derived using the proposed framework that employs multiple sets of equalization filters (with each set corresponding to a subspace) to achieve better equalization irrespective of modeling delay. A method to choose an optimal set of equalization filters corresponding the adequate modeling delay is also proposed. The above contributions have resulted in [10], [11].
- 2) A set of adaptive MCEQ algorithms is proposed to achieve equalization with reduced complexity. The sparsity of target response has been exploited for the proposed MCEQ algorithm. These contributions have resulted in [12]–[14].
- 3) A frequency-domain MCEQ algorithm that achieves significant reduction in computational complexity is proposed. The proposed algorithm achieves fast equalization of higher-order AIRs with

good equalization quality. To reduce the complexity further, a partitioned convolution technique is applied to the proposed algorithm. The modified algorithm achieves significant reduction in the computational complexity without noticeable performance degradation. An adaptive version of the proposed frequency-domain MCEQ algorithm is derived to achieve further reduction in the complexity. An optimal step-size is derived for the adaptive algorithm such that it converges in a single step. These contributions have resulted in [11, 15, 16].

- 4) A new BCI algorithm is proposed to estimate AIRs in a noisy environment. The reverberation process is modeled using the partitioned convolution thereby establishing a convolutive relation for each frequency bin. A cost function is then derived for each frequency bin from the cross-relation between the channels. An analysis on the misconvergence of least-mean-square (LMS) based BCI algorithms in noisy environments is also provided. Based on the analysis, the spectral dispersiveness of the AIRs is used as a constraint to prevent misconvergence of the proposed algorithm in the presence of noise.

Chapter 2

Background of Speech Dereverberation

2.1 Brief introduction to reverberation

A speech signal produced inside an echoic room not only propagates via a direct path from the source to the microphone, but also gets reflected by surfaces of that room. This effect of multipath propagation is often referred to as reverberation as illustrated in Fig. 2.1, where the solid lines denote the direct paths while the dashed lines denote the reflected paths. Since reflected waves with different amplitudes arrive the microphones at different times the received reverberant signal suffers from smearing both in the temporal and spectral domains. To illustrate this the waveform and spectrogram of a clean speech signal and its reverberant version are shown in Fig. 2.2. It can be seen from the spectrogram of the clean speech signal shown in Fig. 2.2 (a) that the formant frequencies which correspond to the resonance of the vocal tract [17] clearly appear as separated dark bands. On the other hand the spectrogram of the reverberant signal shown in Fig. 2.2 (b) exhibits temporal smearing; the reverberant energy fills the gaps between adjacent phonemes and words. This results in the degradation of speech intelligibility.

The effect of reverberation can be modeled as the convolution between the AIR $h_m(n)$ and the original signal $s(n)$, expressed as

$$x_m(n) = s(n) * h_m(n), \quad (2.1)$$

where $x_m(n)$ is the signal received by the m th microphone and $*$ denotes linear convolution. In this model, the phenomenon of reverberation is entirely characterized by the AIR. A typical AIR measured in a room can be divided into the direct response, initial or early reflections, and late reverberation as illustrated in Fig 2.3. The segment before the direct response is contributed by the propagation delay between the speech source and the microphone. The direct response corresponds to the direct path signal and its amplitude (relative to that of the reflections) depends on factors such as the separation between

Fig. 2.1: Sound propagation in a reverberant room.

the source and the microphone, and the absorption coefficient of the reflective surfaces in the room. The reflected waves arrive at the microphone after a delay depending on the source-microphone configuration, room dimension and positions of reflecting objects. The reflected sounds received within approximately 50 – 80 ms with respect to the arrival time of the direct sound are referred to as the early reflections while the late reverberation results from reflections which arrive with longer delays. The amplitude of the late reflections is, in general, smaller compared to the early reflections as they suffer more absorption due to multiple reflections.

The structure of the AIR affects the perceptual quality of the speech signal. The amount of energy and the delay of each reflection cause different psychoacoustic effects. Early reflections are acoustically integrated by the ears, reinforcing the direct sound. Hence they are not perceived as echoes due to the psychoacoustic-masking property of ears. This effect is known as precedence effect [18]. While the early reflections improve speech intelligibility [19]–[22], it may result in coloration of the speech spectrum which degrades speech quality [2, 20]. Late reverberation, on the other hand, causes temporal smearing of the speech energy which will lead to the masking of low energy sounds [23]. This effect is known as overlap masking [24, 25] and introduces an ‘echoing’ effect. As can be seen from Fig. 2.3, the reverberation energy decays exponentially and is quantified by the reverberation time T_{60} [26], defined as the time required by the sound energy to decay by 60 dB after the source is turned off. Another objective metric to quantify the amount of reverberation is the early-to-late energy ratio (ELR) [20, 27], defined

Fig. 2.2: Spectrogram and waveform of (a) clean speech signal, and (b) reverberant speech signal.

(in dB) as

$$C_T = 10 \log_{10} \left[\frac{\sum_{n=0}^{n_d+n_c-1} h^2(n)}{\sum_{n=n_d+n_c}^{\infty} h^2(n)} \right], \quad (2.2)$$

where n_d is direct-path sample delay, $n_c = T f_s$ is the cutoff time delay for the early reflections while f_s denotes the sampling frequency. For speech, the choice of T is usually 50 ms and hence the ELR is

Fig. 2.3: A typical AIR.

denoted by C_{50} .

2.2 Review of existing dereverberation methods

Reverberation has an adverse effect on applications such as hands-free telephony, voice conferencing, automatic speech recognition system and hearing aids. Therefore, dereverberation algorithms have been proposed to mitigate the effects of reverberation. However, speech dereverberation is a challenging problem since both the source signal and AIRs are not known a priori. Some of these algorithms perform reverberation suppression without the knowledge of AIRs while others achieve reverberation cancellation via AIR estimation and inverse filtering. One way of classifying these algorithms is based on number of microphones used, i.e., single- or multichannel methods. Based on the underlying principle one may also classify speech dereverberation methods as speech enhancement via spatial processing, source model-based speech dereverberation, speech dereverberation via homomorphic transformation, or speech dereverberation by channel estimation and equalization. Some of these methods are briefly reviewed in the following sections.

2.2.1 Speech enhancement via spatial processing

Spatial processing algorithms process signals captured by multiple microphones so as to enhance or suppress signals arriving from certain directions. While some techniques assume the source direction, others estimate it. The basic idea behind speech enhancement via spatial processing is to enhance signals only from the source direction and suppress the reflected signals emanating from other directions.

One of the earliest speech enhancement techniques based on spatial processing is beamforming [28, 29] which enhances sound from a particular direction. For a far-field system an acoustic plane wave reaches different microphones at different times depending on the array geometry. The simplest technique to achieve an unidirectional response is the delay-and-sum beamforming (DSB) [30, 31], where delays are selected such that the source signals arriving from a particular direction are coherently added at the microphone constructively giving the maximum array response for a chosen direction. This in turn enhances the direct path component of the speech while suppressing the incoherent reverberant part. Another variant of DSB is the filter-and-sum beamforming (FSB) [30]. In FSB, the delays in DSB are replaced by finite impulse response (FIR) filters. Frequency-domain beamforming algorithms have also been proposed [32, 33, 34], where each monochromatic component is treated independently and a beam is formed at each frequency by applying a phase shift corresponding to the desired delay. Nevertheless, simple beamforming techniques are not highly efficient in reverberation suppression because of the fact that the reverberant waves usually enter into the beam from all directions because of their diffused nature.

Adaptive beamforming algorithms [35]–[38] are more popular for achieving beamforming in non-stationary environments. Most of these algorithms adapt the weight parameters by solving a constrained mean-squared optimization problem and assume statistical independence between the interferences and the source signal. Such an assumption of statistical independence is not fully applicable to reverberation since the reverberant speech results from the superposition of delayed and attenuated versions of the source signal. Although algorithms such as proposed in [39]–[44] relax this constraint to some extent they have not shown to achieve significant reverberation suppression. A theoretical analysis on the trade-off between speech denoising and dereverberation achievable using the well-known minimum variance distortionless response beamformer (MVDR) [45] can be found in [46, 47]. In order to achieve some performance improvement post-processing techniques [48, 49] are also employed at the beamformer’s output.

2.2.2 Source model-based speech dereverberation

Source model-based speech dereverberation methods rely on the characteristics and underlying structure of the clean speech signal to achieve reverberation suppression. The speech signal is enhanced by

exploiting the knowledge of distortion introduced by reverberation into the original structure of the clean speech signal. These algorithms may exploit the speech production model [50], statistical modeling [51], or temporal/spectral characteristics.

Speech enhancement using the dual excitation model

In the dual excitation (DE) speech model [52], the speech signal is separated into voice and unvoiced components by utilizing the harmonic nature of voice part. The voice and unvoiced components are then processed separately to suppress noise. Following [52], and taking pitch variation into account, a generalized dual excitation (GDE) speech model [53] has been proposed to achieve better performance. This method employed an analysis window with an adaptively varied length that is dependent on the speech characteristics. In [54, 55], Brandstein fused spatial filtering with the DE model to enhance the speech quality in a reverberant environment.

LP residual enhancement

The speech production model [50, 56] models the human vocal tract as a time-varying all-pole filter and hence the speech production process is modeled as an autoregressive (AR) process. A speech signal is produced when this all-pole filter is excited with random noise (for unvoiced speech) and quasi-periodic pulses (for voiced speech). This can be mathematically represented as

$$s(n) = \sum_{o=1}^O a_o s(n-o) + b(n), \quad (2.3)$$

where O is the all-pole filter order, a_o is the filter coefficient and $b(n)$ is the excitation signal. The poles of the all-pole filter, which are commonly known as linear prediction coefficients (LPC) or AR coefficients, can be estimated using linear prediction analysis of the speech signal. The LP residual enhancement methods are formulated on the assumption that, since an acoustic channel is usually modeled as an FIR filter with a transfer function that consists of only zeros, room reverberation would introduce only zeros into the received signals and hence has little effect on the all-pole filter [2, 57]. This implies that additional peaks are introduced in the excitation sequence by reverberation and that a smeared structure is

exhibited. To achieve dereverberation, the LP coefficients of a clean speech signal are first estimated from the received signals [58] and the excitation sequence is then obtained by inverse filtering. Since the all-pole filter is assumed to be unaffected by reverberation the aim of LP residual enhancement technique is to suppress those additional peaks in the excitation sequence and resynthesize the enhanced speech using the AR coefficients obtained from the reverberant speech and the modified LP residual. Figure 2.4

Fig. 2.4: General block diagram of LP enhancement method.

depicts a general structure of the LP residual enhancement method. As will be described below, a number of LP residual enhancement techniques have been proposed.

In one of the pioneering works [59], the author proposed to synthesize the clean speech signal by estimating the vocal tract transfer function, period of the voice speech, and the gain parameter from the reverberant speech signal. This essentially required a voiced/unvoiced speech detector, a pitch estimator to determine the voiced speech period from the LP residual and a gain estimator. In [60, 61], a multichannel algorithm based on the LP residual enhancement using wavelets has been proposed. This algorithm assumes that the local extrema in the wavelet transform of the reverberant speech correspond to the impulses of the LP residual. The desired extrema corresponding to the clean LP residual impulses are then estimated by clustering the wavelet extrema across multiple channels from which the enhanced speech is synthesized. The proposed algorithm is capable of discriminating impulses of the LP residual generated by the clean speech signal from those introduced by reverberation and uncorrelated noise. It has been reported in [61] that this method is also capable of reconstructing a noise-like residual signal. In [62] the same authors further proposed the computation of a coarse estimate of each channel response from the received data through short-term analysis and modifying the LP residuals of the received signals in order to reduce effects of reverberation. The coarse estimates are obtained by applying the phase transform version of generalized cross-correlation (GCC-PHAT) [63]. A matched filter-based function

derived from these coarse estimates are then applied to each channel residual to derive an estimate of the clean speech residual. The synthesized enhanced speech signals are then applied to a beamformer to obtain the dereverberated speech.

A single microphone dereverberation algorithm has been proposed by Yegnanarayana et al. [64, 65] where short segments of reverberant data were processed in order to enhance speech segments corresponding to low signal-to-reverberation ratio (SRR) components. The speech enhancement process involves identifying and manipulating the linear prediction residual signal in three different regions of the speech signal, namely, high SRR region, low SRR region, and only reverberation component region. These regions are identified by computing the entropy using the probability density function (pdf) of 20 ms segments of the LP residual signal. The entropy function is next smoothed and a gross weighting function is derived using a non-linear mapping function. Noting that the entropy is high in the low SRR regions, regions corresponding to the low weighting function constitute the reverberant component regions and this function is then applied to modify the LP residual signal for enhanced speech synthesis. This method is reported to be robust to noise present in the received speech signal.

Features of the source excitation in speech production have been exploited for the enhancement of multichannel received signals degraded by noise and reverberation [66, 67]. The underlying principle is that by locating the glottal closure instant in each cycle, it is possible to enhance speech in that region. Due to high strength of excitation, the signal-to-noise ratio (SNR) is high in the neighborhood of these instants compared to other regions. The Hilbert envelope of the LP residual is used to determine the strength of excitation and a weighting function is derived by coherently adding the delay compensated Hilbert envelopes of the LP residual signals from multiple microphones. The enhanced speech signal is generated by exciting the time-varying all-pole filter with the weighted LP residual. Although the proposed method improves the quality of the speech signal, significant distortion is still present due to noise and poor representation of the all-pole filter.

While the above methods achieve reduction in speech degradation due to early reflections, they fail to suppress late reverberation due to the tail of the AIR. This results in a reduction of speech intelligibility. Moreover, the assumption that the LP coefficients are unaffected by reverberation is valid only in a spatially averaged sense and cannot be guaranteed at a single point in space for a given room [2, 68].

Since these methods do not consider the actual shape of the signal, they introduce speech artifacts which cause the dereverberated signals to sound unnatural. Therefore, these LP residual enhancement based speech dereverberation methods can only achieve moderate improvement in relatively low-reverberant environments.

It is well-known that, based on central limit theorem, the distribution of the LP residual signal of a reverberant speech becomes more Gaussian compared to the LP residual of the clean speech [65]. Kurtosis maximization of the LP residual using a subband adaptive filtering structure has been proposed in [69] by Gillespie et al. to achieve reverberation suppression. In this approach, the filter is directly applied on the received signals without performing any LPC analysis. Although the proposed algorithm achieves spectral improvement and significant reduction in reverberation, a stable performance may not be guaranteed due to the inaccurate computation of kurtosis and its derivatives [70]. In order to minimize the sensitivity of these algorithms a maximum likelihood (ML) approach [71] and its extension into multichannel case [72] have been proposed by Tonelli et al. Analysis of the speech modeling and LPC-based speech enhancement have been presented in [4, 73]. With the aim of preserving the naturalness of LP residual enhanced speech, Gaubitch et al. proposed the spatiotemporal averaging method for enhancement of reverberant speech (SMERSH) based on the LP residual processing [74, 75]. In SMERSH the received signals are spatially averaged before temporal larynx cycle averaging of the LP residual corresponding to the voiced speech is performed. The enhanced speech is then synthesized using the processed residual.

Most of the conventional deconvolution techniques model the target signal as an independent identically distributed (i.i.d) process which is often untrue for speech-like signals. Hence application of such techniques may cause excessive whitening of the speech signal. Spatial, temporal and spectral diversities have been exploited in [76, 77] to pre-whiten the received signals. An equalizer is then estimated using the multichannel linear prediction (MCLP) technique. This technique aims to address the source whitening problem associated with conventional dereverberation techniques. Although this technique achieves better dereverberation by exploiting spatial diversity, its performance is dependent on the order of the whitening filter applied. Similar approach is used to define the linear-predictive multi-input equalization (LIME) algorithm in [78]–[81]. However, it has been described in [80] that overlapping zeros among all

channels are not compensated for by LIME and this results in speech distortion. Delcroix et al. further showed that if the number of microphones is increased, the diversity among the channels also increases and this, in turn, results in fewer overlapping zeros leading to reduced spectral distortion. Nakatani et al. proposed to apply MCLP and short-time Fourier transform (STFT) for speech dereverberation [82]–[84]. In [85] MCLP is employed to estimate the spectrum of late reverberation and speech enhancement is achieved using spectral subtraction. The LP residual processing has also been combined with spectral processing in [86].

Harmonic filtering

Harmonicity is considered as a robust feature of a speech signal even in the presence of background noise. The spectrum of a clean voiced speech is composed of a fundamental frequency and a series of harmonics. Reverberation disturbs this harmonic structure by adding non-harmonic components. Motivated by this, the harmonicity-based dereverberation (HERB) [87, 88] first determines the fundamental frequency and the harmonic components are extracted from the voiced speech segments of the reverberant speech spectrum to estimate the enhanced speech spectrum. A dereverberation filter is computed for each voiced speech segment by comparing the enhanced speech spectrum with that of the reverberant signal. A mean dereverberation filter is subsequently obtained by averaging the dereverberation filters of different voiced speech segments. Dereverberated speech signal is then achieved by applying the mean dereverberation filter to the received reverberant speech signal. The conventional HERB does not provide an analytical framework which can be useful in performance optimization. The authors reformulated HERB as a maximum a posteriori (MAP) problem in [89]. Given the observed speech data, the equalization filter is determined as one that maximizes the a posteriori probability. A closed-form solution for HERB is achieved by assuming that the a posteriori probability can be modeled by means of a Gaussian pdf under a harmonicity constraint. The proposed solution is identical to that of conventional HERB when a particular type of pdf is assumed, and it manifests the physical meaning of the weight with which conventional HERB calculates the average transfer function. It has been stated in [89] that the conventional HERB gives rise to undesirable stationary random noise during the unvoiced periods. Experimental results have shown that this stationary background noise can be reduced by adjusting the variance of the pdf

adequately.

Results presented in [90] showed that HERB, when employed as a preprocessor, can improve the accuracy of speech recognition systems. In [88], the dereverberation filter obtained with HERB has shown to reduce the reverberation in terms of energy decay curves. However, it may be noted that this computationally complex algorithm requires a large amount of data to estimate the dereverberation filter and it assumes that the system is stationary for the duration of the entire data used. To address this the authors proposed the fast HERB algorithm [91] which employs a noise reduction algorithm and estimates the inverse filter using a smaller amount of training data. Another limitation of this method is that it requires perfect estimation of the fundamental frequency. Moreover, it introduces artifacts in the enhanced signal due to the elimination of non-harmonic components present in the voiced speech, which contain both clean speech and reverberant speech. In addition, this method has been proposed for voiced segments and its effect on the unvoiced segments has not been discussed. Since this method ignores the effect of reverberation on the harmonic components, perfect dereverberation even in an ideal case is challenging. In [92], frame-wise blind dereverberation is achieved using correlation coefficients by considering the periodicity of voiced speech. This method involves the derivation of a reference signal by reducing speech periodicity from a recovered signal of previous frame. Using this reference signal the impulse response is estimated in each frame.

Spectral enhancement

In spectral enhancement techniques the short-time spectrum of the received signal(s) is processed to achieve dereverberation. Many single- and multi-microphone speech dereverberation algorithms have been proposed and in these approaches, reverberation is modeled as an additive noise [93]–[96]. Therefore, dereverberation is achieved via spectral subtraction.

Figure 2.5 shows a block diagram of the spectral subtraction technique. The noisy signal $x(n)$ is windowed and its discrete Fourier transform (DFT) $\underline{x}(n)$ is computed. The gain function is then estimated by using the current noisy magnitude samples, previous enhanced magnitude signal and noise statistics. The enhanced signal is obtained by associating the enhanced magnitude with the phase information $\angle \underline{x}(n)$ and processing them using the IDFT block along with an overlap-and-add operation. The gain

Fig. 2.5: Block diagram of spectral subtraction technique.

and noise estimation blocks are most crucial and the success of this technique is strongly dependent on the ability to achieve appropriate gains.

In [93, 97], the late reverberation energy of the received signal is estimated based on the Polack's statistical reverberation model [98]. Following this, many algorithms have been proposed to estimate the reverberant spectra [85, 95, 99, 100, 101, 102]. Most of these algorithms rely on probabilistic and statistical modeling of speech source and reverberation [51, 96, 103]. A single microphone approach exploiting the long-term correlation in the DFT domain has been proposed in [104] for the suppression of late reverberation. Wu and Wang proposed a single-microphone algorithm [105] in which the reverberation time is estimated by a pitch-based measure and subsequently, the late reverberation spectrum is estimated followed by spectral subtraction. Spectral subtraction has shown to be effective in the suppression of late reverberation. Many algorithms which apply inverse filtering to suppress early reflections employ spectral subtraction as a second stage to reduce the late reverberation energy [86, 106, 107, 108, 109]. It may be noted that the reverberant spectrum and the gain factor should be carefully estimated since inappropriate subtraction may cause varying levels of narrowband attenuation resulting in random spectral fluctuations. This can give rise to non-stationary tones at random frequencies and this annoying effect is often referred to as musical noise.

Temporal envelope filtering

Temporal envelope filtering techniques exploit the relationship between the temporal envelopes of the clean and reverberant speech signals to achieve speech dereverberation. Houtgast and Steeneken have

Fig. 2.6: Concept of temporal envelope filtering.

shown that the concept of modulation transfer function (MTF) can be applied to evaluate the speech quality in a reverberant environment since reverberation decreases the modulation index of the intensity envelope of the reverberant signal relative to that of the original signal [110]. Motivated by this observation, many algorithms have been proposed. In a typical temporal envelope filtering system, such as shown in Fig. 2.6, the temporal envelope of a received signal is first extracted. The envelope signal is next filtered to estimate the anechoic envelope. The anechoic speech signal is then reconstructed using the estimated anechoic envelope and the fine structure of the reverberant signal without considering the phase modifications in the fine structure. It is worth noting that some techniques require other parameters such as reverberation time from the temporal envelope.

Langhans and Strube proposed to estimate the energy envelope of an anechoic speech by applying a theoretically derived inverse MTF, where the envelopes were filtered in critical frequency bands in the STFT domain [111]. With the same objective Hirsch applied an ad-hoc high-pass filtering process [112]. A multiband processing technique based on the envelope function convolution relationship has been introduced in [113] for the enhancement of single-microphone recorded speech signals. According to this model the envelope of a reverberant signal can be approximated as the convolution between the clean speech envelope and the envelope of the AIR (which is assumed to be known). Hence speech enhancement can be achieved by deconvolution in each subband followed by the reconstruction of the speech signal using the original phase function.

In a different approach, Hirobayashi et al. proposed inverse filtering of power envelope transfer function [114]. Speech dereverberation using a multi-microphone subband envelope estimation (M-SBEE) has been proposed in [115]. In M-SBEE the signals are first divided into 256 subbands and dereverberation filter of each microphone is computed for each subband using least-square approximation.

For each subband the recovered signal corresponding to the microphone that has the minimum estimation error is selected as the recovered subband signal. Avendano and Hermansky applied a data designed filter bank technique, which was originally developed for denoising [116], into dereverberation [117]. They derived a filter bank using training data for the critical bands of reverberant speech. The short-term power spectrum trajectories (STPTs) of the reverberant signal are then filtered using these data-designed filters from which the recovered signal was constructed. Simulation results have shown that although the effect of reverberation is reduced perceptually, significant amount of artifacts are introduced. Unoki et al. [118]–[121] proposed an improvement of the basic power envelope model described in [114] which disregarded the phase information. The improved model blindly estimates AIR parameters and the carrier from the reverberant signal as well as the power envelope. Results showed that the phase information should be incorporated in the envelope filtering technique to improve the speech intelligibility.

2.2.3 Dereverberation via homomorphic transformation

Homomorphic transformation transforms a convolution product into an additive mixture. Cepstrum analysis is a commonly used homomorphic transformation to process signals obtained by convolution. To achieve dereverberation via homomorphic techniques, the signals are first transformed into the cepstral domain before processing. The complex cepstrum of a reverberant signal $x(n) = s(n) * h(n)$ is obtained by [122]

$$\begin{aligned}\mathcal{X}(n) &= \text{IDFT}\{\ln[\text{DFT}\{x(n)\}]\} \\ &= \mathcal{S}(n) + \mathcal{H}(n),\end{aligned}\tag{2.4}$$

where $\mathcal{X}(n)$, $\mathcal{S}(n)$, and $\mathcal{H}(n)$ are the cepstral-domain counterparts of $x(n)$, $s(n)$, and $h(n)$, respectively.

Averaging in the complex cepstral domain was exploited in [123] in order to increase the separation between the speech and the impulse response. In this method, an estimate of the AIR is obtained by cepstral averaging across the signal frames, and a least-squared error inverse filter is computed to remove the estimated impulse response from the reverberant speech. In this algorithm, however, the impulse response cepstrum is obtained empirically and the segmentation must be performed with care. These

limitations make this algorithm less suitable for practical applications. In [124], the authors recover the minimum-phase components of source waveforms by blind dereverberation using the minimum-phase complex cepstrum. An algorithm that combined spatial and cepstral processing of the microphone signals has been proposed in [125]. This algorithm was motivated by the observation that the minimum-phase and allpass components of the received signals are affected differently by the AIRs. In this method, the microphone signals are first delay-steered and then decomposed into minimum-phase and allpass components. The minimum-phase components are processed in the cepstral domain, where spatial averaging is done before low-time filtering is applied. The allpass components are processed in the frequency domain by performing spatial averaging and retaining only the allpass component of the resultant output.

A blind deconvolution-based approach for the restoration of speech degraded by the acoustic environment has been proposed in [126]. This method processes the outputs of two microphones using cepstra operations and phase information only. Under mild assumptions, it reconstructs the AIR associated with each microphone and estimates the source signal. An inverse filtering method based on cepstral analysis has been proposed for achieving near-perfect equalization of non-minimum phase AIRs in [127]. The algorithm performs an iterative flattening of the AIR magnitude spectrum prior to the inversion of the allpass component. Smoothing of the filter gain function in the cepstral domain has also been proposed in [128] to reduce musical noise introduced by the adaptive spectral filters during speech enhancement. In [129], cepstral processing was employed to obtain a priori estimates of the AIRs. Recently, Kumar and Stern approximated reverberation as a filtering operation in the cepstral feature domain and derived a maximum likelihood-based inverse filter for dereverberation [130]. Most of the algorithms in this class are based on some heuristics assumptions and hence may not be suitable for real applications. These algorithms may also introduce additional distortion to the speech signal.

2.2.4 Dereverberation via blind channel identification and inverse filtering

Reverberation is modeled as the convolution between the source signal and the AIR. Hence it is expected that, if the AIRs are correctly estimated, a deconvolution process can recover the original signal. Based on this assumption, speech dereverberation can be achieved via a two-stage process where AIRs are first estimated from the reverberant received signals using blind channel identification (BCI) techniques such

Fig. 2.7: Blind channel identification and equalization system.

as proposed in [8, 131, 132]. The received signals are then equalized by the inverse filtering process, where the received signals are convolved with inverse filters computed from the estimated AIRs. These processes are illustrated as shown in Fig. 2.7. Perfect equalization, in theory, is achievable with this technique provided the AIRs are known a priori. Many single- as well as multi-channel algorithms have been proposed to achieve reverberation cancellation via inverse filtering [133] and, in addition, many algorithms reviewed in the previous subsections [86, 106, 107, 108, 109, 127] can also be included in this category. Existing algorithms in this category face challenges such as:

- 1) The channels share common zeros;
- 2) The correlation matrix of the received signal is not a full rank matrix;
- 3) The AIRs are time varying and typically of high order;
- 4) AIR order is unknown;
- 5) Exact estimation is challenging, solutions are generally found to within a scale factor.

One of the first BCI algorithms has been proposed in [134] following which many BCI algorithms have been proposed. In Sections 2.2.2 and 2.2.3 several algorithms that estimate AIRs by exploiting speech characteristics and acoustic properties have been reviewed. Algorithms exploiting higher-order statistics (HOS) [35, 135, 136] require a large number of observations to estimate the channel parameters and these algorithms suffer from slow convergence. Since the introduction of second-order statistics (SOS) for BCI in [137], SOS-based BCI algorithms have gained increased interest in recent years. While algorithms presented in [138, 139] utilize the cross relationship (CR) between a channel output pair, the

subspace (SS) algorithm [140] exploits the orthogonality between the signal and noise subspaces. Hua proposed the two-step maximum likelihood (TSML) method for the estimation of AIRs [141, 142] and this method has shown to outperform SS and CR methods. A review of different SOS methods has been presented in [8]. Based on the eigenvalue decomposition of a sample correlation matrix, Gurelli and Nikias proposed a technique to estimate the order and root locations of the channel transfer functions for deconvolution [143]. This technique has further been extended for speech enhancement [132, 144]. These subspace methods estimate the null subspace of the data matrix and the null subspace vectors are then used to extract the AIRs.

A two-channel blind deconvolution algorithm has also been proposed in [145]. This method computes the eigenvector corresponding to the smallest eigenvalue of the input correlation matrix and the order of the AIR is determined using a cost function. To reduce sensitivity of equalization algorithms to the order mismatch of the transfer functions Hikichi et al. proposed to first estimate the AIRs with overestimated order using eigenvalue decomposition before computing their inverse filters [146, 147]. The second stage of the algorithm involves the estimation of a common polynomial across the estimated channels so as to compensate for its effect. However, all such eigenvalue decomposition based techniques are sensitive to errors in the correlation matrix estimation. An iterative approach based on the Godard cost has been described in [148] for multiple-input multiple-output (MIMO) impulse response estimation and spatiotemporal equalization. A deterministic time-domain multichannel blind deconvolution algorithm which estimates the AIRs in a least-squares sense and computes deconvolution filters in a minimum-norm sense has been proposed in [149]. As stated by the author, this algorithm is computationally intensive, particularly for high-order AIRs. In [150] Hopgood modeled speech using block stationary AR process whereas the AIRs were described by an all-pole model. The channel parameters were then estimated using Bayesian parameter estimation techniques. An SOS based counterpart of the joint estimation algorithm proposed in [136] has been presented in [151]. This algorithm estimates separate inverse filters for AIRs and a speech production system so that the output samples are uncorrelated. The proposed algorithm has been shown to work in the presence of additive noise.

In addition to the above, the triple-N independent component analysis for convolutive mixtures (TRINICON) framework that uses both SOS and HOS has been presented in [152] to achieve blind

source separation and dereverberation. This framework captures the statistical description of the temporal structure of the source signals using multivariate models in the cost function. However, the dereverberation performance of this framework has not been systematically evaluated. Although these algorithms can achieve reasonably good estimates in relatively high SNR conditions, most of them are not only computationally expensive but also difficult to implement.

In order to perform BCI with relatively less complexity, the time-domain multichannel least-mean-square (MCLMS) algorithm has been proposed in [131]. In MCLMS the unknown channels are estimated by minimizing an error function derived from the cross-relation between different channels. This algorithm, however, suffers from slow convergence. In order to accelerate its convergence, a variable step-size, derived from the inverse Hessian matrix of the cost function, has been employed in [131]. The inversion of a non-diagonal Hessian matrix, however, makes the resultant multichannel Newton (MCN) algorithm computationally intensive. In [153] the authors proposed the multichannel frequency-domain LMS (MCFLMS) and the normalized MCFLMS (NMCFLMS) algorithms. Compared to MCLMS and MCN, these algorithms achieve significant reduction in complexity and are able to achieve better channel estimates under high SNRs. Exploiting fast convergence achieved by proportionate-type algorithms for quasi-sparse AIRs, Ahmad et al. proposed the fast converging improved proportionate NMCFLMS (IPNMCFLMS) and multichannel multi-delay filter (MCMDF) algorithms [154].

One of the main challenges of the LMS-based adaptive BCI algorithms is the misconvergence problem [155]. Results presented in [156]–[159] show that, for LMS-based algorithms, the estimated AIRs first converge towards the desired solution after which they start misconverging. The effect of misconvergence becomes more significant under low SNR conditions. In an initial attempt to mitigate this misconvergence problem, a direct-path constraint was imposed on NMCFLMS [156]. In this technique, the direct paths of the estimated AIRs are substituted with the direct-path components of the true AIRs. This substitution is equivalent to providing NMCFLMS a better initial value at each iteration for updating in the next iteration. Since the direct-path components are not known a priori, a method to estimate them using the generalized cross-correlation (GCC) and the NMCFLMS cost function has been presented in [160]. In order to prevent misconvergence, a power constraint has been introduced to NMCFLMS in addition to the direct-path constraint [159]. Recently, a spectral constraint has been introduced to

NMCFLMS so as to improve its convergence performance in the presence of noise [158]. Lin et al. [161] introduced the concept of forced spectral diversity into the cross-relation based BCI in order to address the near-common zero problem [162, 163]. Adaptive approaches to AIR estimation have gained much interest due to the dynamic nature of the AIRs; these algorithms have the ability to track changes in the unknown AIRs. Many BCI algorithms have been proposed [164]–[170] and extensive research in the area of BCI is still ongoing.

2.3 Inverse filtering techniques

As described in the previous section, most of the BCI algorithms aim to estimate the AIRs and these AIRs are subsequently inverse filtered with the reverberant signals for reverberation suppression or cancellation. The inverse filtering operation involves the computation of stable inverse filters from the estimated AIRs. Since the dereverberation quality fully depends on the inverse filters, it is necessary to compute stable and casual finite length inverse filters.

A straightforward solution for AIR inversion is obtained by computing the direct inverse of the channel transfer function, i.e.,

$$G(z) = \frac{1}{H(z)}, \quad (2.5)$$

where $H(z)$ and $G(z)$ are the AIR and its inverse filter in the z -domain, respectively. It is well-known that for a filter to be stable, its poles in the z -plane should be inside the unit circle [171]. If an AIR has zeros outside the unit circle, then its direct inversion will result in poles outside the unit circle leading to an unstable filter. Hence direct inversion can be applied only to minimum phase systems. It is also well-known that a typical AIR is non-minimum phase with zeros outside the unit circle and therefore its direct inversion will result in an unstable inverse filter [5]. This makes acoustic channel equalization intrinsically challenging. Several algorithms have been proposed for the inversion of mixed-phase AIRs and these methods can be broadly dichotomized into single-channel or multiple-channels. Approaches for single-channel equalization include techniques such as homomorphic (Section 2.2.3) and least-squares [172] techniques. In the homomorphic method, the AIR is first decomposed into minimum phase and allpass components [127, 172]. Equalization is then achieved on these individual components.

On the other hand, the least-square method is applied directly on the time-domain sequence where the inverse filters are estimated by minimizing the squared error between a reference and the output of the inverse filtering system recursively. This method, in fact, estimates the filter coefficients so that the filtered output is close to the reference signal. It may be noted that this method achieves partial equalization by inverting only zeros inside the unit circle. Moreover, it requires a reference signal to estimate the inverse filter. In general, these single-channel methods result in an approximate equalization and hence achieving perfect signal recovery is still a challenging task [172, 173].

In order to address the drawbacks of single-channel techniques a multichannel equalization algorithm based on the multiple-input/output inverse theorem (MINT) has been proposed by Miyoshi and Kaneda in [173]. The MINT solution which is closely related to the Bezout identity achieves perfect inversion, in theory, if the channels have no common zeros. However, its equalization performance reduces in the presence of near-common zeros [162], AIR estimation errors, undermodeling, and noise. Numerical analysis of multiple input inverse filtering, presented in [174], states that the finite precision of the estimated AIRs along with the inversion of poorly conditioned matrix may lead to numerical limitations of MINT. It has been further shown that as the number of channels increases the condition number decreases and this, in turn, enhances the numerical performance. The MINT algorithm will be reviewed in more detail in Chapter 3.

Perfect estimation of AIRs is still a challenging problem. Since, in practice, inverse filters are computed using inaccurate estimated AIRs, it is desirable for equalization algorithms to be robust to such AIR inaccuracies. Hikichi et al. stated that the inverse filter may amplify noise and small fluctuations in AIRs causing significant distortions in the filter output [6]. They proposed to reduce the inverse filter energy to make it less sensitive to AIR fluctuations and noise. To achieve this a regularized least-squares approach which relies on the regularization parameter, filter length, and modeling delay [6, 175, 176] has been proposed. Results presented in [6, 175, 176] show that the introduction of the regularization factor improves the equalization performance by reducing the filter energy. However, the regularization parameter has been empirically chosen and no mathematical formulation has been provided to compute its optimal value. In [177] a truncated singular value decomposition (TSVD) approach has been introduced to improve the robustness of MINT to channel estimation errors. Since late rever-

beration affects speech intelligibility more significantly, one approach is to partially equalize the AIR such that only late reverberation is suppressed instead of computing the exact inverse filters. Works by Kallinger and Mertins [178]–[180] showed that shaping the AIRs to a shorter reverberation time improved the performance of the equalization system. In [181] a mathematical link between MINT and channel shortening (CS) has been derived and the relaxed multichannel least-squares (RMCLS) algorithm has been proposed. Simulation results showed that RMCLS outperforms CS in the presence of channel estimation errors. Kodrasi et al. incorporated regularization to RMCLS and CS for improving their performance [182]. The same authors proposed partial multichannel equalization using MINT (P-MINT) and regularized P-MINT and these algorithms have shown to outperform the MINT and CS algorithms [183]. Within the RMCLS framework Lim and Naylor employed partial relaxation of the early taps of the equalized response to increase robustness to channel estimation errors, while constraining the initial taps to avoid undesirable coloration of the equalized signal [184].

Another concern of MINT is that it requires the inversion of a matrix with dimension that is proportional to the length of AIRs. Since a typical AIR can be of hundreds of samples in length, direct application of MINT in practical scenarios remains a challenge. Adaptive MCEQ algorithms have also been proposed with the aim of estimating inverse filters iteratively without matrix inversion, thereby reducing the complexity [185]–[188]. These algorithms minimize a cost function that has been formulated similar to that of MINT. Although these algorithms can achieve reasonable estimates, they suffer from slow convergence. To achieve higher rate of convergence, the sparsity of the desired response (corresponding to the convolution between the inverse filters and the estimated AIRs) has been incorporated to A-MINT [189]. A sub-system based approach where a cost function is formulated using the output of two sub-systems has been proposed in [190, 191]. Although the techniques of [189]–[191] improve the rate of convergence for adaptive equalization, they still require substantial computational cost.

Another approach to reduce the complexity of MINT is the use of subband methods [192]–[195]. In [192, 195], reverberant signals are first divided into subbands and each subband is decimated and processed by MINT. The subband algorithm presented in [193] applies, for each channel, a single least-square inverse filter on each subband and the channel with the minimum estimation error is chosen as the best invertible channel of that subband. The recovered signal corresponding to the best invertible

channel is then used as the retrieved subband signal. Although these subband methods introduce modest level of spectral distortion, they are computationally more efficient than MINT. In [194], the subband approach is combined with the adaptive multichannel equalization technique to achieve faster convergence and complexity reduction than MINT. The computational complexity of subband approaches is often determined by the number of subbands and decimation factor while the equalization performance is often dependent on the analysis and synthesis filter bank design [194, 196]. Therefore, as with most subband algorithms, there is often a tradeoff between performance and computational cost. It is also important to note that subband methods require the time alignment of received signals as well as AIRs to achieve dereverberation. Therefore, any time delays introduced due to the source-sensor distance must be estimated and compensated in the analysis filter bank [192]. To reduce the complexity Schmid and Enzner proposed to transform the matrix-based solutions of MINT back into lightweight digital filters with less resource consumption and guaranteed numerical stability [197]. They also showed that, with a matched filter array (MFA) and subsequent single-channel equalization utilizing parametric all-pole modeling of the room acoustics, broadband least-squares equalization can be efficiently approximated.

In addition to the above, Amari et.al presented an equalization algorithm based on a maximum entropy formulation for multichannel blind deconvolution [198]. They also extended the natural gradient method [199] for MCEQ to achieve a fast and accurate adaptation that is independent of the characteristics of the unknown channel. Norcross et al. proposed the use of a regularization weighting as the magnitude of the desired function [200]. The phase of the target function was chosen to be minimum-phase to reduce any pre-ringing in the corrected impulse response. This avoids audible artifacts introduced by the non-causal filters. A blind channel equalization algorithm has been presented in [201] with the objective of selecting the best equalizer. This method computes equalizers with all possible delays using second-order statistics of the received signals and the best equalizer is selected based on an evaluation index. A multiple position mixed-phase equalizer, which is a combination of a room amplitude equalizer and a group delay compensator, has been proposed in [202]. Based on informal subjective listening tests, the proposed method is reported to be capable of improving the clarity index without introducing artifacts. The improved proportionate normalized LMS algorithm has been extended to multichannel equalization in [203]. Simulation results presented in the manuscript show that the

Table 2.1: ITU-T recommended speech quality measurement scales [1].

Listening-Quality Scale:	
Quality of the speech	Score
Excellent	5
Good	4
Fair	3
Poor	2
Bad	1

Listening-Effort Scale:	
Effort required to understand the meaning of sentences	Score
Complete relaxation possible; no effort required	5
Attention necessary; no appreciable effort required	4
Moderate effort required	3
Considerable effort required	2
No meaning understood with any feasible effort	1

proposed adaptive algorithm outperforms the conventional LMS-based algorithms.

2.4 Speech dereverberation quality measures

It is important to evaluate the quality of equalized (dereverberated) speech signals so as to assess the performance of dereverberation algorithms. The improvement in speech quality can be determined by comparing the speech quality before and after dereverberation. To this end, both subjective and objective measures have been introduced for speech quality assessment. In this section, some of the common quality measures are briefly discussed.

2.4.1 Subjective measures

Subjective measures deal with the perceptual quality of speech and involve human test subjects to assess the quality of a speech sample. According to the International Telecommunications Union (ITU-T) standard a five-point listening quality scale is used to rate the speech quality for both listening-only and conversational tests [1]. An alternative speech quality scale used in listening-only tests is the listening-effort scale. These scales are listed in Table 2.1.

During the evaluation process, a number of subjects listen to the recordings of reverberant speech as well as the speech recovered by the algorithm under test. The subjects rate the quality of each signal, or

the effort required to understand it, using the listening-quality scale or listening-effort scale, respectively. The average of all scores which indicates the subjective quality of the system or algorithm under test is known as the mean opinion score (MOS). A large number of subjects is desired to obtain a realistic MOS.

Diagnostic acceptability measure (DAM) [204] is another procedure that is used to assess the speech quality on a multidimensional metric. The DAM uses sixteen scales, belonging to one of three categories: signal quality, background quality, and overall quality and a weighted average of all these scales is used as the metric to assess the performance of the algorithm under test.

Although subjective quality assessment measures are very popular a large number of subjects is required to obtain a reliable MOS. Hence subjective measures are time consuming and expensive to implement. They also require specialized methodologies and personnel with specific expertise.

2.4.2 Objective measures

Many objective measures have also been proposed to evaluate the performance of dereverberation algorithms. These measures are derived from formulations and hence do not involve human listeners. This in turn avoids human errors and reduces the time and cost involved. Objective speech quality measures can generally be categorized into (a) time domain, (b) spectral domain, and (c) perceptual domain. In the following, some of the popular objective measures that have been used in thesis are briefly discussed. While objective measures are fast and inexpensive to implement, it is useful to note that these measures require the use of the clean speech signal as reference. Hence such measures are applicable for synthetic simulations unless an additional microphone is placed directly in-front-of the sound source to capture the reference signal for comparison.

Intrusive quality measures

Segmental signal to reverberation ratio: Defining the direct signal

$$s_d(n) = s(n) * h_d(n)$$

where $h_d(n)$ (2.6) is the direct path response, the instantaneous segmental signal-to-reverberation ratio (SRR_{seg}) [205] of the k th frame is defined as

$$\text{SRR}_{\text{seg}}(k) = 10 \log_{10} \left\{ \frac{\sum_{n=kN}^{kN+L-1} s_d^2(n)}{\sum_{n=kN}^{kN+L-1} [s_d(n) - \hat{s}_d(n)]^2} \right\}, \quad (2.7)$$

where L is the frame size, N is the frame shift and $\hat{s}_d(n)$ is the signal enhanced by the dereverberation algorithm. The mean SRR_{seg} is then obtained by

$$\text{SRR}_{\text{seg}} = \frac{1}{K} \sum_{k=0}^{K-1} \text{SRR}_{\text{seg}}(k), \quad (2.8)$$

where K is the total number of frames. A high value of SRR_{seg} is desirable as it implies good dereverberation performance.

Intrusive perceptually-based measures

Bark spectral distortion: The Bark spectral distortion (BSD) [206] is a perceptual measure that incorporates human auditory models [207] with emphasis on the perceptual quality. The BSD is calculated using the Bark spectra of $s_d(n)$ and the enhanced signal $\hat{s}_d(n)$. Defining $B_{s_d}(k, l)$ and $B_{\hat{s}_d}(k, l)$ as values corresponding to the l th Bark frequency bins of $s_d(n)$ and $\hat{s}_d(n)$, respectively, the BSD is computed using

$$\text{BSD} = \frac{1}{K} \sum_{k=0}^{K-1} \frac{\sum_{l=1}^{N_c} [B_{s_d}(k, l) - B_{\hat{s}_d}(k, l)]^2}{\sum_{l=1}^{N_c} [B_{s_d}(k, l)]^2}, \quad (2.9)$$

where N_c is the total number of critical bands in the Bark spectrum.

Perceptual Evaluation of Speech Quality (PESQ): The ITU-T Recommendation P.862 (February 2001) [208] defines an objective perceptual quality measure known as PESQ. This objective measurement tool predicts the outcome of subjective listening tests. The PESQ compares the reference clean signal with the enhanced signal using a sensory (perceptual) model to give a quality score that is analo-

gous to the subjective MOS measured according to ITU-T P.800. The PESQ scores are calibrated with the help of a large database of subjective tests. It also involves the transformation of both the reference and test signals to a representation that is analogous to the psychophysical representation of audio signals in the human auditory system. This transformation takes both the perceptual Bark frequency and loudness into account. Such an internal representation process, in turn, allows the application of a small number of quality indicators to model all subjective effects. The PESQ computes two error parameters in the cognitive model which are then combined to give an objective listening quality score.

2.5 Conclusion

Reverberation cause detrimental effects on the quality of a received speech signal. Speech dereverberation algorithms aim to recover the clean speech signal by mitigating these effects. Some of the popular speech dereverberation techniques were briefly discussed in this chapter. Although reverberation cancellation can be achieved using inverse filtering, perfect estimation of AIRs is a challenging task. Hence speech dereverberation is generally achieved using inaccurate AIRs estimated by BCI algorithms. In addition, while different BCI techniques have been proposed so far, AIR estimation in a noisy environment is still challenging.

Direct inversion of a typical non-minimum phase AIR results in an unstable solution. Therefore, in order to achieve equalization of non-minimum phase AIRs using stable inverse filters, single- and multichannel algorithms have been developed with the latter being more popular since they can achieve perfect equalization. While the MINT algorithm remains a popular MCEQ algorithm, it is not robust to channel estimation errors and is computationally expensive. Many algorithms have been proposed with the aim of addressing these issues. A fast and robust MCEQ algorithm is desirable to achieve equalization in practical scenarios.

This chapter also reviews some of the speech quality measures that have been generally employed to assess the performance of dereverberation algorithms. Some of the popular subjective and objective measures were discussed in this chapter. It has been noted that while the objective measures are fast and easy to implement, they require a reference signal.

Chapter 3

A Generalized Framework for Speech Dereverberation with Insights into the Multichannel Equalization Problem

3.1 Introduction

Multichannel equalization (MCEQ) is an important component of the blind channel identification (BCI) based speech dereverberation system. Most of the existing MCEQ algorithms achieve equalization in the time domain regardless of the transform domain in which the BCI algorithm has been formulated. Little studies have been conducted to investigate the suitability of equalization in different transform domains. As will be shown in this chapter, the advantage of processing signals in the transform domain is that signals can be better represented and equalization can be performed in each transform subspace by designing dedicated inverse filters.

One of the main contributions of this thesis is the formulation of a generalized MCEQ framework. This generalized framework is designed specifically to allow different transform domains to be applied for equalization. The motivation behind formulating a generalized framework is the flexibility of using various transforms to perform channel equalization without modifying the underlying equalization technique such as MINT. As will be shown, this generalized framework not only allows one to develop new algorithms, it can also be exploited to gain insights into the performance of these algorithms. In this chapter, with the help of the proposed framework, the time (MCEQ-T), and Karhunen-Loève transform (MCEQ-KL) domain equalization algorithms are developed. For the time-domain approach, the proposed framework is exploited to gain insights into the MCEQ algorithm. This is achieved by illustrating the

Fig. 3.1: A single-input multiple-output (SIMO) model.

variation in equalization performance with the choice of modeling delay. The insights subsequently lead to the development of two time-domain MCEQ-T algorithms (one that employs multiple delays and the other that employs a single optimal delay). In addition, existing formulation of MCEQ algorithms does not take into account equalization in the presence of noise. This forms the motivation for integrating MCEQ-KL with an existing KLT-domain denoising algorithm so as to achieve joint dereverberation and denoising. In this chapter, since the equalization algorithms are proposed in the time and KLT domains, the superscripts I and KL are, respectively, used to denote these domains.

This chapter is organized as follows: In Section 3.2 the conventional MCEQ is reviewed. The generalized framework is derived in Section 3.3. In Section 3.4, the proposed time- and KLT-domain MCEQ algorithms are presented. In Section 3.5 the performance of these algorithms is analyzed. Finally, conclusion is presented in Section 3.6.

3.2 Review of multichannel equalization

An M -channel single-input multiple-output (SIMO) finite impulse response (FIR) system is considered here in the context of room acoustics, as shown in Fig. 3.1. The signal $x_m(n)$ received by the m th

microphone is the convolution of the clean speech signal $s(n)$ and the AIR h_m between the source and the m th microphone, such that

$$\begin{aligned} x_m(n) &= h_m * s(n) + v_m(n) \\ &= y_m(n) + v_m(n), \quad m = 1, \dots, M, \end{aligned} \quad (3.1)$$

where $*$ is the linear convolution operator, $v_m(n)$ is additive noise and $y_m(n) = h_m * s(n)$. Under noiseless condition, (3.1) can be rewritten as $x_m(n) = h_m * s(n)$.

Given a set of M AIRs $\mathbf{h}_m = [h_{m,0}, \dots, h_{m,L_h-1}]^T$, $m = 1, \dots, M$, where L_h is the length of $\mathbf{h}_m$, the aim of MCEQ is to estimate inverse filters $\mathbf{g}_m = [g_{m,0}, \dots, g_{m,L_g-1}]^T$, each of length L_g , such that the inverse filters satisfy

$$\sum_{m=1}^M \mathbf{H}_m^T \mathbf{g}_m = \mathbf{d}, \quad (3.2)$$

where

$$\mathbf{H}_m = \begin{bmatrix} h_{m,0} & \dots & h_{m,L_h-1} & 0 & 0 & \dots & 0 \\ 0 & h_{m,0} & \dots & h_{m,L_h-1} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & h_{m,0} & \dots & h_{m,L_h-1} \end{bmatrix} \quad (3.3)$$

is a $L_g \times L_d$ filtering matrix constructed from $\mathbf{h}_m$ and the target response

$$\mathbf{d} = [\underbrace{0, \dots, 0}_{\tau}, 1, 0, \dots, 0]^T \quad (3.4)$$

is an $L_d \times 1$ vector with the $(\tau + 1)$ th element equal to 1, corresponding to a modeling delay τ , and $L_d = L_g + L_h - 1$. It may be noted that the dimension of $\mathbf{H}_m$, as will be shown in Section 3.3, varies with the frame size of the proposed multichannel equalization framework. From (3.2), an optimization problem can be formulated and the MCEQ filters can be computed using [209]

$$\arg \min_{\mathbf{g}} \|\mathbf{H}^T \mathbf{g} - \mathbf{d}\|_2^2 = (\mathbf{H}^T)^+ \mathbf{d}, \quad (3.5)$$

where $(.)^+$ is the matrix pseudo-inverse operator,

$$\begin{aligned}\mathbf{H} &= [\mathbf{H}_1^T, \dots, \mathbf{H}_M^T]^T, \\ \mathbf{g} &= [\mathbf{g}_1^T, \dots, \mathbf{g}_M^T]^T\end{aligned}\quad (3.6)$$

and $\|\cdot\|_p$ is the p -norm. In practice, BCI algorithms achieve an erroneous estimate $\hat{\mathbf{h}}_m$ of $\mathbf{h}_m$. Defining $\hat{\mathbf{H}} = [\hat{\mathbf{H}}_1^T, \dots, \hat{\mathbf{H}}_M^T]^T$, where $\hat{\mathbf{H}}_m$ is constructed from $\hat{\mathbf{h}}_m$ similar to (3.3), the corresponding inverse filters are obtained as

$$\hat{\mathbf{g}} = [\hat{\mathbf{g}}_1^T, \dots, \hat{\mathbf{g}}_M^T]^T = (\hat{\mathbf{H}}^T)^+ \mathbf{d}. \quad (3.7)$$

The inverse of a mixed-phase sequence is, in theory, infinite in length and non-causal [210]. In contrast to this, FIR filters are used for MCEQ. The value of L_g is selected such that, for minimum-norm solutions, $L_g \geq L_c$ when $M \geq 2$ [211], where the critical length

$$L_c = \left\lceil \frac{(L_h - 1)}{(M - 1)} \right\rceil \quad (3.8)$$

and $\lceil \cdot \rceil$ denotes the ceiling operator. In addition, it is desirable to incorporate a modeling delay $0 \leq \tau < L_d - 1$ in $\mathbf{d}$ so as to compensate for the non-causality property of $\mathbf{g}$ due to acoustic path delay [172, 210, 212]. In addition, the equalization performance is, to a large extent, dependent on τ for $L_g \leq L_h$ [6, 175, 176]. Since determining the optimal value of τ is challenging, an arbitrary delay $\tau \geq 0$ is often chosen in practice [6, 213].

3.3 The proposed generalized MCEQ framework

A generalized MCEQ framework that facilitates the development of algorithms in different transform domains is proposed here. This framework operates on signals in their corresponding transform domains for the development of MCEQ inverse filters. Each of these transform domains is characterized by a unitary transform matrix

$$\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_L], \quad (3.9)$$

such that $\mathbf{U}^H \mathbf{U} = \mathbf{I}_{L \times L}$, where $\mathbf{I}_{L \times L}$ is the $L \times L$ identity matrix and $\{\cdot\}^H$ denotes Hermitian transposition, with $\mathbf{u}_l = [u_{l,1}, \dots, u_{l,L}]^T$, $l = 1, \dots, L$. Each of the mutually orthonormal vectors $\mathbf{u}_1, \dots, \mathbf{u}_L$, which are the basis vectors of an L dimensional space, defines a subspace and has unit norm.

The relationship between the time and transform domains can be observed by expressing the clean speech vector $\mathbf{s}(n) = [s(n), \dots, s(n-L+1)]^T$ in the transform domain as

$$\begin{aligned} \underline{\mathbf{s}}^D(n) &= \mathbf{U}^H \mathbf{s}(n) \\ &= [\underline{s}^D(n, 1), \dots, \underline{s}^D(n, L)]^T, \end{aligned} \quad (3.10)$$

such that the l th element of this vector is given by

$$\underline{s}^D(n, l) = \mathbf{u}_l^H \mathbf{s}(n), \quad l = 1, \dots, L. \quad (3.11)$$

The superscript D is used to denote an arbitrary transform domain - for example $(\cdot)^I$ denotes the time domain while $(\cdot)^{KL}$ represents the KLT domain. Equations (3.10) and (3.11) describe the analysis of $\mathbf{s}(n)$. The synthesis process aims to reconstruct the time-domain signal $\mathbf{s}(n)$ from the transform domain and is described by

$$\begin{aligned} \mathbf{s}(n) &= \sum_{l=1}^L \underline{s}^D(n, l) \mathbf{u}_l \\ &= \mathbf{U} \underline{\mathbf{s}}^D(n). \end{aligned} \quad (3.12)$$

With the above definitions, a generalized MCEQ framework that is applicable to various transform domains defined by $\mathbf{U}$ can be derived. Here, received signals are first partitioned into overlapping frames, each of length L_x , with a frame shift of N samples, such that $1 \leq N \leq L$ where $L = L_x + L_h - 1$. Therefore when $N \geq L_x$, no overlapping occurs. Defining $k = 0, \dots, K-1$ as the frame index with K being the total number of frames, the k th frame of the m th channel received signal can be represented by

$$\begin{aligned} \mathbf{x}_m(k) &= [x_m(kN + L_x - 1), \dots, x_m(kN)]^T \\ &= \mathbf{H}_m \mathbf{s}(k), \end{aligned} \quad (3.13)$$

where the $L \times 1$ clean speech frame

$$\mathbf{s}(k) = [s(kN + L_x - 1), \dots, s(kN + L_x - L)]^T, \quad (3.14)$$

such that $s(n) = 0, \forall n < 0$ and $\mathbf{H}_m$ is the $L_x \times L$ filtering matrix. Let $\underline{\mathbf{s}}^D(k) = [\underline{s}^D(k, 1), \dots, \underline{s}^D(k, L)]^T$ such that it corresponds to $\mathbf{s}(k)$ in the transform domain. The transform-domain counterpart of (3.13) can then be obtained using $\mathbf{U}$, such that the $L \times 1$ vector

$$\begin{aligned} \underline{\mathbf{x}}_m^D(k) &= \underline{\mathbf{H}}_m^D \underline{\mathbf{s}}^D(k), \quad m = 1, \dots, M, \\ &= \mathbf{U}^H \begin{bmatrix} \mathbf{x}_m(k) \\ \mathbf{0}_{(L-L_x) \times 1} \end{bmatrix} \\ &= [\underline{x}_m^D(k, 1), \dots, \underline{x}_m^D(k, L)]^T, \end{aligned} \quad (3.15)$$

where $\mathbf{0}_{(L-L_x) \times 1}$ is the $(L - L_x) \times 1$ null vector and the m th channel convolution matrix of dimension $L \times L$ is given, in the transform domain, by

$$\underline{\mathbf{H}}_m^D = \mathbf{U}^H \begin{bmatrix} \mathbf{H}_m \\ \mathbf{0}_{(L-L_x) \times L} \end{bmatrix} \mathbf{U}. \quad (3.16)$$

To derive the inverse filters in the transform domain, the m th channel inverse filter that lies in the subspace of $\mathbf{u}_l$ is defined by $\underline{\mathbf{g}}_{m,l}^D = [g_{m,l}^D(1), \dots, g_{m,l}^D(L)]^T$, which is of length $L_g = L$. By filtering the M microphone outputs with M inverse filters $\underline{\mathbf{g}}_{m,l}^D$, it is obtained, for the k th frame, the estimate of the l th coefficient of $\underline{\mathbf{s}}^D(k)$

$$\begin{aligned} \hat{\underline{s}}^D(k, l) &= \sum_{m=1}^M (\underline{\mathbf{g}}_{m,l}^D)^T \underline{\mathbf{x}}_m^D(k) \\ &= \left[\sum_{m=1}^M (\underline{\mathbf{g}}_{m,l}^D)^T \underline{\mathbf{H}}_m^D \right] \underline{\mathbf{s}}^D(k), \end{aligned} \quad (3.17)$$

where (3.15) and (3.16) have been employed. Assuming that the AIRs are co-prime, perfect estimation

of $\underline{s}^D(k, l)$ can therefore be achieved, in theory, if

$$\sum_{m=1}^M (\underline{\mathbf{g}}_{m,l}^D)^T \underline{\mathbf{H}}_m^D = \mathbf{d}_l^T, \quad (3.18)$$

where

$$\mathbf{d}_l = [0, \dots, 0, \underbrace{1}_{l-1}, 0, \dots, 0]^T \quad (3.19)$$

is an $L_d \times 1$ vector with the l th element equal to 1 (with a modeling delay of $l - 1$). The $ML \times 1$ inverse filter in the transform domain, corresponding to the l th subspace across all channels, can then be obtained using

$$\begin{aligned} \underline{\mathbf{g}}_l^D &= [(\underline{\mathbf{g}}_{1,l}^D)^T, \dots, (\underline{\mathbf{g}}_{M,l}^D)^T]^T \\ &= [(\underline{\mathbf{H}}^D)^T]^+ \mathbf{d}_l, \end{aligned} \quad (3.20)$$

where $\underline{\mathbf{H}}^D = [\underline{\mathbf{H}}_1^{D^T}, \underline{\mathbf{H}}_2^{D^T}, \dots, \underline{\mathbf{H}}_M^{D^T}]^T$ is the $ML \times L$ concatenated AIR matrix. Alternatively, with $\mathbf{d}_l$ forming the l th column of an $L \times L$ identity matrix, the equalization filters in the transform domain can be computed across all L subspaces using a $ML \times L$ matrix given by

$$\begin{aligned} \underline{\mathbf{G}}^D &= [\underline{\mathbf{g}}_1^D, \dots, \underline{\mathbf{g}}_L^D] \\ &= [(\underline{\mathbf{H}}^D)^T]^+ \mathbf{I}_{L \times L}. \end{aligned} \quad (3.21)$$

Equation (3.21) provides a generalized framework for the MCEQ filters in the transform domain. Hence, for an M -channel system, the coefficients $\underline{s}^D(k, l)$, $l = 1, \dots, L$, in the transform domain are estimated framewise using L sets of M inverse filters, each of order L defined in (3.21). The l th coefficient of the k th frame (each of dimension $L \times 1$) in the transform domain can be recovered using (3.17), hence obtaining the estimate of the transform-domain $L \times 1$ vector

$$\hat{\underline{\mathbf{s}}}^D(k) = \underline{\mathbf{G}}^{D^T} [\underline{\mathbf{x}}_1^{D^T}(k), \dots, \underline{\mathbf{x}}_M^{D^T}(k)]^T. \quad (3.22)$$

Fig. 3.2: The generalized MCEQ framework.

Applying the inverse transform defined in (3.12), the estimate $\hat{s}(k)$ of $s(k)$ can then be obtained as

$$\begin{aligned}\hat{s}(k) &= \mathbf{U} \hat{\mathbf{s}}^D(k) \\ &= [\hat{s}(kN + L_x - 1), \dots, \hat{s}(kN + L_x - L)]^T.\end{aligned}\quad (3.23)$$

Figure 3.2 shows a conceptual block diagram that defines the proposed framework.

It can be noted from (3.22) and (3.23) that $s(k)$ of length L is estimated from $\mathbf{x}_m(k)$ of length L_x , where $L > L_x$. Hence it is possible to recover the original signal with a frame shift $1 \leq N \leq L$. In addition, it is common to employ windowed and overlapped frames to achieve better reconstruction. If an overlapped rectangular window is used, reconstruction of the clean speech signal is performed by truncating redundant data in $\hat{s}(k)$, which is a direct consequence of frame overlapping [214]. From the k th estimated frame, a segment $\hat{s}[kN + L_x + (\varrho - 1)L], \dots, \hat{s}[kN + L_x + (\varrho - 1)L + N - 1]$ of only N samples can be chosen for reconstruction, where $\varrho = (L - N)/L$. In addition, if necessary, ϱL and $L(1 - \varrho) - N$ zeros must be padded to the beginning and the end of $\mathbf{x}_m(n)$, respectively, so as to prevent information loss due to truncation. It can also be seen from (3.15) and (3.16) that $L - L_x$ zeros are appended for constructing an L -dimensional vector or matrix in this framework.

One of the main advantages of the proposed framework is that it allows one to formulate and develop MCEQ algorithms using a variety of transform domains. More importantly, as will be shown, this

framework allows one to gain insights into and compare the intrinsic properties of these algorithms.

3.4 Proposed framework-based algorithms

3.4.1 MCEQ-T

When $\mathbf{U} = \mathbf{I}_{L \times L}$, the algorithm operates in the time domain. For this time-domain MCEQ-T algorithm, each column of $\mathbf{I}$ constitutes a basis vector, i.e., $\mathbf{I} = [\mathbf{i}_1, \dots, \mathbf{i}_L]$, where

$$\mathbf{i}_l = [\underbrace{0, \dots, 0}_{l-1}, 1, \underbrace{0, \dots, 0}_{L-l}]^T, \quad l = 1, \dots, L. \quad (3.24)$$

As can be seen from (3.15) and (3.16), when $\mathbf{U} = \mathbf{I}_{L \times L}$, the last $L - L_x$ rows of the transform domain variables $\underline{\mathbf{x}}_m^{\text{D}}(k)$ and $\underline{\mathbf{H}}_m^{\text{D}}$ are zeros. As noted from (3.22) and (3.23), this implies that such zeros will not contribute to the reconstruction of $s(k)$. Therefore, the length of the equalization filters $\underline{\mathbf{g}}_{m,l}^{\text{I}}$, where $(\cdot)^{\text{I}}$ denotes for the time domain in the context of MCEQ-T, can be set to $L_g = L_x$ instead of L . In addition, the equalization filters in (3.21) can be expressed by the $ML_x \times L$ matrix

$$\begin{aligned} \underline{\mathbf{G}}^{\text{I}} &= [\underline{\mathbf{g}}_1^{\text{I}}, \dots, \underline{\mathbf{g}}_L^{\text{I}}] \\ &= [(\underline{\mathbf{H}}^{\text{I}})^T]^+ \mathbf{I}_{L \times L} \\ &= (\mathbf{H}^T)^+ \mathbf{I}_{L \times L}, \end{aligned} \quad (3.25)$$

where $\underline{\mathbf{H}}^{\text{I}} = \mathbf{H}$ is of dimension $ML_x \times L$ and the $ML_x \times 1$ vectors $\underline{\mathbf{g}}_l^{\text{I}} = [(\underline{\mathbf{g}}_{1,l}^{\text{I}})^T, \dots, (\underline{\mathbf{g}}_{M,l}^{\text{I}})^T]^T$. Employing the MCEQ filters obtained in (3.25), the l th coefficient of the k th frame is estimated using the $L_x \times 1$ received signal $\underline{\mathbf{x}}_m^{\text{I}}(k)$ given by

$$\begin{aligned} \hat{s}^{\text{I}}(k, l) &= \sum_{m=1}^M (\underline{\mathbf{g}}_{m,l}^{\text{I}})^T \underline{\mathbf{x}}_m^{\text{I}}(k) \\ &= (\underline{\mathbf{g}}_l^{\text{I}})^T \{[\underline{\mathbf{x}}_1^{\text{I}}(k)]^T, \dots, [\underline{\mathbf{x}}_M^{\text{I}}(k)]^T\}^T \\ &= \hat{s}(kN + L_x - L + l - 1). \end{aligned} \quad (3.26)$$

Fig. 3.3: Equalization using MCEQ-T.

An estimate of the k th frame of $\mathbf{s}(n)$, of length L , is finally obtained using

$$\hat{\mathbf{s}}^I(k) = \sum_{l=1}^L \hat{\mathbf{s}}^I(k, l) \mathbf{i}_l. \quad (3.27)$$

The above implies that L sets of M equalization filters, each of length L_x , are used for recovering $\mathbf{s}(k)$ frame-by-frame. However, as described in Section 3.3 after (3.23), it is sufficient to recover only N samples corresponding to N sets of inverse filters

$$[\underline{\mathbf{g}}_{l_s}^I, \dots, \underline{\mathbf{g}}_{l_e}^I] = (\mathbf{H}^T)^+ [\mathbf{d}_{l_s}, \dots, \mathbf{d}_{l_e}], \quad (3.28)$$

where the start and end values $1 \leq l_s \leq L - N + 1$ and $l_e = l_s + N - 1$, per frame for reconstruction since overlapped framing with a frame shift of N samples is employed. Hence only N corresponding sets of inverse filters need to be estimated as depicted in Fig. 3.3.

3.4.2 Insights into the MCEQ-T algorithm

It can be noted that, due to the intrinsic property of $\mathbf{I}$, $\underline{\mathbf{H}}_m^I = \mathbf{H}_m$, $\underline{\mathbf{x}}_m^I(k) = \mathbf{x}_m(k)$ and $\hat{\underline{\mathbf{s}}}^I(k, l) = \hat{\mathbf{s}}(kN + b)$, such that $\hat{\mathbf{s}}(kN + b) = 0$ for $kN + b < 0$. Here $b = L_x - L + l - 1$. The column vector $\mathbf{i}_l$, defined in (3.24), can be interpreted as the subspace that contains time delay information of $\hat{\underline{\mathbf{s}}}^I(k, l)$. Therefore, while MINT employs a single inverse filter of length L_g per channel to recover $\mathbf{s}(n)$, MCEQ-

T employs a dedicated filter $\underline{\mathbf{g}}_{m,l}^{\text{I}}$ for *each* coefficient $\hat{s}(kN + l), l = L_x - L, \dots, L_x - 1$, per channel described by (3.26). The link between MCEQ-T and MINT can be established from the equality $\mathbf{U} = \mathbf{I}$ by letting the frame shift parameter $N = 1$ in (3.26) and recovering the l th sample in each frame using

$$\begin{aligned}\hat{\underline{s}}^{\text{I}}(k, l) &= \hat{s}(k + L_x - L + l - 1) \\ &= \sum_{m=1}^M \left(\underline{\mathbf{g}}_{m,l}^{\text{I}} \right)^T \mathbf{x}_m(k) \\ &= \left(\underline{\mathbf{g}}_l^{\text{I}} \right)^T [\mathbf{x}_1^T(k), \dots, \mathbf{x}_M^T(k)]^T,\end{aligned}\tag{3.29}$$

where $\underline{\mathbf{g}}_l^{\text{I}}$ is the M -channel concatenated inverse filter corresponding to $\tau = l - 1$. The recovered signal $\hat{s}(n)$ can be obtained by concatenating $\hat{\underline{s}}^{\text{I}}(k, l)$, $k = 0, \dots, K - 1$. This implies that when MCEQ-T is equivalent to MINT with $\tau = l - 1$, $s(n)$ is recovered using a single set of M inverse filters $\underline{\mathbf{g}}_l^{\text{I}}$, corresponding to the l th subspace.

As described in Section 3.2, a properly selected delay $0 \leq \tau < L$ should be incorporated in $\mathbf{d}$ so as to minimize the least-squared error in the solution of (3.5). It has been shown in [210] that the optimal modeling delay τ_{opt} depends on the phase delay of the AIRs. An arbitrarily chosen τ may not guarantee improved equalization performance. To elaborate the importance of τ and its effect on MINT, the equalization performance was evaluated using $\hat{\mathbf{g}}$ computed for $0 \leq \tau \leq L_d - 1$. In this simulation, $M = 5$ synthetic AIRs, each of length $L_h = 512$, were generated using the method of images [215] at $f_s = 8$ kHz. Three sets of AIRs were generated inside a room of dimension $6 \text{ m} \times 5 \text{ m} \times 3 \text{ m}$ for source-sensor distance of $r = 0.3 \text{ m}$, 3.2 m and 7.1 m . To model equalization of imperfect AIRs, the estimated AIRs were then simulated by perturbing $\mathbf{h}_m$ as

$$\hat{\mathbf{h}}_m = (\mathbf{I}_{L_h \times L_h} + \boldsymbol{\epsilon}_m) \mathbf{h}_m,\tag{3.30}$$

where $\boldsymbol{\epsilon}_m = \text{diag}(\epsilon_{m,0}, \dots, \epsilon_{m,L_h-1})$ such that

$$\mathcal{M}_m = 10 \log_{10} \sigma_{\epsilon}^2\tag{3.31}$$

Fig. 3.4: (a) Variation of $e(\tau)$ with τ , and (b) variation of $\rho(l)$ with l , for different AIR sets.

is the system mismatch and $\epsilon_{m,i}$ is a zero-mean white Gaussian random variable of variance σ_ϵ^2 . In this illustrative example, $\mathcal{M}_m = -40$ dB, $L_g = L_c = 128$ giving $L_d = L_g + L_h - 1 = 639$ and $\hat{\mathbf{g}}$ was computed from $\hat{\mathbf{H}}$ using $\mathbf{d}$ with a modeling delay τ and (3.7). For each set of AIRs, the Euclidean distance between the target response and the reconstructed response, i.e.,

$$e(\tau) = 10 \log_{10} \|\mathbf{d} - \mathbf{H}^T \hat{\mathbf{g}}\|_2^2 \text{ dB}, \quad (3.32)$$

was computed for $0 \leq \tau \leq L_d - 1$ in order to quantify the equalization of $\mathbf{H}$.

It can be seen from Fig. 3.4(a) that the optimal value of τ is dependent on the AIRs. More specifically, for the set of AIRs with $r = 0.3$ m, $e(\tau) = -45$ dB is the lowest when $\tau = 29$. However this choice of τ will increase $e(\tau)$ significantly by 70 dB and 85 dB when $r = 3.2$ m and 7.1 m, respectively. In addition, for an arbitrarily chosen $\tau = L_h/2$, the set of AIRs for $r = 7.1$ m achieves the lowest $e(\tau) \approx -4$ dB compared to $e(\tau) \approx 26$ dB and 30 dB for the case of $r = 0.3$ m and 3.2 m, respectively. These results explicitly show that the selection of τ is crucial in the process of AIR equalization when $L_g < L_h$. In practice, $\mathbf{H}$ is unknown and hence evaluating $e(\tau)$ to obtain an optimal τ corresponding to the lowest $e(\tau)$ is not possible.

Defining the concatenated (estimated) inverse filters as

$$\begin{aligned}\hat{\underline{\mathbf{G}}}^I &= [\hat{\underline{\mathbf{g}}}_1^I, \dots, \hat{\underline{\mathbf{g}}}_L^I] \\ &= (\hat{\mathbf{H}}^T)^+ \mathbf{I}_{L \times L},\end{aligned}\tag{3.33}$$

it will now be shown why MCEQ-T outperforms MINT with an inappropriate choice of τ . It can be noted that $\mathbf{I}_{L \times L}$ consists of all L possible delays and $\hat{\underline{\mathbf{g}}}^I(k)$ is obtained using multiple sets of filters $\hat{\underline{\mathbf{g}}}_l^I, l = l_s, \dots, l_e$, with each set corresponding to $\tau = l - 1$ across the M channels. Application of multiple delays in MCEQ-T results in a smaller mean-squared error (MSE) than MINT with an inappropriate choice of τ since, in MCEQ-T, each coefficient of $\mathbf{s}(k)$ is estimated using a dedicated set of inverse filters. While the estimation error is common across all the elements in $\mathbf{s}(k)$ for MINT due to the application of a single set of inverse filters, the error associated with each element is different for MCEQ-T. Therefore, MCEQ-T is expected to yield a lower MSE compared to MINT with an inappropriate τ .

To quantify the improvement of MCEQ-T over MINT that uses an inappropriate τ in the presence of BCI error, let

$$\hat{\mathbf{H}} = \mathbf{H} + \mathcal{E},\tag{3.34}$$

where $\mathcal{E} = [\mathcal{E}_1^T, \dots, \mathcal{E}_M^T]^T$ is, similar to $\mathbf{H}$, a $ML_g \times L$ concatenated filtering matrix constructed from the estimation error. For MINT, with $N = 1$ and defining $b' = k + L_x - l + 1$, (3.26) can be expressed as

$$\begin{aligned}\hat{s}(b') &= \sum_{m=1}^M (\hat{\underline{\mathbf{g}}}_{m,l}^I)^T [(\hat{\mathbf{H}}_m - \mathcal{E}_m) \mathbf{s}(k)] \\ &= s(b') - \wp(b'),\end{aligned}\tag{3.35}$$

where

$$\begin{aligned}\wp(b') &= \sum_{m=1}^M (\hat{\underline{\mathbf{g}}}_{m,l}^I)^T \mathcal{E}_m \mathbf{s}(k) \\ &= (\hat{\underline{\mathbf{g}}}_l^I)^T \mathbf{\check{s}}(k)\end{aligned}\tag{3.36}$$

Fig. 3.5: Filter-norm ρ for MINT and MCEQ-T.

denotes the reconstruction error, $\check{\mathbf{s}}(k) = [\check{\mathbf{s}}_1^T(k), \dots, \check{\mathbf{s}}_M^T(k)]^T$ and $\check{\mathbf{s}}_m(k) = \mathcal{E}_m \mathbf{s}(k)$. For MINT, the mean error corresponding to equalization of $\mathbf{s}(n)$ (using a specific τ which results in a $\hat{\mathbf{g}}_l^I$ where $l = \tau + 1$) is given by

$$\mathbb{E}_{\text{MINT}} = \left(\hat{\mathbf{g}}_{\tau+1}^I \right)^T \mathcal{E} E[\mathbf{s}(k)], \quad (3.37)$$

where $E(\cdot)$ denotes the expectation operator. As opposed to the above, for MCEQ-T with $N = l_e - l_s + 1$ sets of inverse filters,

$$\mathbb{E}_{\text{MCEQ-T}} = \frac{1}{N} \sum_{l=l_s}^{l_e} \hat{\mathbf{g}}_l^I \mathcal{E} E[\mathbf{s}(k)]. \quad (3.38)$$

Denoting $E[\mathbf{s}(k)] = \mathbf{v}$ for clarity of presentation,

$$\begin{aligned} \mathbb{E}_{\text{MINT}} &= \left(\hat{\mathbf{g}}_{\tau+1}^I \right)^T \mathcal{E} \mathbf{v}, \\ \mathbb{E}_{\text{MCEQ-T}} &= \left(\hat{\mathbf{g}}^I \right)^T \mathcal{E} \mathbf{v}, \end{aligned} \quad (3.39)$$

where

$$\hat{\mathbf{g}}^I = \frac{1}{N} \sum_{l=l_s}^{l_e} \hat{\mathbf{g}}_l^I. \quad (3.40)$$

To show that MCEQ-T can achieve a lower MSE than MINT, $\rho = \left\| \hat{\mathbf{g}}_l^I \right\|_2$ and $\left\| \hat{\mathbf{g}}^I \right\|_2$ were plotted in Fig. 3.5 for MINT and MCEQ-T, respectively. In this simulation, the same set of AIRs corresponding to $r = 7.1$ m as described after (3.30) were used. The horizontal lines which denote ρ for $\hat{\mathbf{g}}^I$ also show the range of l used for MCEQ-T. It can be seen from this result that $\hat{\mathbf{g}}^I$ achieves a lower energy than $\hat{\mathbf{g}}_l^I$ with an inadequate choice of τ . Therefore, MCEQ-T can achieve a lower MSE compared to MINT with an

inappropriate choice of τ .

3.4.3 Equalization using optimal filter (MCEQ-T_{opt})

Using the inherent property of MCEQ-T, it is also possible to derive a *single* set (out of the L sets) of M inverse filters, from $\hat{\mathbf{G}}^T$, that yields the minimum equalization error. Compared to performing an iterative search for the optimal τ using MINT, the proposed algorithm is a better implementation.

It will be now shown that the optimal filter $\hat{\mathbf{g}}_{l_{\text{opt}}}^I$ corresponding to the τ_{opt} achieves the smallest energy and this is in accordance with results presented in [6, 176]. Letting $\mathbf{d}_{l_{\text{opt}}}$ denote the optimal target response, the least-squared error obtained with an arbitrarily chosen $\hat{\mathbf{g}}_l^I$ is then given by

$$\left\| \mathbf{H}^T \hat{\mathbf{g}}_l^I - \mathbf{d}_{l_{\text{opt}}} \right\|_2^2 = \left\| \mathbf{H}^T \left(\hat{\mathbf{g}}_l^I - \hat{\mathbf{g}}_{l_{\text{opt}}}^I \right) + \left(\mathbf{H}^T \hat{\mathbf{g}}_{l_{\text{opt}}}^I - \mathbf{d}_{l_{\text{opt}}} \right) \right\|_2^2. \quad (3.41)$$

For the optimal solution $\hat{\mathbf{g}}_{l_{\text{opt}}}^I$ we achieve the least squared error $\mathbf{H}^T \hat{\mathbf{g}}_{l_{\text{opt}}}^I - \mathbf{d}_{l_{\text{opt}}}$, which is in the null space of $\mathbf{H}$ [35] giving

$$\mathbf{H} \left(\mathbf{H}^T \hat{\mathbf{g}}_{l_{\text{opt}}}^I - \mathbf{d}_{l_{\text{opt}}} \right) = \mathbf{0}_{ML_g \times 1}. \quad (3.42)$$

As a result,

$$\left[\mathbf{H}^T \left(\hat{\mathbf{g}}_l^I - \hat{\mathbf{g}}_{l_{\text{opt}}}^I \right) \right]^T \left(\mathbf{H}^T \hat{\mathbf{g}}_{l_{\text{opt}}}^I - \mathbf{d}_{l_{\text{opt}}} \right) = 0, \quad (3.43)$$

which implies that $\mathbf{H}^T \left(\hat{\mathbf{g}}_l^I - \hat{\mathbf{g}}_{l_{\text{opt}}}^I \right) \perp \left(\mathbf{H}^T \hat{\mathbf{g}}_{l_{\text{opt}}}^I - \mathbf{d}_{l_{\text{opt}}} \right)$. Therefore

$$\left\| \mathbf{H}^T \hat{\mathbf{g}}_l^I - \mathbf{d}_{l_{\text{opt}}} \right\|_2^2 = \left\| \mathbf{H}^T \left(\hat{\mathbf{g}}_l^I - \hat{\mathbf{g}}_{l_{\text{opt}}}^I \right) \right\|_2^2 + \left\| \mathbf{H}^T \hat{\mathbf{g}}_{l_{\text{opt}}}^I - \mathbf{d}_{l_{\text{opt}}} \right\|_2^2. \quad (3.44)$$

For $\hat{\mathbf{g}}_l^I \neq \hat{\mathbf{g}}_{l_{\text{opt}}}^I$,

$$\left\| \mathbf{H}^T \hat{\mathbf{g}}_{l_{\text{opt}}}^I - \mathbf{d}_{l_{\text{opt}}} \right\|_2^2 < \left\| \mathbf{H}^T \hat{\mathbf{g}}_l^I - \mathbf{d}_{l_{\text{opt}}} \right\|_2^2 \quad (3.45)$$

since $\mathbf{H}^T \left(\hat{\mathbf{g}}_l^I - \hat{\mathbf{g}}_{l_{\text{opt}}}^I \right) \neq \mathbf{0}_{L \times 1}$. From (3.45) it can be deduced that, for the optimal least-square solution, $\left\| \hat{\mathbf{g}}_{l_{\text{opt}}}^I \right\|_2^2 < \left\| \hat{\mathbf{g}}_l^I \right\|_2^2, l = 1, \dots, L, l \neq l_{\text{opt}}$. Hence the inverse filter corresponding to τ_{opt} has the lowest energy and it is therefore expected that $\hat{\mathbf{g}}_{l_{\text{opt}}}^I$ can achieve the smallest equalization error.

The relationship between the estimation error and filter energy is now investigated by plotting the inverse-filter norm $\rho(l) = \left\| \hat{\mathbf{g}}_l^I \right\|_2$ in Fig. 3.4(b) for the same sets of AIRs described in Section 3.4.2.

Comparing Figs. 3.4(a) and 3.4(b), it can be seen that the plots look almost identical, except for the scale, i.e., the range of τ for which $e(\tau)$ and $\rho(l)$ is minimum is the same. This implies that the delay associated with the l' th subspace giving the smallest estimation error $e(\tau)$ is equivalent to that giving the smallest $\rho(l)$, and the corresponding inverse filter set

$$\underline{\hat{\mathbf{g}}}_{l'}^I = \left[\left(\underline{\hat{\mathbf{g}}}_{1,l'}^I \right)^T, \dots, \left(\underline{\hat{\mathbf{g}}}_{M,l'}^I \right)^T \right]^T \quad (3.46)$$

achieves the best equalization performance. In MCEQ-T_{opt}, as opposed to MCEQ-T where each element of $\mathbf{s}(k)$ is estimated using a dedicated filter, the full length $\mathbf{s}(n)$ is estimated using an optimal set of filters $\underline{\hat{\mathbf{g}}}_{l'}^I$ as

$$\hat{\mathbf{s}}(n) = \sum_{m=1}^M \left(\underline{\hat{\mathbf{G}}}_{m,l'}^I \right)^T \mathbf{x}_m(n), \quad (3.47)$$

where $\underline{\hat{\mathbf{G}}}_{m,l'}^I$ is the filtering matrix of $\underline{\hat{\mathbf{g}}}_{m,l'}^I$ and $\rho(l) > \rho(l'), l \neq l'$. While MCEQ-T achieves good equalization performance via the application of multiple sets of filters MCEQ-T_{opt} achieves the same by applying a single set of filters corresponding to an optimal modeling delay that gives the minimum equalization error.

To further illustrate the equalization performance of MCEQ-T and MCEQ-T_{opt}, an example using five recorded AIRs obtained from the MARDY database [216] is provided. The AIRs were re-sampled to 16 kHz and truncated to $L_h = 1600$. The equalization performance of MINT, subband MCEQ (SB-MCEQ) [195], and MCEQ-T is compared here. Similar to [195], $L_g = L_c$, where L_c is defined in (3.8), giving $L_g = 400$ and $L = 1999$. For SB-MCEQ, the decimation factor $N_d = 24$, number of subbands $N_s = 32$, and the length of prototype filter $L_{pr} = 512$ [195]. As before, a system mismatch $\mathcal{M}_m = -40$ dB is used in this simulation.

The performance of these algorithms is illustrated by computing the equalized response $\hat{\mathbf{d}}$ using $\mathbf{h}_m$ and the inverse filters. These inverse filters are estimated using $\hat{\mathbf{h}}_m$ for each algorithm. For MINT, $\hat{\mathbf{g}}$ was estimated with $\tau = \lceil L_h/2 \rceil$ and

$$\hat{\mathbf{d}} = \sum_{m=1}^M \mathbf{H}_m^T \hat{\mathbf{g}}_m. \quad (3.48)$$

For MCEQ-T, $\left[\underline{\hat{\mathbf{g}}}_{200}^I, \dots, \underline{\hat{\mathbf{g}}}_{300}^I \right]$ was first estimated using $\underline{\hat{\mathbf{H}}}^I$ where $\underline{\hat{\mathbf{H}}}^I = \hat{\mathbf{H}}$ was formed from $\hat{\mathbf{h}}_m, m =$

$1, \dots, M$. For the computation of $\hat{\mathbf{d}}^I$, each $\mathbf{h}_m$ was then partitioned into frames of length $L_x = L_g$ after zero padding to a length $(\lceil L_h/N \rceil - 1)N + L_g$, where N is the number of subspaces used for reconstruction in this example. Frame-wise equalization was then performed using

$$\begin{aligned} \hat{d}^I(k, l) &= \left(\hat{\mathbf{g}}_l^I \right)^T [\mathbf{h}_1^T(k), \dots, \mathbf{h}_M^T(k)]^T, \quad l = 200, \dots, 300, \\ \hat{\mathbf{d}}^I(k) &= \sum_{l=l_s}^{l_e} \hat{d}^I(k, l) \mathbf{i}_l, \end{aligned} \quad (3.49)$$

where $\mathbf{h}_m(k)$ and $\hat{\mathbf{d}}^I(k)$ denote the k th frame of $\mathbf{h}_m$ and $\hat{\mathbf{d}}^I$, respectively. The equalized response $\hat{\mathbf{d}}^I$ was reconstructed using $\hat{\mathbf{d}}^I(k)$ with $N = 101$, similar to how $\hat{s}(k)$ is reconstructed after (3.23). Hence, in effect, $\hat{\mathbf{g}}_l^I, l = 200, \dots, 300$, corresponding to $199 \leq \tau \leq 299$ were used for reconstruction. For MCEQ-T_{opt}, equalization was achieved using a single set of inverse filters $\hat{\mathbf{g}}_{l'}^I$ estimated using MCEQ-T, that yielded the smallest $\rho(l)$, i.e., the optimal filter set.

Performance of the algorithms can be evaluated by noting that, in the case of perfect equalization, the equalized response $\hat{\mathbf{d}}$ should be a Kronecker delta function with magnitude response of 0 dB over the entire frequency range. The performance can be quantified using the spectral flatness measure (SFM) $-\infty \leq \phi \leq 0$ dB [217] defined as the ratio of the geometric mean of a spectrum to its arithmetic mean such that $\phi = 0$ dB indicates a flat spectrum with uniformly distributed spectral power. The performance of the algorithms is also quantified by employing the sparseness measure of $\hat{\mathbf{d}}$ defined by [218]

$$\xi = \frac{L_{\hat{\mathbf{d}}}}{L_{\hat{\mathbf{d}}} - \sqrt{L_{\hat{\mathbf{d}}}}} \left(1 - \frac{\|\hat{\mathbf{d}}\|_1}{\sqrt{L_{\hat{\mathbf{d}}}} \|\hat{\mathbf{d}}\|_2} \right), \quad (3.50)$$

where $L_{\hat{\mathbf{d}}}$ is the length of $\hat{\mathbf{d}}$. Note that $0 \leq \xi \leq 1$, and that $\xi = 1$ when $\hat{\mathbf{d}}$ is a Kronecker delta function and $\xi = 0$ if all elements in $\hat{\mathbf{d}}$ have the same magnitude. Therefore, $\xi = 1$ and $\phi = 0$ dB imply that the equalization algorithm has perfect performance.

Figures 3.6(a) – (d) show the equalization performance of MINT, SB-MCEQ, MCEQ-T and MCEQ-T_{opt}, respectively. It can be observed from the magnitude frequency response that, being close to the desired 0 dB across all frequencies, MCEQ-T and MCEQ-T_{opt}, with $\phi = -0.02$ dB and -0.004 dB,

Fig. 3.6: The frequency-domain magnitude response of equalized response $\hat{\mathbf{d}}$. (a) MINT. (b) SB-MCEQ. (c) MCEQ-T. (d) MCEQ-T_{opt}. (e) MCEQ-KL.

respectively, outperform MINT and SB-MCEQ, which give $\phi = -0.54$ dB and -0.03 dB, respectively. This is further evident from $\xi = 0.749, 0.952, 0.963$, and 0.981 for MINT, SB-MCEQ, MCEQ-T and MCEQ-T_{opt}, respectively.

3.4.4 MCEQ-KL

Speech signals received within an enclosed space are often contaminated with reverberation as well as additive noise. One possible approach for mitigating such undesired effects is to perform dereverberation and noise reduction. It has been shown in [7] that the KLT yields the best noise reduction performance compared to the Fourier, cosine, Hadamard, and identity transforms. Hence, it is desirable to perform

MCEQ in the KLT domain since this provides a useful platform for noise reduction as well as dereverberation.

To describe MCEQ-KL, let

$$\mathbf{R}(k) = E \left[\tilde{\mathbf{X}}(k) \tilde{\mathbf{X}}^T(k) \right] \quad (3.51)$$

be defined as the $L \times L$ correlation matrix of the zero-appended received signal $\tilde{\mathbf{X}}(k) = [\mathbf{X}(k), \mathbf{0}_{M \times (L-L_x)}]^T$, where the $M \times L_x$ matrix $\mathbf{X}(k) = [\mathbf{x}_1(k), \dots, \mathbf{x}_M(k)]^T$. The correlation matrix $\mathbf{R}(k)$ can be diagonalized using

$$\mathbf{Q}^T \mathbf{R}(k) \mathbf{Q} = \mathbf{\Lambda}, \quad (3.52)$$

where $\mathbf{Q} = [\mathbf{q}_1, \dots, \mathbf{q}_L]$ is a $L \times L$ matrix containing the $L \times 1$ eigenvectors $\mathbf{q}_l$ and the diagonal elements of the $L \times L$ matrix $\mathbf{\Lambda}$ correspond to the eigenvalues of $\mathbf{R}(k)$. Letting $\mathbf{U} = \mathbf{Q}$ for the proposed MCEQ-KL, the KLT-domain AIR filtering matrix is obtained as

$$\underline{\mathbf{H}}_m^{\text{KL}} = \mathbf{Q}^T \begin{bmatrix} \mathbf{H}_m \\ \mathbf{0}_{(L-L_x) \times L} \end{bmatrix} \mathbf{Q}, \quad (3.53)$$

where $(\cdot)^{\text{KL}}$ denotes for the KLT domain. The MCEQ-KL algorithm achieves equalization by processing each frame $\mathbf{x}_m(k)$ of length L_x such that the KLT coefficients of the k th frame, for the m th channel, are given by

$$\begin{aligned} \underline{\mathbf{x}}_m^{\text{KL}}(k) &= \mathbf{Q}^T [\mathbf{x}_m^T(k), \mathbf{0}_{1 \times L-L_x}]^T \\ &= [\underline{x}_m^{\text{KL}}(k, 1), \dots, \underline{x}_m^{\text{KL}}(k, L)]^T, \end{aligned} \quad (3.54)$$

where $\underline{x}_m^{\text{KL}}(k, l) = \mathbf{q}_l^T [\mathbf{x}_m^T(k), \mathbf{0}_{1 \times L-L_x}]^T$.

It may be noted that the appendage of $L - L_x$ columns of zeros to $\mathbf{X}(k)$ to form $\tilde{\mathbf{X}}(k)$ shown above

(3.52) will result in $L - L_x$ rows of zeros in $\underline{\mathbf{H}}_m^{\text{KL}}$ and $L - L_x$ zeros in $\underline{\mathbf{x}}_m^{\text{KL}}(k)$ since

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}'_{L_x \times L_x} & \mathbf{0}_{L_x \times (L-L_x)} \\ \mathbf{0}_{(L-L_x) \times L_x} & \mathbf{I}_{(L-L_x) \times (L-L_x)} \end{bmatrix}, \quad (3.55)$$

where $\mathbf{Q}'_{L_x \times L_x}$ is a matrix containing eigenvectors of the correlation matrix formed by $\mathbf{X}(k)$. Hence, the dimension of $\underline{\mathbf{H}}_m^{\text{KL}}$ and $\underline{\mathbf{x}}_m^{\text{KL}}(k)$ can be reduced to $L_x \times L$ and $L_x \times 1$ respectively by truncating the non-essential zeros. Thus, with $L_x = L_g$, it can be written that

$$\underline{\mathbf{H}}_m^{\text{KL}} = (\mathbf{Q}')^T \mathbf{H}_m \mathbf{Q}, \quad (3.56)$$

$$\underline{\mathbf{x}}_m^{\text{KL}}(k) = (\mathbf{Q}')^T \mathbf{x}_m(k). \quad (3.57)$$

Then the $ML_g \times L$ concatenated MCEQ-KL inverse filters given by

$$\begin{aligned} \underline{\mathbf{G}}^{\text{KL}} &= [\underline{\mathbf{g}}_1^{\text{KL}}, \dots, \underline{\mathbf{g}}_L^{\text{KL}}] \\ &= [(\underline{\mathbf{H}}^{\text{KL}})^T]^+ \mathbf{I}_{L \times L} \end{aligned} \quad (3.58)$$

are obtained, where the $ML_g \times 1$ inverse filter vector $\underline{\mathbf{g}}_l^{\text{KL}} = \left[(\underline{\mathbf{g}}_{1,l}^{\text{KL}})^T, \dots, (\underline{\mathbf{g}}_{M,l}^{\text{KL}})^T \right]^T$ corresponds to the l th subspace, and $\underline{\mathbf{H}}^{\text{KL}} = \left[(\underline{\mathbf{H}}_1^{\text{KL}})^T, \dots, (\underline{\mathbf{H}}_M^{\text{KL}})^T \right]^T$ is a $ML_g \times L$ matrix with $\underline{\mathbf{H}}_m^{\text{KL}}$ being defined in (3.56). Finally, defining $\underline{\mathbf{s}}^{\text{KL}}(k) = [\underline{s}^{\text{KL}}(k, 1), \dots, \underline{s}^{\text{KL}}(k, L)]^T$ as the KLT-domain representation of $\mathbf{s}(k)$, the KLT coefficients $\underline{s}^{\text{KL}}(k, l)$ of the clean speech signal can then be recovered using

$$\hat{\underline{s}}^{\text{KL}}(k, l) = \sum_{m=1}^M \left(\underline{\mathbf{g}}_{m,l}^{\text{KL}} \right)^T \underline{\mathbf{x}}_m^{\text{KL}}(k), \quad l = 1, \dots, L. \quad (3.59)$$

The desired k th frame of length L in the time domain is synthesized as a $L \times 1$ vector by

$$\begin{aligned} \hat{\mathbf{s}}^{\text{KL}}(k) &= \sum_{l=1}^L \hat{\underline{s}}^{\text{KL}}(k, l) \mathbf{q}_l \\ &= \mathbf{Q} \hat{\underline{\mathbf{s}}}^{\text{KL}}(k), \end{aligned} \quad (3.60)$$

Fig. 3.7: Equalization using MCEQ-KL.

where $\hat{\underline{s}}^{\text{KL}}(k, l)$ and $\hat{\underline{s}}^{\text{KL}}(k) = [\hat{s}^{\text{KL}}(kN + L_x - 1), \dots, \hat{s}^{\text{KL}}(kN + L_x - L)]^T$ are the estimates of $\underline{s}^{\text{KL}}(k, l)$ and $\underline{s}(k)$, respectively, and $\hat{\underline{s}}^{\text{KL}}(k) = [\hat{\underline{s}}^{\text{KL}}(k, 1), \dots, \hat{\underline{s}}^{\text{KL}}(k, L)]^T$.

In the KLT domain, $\underline{\mathbf{g}}_l^{\text{KL}}$, containing M inverse filters each of dimension $L_g \times 1$, is designed for each subspace defined by $\mathbf{q}_l$, $l = 1, \dots, L$. This implies that each $\hat{\underline{s}}^{\text{KL}}(k, l)$ is recovered using a set of dedicated inverse filters $\underline{\mathbf{g}}_{m,l}^{\text{KL}}$, $m = 1, \dots, M$, each of length L_g . Figure 3.7 illustrates the proposed MCEQ-KL algorithm. Similar to MCEQ-T, the performance of MCEQ-KL was evaluated as described in Section 3.4.3. It can be noted from the magnitude frequency response given in Fig. 3.6(e) that, similar to MCEQ-T, MCEQ-KL with $\phi = -0.02$ dB and $\xi = 0.964$ outperforms both MINT and SB-MCEQ.

An application of MCEQ-KL for achieving equalization in conjunction with denoising is now formulated. In a noisy and reverberant environment,

$$\begin{aligned} \mathbf{x}_m(k) &= \mathbf{H}_m \mathbf{s}(k) + \mathbf{v}_m(k) \\ &= \mathbf{y}_m(k) + \mathbf{v}_m(k), \end{aligned} \quad (3.61)$$

where $\mathbf{H}_m$ has been defined in (3.3), $\mathbf{y}_m(k) = [y_m(kN + L_x - 1), \dots, y_m(kN)]^T$ and $\mathbf{v}_m(k) = [v_m(kN + L_x - 1), \dots, v_m(kN)]^T$. Noise reduction in the KLT domain can be achieved by applying an

optimal Wiener filter given by [7]

$$\underline{w}^{\text{KL}}(k, l) = 1 - \frac{E \left\{ \hat{\underline{v}}_m^{\text{KL}}(k, l) [\hat{\underline{v}}_m^{\text{KL}}(k, l)]^* \right\}}{E \left\{ \underline{x}_m^{\text{KL}}(k, l) [\underline{x}_m^{\text{KL}}(k, l)]^* \right\}}, \quad (3.62)$$

where $(.)^*$ denotes complex conjugate,

$$\hat{\underline{v}}_m^{\text{KL}}(k, l) = \mathbf{q}_l^T \hat{\mathbf{v}}_m(k), \quad l = 1, \dots, L_x, \quad (3.63)$$

$$\hat{\underline{\mathbf{v}}}_m^{\text{KL}}(k) = \left[\hat{\underline{v}}_m^{\text{KL}}(k, 1), \dots, \hat{\underline{v}}_m^{\text{KL}}(k, L_x) \right]^T \quad (3.64)$$

and $\hat{\mathbf{v}}_m(k)$ is an estimate of $\mathbf{v}_m(k)$. The denoised signal $\hat{\underline{\mathbf{y}}}_m^{\text{KL}}(k)$ can be obtained using

$$\begin{aligned} \hat{\underline{y}}_m^{\text{KL}}(k, l) &= \underline{w}^{\text{KL}}(k, l) \underline{x}_m^{\text{KL}}(k, l), \quad l = 1, \dots, L_x, \\ \hat{\underline{\mathbf{y}}}_m^{\text{KL}}(k) &= \left[\hat{\underline{y}}_m^{\text{KL}}(k, 1), \dots, \hat{\underline{y}}_m^{\text{KL}}(k, L_x) \right]^T. \end{aligned} \quad (3.65)$$

By applying MCEQ-KL and the Wiener filter, it is now possible to achieve denoising and channel equalization. Substituting (3.65) for $\underline{\mathbf{x}}_m^{\text{KL}}(k)$ in (3.59) and using (3.60), the k th frame of reconstructed speech signal is given by

$$\hat{\mathbf{s}}^{\text{KL}}(k) = \sum_{l=1}^L \left\{ \sum_{m=1}^M \left(\underline{\mathbf{g}}_{m,l}^{\text{KL}} \right)^T \left[\underline{\mathbf{w}}^{\text{KL}}(k) \odot \hat{\underline{\mathbf{x}}}_m^{\text{KL}}(k) \right] \right\} \mathbf{q}_l, \quad (3.66)$$

where $\odot$ denotes the Hadamard product and $\underline{\mathbf{w}}^{\text{KL}}(k) = [\underline{w}^{\text{KL}}(k, 1), \dots, \underline{w}^{\text{KL}}(k, L_x)]^T$. The improvement in the equalization performance achieved using this proposed technique will be shown in the next section.

3.5 Simulation results

In this section, the performance of the proposed algorithms in terms of speech dereverberation is compared with that of MINT. Similar to that used to generate Fig. 3.6, a segment of $N = 101$ samples from each frame $k = 0, \dots, K - 1$ was used during reconstruction.

Fig. 3.8: SRR_{seg} performance of the proposed algorithms for different L_g , as compared to MINT.

3.5.1 Effect of L_g

The effect of L_g on dereverberation performance is first investigated. Synthetic AIRs with reverberation time $T_{60} = L_h/f_s$ were generated using the method of images [215]. Twenty sets of five AIRs were simulated inside a $6 \text{ m} \times 5 \text{ m} \times 3 \text{ m}$ room at $f_s = 16 \text{ kHz}$ for a linear array of $M = 5$ microphones spaced 0.1 m apart. A 0.5 s long male speech signal was convolved with the AIRs, and zero-mean white Gaussian noise (WGN) was added to each reverberant signal, giving an $\text{SNR} = 30 \text{ dB}$. With system mismatch $\mathcal{M}_m = -40 \text{ dB}$, equalization was then performed on these noisy reverberant signals using the equalization algorithms.

Here, dereverberation was achieved with $400 \leq L_g \leq 800$ for AIRs of length $L_h = 1600$. For each L_g , the segmental signal-to-reverberant ratio (SRR_{seg}) [4] of the equalized speech signal, averaged over the twenty sets of AIRs, was computed for each algorithm and plotted in Fig. 3.8. For good equalization performance, a high SRR_{seg} is expected. It can be seen that the dereverberation performance improves with L_g and these observations are consistent with that presented in [6]. It may be noted that, under noisy conditions and in the presence of AIR mismatch, the performance of MINT with $\tau = L_h/2$ degrades significantly for $L_g < L_h$. On the other hand, the proposed algorithms and MINT with $\tau = 0$ exhibit better performance with the proposed algorithms achieving higher SRR_{seg} than MINT.

3.5.2 Effect of L_h

The performance of the proposed algorithms, MINT and SB-MCEQ for various L_h is also assessed using the above simulation setup. Since $L_g = \lceil L_h/2 \rceil$ gives better performance as illustrated in Fig. 3.8,

Fig. 3.9: SRR_{seg} of the proposed algorithms for different L_h , as compared to MINT and SB-MCEQ.

$L_g = \lceil L_h/2 \rceil$ is chosen for all subsequent simulations in this section henceforth. Also, similar to [195], [219], a zero modeling delay is used for MINT since, as shown in Fig. 3.9, $\tau = 0$ achieves better equalization compared to $\tau = L_h/2$. Figure 3.9 shows the average SRR_{seg} for $600 \leq L_h \leq 1600$ in which MINT outperforms SB-MCEQ. These results show that, for all L_h , MCEQ-T, MCEQ-T_{opt}, and MCEQ-KL achieve better dereverberation performance over MINT, with an average improvement of 9 dB, 10.3 dB, and 4.8 dB in SRR_{seg} , respectively.

3.5.3 Robustness to AIR estimation errors

Perfect estimation of AIRs is still a challenging problem. Since, in practice, speech dereverberation is realized by equalizing inaccurate estimated AIRs, i.e., $\hat{\mathbf{h}}_m \neq \mathbf{h}_m$, it is desirable for equalization algorithms to be robust to such AIR inaccuracies. In this simulation, the robustness of MCEQ-T, MCEQ-T_{opt}, MCEQ-KL, SB-MCEQ and MINT to inaccurate AIR estimates is examined. As before, noise was added to achieve $\text{SNR} = 30$ dB and perfect knowledge of $L_h = 1600$ was assumed. A mismatch $\mathcal{M}_m$ was then introduced to $\mathbf{h}_m$ using (3.30) to obtain inaccurate AIR estimates $\hat{\mathbf{h}}_m$. The equalization filters were estimated from $\hat{\mathbf{h}}_m$ using the algorithms, from which equalization of the noisy $\mathbf{x}_m(n)$ was then performed. Comparing the SRR_{seg} performance of the algorithms in Fig. 3.10, it can be noted that SB-MCEQ exhibits the worst performance. The MCEQ-T algorithm yields an average performance improvement over MINT by approximately 8.3 dB, and 9.1 dB, respectively, for optimal delay filters and multiple filters. Under mismatch condition, MCEQ-KL also achieves an average SRR_{seg} improvement of 2.4 dB.

Fig. 3.10: Variation of SRR_{seg} with $\mathcal{M}_m$ for MINT, SB-MCEQ, MCEQ-KL, MCEQ-T, and MCEQ-T_{opt}.

3.5.4 Dereverberation

Speech dereverberation performance of the proposed algorithms and MINT is next compared using the recorded AIRs described in Section 3.4.3. Since SB-MCEQ achieves the lowest SRR_{seg} compared to other algorithms, as can be seen from the above results, it has not been included in this comparison. A 3.5 s long male speech signal was used as the source. Dereverberation was achieved with $\mathcal{M}_m = -40$ dB at $\text{SNR} = 30$ dB. Comparing Figs. 3.11(a) and (b) it can be seen that reverberation has affected the temporal as well as spectral structure of the original source signal giving an average $\text{SRR}_{\text{seg}} = 0.04$ dB across the channels. Referring to Fig. 3.11(c) it can be seen that although MINT with $\text{SRR}_{\text{seg}} = 14.7$ dB achieved some degree of dereverberation it could not successfully recover some of the high frequency spectral components. However the proposed algorithms achieve a better dereverberation performance yielding a recovered signal that is close to the original speech signal. The proposed MCEQ-T and MCEQ-T_{opt} achieve $\text{SRR}_{\text{seg}} = 27.5$ dB and 27.6 dB, respectively, while MCEQ-KL achieves $\text{SRR}_{\text{seg}} = 26.8$ dB.

3.5.5 Comparison of time-domain algorithms with RMCLS

Channel shortening (CS) techniques have been introduced to improve the robustness of MCEQ to system mismatch. In this section, the equalization performance of the proposed MCEQ-T and MCEQ-T_{opt} algorithms is compared with the recently proposed relaxed multichannel least squares (RMCLS) algorithm [181, 220], which is a CS-based approach, and MINT. Two recorded AIRs from the MARDY database [216], which were re-sampled to 8 kHz and truncated to $L_h = 2000$ were used in this sim-

Fig. 3.11: Speech dereverberation performance comparison: Waveform and spectrogram of (a) original source signal, (b) reverberant signal at channel 1, (c) signal recovered using MINT, (d) signal recovered using MCEQ-T, (e) signal recovered using MCEQ-T_{opt} and (f) signal recovered using MCEQ-KL.

Fig. 3.12: Equalization performance measured using EDCs for MINT, RMCLS, MCEQ-T and MCEQ-T_{opt}.

Table 3.1: PESQ scores.

Method	$\mathcal{M}_m = -40$ dB	$\mathcal{M}_m = -20$ dB
MINT	2.48	2.19
RMCLS	2.86	2.73
MCEQ-T	3.3	2.74
MCEQ-T _{opt}	3.36	3.1

ulation. Equalization was then achieved using the inverse filters estimated from $\hat{\mathbf{h}}_m, m = 1, 2$ with $\mathcal{M}_m = -20$ dB. A relaxation window length of $L_w = 50$ ms was chosen for RMCLS [220].

In Fig. 3.12, the energy decay curve (EDC) [4] of $\hat{\mathbf{d}}$ achieved with each algorithm, truncated to L_h samples, are shown along with the EDC of the recorded AIRs (averaged across the channels). This result shows that, in the presence of system mismatch, MINT achieves a lower decay rate of the EDC than that of the AIRs. This implies that MINT with an arbitrary choice of $\tau = 0$ fails to achieve equalization. The RMCLS algorithm achieves better suppression of the late reverberation, while the early reflections are not fully equalized. On the other hand, in addition to achieving better equalization of the early reverberation than existing techniques, the proposed MCEQ-T and MCEQ-T_{opt} algorithms achieve comparable suppression of the late reflections to RMCLS.

The perceptual quality of the equalized signals with the clean speech signal as the reference is also compared using PESQ [208] for the same set of recorded AIRs. Equalization was performed on the reverberant data with $\mathcal{M}_m = -40$ and -20 dB at $\text{SNR} = 40$ dB with $L_w = 50$ ms for RMCLS. The PESQ for signal corresponding to the first channel was computed as 2.52. From Table 3.1 it can be noted that MINT results in the lowest PESQ scores indicating further degradation in the perceptual quality. On the other hand, the proposed MCEQ-T and MCEQ-T_{opt} algorithms achieve the highest PESQ scores as a result of better equalization performance.

Fig. 3.13: Comparison of SRR_{seg} against SNR for MINT, MCEQ-KL only, and MCEQ-KL with the Wiener filter.

3.5.6 Equalization in noisy environment using MCEQ-KL with denoising filter

The equalization performance of MINT, and MCEQ-KL with a denoising filter is assessed in a noisy environment with $\mathcal{M}_m = -10$ dB. A relatively high $\mathcal{M}_m$ is chosen since it is challenging to estimate AIRs with high accuracy at low SNRs. A set of five synthetic AIRs with $L_h = 1600$ were generated using the method of images. Noisy speech signals were obtained by adding zero-mean WGN to the signals obtained after filtering by the AIRs. For each SNR, equalization was achieved using (a) MINT, (b) MCEQ-KL, and (c) MCEQ-KL and Wiener filter, as described in Section 3.4.4. To demonstrate the effect of denoising, the Wiener denoising filter described by (3.62) and (3.65) [7] was used.

The variation of SRR_{seg} with SNR for each algorithm is shown in Fig. 3.13 along with the SRR_{seg} of the unprocessed noisy reverberant signals averaged across the channels. It can be noted that at low SNRs equalization performance of the algorithms that do not incorporate denoising filter becomes worse in the presence of AIR estimation error. On the other hand, joint application of denoising and MCEQ-KL improves the equalization performance under low SNR conditions. For example, when $\text{SNR} = 10$ dB, MINT only achieves $\text{SRR}_{\text{seg}} = -5.5$ dB, whereas MCEQ-KL achieves -3.1 dB and 2.4 dB, respectively, without and with the denoising filter. When $\text{SNR} = 0$ dB, the SRR_{seg} values obtained for the above cases are -11.2 dB, -10.1 dB, and 0.4 dB, respectively. The MCEQ-KL algorithm with denoising thus achieves an improvement of 7.9 dB and 11.6 dB over MINT, respectively, for $\text{SNR} = 10$ dB and 0 dB. These results imply that the application of denoising and MCEQ-KL provides better equalization under noisy conditions.

3.6 Conclusion

Speech dereverberation via BCI and inverse filtering has become increasingly popular in recent years. This technique employs an MCEQ algorithm such as MINT to derive the equalization filters using the AIRs estimated by BCI algorithms. Although the popular MINT algorithm can achieve good equalization under ideal conditions, it is not robust to AIR estimation errors. Moreover, the equalization performance of MINT deteriorates if an appropriate modeling delay is not introduced to the target response. A robust MCEQ algorithm is essential for achieving good equalization.

Most of the existing MCEQ algorithms operate in the time domain. This formed the motivation behind formulating an MCEQ framework that can be applied in different transform domains to achieve equalization. Based on the proposed framework, three MCEQ algorithms (two in the time domain and one in the KLT domain) were introduced. By applying multiple sets of inverse filters, the proposed MCEQ-T outperforms MINT using an inappropriate modeling delay. In the proposed MCEQ-T_{opt} algorithm, a filter set having the lowest energy among L possible sets is selected as the optimal filter set and equalization is performed using this. Both the proposed time-domain algorithms achieve higher robustness to BCI errors and better performance when compared to MINT modeled with an inappropriate arbitrary delay and SB-MCEQ. The MCEQ-KL algorithm presents a platform to achieve equalization in conjunction with denoising. Simulation results show that the proposed algorithm with denoising can achieve better equalization performance than MCEQ alone.

Chapter 4

Subspace-based Adaptive Multichannel Equalization

4.1 Introduction

The MINT algorithm involves inversion of a filtering matrix. Since the proposed equalization algorithms in Chapter 3 are based on MINT relation, they also require matrix inversion. For a high-order AIR the dimension of the filtering matrix becomes very large which in turn makes these algorithms computationally expensive. This prevents them from being directly applied in practical scenarios where the length of a typical AIR is of several hundred samples.

An adaptive version of MINT (A-MINT) [188] has been proposed with the intention of reducing the computational complexity. However, A-MINT not only suffers from the shortcomings of MINT, but suffers from slow convergence. As described in the previous chapter, MCEQ-T can provide better equalization performance compared to MINT. In this chapter, in order to estimate the multiple sets of inverse filters with reduced complexity, an adaptive solution for the time-domain subspace-based multichannel equalization is proposed.

In the proposed time-domain adaptive MCEQ (TDA-MCEQ) algorithm, the inverse filters corresponding to each subspace are estimated adaptively similar to A-MINT. The cost function, for each subspace, is derived to minimize the error between a Kronecker delta function with a delay corresponding to that subspace and the convolution between the inverse filters for that subspace and the AIRs. As described in Chapter 3, the desired Kronecker delta functions with delays corresponding to the subspaces form a sparse identity matrix. Hence, to further improve the convergence performance of the proposed algorithm further the sparseness measure is introduced as a constraint to the cost function. Since this

chapter deals with the time domain only, unlike MCEQ-T in Chapter 3, $(\cdot)^{\dagger}$ and the underscore notations are omitted.

This chapter is organized as follow: In Section 4.2 the A-MINT algorithm is briefly reviewed and Section 4.3 introduces the proposed algorithms. Simulation results are presented in Section 4.4 followed by conclusion in Section 4.5.

4.2 Review of A-MINT

The MINT algorithm described in Chapter 3 computes the inverse filter using the relation

$$\mathbf{g} = (\mathbf{H}^T)^+ \mathbf{d}. \quad (4.1)$$

For high-order AIRs the dimension of $\mathbf{H}$ becomes large making the computation of $(\mathbf{H}^T)^+$ in (4.1) challenging. An adaptive MINT (A-MINT) algorithm has been proposed in [188] to estimate $\mathbf{g}$ with reduced complexity. This algorithm computes $\hat{\mathbf{g}}$ by minimizing the cost function

$$J_{\text{A-MINT}}(n) = \|\mathbf{d} - \mathbf{H}^T \hat{\mathbf{g}}(n)\|_2^2. \quad (4.2)$$

The gradient of A-MINT is given by

$$\begin{aligned} \nabla J_{\text{A-MINT}}(n) &= \frac{\partial J_{\text{A-MINT}}(n)}{\partial \hat{\mathbf{g}}(n)} \\ &= -2\mathbf{H}\mathbf{d} + 2\mathbf{H}\mathbf{H}^T \hat{\mathbf{g}}(n). \end{aligned} \quad (4.3)$$

Employing this gradient, the update equation of A-MINT is given by

$$\hat{\mathbf{g}}(n+1) = \hat{\mathbf{g}}(n) - \mu \nabla J_{\text{A-MINT}}(n)|_{\hat{\mathbf{g}}=\hat{\mathbf{g}}(n)}, \quad (4.4)$$

where μ is the step size.

Since A-MINT does not require the inversion of $\mathbf{H}$ it can be applied to achieve equalization of large-order AIRs with reduced complexity. In addition, the adaptive nature of the algorithm makes it suitable

for time-varying AIRs. However, it has been shown that A-MINT suffers from slow convergence [189, 190, 221]. In order to achieve faster convergence a sparseness constrained has been introduced to A-MINT in the recently proposed sparseness constrained MINT (SC-MINT) algorithm [221].

4.3 The proposed TDA-MCEQ algorithm

From (3.20) and (3.21), it can be noted that, for MCEQ-T,

$$\begin{aligned}\mathbf{H}^T \mathbf{g}_l &= \mathbf{d}_l, \\ \mathbf{H}^T \mathbf{G} &= \mathbf{D}.\end{aligned}\tag{4.5}$$

The inverse filters corresponding to the l th subspace is estimated using

$$\hat{\mathbf{g}}_l = (\mathbf{H}^T)^+ \mathbf{d}_l.\tag{4.6}$$

In TDA-MCEQ, for an assumed $\mathbf{H}$, an optimization problem

$$\min_{\mathbf{G}} \|\mathbf{D} - \mathbf{H}^T \mathbf{G}\|_F = \min_{\mathbf{G}} \sum_{l=1}^L \|\mathbf{d}_l - \mathbf{H}^T \mathbf{g}_l\|_2^2\tag{4.7}$$

is formulated from (4.5), where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. Since the terms in (4.7) corresponding to L subspaces are independent of each other, for simplicity, L cost functions can be defined, one for each subspace. These cost functions can be minimized from which the inverse filters for L subspaces are estimated. Similar to A-MINT, a cost function, for the l th subspace of the time domain, is defined as

$$J_l(n) = \|\mathbf{d}_l - \mathbf{H}^T \hat{\mathbf{g}}_l(n)\|_2^2,\tag{4.8}$$

Table 4.1: The Proposed TDA-MCEQ Algorithm

Initialization:
Initialize $\hat{\mathbf{G}}$
$\mathbf{D} = \mathbf{I}_{L \times L}$
Computation:
for $n = 0, 1, \dots$
$\mathbf{E}(n) = \mathbf{D} - \mathbf{H}^T \hat{\mathbf{G}}(n),$
where $\mathbf{E}(n) = [\mathbf{e}_1(n), \mathbf{e}_2(n), \dots, \mathbf{e}_L(n)]$
$\nabla J(n) = -2\mathbf{H}\mathbf{E},$
where $\nabla J(n) = [\nabla J_1(n), \nabla J_2(n), \dots, \nabla J_L(n)]$
$\hat{\mathbf{G}}(n+1) = \hat{\mathbf{G}}(n) - \mu \nabla J(n) _{\hat{\mathbf{G}}=\hat{\mathbf{G}}(n)},$
where $\hat{\mathbf{G}}(n) = [\hat{\mathbf{g}}_1(n), \hat{\mathbf{g}}_2(n), \dots, \hat{\mathbf{g}}_L(n)]$
end for

such that by minimizing $J_l(n)$, an estimate of the inverse filter for the l th subspace can be obtained, where $l = 1, \dots, L$. The gradient of this iterative algorithm is given by the $ML_g \times 1$ vector

$$\begin{aligned} \nabla J_l(n) &= \frac{\partial J_l(n)}{\partial \hat{\mathbf{g}}_l(n)} \\ &= -2\mathbf{H}\mathbf{e}_l(n), \end{aligned} \quad (4.9)$$

where the l th subspace error is defined by the $L \times 1$ vector

$$\mathbf{e}_l(n) = \mathbf{d}_l - \mathbf{H}^T \hat{\mathbf{g}}_l(n). \quad (4.10)$$

Subsequently, by employing (4.9), the TDA-MCEQ update equation for estimating the l th subspace inverse filter is given by

$$\hat{\mathbf{g}}_l(n+1) = \hat{\mathbf{g}}_l(n) - \mu \nabla J_l(n)|_{\hat{\mathbf{g}}_l=\hat{\mathbf{g}}_l(n)}. \quad (4.11)$$

The proposed TDA-MCEQ algorithm is summarized in Table 4.1.

Compared to MINT and MCEQ-T, the proposed TDA-MCEQ algorithm reduces the computational load by avoiding the need to solve the closed-form equation directly. Due to the application of multiple sets of filters, TDA-MCEQ can achieve similar equalization performance as MCEQ-T.

4.3.1 TDA-MCEQ-SP using the sparseness constraint

To improve the convergence rate of TDA-MCEQ, a sparseness constraint is introduced to the update equation given in (4.11). The resulting algorithm is named as TDA-MCEQ-SP. Equating the sparseness measure defined in (3.50) to one implies that $\|\mathbf{d}_l\|_1/\|\mathbf{d}_l\|_2 = 1$ and hence to achieve a perfectly sparse $\hat{\mathbf{d}}_l$ for equalization, a constraint

$$\frac{\|\hat{\mathbf{d}}_l(n)\|_1}{\|\hat{\mathbf{d}}_l(n)\|_2} = \frac{\|\mathbf{H}^T \hat{\mathbf{g}}_l(n)\|_1}{\|\mathbf{H}^T \hat{\mathbf{g}}_l(n)\|_2} = 1 \quad (4.12)$$

can be introduced to (4.8). This in turn aids the adaptive algorithm to achieve fast convergence by suppressing all non-zero coefficients of $\hat{\mathbf{d}}_l(n)$ except for the l th coefficient thereby improving the convergence rate of TDA-MCEQ-SP. The modified cost function incorporating the constraint is

$$J_l^c(n) = \|\mathbf{d}_l - \mathbf{H}^T \hat{\mathbf{g}}_l(n)\|_2^2 + \beta \left(1 - \frac{\|\hat{\mathbf{d}}_l(n)\|_1}{\|\hat{\mathbf{d}}_l(n)\|_2} \right)^2, \quad (4.13)$$

where β is a scale factor. It may be noted that (4.13) is equivalent to (4.8) when $\beta = 0$. From (4.13) the new gradient

$$\begin{aligned} \nabla J_l^c(n) = & -2\mathbf{H}\mathbf{e}_l(n) - 2\beta \left(1 - \frac{\|\hat{\mathbf{d}}_l(n)\|_1}{\|\hat{\mathbf{d}}_l(n)\|_2} \right) \\ & \left\{ \frac{\mathbf{H}}{\|\hat{\mathbf{d}}_l(n)\|_2} \text{sgn} [\hat{\mathbf{d}}_l(n)] - \frac{\|\hat{\mathbf{d}}_l(n)\|_1}{2\|\hat{\mathbf{d}}_l(n)\|_2^3} \mathbf{H}\mathbf{H}^T \hat{\mathbf{g}}_l(n) \right\} \end{aligned} \quad (4.14)$$

can be obtained, where

$$\text{sgn} [\hat{d}_{l,i}(n)] = \begin{cases} \hat{d}_{l,i}(n)/|\hat{d}_{l,i}(n)| & \hat{d}_{l,i}(n) \neq 0, \\ 0 & \hat{d}_{l,i}(n) = 0. \end{cases}$$

This modified gradient takes the sparseness of $\hat{\mathbf{d}}_l(n)$ into account which, in turn, confines the search space of TDA-MCEQ-SP such that only one coefficient of $\hat{\mathbf{d}}_l(n)$ is non-zero. This constraint further improves the convergence rate as will be seen later. It may be noted that β is empirically selected as it is

challenging to obtain a closed-form solution from the non-linear equation of (4.14).

The update equation of TDA-MCEQ-SP for estimating $\mathbf{G}$ is then given, across all L subspaces, by

$$\hat{\mathbf{G}}(n+1) = \hat{\mathbf{G}}(n) - \mu \nabla J^c(n)|_{\hat{\mathbf{G}}=\hat{\mathbf{G}}(n)}, \quad (4.15)$$

where $\nabla J^c(n) = [\nabla J_1^c(n), \nabla J_2^c(n), \dots, \nabla J_L^c(n)]$. It may be noted that if the algorithm is initialized with $\hat{\mathbf{G}}(n+1) = \mathbf{0}_{ML_g \times L}$, then $\|\hat{\mathbf{d}}(0)\|_2 = 0$ resulting in numerical error in (4.14) due to division by zero. This can be avoided by setting $\beta = 0$ for the first iteration so that the constraint will not be imposed.

As described in Chapter 3, subspace-based equalization using TDA-MCEQ and TDA-MCEQ-SP require $x_m(n)$ to be partitioned into frames of length L_g with a frame shift N , i.e., $\mathbf{x}_m(k) = [x_m(kN + L_g - 1), \dots, x_m(kN)]^T$. Each resultant frame is then equalized using $\hat{\mathbf{G}}$ given by

$$\hat{\mathbf{s}}(k) = \hat{\mathbf{G}}^T \mathbf{x}(k), \quad (4.16)$$

where $\hat{\mathbf{s}}(k) = [\hat{s}(kN + L_g - 1), \dots, \hat{s}(kN + L_g - L)]^T$, and the recovered signal is reconstructed using the equalized frames appropriately as described in Section 3.3.

4.4 Simulation results

4.4.1 Convergence performance

First, the convergence performance of A-MINT, SC-MINT and the proposed TDA-MCEQ and TDA-MCEQ-SP algorithms is illustrated in terms of the normalized misalignment $\eta(n)$. For the subspace-based algorithms $\eta(n)$ is computed across the L subspaces using

$$\eta(n) = 20 \log_{10} \left(\frac{1}{L} \sum_{l=1}^L \frac{\|\mathbf{d}_l - \hat{\mathbf{d}}_l(n)\|_2}{\|\mathbf{d}_l\|_2} \right) \text{ dB}, \quad (4.17)$$

where $\mathbf{d}_l(n) = \mathbf{H}^T \hat{\mathbf{g}}_l$. For A-MINT and SC-MINT, the equalized response is achieved using $\hat{\mathbf{d}}(n) = \mathbf{H}^T \hat{\mathbf{g}}(n)$ and $\eta(n)$ is computed as

$$\eta(n) = 20 \log_{10} \frac{\|\mathbf{d} - \hat{\mathbf{d}}(n)\|_2}{\|\mathbf{d}\|_2} \text{ dB}. \quad (4.18)$$

A set of five AIRs, each with length $L_h = 1600$, were synthetically generated using the method of images [215] for a room of dimension $6 \text{ m} \times 5 \text{ m} \times 3 \text{ m}$. A linear microphone array of $M = 5$ microphones spaced 0.1 m apart was placed at a height of 1.6 m while the array centroid was located at $(2, 2) \text{ m}$. The reverberation time T_{60} was chosen as $T_{60} = L_h/f_s$ where $f_s = 16 \text{ kHz}$ is the sampling frequency. For all the simulations in this section the length of the inverse filters is equal to $L_h/2$.

Figure 4.1 shows the convergence performance of A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP with $\beta = 0.01$ for $\mu = 0.2$ and 0.5 . It can be seen that the convergence rate of the algorithms increases with the step-size as expected and the proposed algorithms achieve a higher rate of convergence than A-MINT and SC-MINT. Comparing the performance between TDA-MCEQ and TDA-MCEQ-SP, it can be seen that TDA-MCEQ-SP converges at a rate that is modestly higher than TDA-MCEQ. These simulation results verify that the sparseness constraint further increases the convergence rate of TDA-MCEQ by confining the search space such that each column of $\mathbf{D}$ is sparse. It can also be noted that, compared to TDA-MCEQ, TDA-MCEQ-SP achieves an improvement of approximately 5 dB in η at steady-state convergence for both step-sizes.

The convergence performance of TDA-MCEQ-SP with different β for $\mu = 0.5$ is also shown in Fig. 4.2. Although the convergence rate increases with β , the performance of TDA-MCEQ-SP is relatively insensitive to the choice of β within the range 0.005 to 0.01 .

Next, the convergence performance of the algorithms using recorded AIRs is shown. The AIRs were recorded inside a $6.5 \text{ m} \times 8.75 \text{ m} \times 2.65 \text{ m}$ rectangular room at $f_s = 8 \text{ kHz}$ using an array of five microphones spaced 0.14 m apart. These recorded AIRs are of length $L_h = 1024$ and have $T_{60} \approx 300 \text{ ms}$. The inverse filters were estimated using A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP with $\mu = 0.5$. The convergence performance of the algorithms for these recorded AIRs is shown in Fig. 4.3. Both A-MINT and SC-MINT achieve $\eta = -20.6 \text{ dB}$ after 5×10^3 iterations. On the other hand,

Fig. 4.1: Convergence performance for A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP with different step-size μ .

Fig. 4.2: Convergence performance of TDA-MCEQ-SP for different β .

TDA-MCEQ and TDA-MCEQ-SP achieve $\eta = -29.3$ dB and -36.1 dB, respectively. It can be seen from Figs. 4.1 and 4.3 that TDA-MCEQ-SP achieves better convergence performance compared to other algorithms for both recorded as well as synthetic AIRs.

Fig. 4.3: Convergence performance of A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP for recorded AIRs.

4.4.2 Equalization performance

To quantify the equalization performance, the sparseness measure $0 \leq \xi(n) \leq 1$ of the equalized response $\hat{\mathbf{d}}(n)$ is used. To show equalization quality

$$\xi'(n) = 20 \log_{10}[1 - \xi(n)] \text{ dB} \quad (4.19)$$

is plotted for each algorithm such that $\xi'(n) = -\infty$ corresponds to perfect equalization.

In this simulation, the target response is estimated using A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP at each iteration and $\xi'(n)$ is computed for each algorithm. For TDA-MCEQ and TDA-MCEQ-SP, a frame shift $N = L_g$ is used for constructing $\hat{\mathbf{d}}(n)$. Figure 4.4 shows how $\xi'(n)$ varies with number of iterations for synthetic AIRs which were generated using the same simulation setup as that used to generate Fig. 4.1. As can be seen from Fig. 4.4, after 2×10^3 iterations, A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP achieve $\xi'(n) \approx -28.3$ dB, -28.2 dB, -34.8 dB and -41.4 dB, respectively. This clearly shows that the proposed TDA-MCEQ and TDA-MCEQ-SP algorithms outperform A-MINT and SC-MINT in terms of equalization quality under the given condition.

The sparseness performance of A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP is next evaluated using the same set of recorded AIRs used to generate Fig. 4.3. While $\xi'(n) = -26.8$ dB, -26.6 dB and -42.1 dB for A-MINT, SC-MINT and TDA-MCEQ, respectively, TDA-MCEQ-SP achieves $\xi'(n) = -48.2$ dB within 5×10^3 iterations. This shows that TDA-MCEQ-SP exploiting the sparseness constraint

Fig. 4.4: Sparseness performance comparisons for A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP with synthetic AIRs.

Fig. 4.5: Sparseness performance of A-MINT, SC-MINT, TDA-MCEQ and TDA-MCEQ-SP with recorded AIRs.

outperforms the other algorithms in terms of convergence and equalization performance.

In addition, to further illustrate the performance improvement, $\hat{\mathbf{d}}(n)$ that is achieved after 5×10^3 iterations for each algorithm is plotted in Fig. 4.6. Each of these target responses was normalized such that its maximum corresponds to unity. For clarity of presentation maximum magnitude in Fig. 4.6 is limited to 0.02. Comparing Figs. 4.6(a) and (b) with Figs. 4.6(c) and (d), it can be noted that both A-MINT and SC-MINT result in a less sparse $\hat{\mathbf{d}}(n)$ with the lowest $\xi(n) \approx 0.95$ compared to TDA-MCEQ and TDA-MCEQ-SP. The proposed TDA-MCEQ-SP and TDA-MCEQ algorithms, on the other hand, result in better equalization yielding $\xi(n) = 0.996$ and $\xi(n) = 0.992$, respectively.

Finally, speech dereverberation performance of the proposed algorithms and A-MINT is illustrated

using the same set of recorded AIRs and the source signal as that used in Section 3.5.4. Since both A-MINT and SC-MINT exhibit approximately same equalization performance, only A-MINT is included in this simulation. While SRR_{seg} averaged across $M = 5$ channels is 0.04 dB for the received signals, A-MINT achieves $\text{SRR}_{\text{seg}} = 14.9$ dB. On the other hand, both the proposed algorithms exhibit approximately equal dereverberation performance achieving $\text{SRR}_{\text{seg}} \approx 27.8$ dB. Figure 4.4.2 shows the waveform and spectrogram of the source signal, signal received at the first channel, and the signals recovered by the algorithms. It can be noted the signal recovered by A-MINT still suffers from spectral and temporal smearing effects caused by reverberation. Moreover, A-MINT introduces some high-frequency artifacts to the recovered signal. On the other hand, the proposed TDA-MCEQ and TDA-MCEQ-SP algorithms achieve, similar to MCEQ-T, significant suppression of time-spreading effects due to reverberation thereby achieving good speech dereverberation.

4.5 Conclusion

Similar to MINT, the MCEQ-T algorithm proposed in Chapter 3 is also computationally complex due to the matrix inversion. Hence, in this chapter, adaptive filtering techniques were employed to estimate the multiple sets of filters with reduced complexity. Two time-domain subspace-based adaptive multichannel equalization algorithms are proposed to achieve equalization of high-order AIRs with less computational complexity. In addition to exploiting the computational efficiency of adaptive filters, the sparseness of the target response has also been exploited as a constraint to the objective function for improved convergence performance. Simulation results showed that the proposed algorithms achieve faster convergence and better equalization compared to A-MINT and SC-MINT. By virtue of application of the sparseness constraint, the proposed TDA-MCEQ-SP algorithm achieves faster convergence as well as better equalization performance.

(a) A-MINT

(b) SC-MINT

(c) TDA-MCEQ

(d) TDA-MCEQ-SP

Fig. 4.6: Equalized response $\hat{\mathbf{d}}$ achieved using (a) A-MINT, (b) TDA-MCEQ and (c) TDA-MCEQ-SP with $\mu = 0.5$.

Fig. 4.7: Speech dereverberation performance comparison: Waveform and spectrogram of (a) Original source signal, (b) Reverberant signal at channel 1, (c) signal recovered using A-MINT, (d) signal recovered using TDA-MCEQ, (e) signal recovered using TDA-MCEQ-SP

Chapter 5

Variable Step-size Acoustic Multichannel Equalization Algorithms Exploiting Sparseness Measure

5.1 Introduction

In Chapter 4 two adaptive algorithms were proposed to achieve subspace-based multichannel equalization. The cost functions of the proposed TDA-MCEQ and TDA-MCEQ-SP algorithms were derived similar to the A-MINT algorithm [188]. In this chapter, a new approach to achieve adaptive MCEQ is introduced. The proposed adaptive MCEQ (A-MCEQ) algorithms are derived by imposing a constraint such that the response after equalization is in the same direction as the target Kronecker delta function, i.e., the equalized response is obtained as a projection on the desired response.

The step-size of an adaptive algorithm must be carefully selected since it determines the rate of convergence. Although the choice of an appropriate step-size is dependent on the statistical properties of the input data it is common to employ a fixed step-size. For algorithms employing a fixed step-size, the rate of convergence varies largely with the input data. With the aim of achieving faster convergence and better equalization performance, a variable step-size approach is formulated in this chapter. For each iteration an upper bound for the step-size is derived from the sparseness constraint. As will be shown, the proposed variable step-size technique can be applied for A-MINT and the proposed adaptive MCEQ algorithms. Simulation results show that the proposed A-MCEQ algorithms achieve better convergence performance than A-MINT yielding better equalization quality.

The rest of this chapter is organized as follows: In Section 5.2 the proposed A-MCEQ and A-

Fig. 5.1: The angle between $\mathbf{a}$ and $\mathbf{b}$.

MCEQ-N algorithms are derived. The convergence analysis of A-MCEQ is presented in Section 5.2.1. Simulations analyzing the performance of the proposed algorithms are described in Section 5.3 while Section 5.4 concludes this chapter.

5.2 The proposed adaptive MCEQ (A-MCEQ) algorithms

In A-MINT the inverse filters are estimated by minimizing the difference between the target response and the equalized response. A different approach is introduced here for the adaptive estimation of the inverse filters $\mathbf{g}$. The proposed adaptive MCEQ (A-MCEQ) algorithm estimates $\mathbf{g}$ such that the equalized response

$$\hat{\mathbf{d}}(n) = \mathbf{H}^T \hat{\mathbf{g}}(n) \quad (5.1)$$

is parallel to $\mathbf{d}$, where $\mathbf{H}$ has been defined in (3.6). This algorithm therefore minimizes the angle between $\mathbf{d}$ and $\hat{\mathbf{d}}(n)$ at each iteration resulting in faster convergence compared to A-MINT.

The cosine of an angle θ between two arbitrary vectors (see Fig. 5.1), $\mathbf{a}$ and $\mathbf{b}$, is given by

$$\cos \theta = \frac{\mathbf{a}^T \mathbf{b}}{\|\mathbf{a}\|_2 \|\mathbf{b}\|_2} \quad (5.2)$$

such that when $\theta = 0$ both vectors lie in the same direction giving $\cos \theta = 1$. Therefore, here, it is proposed to estimate $\hat{\mathbf{g}}(n)$ iteratively such that the angle between $\mathbf{d}$ and $\hat{\mathbf{d}}(n)$ is minimized towards zero at each iteration, i.e., $\cos \theta$ is maximized. The algorithm can be formulated from the fact that perfect

equalization requires $\hat{\mathbf{d}}(n) = \mathbf{d}$ giving

$$\frac{\mathbf{d}^T \hat{\mathbf{d}}(n)}{\|\mathbf{d}\|_2 \|\hat{\mathbf{d}}(n)\|_2} = 1. \quad (5.3)$$

Equation (5.3) implies that both $\mathbf{d}$ and $\hat{\mathbf{d}}(n)$ are aligned in the same direction. However, since (5.3) does not take vector magnitudes into account $\|\hat{\mathbf{d}}(n)\|_2 = \alpha \|\mathbf{d}\|_2$ after convergence, where α is a scale factor. It is important to note that this scale factor only affects the energy of the reconstructed speech which may not be crucial for enhancing the intelligibility of the speech.

From (5.3) a minimization criterion given by

$$\hat{\mathbf{g}}(n) = \arg \min_{\hat{\mathbf{g}}(n)} \left\{ 1 - \frac{\mathbf{d}^T [\mathbf{H}^T \hat{\mathbf{g}}(n)]}{\|\mathbf{d}\|_2 \|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_2} \right\}^2 \quad (5.4)$$

is defined. Since $\|\mathbf{d}\|_2 = 1$, the cost function results in

$$J(n) = \left\{ 1 - \frac{\mathbf{d}^T [\mathbf{H}^T \hat{\mathbf{g}}(n)]}{\|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_2} \right\}^2. \quad (5.5)$$

This cost function is then differentiated w.r.t. $\hat{\mathbf{g}}(n)$ giving

$$\nabla J(n) = -2e(n) \left[\frac{\mathbf{H}\mathbf{d}}{\|\hat{\mathbf{d}}(n)\|_2} - \frac{\mathbf{d}^T \hat{\mathbf{d}}(n) (\mathbf{H}\mathbf{H}^T) \hat{\mathbf{g}}(n)}{\|\hat{\mathbf{d}}(n)\|_2^3} \right], \quad (5.6)$$

where

$$e(n) = 1 - \frac{\mathbf{d}^T [\mathbf{H}^T \hat{\mathbf{g}}(n)]}{\|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_2}. \quad (5.7)$$

Employing (5.6) an update equation is formulated for A-MCEQ as

$$\hat{\mathbf{g}}(n+1) = \hat{\mathbf{g}}(n) - \mu \nabla J(n)|_{\hat{\mathbf{g}}=\hat{\mathbf{g}}(n)}. \quad (5.8)$$

Initialization using a null vector such that $\hat{\mathbf{g}}(0) = \mathbf{0}_{ML_g \times 1}$ will result in numerical errors in (5.6) and (5.7) and therefore, $\hat{\mathbf{g}}(0) = [1, \mathbf{0}_{1 \times ML_g-1}]^T$ is used for initialization.

5.2.1 Convergence analysis

The convergence behavior of the proposed A-MCEQ algorithm is now analyzed. Defining

$$\Delta \hat{\mathbf{g}}(n+1) = \hat{\mathbf{g}}(n+1) - \mathbf{g}, \quad (5.9)$$

where $\mathbf{g}$ is the true solution, it can be written that

$$\begin{aligned} \Delta \hat{\mathbf{g}}(n+1) &= \Delta \hat{\mathbf{g}}(n) - \mu \nabla J(n) \\ &= \Delta \hat{\mathbf{g}}(n) + \mu \frac{2e(n)}{\|\hat{\mathbf{d}}(n)\|_2} \left[\mathbf{H}\mathbf{d} - \frac{\mathbf{d}^T \hat{\mathbf{d}}(n) \mathbf{H} \mathbf{H}^T \hat{\mathbf{g}}(n)}{\|\hat{\mathbf{d}}(n)\|_2^2} \right] \\ &= \Delta \hat{\mathbf{g}}(n) + \mu' \left\{ \mathbf{H}\mathbf{d} - \frac{\mathbf{d}^T \hat{\mathbf{d}}(n) \mathbf{H} \mathbf{H}^T [\mathbf{g} + \Delta \hat{\mathbf{g}}(n)]}{\|\hat{\mathbf{d}}(n)\|_2^2} \right\}, \end{aligned} \quad (5.10)$$

where $\mu' = \mu \frac{2e(n)}{\|\hat{\mathbf{d}}(n)\|_2}$. Defining $\mathfrak{R} = E \left[\frac{\mathbf{d}^T \hat{\mathbf{d}}(n)}{\|\hat{\mathbf{d}}(n)\|_2^2} \right] \mathbf{H} \mathbf{H}^T$ and taking the expectation on both sides of (5.10) the convergence behavior of A-MCEQ is obtained, which is given by

$$\begin{aligned} E[\Delta \hat{\mathbf{g}}(n+1)] &= [\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R}] E[\Delta \hat{\mathbf{g}}(n)] + \mu' \mathbf{H} \left\{ \mathbf{d} - E \left[\frac{\mathbf{d}^T \hat{\mathbf{d}}(n)}{\|\hat{\mathbf{d}}(n)\|_2^2} \right] \mathbf{H}^T \mathbf{g} \right\} \\ &= [\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R}] E[\Delta \hat{\mathbf{g}}(n)] + \mu' \mathbf{H} \psi, \end{aligned} \quad (5.11)$$

where $\psi = \mathbf{d} - E \left[\frac{\mathbf{d}^T \hat{\mathbf{d}}(n)}{\|\hat{\mathbf{d}}(n)\|_2^2} \right] \mathbf{H}^T \mathbf{g}$. Equation (5.11) can be expanded as

$$\begin{aligned} E[\Delta \hat{\mathbf{g}}(n+1)] &= [\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R}] E[\Delta \hat{\mathbf{g}}(n)] + \mu' \mathbf{H} \psi \\ &= [\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R}]^2 E[\Delta \hat{\mathbf{g}}(n-1)] + \mu' \mathbf{H} \psi [\mathbf{I}_{ML_g \times ML_g} + (\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R})] \\ &= [\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R}]^3 E[\Delta \hat{\mathbf{g}}(n-2)] + \mu' \mathbf{H} \psi [\mathbf{I}_{ML_g \times ML_g} \\ &\quad + (\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R}) + (\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R})^2] \\ &= [\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R}]^{n+1} E[\Delta \hat{\mathbf{g}}(0)] + \mu' \mathbf{H} \psi \sum_{i=0}^n (\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R})^i. \end{aligned} \quad (5.12)$$

Fig. 5.2: Target response $\mathbf{d}$ and estimated response $\hat{\mathbf{d}}(n)$ achieved using A-MCEQ without normalization for $\mu = 0.2$.

As $n \rightarrow \infty$, $[\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R}]^{n+1} \rightarrow \mathbf{0}_{ML_g \times ML_g}$ and $\sum_{i=0}^n (\mathbf{I}_{ML_g \times ML_g} - \mu' \mathfrak{R})^i \rightarrow (\mu' \mathfrak{R})^{-1}$ giving

$$E[\Delta \hat{\mathbf{g}}(\infty)] = (\mu' \mathfrak{R})^{-1} \mu' \mathbf{H} \psi.$$

Since $\mu'^{-1} \mu' = 1$,

$$\begin{aligned} E[\Delta \hat{\mathbf{g}}(\infty)] &= \mathfrak{R}^{-1} \mathbf{H} \psi \\ &= (\mathbf{H}^T)^+ \left\{ \mathbf{d} - E \left[\frac{\mathbf{d}^T \hat{\mathbf{d}}(n)}{\|\hat{\mathbf{d}}(n)\|_2^2} \right] \mathbf{H}^T \mathbf{g} \right\} \\ &= (\mathbf{H}^T)^+ \mathbf{d} - \mathbf{g} E \left[\frac{\mathbf{d}^T \hat{\mathbf{d}}(n)}{\|\hat{\mathbf{d}}(n)\|_2^2} \right]. \end{aligned} \quad (5.13)$$

Since $E[\hat{\mathbf{g}}(n)] = E[\Delta \hat{\mathbf{g}}(n)] + \mathbf{g}$, A-MCEQ-SP can converge with the limiting vector given by

$$E[\hat{\mathbf{g}}(\infty)] = \frac{(\mathbf{H}^T)^+ \mathbf{d}}{E \left[\frac{\mathbf{d}^T \hat{\mathbf{d}}(\infty)}{\|\hat{\mathbf{d}}(\infty)\|_2^2} \right]}. \quad (5.14)$$

After having analyzed the convergence behavior, the performance of the proposed A-MCEQ algorithm is now illustrated using a set of five recorded AIRs from the MARDY database [216]. These AIRs are downsampled to 16 kHz and truncated to $L_h = 800$ before being equalized using inverse filters obtained from (5.8) with $\mu = 0.2$, $\tau = 49$ and $L_g = L_h/2$ after 5×10^4 iterations. From the reconstructed response $\hat{\mathbf{d}}(n)$ as shown in Fig. 5.2, it can be seen that the directions of both $\hat{\mathbf{d}}(n)$ and

$\mathbf{d}(n)$ are the same while their magnitudes differ.

For applications where $\|\hat{\mathbf{d}}(n)\|_2 = \|\mathbf{d}\|_2$ is desired, one can further impose a magnitude constraint in the algorithm. Since $\|\mathbf{d}\|_2 = 1$, normalization of $\hat{\mathbf{d}}(n)$ achieved using

$$\hat{\mathbf{d}}(n) = \frac{\mathbf{H}^T \hat{\mathbf{g}}(n)}{\|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_2} \quad (5.15)$$

ensures $\|\hat{\mathbf{d}}(n)\|_2 = \|\mathbf{d}\|_2$. In order to incorporate the magnitude constraint in the estimation process this normalization operation is incorporated into the update equation in (5.8) as

$$\begin{aligned} \tilde{\mathbf{g}}(n+1) &= \hat{\mathbf{g}}(n) - \mu \nabla J(n)|_{\hat{\mathbf{g}}=\hat{\mathbf{g}}(n)}, \\ \hat{\mathbf{g}}(n+1) &= \frac{\tilde{\mathbf{g}}(n+1)}{\|\mathbf{H}^T \tilde{\mathbf{g}}(n+1)\|_2}, \end{aligned} \quad (5.16)$$

while $\hat{\mathbf{d}}(n+1) = \mathbf{H}^T \hat{\mathbf{g}}(n+1)$ giving the normalized A-MCEQ (A-MCEQ-N) algorithm. This unit-norm constraint helps to simplify the gradient of A-MCEQ-N as

$$\nabla J(n) = -2e(n) \left[\mathbf{H}\mathbf{d} - \mathbf{d}^T \hat{\mathbf{d}}(n) (\mathbf{H}\mathbf{H}^T) \hat{\mathbf{g}}(n) \right], \quad (5.17)$$

where the algorithm should be initialized such that $\|\mathbf{H}^T \hat{\mathbf{g}}(0)\|_2 = 1$.

5.2.2 Adaptation of step-size based on the sparseness measure

Selection of the step-size μ is an important factor that determines the convergence performance of any adaptive algorithm. This leads to the derivation of a variable step-size $\mu(n)$ at each iteration so as to improve the convergence rate of MINT-based adaptive equalization algorithms. As will be shown, this time-varying step-size can be applied to A-MINT, the above proposed A-MCEQ and A-MCEQ-N algorithms. Here, the sparsity of $\hat{\mathbf{d}}(n)$ is exploited to determine $\mu(n)$. The proposed method estimates an upper bound $\mu_{\max}(n)$ for the step-size $\mu(n)$ at each iteration such that $\mu(n) \leq \mu_{\max}(n)$ minimizes a sparseness-based posterior error ensuring faster convergence of the algorithm and better equalization.

From the sparseness measure given in (3.50), a constraint can be derived as

$$\frac{\|\hat{\mathbf{d}}(n)\|_1}{\|\hat{\mathbf{d}}(n)\|_2} = 1, \quad (5.18)$$

such that $\xi(n) \rightarrow 1$ for $\hat{\mathbf{d}}(n)$. From (5.18), the convergence rate of the algorithm can be improved if $\mu(n)$ is computed at each iteration such that the squared posterior error

$$e_s^2(n+1) = \left(1 - \frac{\|\mathbf{H}^T \hat{\mathbf{g}}(n+1)\|_1}{\|\mathbf{H}^T \hat{\mathbf{g}}(n+1)\|_2}\right)^2 \quad (5.19)$$

is minimized. With $J(n)$ being a cost function, which may correspond to that of A-MINT, A-MCEQ or A-MCEQ-N, the update equation

$$\hat{\mathbf{g}}(n+1) = \hat{\mathbf{g}}(n) - \mu(n) \nabla J(n)|_{\hat{\mathbf{g}}=\hat{\mathbf{g}}(n)} \quad (5.20)$$

is substituted into (5.19) giving

$$e_s^2(n+1) = \left(1 - \frac{\|\mathbf{H}^T [\hat{\mathbf{g}}(n) - \mu(n) \nabla J(n)]\|_1}{\|\mathbf{H}^T [\hat{\mathbf{g}}(n) - \mu(n) \nabla J(n)]\|_2}\right)^2. \quad (5.21)$$

Equating (5.21) to zero and rearranging the terms,

$$\|\mathbf{H}^T [\hat{\mathbf{g}}(n) - \mu(n) \nabla J(n)]\|_1 = \|\mathbf{H}^T [\hat{\mathbf{g}}(n) - \mu(n) \nabla J(n)]\|_2. \quad (5.22)$$

Obtaining a closed-form solution of $\mu(n)$ is challenging due to the non-linear characteristic of (5.22).

Using the triangular inequality property of vector norms (5.22) can be expressed as

$$\|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_1 - |\mu(n)| \|\mathbf{H}^T \nabla J(n)\|_1 \geq \|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_2 - |\mu(n)| \|\mathbf{H}^T \nabla J(n)\|_2, \quad (5.23)$$

from which it can be derived that

$$|\mu(n)| \leq \frac{\|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_1 - \|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_2}{\|\mathbf{H}^T \nabla J(n)\|_1 - \|\mathbf{H}^T \nabla J(n)\|_2}. \quad (5.24)$$

Since μ is non-negative the right hand term of (5.24) can be treated as the upper bound of $\mu(n)$ for the n th iteration. This bound is denoted by $\mu_{\max}(n)$ and it ensures stability of the algorithm. In order to prevent division by zero a small value δ is introduced in (5.24) giving

$$\mu_{\max}(n) \leq \frac{\|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_1 - \|\mathbf{H}^T \hat{\mathbf{g}}(n)\|_2}{\|\mathbf{H}^T \nabla J(n)\|_1 - \|\mathbf{H}^T \nabla J(n)\|_2 + \delta}. \quad (5.25)$$

Selection of $\mu(n) = \beta \mu_{\max}(n)$, where $\beta \leq 1$ is a scale factor, results in high convergence rate compared to a pre-determined value of μ while ensuring the convergence of algorithm. It is important to note that, to achieve a non-zero step-size at the first iteration, $\hat{\mathbf{g}}(0)$ should not be a null vector.

5.3 Simulation results

5.3.1 A-MCEQ and A-MCEQ-N

Since the proposed algorithms are based on a new cost function, their performance is compared with only A-MINT, which is the direct adaptive counterpart of MINT. Moreover, simulation results presented in Section 4.4 show that both A-MINT and SC-MINT achieve similar equalization performance on convergence. Hence, SC-MINT is not included in the comparison. The convergence performance of A-MCEQ and A-MCEQ-N with fixed μ is first compared with that of A-MINT using the normalized misalignment defined as

$$\eta(n) = 20 \log_{10} \frac{\|\mathbf{d} - \hat{\mathbf{d}}(n)\|_2}{\|\mathbf{d}\|_2} \text{ dB}. \quad (5.26)$$

A set of AIRs, each of length $L_h = 1600$, are generated using the method of images. In this simulation, a linear array is deployed in a $6 \text{ m} \times 5 \text{ m} \times 3 \text{ m}$ room. The microphone array consists of $M = 5$ microphones with an inter-microphone spacing of 0.1 m . The centroid of the array is located at $(2, 2, 1.6) \text{ m}$. A sampling frequency of $f_s = 16 \text{ kHz}$ and reverberation time $T_{60} = L_h/f_s$ are used. For the A-MCEQ algorithm $\hat{\mathbf{d}}$ is achieved using (5.8) and (5.15) while (5.16) is used for A-MCEQ-N. For all simulations $L_g = L_h/2$.

The convergence performance of the algorithms is compared in Fig. 5.3 where fixed step-size $\mu = 0.2$ and 0.5 for A-MINT while fixed step-size μ for A-MCEQ and A-MCEQ-N are selected so as to achieve

Fig. 5.3: Convergence performance comparison of A-MCEQ, A-MCEQ-N and A-MINT for various μ .

Fig. 5.4: Effect of μ on convergence performance of A-MCEQ and A-MCEQ-N.

approximately the same steady-state normalized misalignment as A-MINT. As seen from Fig. 5.3, both A-MCEQ and A-MCEQ-N exhibit higher convergence rate compared to A-MINT. In addition, A-MINT requires more than 2×10^3 iterations to achieve $\eta(n) < -15$ dB when $\mu = 0.5$ whereas the proposed algorithms require less than 500 iterations to achieve the same $\eta(n)$.

The convergence performance is also analyzed for various μ (which were fixed for all iterations) using the set of five recorded AIRs of $L_h = 1024$ described in Section 4.4.1. It can be observed from Fig. 5.4 that the convergence performance of both algorithms is dependent on μ as expected; the convergence rate of A-MCEQ increases while $\eta(n)$ reduces with increasing step-size for $0.01 \leq \mu \leq 0.5$. For $\mu = 0.9$ and 2, the steady-state of A-MCEQ increases with μ .

The effect of applying variable step-size $\mu(n) = \beta\mu_{\max}(n)$ on the convergence performance is

Fig. 5.5: Convergence performance of A-MINT with variable step-size.

Fig. 5.6: Convergence performance of A-MCEQ with variable step-size.

next examined using the same set of recorded AIRs. The convergence performance of A-MINT with different β values is compared with that for fixed μ . When the variable step-size is used, A-MINT was initialized with $\hat{\mathbf{g}}(0) = [1, \mathbf{0}_{1 \times ML_g - 1}]^T$. Figure 5.5 shows the convergence performance of A-MINT for $\beta = 0.1, 0.5, 1, 1.8$ and 2 . Convergence curves achieved with fixed step-size $\mu = 0.5$ and 0.8 are also shown for comparison. It can be noted that, compared to a fixed μ , the proposed variable step-size method improves the convergence performance significantly yielding lower $\eta(n)$ with improved convergence rate. It may also be noted that the algorithm misconverges for $\mu(n) = 2\mu_{\max}(n)$.

Convergence performance of A-MCEQ, for the same set of recorded AIRs, with variable step-size $\mu(n)$ is illustrated in Fig. 5.6. Since the performance of A-MCEQ-N with variable step-size is almost similar, its convergence performance is not shown here. Comparing Figs. 5.4 and 5.6 it can be seen that

Fig. 5.7: Equalization performance with system mismatch $\mathcal{M}_m$.

application of a variable step-size improves both the convergence rate and equalization quality. It can be noted that $\eta(n)$ converges to less than -30 dB within 2×10^3 iterations.

Equalization performance with noisy channel estimates that may arise from errors in channel estimation is also examined in this section using A-MINT and A-MCEQ with variable step-size. The β value for each algorithm is first selected such that both algorithms achieve approximately equal performance after 5×10^3 iterations in terms of η under no system mismatch condition. These β values correspond to 0.8 and 0.5 for A-MINT and A-MCEQ, respectively. Equalization of $\mathbf{h}_m$ is then achieved using the equalization filters estimated from the perturbed AIRs, for $-10 \text{ dB} \geq \mathcal{M}_m \geq -80 \text{ dB}$, where $\mathcal{M}_m$ has been defined in (3.31). The η values achieved by each algorithm after 5×10^3 iterations are plotted in Fig. 5.7 against $\mathcal{M}_m$. It can be observed that both the algorithms exhibit approximately equal performance. The performance of the algorithms degrades as $\mathcal{M}_m$ increases and the performance degradation becomes significant for $\mathcal{M}_m \geq 40 \text{ dB}$.

Finally, the equalized responses $\hat{\mathbf{d}}$ obtained using A-MINT and A-MCEQ with variable step-size are shown in Fig. 5.8 for the above set of recorded AIRs. It may be noted that the maximum magnitude in Fig. 5.8 is limited to 0.02 for ease of comparison. The β values for the algorithms are selected so as to achieve approximately equal convergence performance, as before, and $\hat{\mathbf{d}}$ is obtained after 1×10^4 iterations. It can be seen that, with variable step-size, both A-MINT and A-MCEQ achieve good equalization performance yielding $\xi = 0.992$ and 0.996 , respectively. It may be noted that, with this

Fig. 5.8: Equalized response $\hat{\mathbf{d}}$ obtained using (a) A-MINT and (b) A-MCEQ with variable step-size.

configuration, both A-MCEQ and A-MINT achieve $\text{SRR}_{\text{seg}} \approx 14.9$ dB for the same simulation setup using the recorded AIRs as described in Section 3.5.4.

5.4 Conclusion

Adaptive MCEQ algorithms address the computational complexity issue associated with the MINT algorithm. The A-MINT algorithm suffers from a slow convergence rate. It is desirable for an adaptive algorithm to achieve better equalization with a fast convergence rate. With this objective, a new multichannel-adaptive approach for equalizing room acoustics was proposed. The equalized response is achieved as a vector projection on the target response. The proposed A-MCEQ and A-MCEQ-N algorithms achieve faster initial convergence than A-MINT while achieving better equalization performance. In order to improve the convergence performance of adaptive MCEQ algorithms based on the MINT relation, the sparsity of the equalized response is exploited to determine the step-size. An upper bound for the step-size is derived for each iteration such that, with the derived step-size, the sparseness of the equalized

response increases significantly. This in turn results in faster convergence. Simulation results show that the application of variable step-size improves the convergence rate as well as equalization quality.

Chapter 6

Multichannel Equalization in the Frequency Domain

6.1 Introduction

As discussed in Chapter 4, existing multiple-input/output inverse theorem (MINT)-based non-adaptive algorithms are computationally expensive since they require the inversion of a matrix with dimension that is proportional to the AIR length. Although adaptive MCEQ algorithms such as A-MCEQ achieve equalization of high-order AIRs with reduced complexity compared to the non-adaptive MINT algorithm, their convergence rate is slow. While subband based algorithms achieve some reduction in complexity, their performance is dependent on various factors such as filter bank design, decimation factor, and number of subbands.

It is well-known that linear convolution in the time domain is transformed into element-wise multiplication in the frequency domain. For long impulse responses, this property makes frequency-domain processing less complex compared to the time domain. In this chapter, this complexity reduction in the frequency domain is exploited and a frequency-domain MCEQ algorithm is proposed. This algorithm is computationally efficient and can thus be employed for the equalization of high-order AIRs in practice.

It has been shown in Chapter 3 that equalization can be achieved in various transform domains with the help of the proposed generalized MCEQ framework. As illustrative examples, three MCEQ algorithms (two in time and one in the KLT domain) have been formulated in Section 3.4. To derive the frequency-domain algorithm an insight into frequency-domain equalization is first established in this chapter. This is then applied to develop the frequency-domain MCEQ-F algorithm which achieves significant complexity reduction in the equalization process. The frequency-domain MCEQ-F algorithm

will be formulated in Section 6.2. The equalization performance and the computational complexity of the proposed algorithm is compared with two well-known MCEQ algorithms. To achieve further reduction in the complexity without significant performance degradation, a modified version of the frequency-domain algorithm which divides the AIRs into frames is also proposed in Section 6.2. Simulation results presented in Section 6.4 show that the proposed algorithms outperform existing MCEQ algorithms in terms of complexity while achieving better equalization performance compared to existing algorithms. These results also reveal that the proposed frequency-domain algorithms are more robust to AIR estimation errors. This chapter is concluded in Section 6.5.

6.2 The proposed MCEQ-F algorithm

With reference to (3.9), the transform matrix $\mathbf{U}$ for the frequency-domain MCEQ (MCEQ-F) algorithm is the $L \times L$ inverse Fourier matrix

$$\mathbf{F}_{L \times L} = [\mathbf{f}_1, \dots, \mathbf{f}_L], \quad (6.1)$$

where each

$$\mathbf{f}_l = \frac{1}{\sqrt{L}} \{1, \exp[j\omega_l], \dots, \exp[j\omega_l(L-1)]\}^T, \quad l = 1, \dots, L, \quad (6.2)$$

is an $L \times 1$ vector defining the l th subspace, $j = \sqrt{-1}$, and $\omega_l = 2\pi(l-1)/L$. The $L \times L$ frequency-domain counterpart of $\mathbf{H}_m$ is obtained as

$$\begin{aligned} \underline{\mathbf{H}}_m &= \mathbf{F}_{L \times L}^H \begin{bmatrix} \mathbf{H}_m \\ \mathbf{0}_{(L-L_x) \times L} \end{bmatrix} \mathbf{F}_{L \times L} \\ &= [\underline{\mathbf{h}}_{m,1}, \dots, \underline{\mathbf{h}}_{m,L}]^T, \end{aligned} \quad (6.3)$$

where $\underline{\mathbf{h}}_{m,l} = [h_{m,l}(1), \dots, h_{m,l}(L)]^T$. Similar to (3.21), the frequency-domain inverse filters of dimension $ML \times L$ can be realized using

$$\begin{aligned} \underline{\mathbf{G}} &= [\underline{\mathbf{g}}_1, \dots, \underline{\mathbf{g}}_L] \\ &= (\underline{\mathbf{H}}^T)^+ \mathbf{I}_{L \times L}, \end{aligned} \quad (6.4)$$

where the $ML \times L$ matrix $\underline{\mathbf{H}} = [\underline{\mathbf{H}}_1^T, \dots, \underline{\mathbf{H}}_M^T]^T$ and the $ML \times 1$ vector $\underline{\mathbf{g}}_l = [\underline{\mathbf{g}}_{1,l}^T, \dots, \underline{\mathbf{g}}_{M,l}^T]^T$.

It may be noted that, unlike the KLT and time domains described in Chapter 3, the appendage of $L - L_x$ zeros to the received signal and $L - L_x$ rows of zeros to $\underline{\mathbf{H}}_m$ in (3.15) and (3.16) is necessary for the frequency-domain case in order to realize linear convolution. In this case, as can be seen from (6.4), a set of inverse filters $\underline{\mathbf{g}}_{m,l}$, $m = 1, \dots, M$, each of length $L_g = L$, are designed for each frequency bin defined by $\mathbf{f}_l$, $l = 1, \dots, L$, for the recovery of $\underline{\mathbf{s}}(k)$ (the frequency-domain representation of $\mathbf{s}(k)$). It can be noted that the computation of (6.4) requires the pseudo-inverse of $\underline{\mathbf{H}}^T$, making its direct application computationally expensive for high-order AIRs, similar to MCEQ-T and MCEQ-KL. Therefore, in this chapter, a method is proposed to accomplish frequency-domain MCEQ with less complexity.

It is well known that the columns of $\mathbf{F}^H$ are the eigenvectors of a circulant matrix having the same dimension as the Fourier matrix, with Fourier coefficients being the eigenvalues of the circulant matrix [222]. This implies that the Fourier matrix can diagonalize a circulant matrix. Although $[\underline{\mathbf{H}}_m^T, \mathbf{0}_{L \times (L-L_x)}]^T$ in (6.3) is not a circulant matrix, it has been proven in Appendix A that when L is large, $\underline{\mathbf{H}}_m$ approximates a diagonal matrix. As an illustrative example, Fig. 6.1 shows the case where the magnitude of $\underline{\mathbf{H}}_m$ is computed using one of the recorded AIRs described in Section 3.4.4 for $L_x = 2L_h$ with $L_h = 1600$. To further illustrate the effect of L on the degree of diagonalization,

$$\xi_D = (\bar{\xi}_r + \bar{\xi}_c)/2 \quad (6.5)$$

is used to quantify the degree of diagonalization, where $\bar{\xi}_r$ and $\bar{\xi}_c$, respectively, define the sparseness measures averaged across the row and column vectors of $\underline{\mathbf{H}}_m$. For a perfectly diagonal matrix, $\xi_D = 1$. It can be seen from Fig. 6.2 that as L increases, ξ_D approaches 1. This in turn implies that better diagonalization of $\underline{\mathbf{H}}_m$ can be achieved when L is large.

Applying the MCEQ framework, the l th Fourier coefficient of the k th frame $\mathbf{x}_m(k)$ can be represented as

$$\underline{x}_m(k, l) = \underline{\mathbf{h}}_{m,l}^T \underline{\mathbf{s}}(k). \quad (6.6)$$

Since $\underline{\mathbf{H}}_m$ can be approximated by a diagonal matrix, the l th row of $\underline{\mathbf{H}}_m$, given by $\underline{\mathbf{h}}_{m,l}$, contains only one significant coefficient $\underline{h}_{m,l}(l)$, corresponding to the l th diagonal element of $\underline{\mathbf{H}}_m$. This effectively

Fig. 6.1: $\underline{\mathbf{H}}_m$ in the frequency domain for $L_h = 1600$ and $L_x = 2L_h$.

Fig. 6.2: Diagonality measure ξ_D of $\underline{\mathbf{H}}_m$ as a function of L .

converts the inner product in (6.6) to a scalar product

$$\underline{x}_m(k, l) = \alpha [\underline{h}_{m,l}(l) \underline{s}(k, l)], \quad (6.7)$$

where α is a uniform scaling factor, which, as shown in (A.6) of Appendix A, is equivalent to $R/L = L_x/L$. This factor only scales the reconstructed signal and therefore, for clarity, the effect of α is ignored for the description of MCEQ-F. Hence, when $\mathbf{U} = \mathbf{F}$ and for large L , $\underline{h}_{m,l}$ and $\underline{s}(k)$ for each bin, in effect, are reduced to $\underline{h}_{m,l}(l)$ and $\underline{s}(k, l)$, respectively, each of unit length. In a similar manner, $\underline{\mathbf{g}}_{m,l} = [\underline{g}_{m,l}(1), \dots, \underline{g}_{m,l}(L)]^T$ contains only one non-zero element $\underline{g}_{m,l}(l)$.

Based on MINT, it is now possible to deduce that for each frequency bin,

$$\underline{\mathbf{h}}_l^T \underline{\mathbf{g}}_l = 1 \quad (6.8)$$

across all M channels, from which the inverse filters for each bin can be derived as

$$\underline{\mathbf{g}}_l = [\underline{\mathbf{h}}_l^T]^+, \quad (6.9)$$

where the $M \times 1$ vector $\underline{\mathbf{h}}_l = [h_{1,l}(l), \dots, h_{M,l}(l)]^T$ and $\underline{\mathbf{g}}_l$, defined below (6.4), is now reduced to the $M \times 1$ vector $\underline{\mathbf{g}}_l = [g_{1,l}(l), \dots, g_{M,l}(l)]^T$. For each bin, the pseudo-inverse computation of $(\underline{\mathbf{h}}_l)^T$ is not computationally demanding: It is shown in Appendix A that $\underline{h}_{m,l}(l)$ is equivalent to the complex conjugate of l th discrete Fourier transform (DFT) coefficient of $\mathbf{h}_m$, scaled by a factor. Since this scaling factor is uniform and can be neglected, $\underline{\mathbf{h}}_l$ can be formed from the complex conjugate of l th DFT coefficients of the AIRs. It is also noted that in order to achieve a diagonal matrix to realize scalar operation, the order L of the DFT should be large. It has been shown in [223] that an infinitely long DFT sequence, corresponding to the optimal inverse filter of infinite length, is required to completely mitigate circular convolution effects. This implies that a larger L reduces errors due to circular convolution artifacts, resulting in better equalization performance. Since L_h is determined by the unknown system, L can be increased by choosing a larger frame-length L_x . The trade-off between L and equalization performance of MCEQ-F will be illustrated later in Section 6.4.

For frame-by-frame equalization, the received signals are first split into overlapping frames of length L_x with a frame shift N using a suitable window function. The frequency-domain representation $\underline{\mathbf{x}}_m(k) = [x_m(k, 1), \dots, x_m(k, L)]^T$ is obtained by taking the L -point DFT of the k th frame, $\mathbf{x}_m(k)$, where $L = L_x + L_h - 1$, i.e., $\underline{\mathbf{x}}_m(k) = \mathbf{F}_{L \times L}^H \mathbf{x}_m(k)$. The L -point DFT of $\mathbf{h}_m$ is computed similarly, from which the coefficients $\underline{h}_{m,l}(l)$ are obtained. The inverse filter $\underline{\mathbf{g}}_l$ for each frequency bin is found using (6.9), followed by frame-wise equalization in each frequency bin using

$$\hat{\underline{s}}(k, l) = \sum_{m=1}^M \underline{g}_{m,l}(l) \underline{x}_m(k, l). \quad (6.10)$$

The recovered frequency-domain source signal frames $\hat{\underline{\mathbf{s}}}(k) = [\hat{s}(k, 1), \dots, \hat{s}(k, L)]^T$ are then transformed into an $L_x \times 1$ time domain frame $\hat{\mathbf{s}}(k) = [\hat{s}(kN + L_x - 1), \dots, \hat{s}(kN)]$ by taking the inverse

Fig. 6.3: Equalization using MCEQ-F.

DFT, followed by truncation to the length L_x , as

$$\hat{\mathbf{s}}(k) = \mathbf{W}_{L_x \times L}^{01} \mathbf{F}_{L \times L} \hat{\mathbf{s}}(k), \quad (6.11)$$

where $\mathbf{W}_{L_x \times L}^{01} = [\mathbf{0}_{L_x \times L - L_x}, \mathbf{I}_{L_x \times L_x}]$. Reconstruction of $\hat{\mathbf{s}}(n)$ is then achieved using $\hat{\mathbf{s}}(k)$ as described after (3.23). Since spectral leakage has little effect on the equalization performance a rectangular window is used in this work and the N samples centered at each frame are chosen for reconstruction. It may be noted that, in practice, one may use $\mathbf{x}_m(k) = [x_m(kN), \dots, x_m(kN + L - 1)]^T$ to directly achieve $\hat{\mathbf{s}}(k) = [s(kN), \dots, s(kN + L_x - 1)]^T$. In this case, $\underline{h}_{m,l}(l)$ is equal to the l th DFT coefficient of $\mathbf{h}_m$ (not the complex conjugate). Figure 6.3 shows a conceptual block diagram of MCEQ-F.

The complexity of MCEQ-F is determined by the DFT length L . Because of the complex conjugate symmetry of the DFT, only $L/2$ or $1 + \lfloor L/2 \rfloor$ frequency bins need to be processed, respectively, for an even or odd values of L , where $\lfloor \cdot \rfloor$ denotes the floor operator. Counting each arithmetic operation as one floating point operation (flop), the maximum number of flops required for computing the inverse filters using MCEQ-F can be approximated by

$$C_{\text{MCEQ-F}} \approx 0.5ML (4L + M + M^2/3), \quad (6.12)$$

Fig. 6.4: Computational complexity comparison in flops.

where the first term corresponds to the complexity involved in the direct computation of L -point DFT sequence, while the remaining terms correspond to the number of flops required to compute the equalization filters for $L/2$ frequency bands using the least-squares approach. In practice, application of the computationally-efficient fast Fourier transform (FFT) for the computation of the DFT significantly reduces the complexity further, as will be seen in Section 6.4. The number of flops required for MINT is approximately

$$C_{\text{MINT}} \approx (ML_g)^2(L_h + L_g - 1) + (ML_g)^3/3, \quad (6.13)$$

while SB-MCEQ requires

$$C_{\text{SB-MCEQ}} \approx 2N_s \left\{ (ML'_g)^2(\tilde{L} + L'_g - 1) + L_r \tilde{L}^2 + [(ML'_g)^3 + \tilde{L}^3]/3 \right\}, \quad (6.14)$$

where $L_r = \lceil (L_h + L_{\text{pr}} - 1)/N_d \rceil$, $\tilde{L} = L_r - \lceil L_{\text{pr}}/N_d \rceil + 1$, and $L'_g = \lceil (\tilde{L} - 1)/(M - 1) \rceil$ [195]. These computational complexities are illustrated in Fig. 6.4 as a function of L_h . In this comparison, $L_g = L_c$ in (6.13) where L_c has been defined in (3.8), $L = 2L_h - 1$ in (6.12), and $N_d = 24$, $N_s = 32$, and $L_{\text{pr}} = 512$ in (6.14). It can be observed that, even with the computationally expensive direct DFT computation, MCEQ-F achieves a significant reduction in computational complexity compared to MINT and SB-MCEQ.

The equalization performance of MCEQ-F is illustrated in Fig. 6.5(e) using the same simulation setup as that used to generate Fig. 3.6. Similar to MCEQ-T and MCEQ-KL as described in Chapter 3, these

results are obtained by performing equalization of $\mathbf{h}_m$ using inverse filters obtained from the perturbed AIRs $\hat{\mathbf{h}}_m$ with $L = 2L_h - 1$. With reference to (3.50) and [217] for the definition of sparseness measure ξ and spectral flatness measure ϕ , the MCEQ-F algorithm with $\xi = 0.998$ and $\phi \approx 0$ dB results in better equalization than the other algorithms. This shows that MCEQ-F performs equalization with good quality even in the presence of AIR mismatch, while offering a significant reduction in computational complexity. It may be noted that MCEQ-F achieves modestly better equalization performance compared to MCEQ-T_{opt}. For MCEQ-T_{opt}, equalization is achieved using a single set of inverse filters estimated from the full-band time-domain AIRs. Equalization error associated with the n th coefficient in the equalized response depends on the inaccuracy in estimating the first n coefficients of the equalization filters due to convolution operation. On the other hand, MCEQ-F computes an independent set of inverse filters for each frequency bin thereby achieving equalization of each bin independent of each other. This yields a flatter spectrum for the equalized response resulting in better equalization as can be seen from Fig. 6.5.

6.3 The proposed partitioned PMCEQ-F algorithm

It can be noted that the complexity of MCEQ-F is proportional to L . With high sampling rates, L_h (and consequently L) will be large. This implies that complexity will be increased and to address this, the proposed partitioned MCEQ-F (PMCEQ-F) algorithm partitions AIRs as well. Each AIR is partitioned into P non-overlapping segments of length L'_h as $\mathbf{h}_m(p) = [h_m(pL'_h + 1), \dots, h_m(pL'_h + L'_h)]^T, p = 0, \dots, P - 1$, where p is the AIR frame index and $P = \lceil L_h/L'_h \rceil$. Defining $L' = L_x + L'_h - 1$ and the $L_x \times L'$ filtering matrix $\mathbf{H}_{m,p}$ of $\mathbf{h}_m(p)$ in (6.15), the relationship in (3.13) can be represented using partitioned convolution as

$$\mathbf{x}_m(k) = \sum_{p=0}^{P-1} \mathbf{H}_{m,p} \mathbf{s}(k-p), \quad (6.16)$$

where $\mathbf{s}(k-p) = \{s[(k-p)N + L_x - 1], \dots, s[(k-p)N + L_x - L']\}^T$ is now a $L' \times 1$ vector. It may be noted from (6.16) that a frame shift of $N = L'_h$ is desired when partitioning the received signals in order to faithfully represent the convolution relation using the partitioned overlap-save method, where $L'_h \leq L_x/2$. By applying the $L' \times L'$ transform matrix $\mathbf{F}_{L' \times L'} = [\mathbf{f}_1, \dots, \mathbf{f}_{L'}]$, (6.16) is transformed into the

Fig. 6.5: The frequency-domain magnitude response of equalized response $\hat{\mathbf{d}}$. (a) MINT. (b) SB-MCEQ. (c) MCEQ-T. (d) MCEQ-T_{opt}. (e) MCEQ-F

$$\mathbf{H}_{m,p} = \begin{bmatrix} h_m(pL'_h + 1) & \dots & h_m(pL'_h + L'_h) & 0 & 0 & \dots & 0 \\ 0 & h_m(pL'_h + 1) & \dots & h_m(pL'_h + L'_h) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & h_m(pL'_h + 1) & \dots & h_m(pL'_h + L'_h) \end{bmatrix} \quad (6.15)$$

frequency domain, i.e.,

$$\mathbf{F}_{L' \times L'}^H \begin{bmatrix} \mathbf{x}_m(k) \\ \mathbf{0}_{(L'_h-1) \times 1} \end{bmatrix} \approx \sum_{p=0}^{P-1} \left(\mathbf{F}_{L' \times L'}^H \begin{bmatrix} \mathbf{H}_{m,p} \\ \mathbf{0}_{(L'_h-1) \times L'} \end{bmatrix} \mathbf{F}_{L' \times L'} \right) \mathbf{F}_{L' \times L'}^H \mathbf{s}(k-p). \quad (6.17)$$

The reason for expressing (6.17) as an approximation is that the error introduced by the time-aliasing effects due to the circular convolution performed by multiplying two DFT sequences has not been compensated here. A common practice to mitigate this time-aliasing effect is to perform an inverse DFT of each product term, set the first $L'_h - 1$ points in the time domain to zero and perform a DFT, prior to summation. Since this error is negligible when $L'_h \leq L_x/2$, the approximation in (6.17) will not introduce significant distortion. For large L' , $\mathbf{F}_{L' \times L'}^H$ in (6.17) approximately diagonalizes $\left[\mathbf{H}_{m,p}^T, \mathbf{0}_{L' \times (L'_h-1)}\right]^T$, as described in Figs 6.1 and 6.2. In addition, the diagonal elements $\underline{h}_{m,l}(p, l)$, being the eigenvalues, are the uniformly scaled complex conjugate of DFT coefficients $\underline{h}_m(p, l)$ of $\mathbf{h}_m(p)$. Hence, (6.17) can be rewritten, for the l th frequency bin, as

$$\underline{x}_m(k, l) \approx \sum_{p=0}^{P-1} \underline{h}_{m,l}(p, l) \underline{s}(k-p, l). \quad (6.18)$$

With $\underline{s}(l) = [\underline{s}(0, l), \dots, \underline{s}(K-1, l)]^T$, and $\underline{x}_m(l) = [\underline{x}_m(0, l), \dots, \underline{x}_m(K-1, l)]^T$, (6.18) can be expressed using matrix notation by

$$\underline{x}_m(l) \approx \underline{\mathbf{H}}_m^T(l) \underline{s}(l), \quad (6.19)$$

where the $K \times K$ filtering matrix $\underline{\mathbf{H}}_m(l)$ is obtained from $\underline{\mathbf{h}}_m(l) = [\hat{\underline{h}}_{m,l}(0, l), \dots, \hat{\underline{h}}_{m,l}(P-1, l)]^T$ while $\underline{s}(k, l)$ and $\underline{x}_m(k, l)$, respectively, are the l th coefficient of the L' -point DFT sequence of $\mathbf{s}(k)$ and $\mathbf{x}_m(k)$.

With (6.19), the inverse filters $\hat{\underline{\mathbf{g}}}_m(l) = [\hat{\underline{g}}_{m,l}(0, l), \dots, \hat{\underline{g}}_{m,l}(P'-1, l)]^T$, each of length P' , can be estimated using $\hat{\underline{\mathbf{h}}}_m(l)$ to recover $\underline{s}(l)$, where $P' \geq \lceil (P-1)/(M-1) \rceil$, $P' \neq 0$. Defining the $(P+P'-1) \times MP'$ matrix $\hat{\underline{\mathbf{H}}}(l) = [\hat{\underline{\mathbf{H}}}_1^T(l), \dots, \hat{\underline{\mathbf{H}}}_M^T(l)]$, where $\hat{\underline{\mathbf{H}}}_m(l)$ is of dimension $P' \times (P+P'-1)$, the $MP' \times 1$ concatenated inverse filter $\hat{\underline{\mathbf{g}}}_m(l)$ can be computed by employing MINT as,

$$\begin{aligned} \hat{\underline{\mathbf{g}}}(l) &= [\hat{\underline{\mathbf{g}}}_1^T(l), \dots, \hat{\underline{\mathbf{g}}}_M^T(l)]^T \\ &= [\hat{\underline{\mathbf{H}}}(l)]^+ \mathbf{d}_p, \end{aligned} \quad (6.20)$$

where $\mathbf{d}_p = [1, \mathbf{0}_{1 \times (P+P'-2)}]^T$ is a $(P+P'-1) \times 1$ vector. Applying $\hat{\underline{\mathbf{g}}}_m(l)$ on $\underline{x}_m(l)$, frequency

Fig. 6.6: Equalization using PMCEQ-F.

bin-wise equalization can be achieved using

$$\hat{\underline{s}}(l) = \sum_{m=1}^M \hat{\underline{G}}_m^T(l) \underline{x}_m(l), \quad (6.21)$$

where $\hat{\underline{s}}(l)$ is the estimate of $\underline{s}(l)$, and $\hat{\underline{G}}_m(l)$ is the $K \times K$ filtering matrix of $\underline{g}_m(l)$. The estimated frame $\hat{\underline{s}}(k)$ is obtained from $\hat{\underline{s}}(l)$ which can be constructed using $\hat{\underline{s}}(l)$. It may be noted that when $L'_h = L_h$ PMCEQ-F is equivalent to MCEQ-F. A graphical illustration of equalization using PMCEQ-F is provided in Fig. 6.6.

For each frequency bin, the dimension of $\hat{\underline{H}}(l)$ in (6.20) is $(P + P' - 1) \times MP'$, which is significantly less than the time-domain full-band version. Similar to (6.12), the computational complexity associated with PMCEQ-F with direct DFT computation can be expressed in terms of flops using

$$C_{\text{PMCEQ-F}} \approx 0.5L' (4ML' + (2P' - 1)(MP')^2 + (MP)^3/3). \quad (6.22)$$

Figure 6.4 shows the complexity of the proposed PMCEQ-F algorithm, computed using (6.22), for $L'_h = L_h/4$ and $L_h/8$, where $L_x = 2L'_h$. It may be noted that PMCEQ-F can achieve equalization with less complexity than the proposed MCEQ-F algorithm by selecting appropriate values for L'_h and L_x . However, there is a trade-off between complexity and performance, as will be illustrated in next Section.

Fig. 6.7: Average processing time comparison. (a) SB-MCEQ. (b) MCEQ-F, configured to achieve the same SRR_{seg} performance as SB-MCEQ. (c) MCEQ-F with $L_x = 2L_h$ and $N = L_h$.

Fig. 6.8: SRR_{seg} for speech signal equalized with MCEQ-F, and SB-MCEQ.

6.4 Simulation results

6.4.1 Performance comparison

Comparative evaluation of MCEQ-F and SB-MCEQ using simulated AIRs is now presented. The same speech signal used in Section 3.5 is used as the source signal. Here, the comparison is made between the two computationally-efficient algorithms, MCEQ-F and SB-MCEQ. The MINT algorithm has not been included in this comparison due to its high computational complexity- see Fig. 6.4. In this work, the FFT is applied for computing the DFT. Twenty sets of AIRs were generated, where $1000 \leq L_h \leq 10000$, and the same simulation setup and evaluation metric as described in Section 3.5 were used. Each simulation is averaged over these 20 sets. The parameters of SB-MCEQ were selected according to that described in [195], and are the same as those used in the computational complexity calculation presented in Section 6.2. In order to compare the computational complexity, MCEQ-F was configured to give approximately the same SRR_{seg} as SB-MCEQ for each L_h under ideal noiseless condition with no

system mismatch. This is achieved by selecting appropriate values for L_x and N for each L_h considered in the simulation. The average processing time (in seconds, computed using an Intel Core i5-2400 3.1-GHz processor with 8 GB RAM) was then noted. Results presented in Fig. 6.7 show that MCEQ-F can achieve the same SRR_{seg} as SB-MCEQ with a significant reduction in processing time. It is worth noting that, as opposed to direct computation of the DFT, the FFT can achieve a significant reduction in computational complexity.

To compare the equalization performance with $\text{SNR} = 30$ dB and $\mathcal{M}_m = -40$ dB, equalization was performed for each set of simulated AIRs, and the average SRR_{seg} is plotted in Fig. 6.8. The average processing time required for MCEQ-F with $L_x = 2L_h$ and $N = L_h$ to achieve equalization is also shown in Fig. 6.7. It is interesting to see that, although SB-MCEQ can yield satisfactory results in a noiseless scenario as illustrated in [195], it suffers performance degradation in the given noisy environment. Comparing Figs. 6.7(a) and (c) it can be seen that, to achieve an average performance improvement of 29.9 dB over SB-MCEQ in Fig. 6.8, MCEQ-F requires less processing time.

6.4.2 Effect of L

Next the effect of L on the equalization performance of MCEQ-F is examined in terms of SRR_{seg} . In this simulation, twenty sets of AIRs, generated using the same setup as described in Section 3.5 with $L_h = 1600$ were used. The reverberant signals were then equalized using MCEQ-F for different L_x with $N = L_x/2$, $\mathcal{M}_m = -40$ dB, and $\text{SNR} = 30$ dB. The SRR_{seg} value corresponding to each $L = L_x + L_h - 1$, averaged over twenty sets of AIRs, is shown in Fig. 6.9. It can be seen that the performance of MCEQ-F increases with L , but beyond $L \approx 6.4 \times 10^3$, i.e., $L_x \approx 4.8 \times 10^3$, its performance levels off under the given condition.

6.4.3 Robustness to AIR estimation errors

In this simulation, the robustness of MCEQ-F to inaccurate AIR estimates in the presence of noise is illustrated. In these simulations, perfect knowledge of L_h is assumed. Synthetic AIRs $\mathbf{h}_m$ with $L_h = 1600$, generated at twenty different positions, as described in Section 3.5 were used for this simulation. Inverse filters were estimated from $\hat{\mathbf{h}}_m$ for each $\mathcal{M}_m$ using MCEQ-F, MCEQ-T_{opt}, MINT, and SB-

Fig. 6.9: Effect of L on SRR_{seg} of MCEQ-F.

MCEQ, from which equalization of the noisy $x_m(n)$ was then performed, where $\text{SNR} = 30$ dB. The SRR_{seg} values averaged over the twenty sets of AIRs are plotted against $\mathcal{M}_m$ in Fig. 6.10 for each algorithm.

Figure 6.10(a) is obtained by configuring MCEQ-F with $L_x = 2L_h$ and $N = L_h$ as described for Fig. 6.7, while Fig. 6.10(b) is obtained by configuring MCEQ-F to achieve approximately the same SRR_{seg} as SB-MCEQ in the absence of noise when $\hat{\mathbf{h}}_m = \mathbf{h}_m$. It can be observed that MCEQ-F achieves higher SRR_{seg} performance, even when $\mathcal{M}_m = 0$ dB, whereas, for this simulation with additive noise, SB-MCEQ achieves the lowest SRR_{seg} . Across these simulations, it is noted that MINT achieves an average SRR_{seg} improvement of 18.4 dB over that of the unprocessed reverberant speech signal. On the other hand, MCEQ-F achieves an average improvement of 26.2 dB and 20.5 dB for Figs. 6.10(a) and (b), respectively, showing better equalization performance with lower computational complexity in the presence of AIR inaccuracies. The MCEQ-T_{opt} algorithm also exhibits higher robustness achieving an average improvement of 25.8 dB.

The robustness of the algorithms to inaccurate estimation of AIR order L_h is also investigated under noisy conditions with $\text{SNR} = 30$ dB. The inverse filters were estimated for orders $\hat{L}_h \leq L_h$ using the same test configurations described above with $L_h = 1600$ and $\mathcal{M}_m = -40$ dB. Here, $\hat{L}_h$ is defined as the estimate of L_h . The average SRR_{seg} value of the equalized speech signal obtained with each algorithm is shown in Fig. 6.11 as a function of $\hat{L}_h$. Figures 6.11(a) and (b) were obtained for MCEQ-F with the same configurations as used for Figs. 6.10(a) and (b), respectively. It is shown that the SRR_{seg} value decreases when $\hat{L}_h < L_h$. However, compared to MINT and SB-MCEQ, MCEQ-F and MCEQ-T_{opt} exhibit higher robustness to such order mismatches, as can be observed from Fig. 6.11.

Fig. 6.10: Robustness to AIR mismatch: SRR_{seg} for speech equalized with (a) MCEQ-F with $L_x = 2L_h$ and $N = L_h$, (b) MCEQ-F with same SRR_{seg} as SB-MCEQ when $\hat{\mathbf{h}}_m = \mathbf{h}_m$, (c) MCEQ- T_{opt} , (d) SB-MCEQ, (e) MINT, and (f) original reverberant speech signal averaged across all channels.

Fig. 6.11: Robustness to AIR order estimation inaccuracy: SRR_{seg} for speech equalized with (a) MCEQ-F with $L_x = 2L_h$ and $N = L_h$, (b) MCEQ-F with same SRR_{seg} as SB-MCEQ when $\hat{\mathbf{h}}_m = \mathbf{h}_m$, (c) MCEQ- T_{opt} , (d) SB-MCEQ, (e) MINT, and (f) original reverberant speech signal averaged across all channels.

6.4.4 PMCEQ-F

In this simulation, the tradeoff between complexity and equalization performance of the PMCEQ-F algorithm is illustrated using a set of five recorded AIRs from the MARDY database. To illustrate the feasibility of equalizing very long AIRs, $L_h = 48 \times 10^3$ samples were used. Noise was added to achieve $\text{SNR} = 30$ dB and $\mathcal{M}_m = -40$ dB. To compare the complexity of the algorithms, the processing time (in seconds) required to achieve the equalized signal, averaged over ten runs is also displayed in Table 6.1. The MCEQ-F algorithm, with $L_x = 2L_h$ and $N = L_h$ achieves $\text{SRR}_{\text{seg}} = 18.8$ dB with an average processing time of 20.7 s. The SRR_{seg} achieved by PMCEQ-F with different L'_h is shown in Table 6.1 along with the average processing time where $L_x = 2L'_h$. When $L'_h = L_h$ PMCEQ-F is equivalent to MCEQ-F and achieves $\text{SRR}_{\text{seg}} = 18.8$ dB with an average processing time of 20.7 s. Note from Table 6.1

Table 6.1: Performance of PMCEQ-F with different L'_h .

L'_h	L'	SRR_{seg} (dB)	Proc. Time (s)
L_h (MCEQ-F)	143999	18.8	20.7
$L_h/2$	71999	18.8	11.5
$L_h/4$	35999	18.12	6.3
$L_h/8$	17999	16.64	4.9
$L_h/12$	11999	16.3	4.6
$L_h/16$	8999	15.72	5.4
$L_h/20$	7199	15.21	6.4

that PMCEQ-F with $L'_h = L_h/2$ shows approximately the same equalization performance as MCEQ-F employing full length AIRs while reducing the processing time by approximately 44%. For $L'_h = L_h/12$, the PMCEQ algorithm reduces processing time by approximately 78% with a degradation of 2.5 dB in SRR_{seg} . However, the processing time begins to increase as L'_h is reduced further while SRR_{seg} still decreases. This is because, although L' is small, the complexity involved in the inversion of $\hat{\mathbf{H}}(l)$ becomes significant for such small values of L'_h since the dimension of $\hat{\mathbf{H}}(l)$ is inversely proportional to L'_h . Hence, with PMCEQ-F good equalization of large-order AIRs with reduced complexity can be achieved by choosing suitable values for L'_h and L_x .

6.4.5 Dereverberation using BCI and MCEQ

Finally, results achieved for recorded AIRs estimated using the normalized multichannel fast least-mean-square (NMCFLMS) algorithm [153] are presented in Table 6.2. Dereverberation is achieved by applying the inverse filters obtained using these estimated AIRs. To compare the performance with MINT and SB-MCEQ, which have significantly higher complexity than MCEQ-F, a set of five AIRs, recorded in a room of dimension $6.5 \text{ m} \times 8.75 \text{ m} \times 2.65 \text{ m}$ at Nanyang Technological University, Singapore, was used. In this simulation, $f_s = 8 \text{ kHz}$, and a microphone spacing of 14 cm was used. The AIRs with $T_{60} \approx 300 \text{ ms}$ were truncated to $L_h = 1024$. The speech signal $s(n)$ was convolved with these AIRs and, in this simulation, a relatively higher $\text{SNR} = 50 \text{ dB}$ was selected to prevent NMCFLMS from misconverging [156]. The AIRs were then estimated from $x_m(n)$ and equalization was performed on the reverberant speech with the estimated $\hat{\mathbf{h}}_m$ using MINT, SB-MCEQ, MCEQ- T_{opt} , and MCEQ-F. To evaluate the dereverberation performance, the SRR_{seg} found by averaging the SRR_{seg} across all channels is compared. The AIR

Table 6.2: Dereverberation performance analysis.

Methods	SRR_{seg} (dB)	PESQ
Clean speech	∞	4.5
Reverberant speech	-4.9	1.6
Speech recovered using MINT	16.9	4.1
Speech recovered using SB-MCEQ	4.3	2.8
Speech recovered using MCEQ- T_{opt}	18.2	4.3
Speech recovered using MCEQ-F	18.8	4.4

estimation error is quantified using the normalized projection misalignment (NPM) [224]. The results obtained for $\text{NPM} = -9.12$ dB are presented in Table 6.2. The tabulated results show that, MCEQ-F outperforms MINT, SB-MCEQ as well as MCEQ- T_{opt} in terms of equalization quality. Figure 6.12 displays the waveform and spectrogram of the original source signal, reverberant signal received by the first microphone and the recovered signals. The temporal and spectral energy spreading caused by reverberation is obvious from 0 to 1 s in the spectrogram of the reverberant signal shown in Fig. 6.12 (b). It can be seen from Fig. 6.12 (d) that reverberation effects are still present in the signal recovered by SB-MCEQ. It can also be noted from the spectrogram that low frequency components (0 to 1 kHz) in the signal equalized by SB-MCEQ suffer from temporal spreading which is not significant in the reverberant signal. On the other hand, MINT, MCEQ- T_{opt} and MCEQ-F manage to recover the original temporal and spectral structures of the signal yielding equalized signals that are close to the source signal.

6.5 Conclusion

Although adaptive algorithms proposed in Chapters 4 and 5 can achieve equalization with reduced complexity, they require many iterations to converge. Hence they are not suitable for applications which require fast processing. To achieve fast equalization of AIRs, a frequency-domain MCEQ algorithm has been derived based on the generalized MCEQ framework defined in Chapter 3. The proposed MCEQ-F algorithm achieves significant reduction in the computational complexity compared to MINT and SB-MCEQ. Simulation results show that MCEQ-F is able to equalize high-order AIRs without noticeable distortion. The MCEQ-F algorithm also exhibits higher robustness to AIR inaccuracies while yielding stable AIR inversion. The proposed partitioned MCEQ-F (PMCEQ-F) algorithm can achieve equaliza-

Fig. 6.12: Speech dereverberation performance comparison: Waveform and spectrogram of (a) Original source signal, (b) Reverberant signal at channel 1, (c) signal recovered using MINT, (d) signal recovered using SB-MCEQ, (e) signal recovered using MCEQ- T_{opt} and (f) signal recovered using MCEQ-F.

tion with further reduction in complexity at the expense of modest degradation in performance when compared to MCEQ-F. The PMCEQ-F algorithm achieves this complexity reduction by partitioning the AIRs into segments, which in turn reduces the required DFT length.

Chapter 7

Adaptive Multichannel Equalization in the Frequency Domain

7.1 Introduction

In Chapter 6, complexity reduction in the frequency domain is exploited to derive the MCEQ-F algorithm. It has been shown that the proposed algorithm achieves equalization of high-order AIRs with significant reduction in computational complexity. Moreover, MCEQ-F exhibits robustness to channel estimation errors while outperforming existing equalization algorithms.

As already mentioned, application of adaptive filtering techniques can further improve the computational efficiency of MCEQ algorithms. In this chapter, computational efficiency of adaptive algorithms is incorporated into MCEQ-F to achieve further reduction in complexity. First, a cost function is derived from the MCEQ-F relation $\underline{\mathbf{h}}_l^T \underline{\mathbf{g}}_l = 1$. The frequency-domain adaptive MCEQ (FDA-MCEQ) algorithm proposed in Section 7.2 estimates the inverse filters by minimizing this cost function iteratively. This algorithm is further extended such that the algorithm converges to the desired solution in a single iteration. To achieve this, an optimal step-size for the first iteration is derived from the initial error function. It will be shown that, with the optimal step-size, the proposed FDA-MCEQ_{fast} algorithm can converge in a single step. Simulation results presented in Section 7.3 will show that the proposed algorithms are the computationally efficient implementations of MCEQ-F. The conclusion of this chapter is presented in Section 7.4.

7.2 The FDA-MCEQ and FDA-MCEQ_{fast} algorithms

In Chapter 6, the overlap-save method has been employed to describe the effect of reverberation in the frequency domain. Received signals are first partitioned into K overlapping frames of length L_x with a frame shift N . In this chapter, for the m th channel, $\mathbf{x}_m(k) = [x_m(kN), \dots, x_m(kN + L - 1)]^T$, which was defined after (6.11). To realize linear convolution, $L - L_x$ zeros are appended to each frame and L -point DFT is computed where $L = L_x + L_h - 1$. The DFT of the zero-appended k th frame $\mathbf{x}_m(k) = [x_m(kN), \dots, x_m(kN + L_x - 1), \mathbf{0}_{1 \times L - L_x}]^T$ is computed using

$$\begin{aligned}\underline{\mathbf{x}}_m(k, l) &= \mathbf{F}_{L \times L}^H \mathbf{x}_m(k) \\ &= [\underline{x}_m(k, 1), \dots, \underline{x}_m(k, L)]^T,\end{aligned}\tag{7.1}$$

where the underscore denotes for the frequency domain. As shown in Chapter 6, neglecting the scaling factor,

$$\underline{x}_m(k, l) = \underline{s}(k, l) \underline{h}_{m, l}(l),\tag{7.2}$$

where $\underline{h}_{m, l}(l)$ is the l th DFT coefficient $\underline{h}_m(l)$ of $\mathbf{h}_m$. With these definitions and defining $\underline{\mathbf{s}}(k) = [\underline{s}(k, 1), \dots, \underline{s}(k, L)]^T$ as the DFT sequence of the $L \times 1$ zero-appended clean speech frame $\mathbf{s}(k) = [s(kN), \dots, s(kN + L_x - 1), \mathbf{0}_{1 \times L - L_x}]^T$,

$$\underline{\mathbf{x}}_m(k) = \underline{\mathbf{s}}(k) \odot \underline{\mathbf{h}}_m,\tag{7.3}$$

where $\underline{\mathbf{x}}_m(k) = [\underline{x}_{m, 1}(k, 1), \dots, \underline{x}_{m, L}(k, L)]^T$, $\underline{\mathbf{h}}_m = [\underline{h}_m(1), \dots, \underline{h}_m(L)]^T$ and $\odot$ denotes the Schür (Hadamard) product. Defining $\oslash$ as the Schür division, $\underline{\mathbf{s}}(k)$ may not be estimated from (7.3) directly using $\widehat{\underline{\mathbf{s}}}(k) = \underline{\mathbf{x}}_m(k) \oslash \widehat{\underline{\mathbf{h}}}_m$ since this may result in large spectral peaks in $\widehat{\underline{\mathbf{s}}}(k)$ due to possible spectral nulls in $\widehat{\underline{\mathbf{h}}}_m$. This may cause narrowband noise amplification. In addition, AIRs are often non-minimum phase and therefore inversion of a single-channel AIR is not feasible as described in Chapter 2. Hence, it is proposed here, similar to the time-domain adaptive MCEQ algorithms, to employ the gradient method for the estimation of inverse filters $\widehat{\underline{\mathbf{g}}}_m = [\widehat{g}_m(1), \dots, \widehat{g}_m(L)]^T$ iteratively, using $\widehat{\underline{\mathbf{h}}}_m = [\widehat{h}_m(1), \dots, \widehat{h}_m(L)]^T$ obtained from the DFT of the erroneous estimate $\widehat{\mathbf{h}}_m$. Ideally, as mentioned in Chapter 6, an infinitely

long $\hat{\underline{\mathbf{g}}}_m$ is required to completely avoid circular convolution artifacts since the optimal $\mathbf{g}_m$ is an infinite length filter [223]. Since an infinite length DFT sequence is not feasible, $L > L_h$ is used to mitigate the effects of circular convolution. The estimated inverse filters are then used to equalize $\underline{\mathbf{x}}_m(k)$ so as to estimate $\underline{\mathbf{s}}(k)$ from which $\mathbf{s}(n)$ is recovered. Exploiting the complex conjugate property of DFT, only $\hat{\underline{g}}_m(l), l = 1, \dots, L_F$, need to be estimated since the remaining coefficients are obtained from the time-reversed complex conjugate of it. Here, $L_F = 1 + \lfloor L/2 \rfloor$ where $\lfloor \cdot \rfloor$ denotes the floor operator.

Defining $\hat{\underline{\mathbf{d}}} = \sum_{m=1}^M \hat{\underline{\mathbf{g}}}_m \odot \hat{\underline{\mathbf{h}}}_m$ and the $L_F \times 1$ vector $\mathbf{1} = [1, \dots, 1]^T$, the condition

$$\hat{\underline{\mathbf{d}}} = \mathbf{1} \quad (7.4)$$

should be satisfied in the frequency domain for perfect equalization, where $\hat{\underline{\mathbf{g}}}_m$ and $\hat{\underline{\mathbf{h}}}_m$ consist of only L_F coefficients. A cost function can be formulated, for each iteration n , as

$$\begin{aligned} \underline{\mathbf{J}}(n) &= \left| \mathbf{1} - \sum_{m=1}^M \hat{\underline{\mathbf{g}}}_m(n) \odot \hat{\underline{\mathbf{h}}}_m \right|^2 \\ &= [\underline{J}(n, 1), \dots, \underline{J}(n, L_F)]^T, \end{aligned} \quad (7.5)$$

where $|\cdot|^2$ denotes element-wise magnitude square. Taking the partial derivative of

$$\underline{J}(n, l) = \left| 1 - \sum_{m=1}^M \hat{\underline{g}}_m(n, l) \hat{\underline{h}}_m(l) \right|^2, l = 1, \dots, L_F, \quad (7.6)$$

with respect to $\hat{\underline{g}}_m^*(n, l)$,

$$\begin{aligned} \nabla_m \underline{J}(n, l) &= \frac{\partial \underline{J}(n, l)}{\partial \hat{\underline{g}}_m^*(n, l)} \\ &= -\underline{e}(n, l) \hat{\underline{h}}_m^*(l), \end{aligned} \quad (7.7)$$

where the scalar

$$\begin{aligned} \underline{e}(n, l) &= 1 - \sum_{m=1}^M \hat{\underline{g}}_{m,l}(n) \hat{\underline{h}}_{m,l}(l) \\ &= 1 - [\hat{\underline{\mathbf{g}}}_l(n)]^T \hat{\underline{\mathbf{h}}}_l \end{aligned} \quad (7.8)$$

with $\hat{\mathbf{h}}_l = [\hat{h}_1(l), \dots, \hat{h}_M(l)]^T$ and $\hat{\mathbf{g}}_l(n) = [\hat{g}_1(n, l), \dots, \hat{g}_M(n, l)]^T$ being the M -channel concatenated inverse filter of the l th frequency bin. Subsequently, the update equation for FDA-MCEQ is obtained as

$$\hat{\mathbf{g}}_m(n+1) = \hat{\mathbf{g}}_m(n) - \boldsymbol{\mu}(n) \odot \nabla_m \mathbf{J}(n), \quad (7.9)$$

where $\nabla_m \mathbf{J}(n) = [\nabla_m J(n, 1), \dots, \nabla_m J(n, L_F)]^T$, and $\boldsymbol{\mu}(n) = [\mu(n, 1), \dots, \mu(n, L_F)]^T$ with each of its elements being the step-size for each frequency bin. The elements in $\hat{\mathbf{g}}_m(n+1)$ is given by

$$\hat{g}_m(n+1, l) = \hat{g}_m(n, l) + \mu \nabla_m J(n, l). \quad (7.10)$$

It is well known that, for a constant μ , the convergence rate of an adaptive algorithm depends on the input sequence [35]. In order to achieve stable convergence irrespective of the input sequence, the gradient in (7.10) is normalized using the power of the l th subspace input sequence $\hat{\mathbf{h}}_l$ giving

$$\hat{g}_m(n+1, l) = \hat{g}_m(n, l) + \frac{\mu}{\|\hat{\mathbf{h}}_l\|_2^2 + \delta} \nabla_m J(n, l), \quad (7.11)$$

where δ is a small value, introduced to avoid the numerical errors when $\|\hat{\mathbf{h}}_l\|_2^2 \approx 0$. Figure 7.1 presents a diagrammatic illustration of the proposed FDA-MCEQ algorithm.

Having derived the FDA-MCEQ algorithm, the next objective is to formulate the fast FDA-MCEQ (FDA-MCEQ_{fast}) which can converge to the desired solution in a single step. To achieve the above, with $\mathbf{e}(n) = [e(n, 1), \dots, e(n, L_F)]^T$, an expression for the step-size vector $\boldsymbol{\mu}'(0)$ can be derived using

$$\mathbf{e}(0) = \mathbf{1} - \sum_{m=1}^M \hat{\mathbf{g}}_m(0) \odot \hat{\mathbf{h}}_m, \quad (7.12)$$

where $\hat{\mathbf{g}}_m(0)$ is the initialization vector. Therefore, to estimate the inverse filters in a single step, each step-size $\mu(0, l)$ is computed so as to achieve $e(1, l) = 0$.

From (7.7), $\nabla_m J(0, l) = -e(0, l) \hat{h}_m^*(l)$ and using (7.9),

$$\hat{\mathbf{g}}_m(1) = \hat{\mathbf{g}}_m(0) - \boldsymbol{\mu}(0) \odot \nabla_m \mathbf{J}(0). \quad (7.13)$$

Fig. 7.1: Equalization using FDA-MCEQ.

Therefore the aim is to find $\boldsymbol{\mu}'(0) = [\mu'(0, 1), \dots, \mu'(0, L_F)]^T$ for (7.13) such that

$$\begin{aligned} \underline{\mathbf{e}}(1) &= \mathbf{1} - \sum_{m=1}^M \hat{\underline{\mathbf{g}}}_m(1) \odot \hat{\underline{\mathbf{h}}}_m \\ &= \mathbf{0}_{L_F \times 1}. \end{aligned} \quad (7.14)$$

Substituting (7.13) in (7.14) with $\boldsymbol{\mu}(0) = \boldsymbol{\mu}'(0)$,

$$\begin{aligned} \underline{\mathbf{e}}(1) &= \mathbf{1} - \sum_{m=1}^M \left[\hat{\underline{\mathbf{g}}}_m(0) - \boldsymbol{\mu}'(0) \odot \nabla_m \underline{\mathbf{J}}(0) \right] \odot \hat{\underline{\mathbf{h}}}_m \\ &= \mathbf{1} - \sum_{m=1}^M \hat{\underline{\mathbf{g}}}_m(0) \odot \hat{\underline{\mathbf{h}}}_m + \boldsymbol{\mu}'(0) \odot \sum_{m=1}^M \nabla_m \underline{\mathbf{J}}(0) \odot \hat{\underline{\mathbf{h}}}_m \\ &= \underline{\mathbf{e}}(0) + \boldsymbol{\mu}'(0) \odot \sum_{m=1}^M \nabla_m \underline{\mathbf{J}}(0) \odot \hat{\underline{\mathbf{h}}}_m. \end{aligned} \quad (7.15)$$

Equating (7.15) to zero and solving for $\boldsymbol{\mu}'(0)$,

$$\boldsymbol{\mu}'(0) = -\underline{\mathbf{e}}(0) \oslash \left(\sum_{m=1}^M \nabla_m \underline{\mathbf{J}}(0) \odot \hat{\underline{\mathbf{h}}}_m \right), \quad (7.16)$$

to achieve $\underline{\mathbf{e}}(1) = \mathbf{0}_{L_F \times 1}$. In addition, elements in $\mu'(0)$ can be expressed as

$$\mu'(0, l) = \frac{-\underline{\mathbf{e}}(0, l)}{[\nabla \underline{\mathbf{J}}_l(0)]^T \hat{\underline{\mathbf{h}}}_l} \quad (7.17)$$

with $\nabla \underline{\mathbf{J}}_l(0) = [\nabla_1 J(0, l), \dots, \nabla_M J(0, l)]^T = -\underline{\mathbf{e}}(0, l)(\hat{\underline{\mathbf{h}}}_l)^*$. It may be noted that division by zero will not occur in (7.17) since $\hat{\underline{\mathbf{h}}}_l \neq \mathbf{0}_{M \times 1}$ in practice [225]. In addition, $\nabla_m J(0, k) \neq 0$ since $\underline{\mathbf{e}}(0, k) \neq 0$ due to the fact that one does not initialize the algorithm with the true solution. Applying (7.16) in (7.13), $\hat{\underline{\mathbf{g}}}_m$ can be computed in a single step.

7.2.1 FDA-MCEQ_{fast} and Wiener solution

The relationship between FDA-MCEQ_{fast} and the Wiener solution is now presented. Defining $\underline{\mathbf{e}}_l^* = 1 - \hat{\underline{\mathbf{h}}}_l^H \hat{\underline{\mathbf{g}}}_l^*$ and using the orthogonality principle [35],

$$\begin{aligned} E(\hat{\underline{\mathbf{h}}}_l \underline{\mathbf{e}}_l^*) &= E[\hat{\underline{\mathbf{h}}}_l (1 - \hat{\underline{\mathbf{h}}}_l^H \hat{\underline{\mathbf{g}}}_l^*)] \\ &= \mathbf{0}_{L_F \times 1}, \end{aligned} \quad (7.18)$$

from which $\hat{\underline{\mathbf{h}}}_l = (\hat{\underline{\mathbf{h}}}_l \hat{\underline{\mathbf{h}}}_l^H)^+ (\hat{\underline{\mathbf{g}}}_l^*)^*$, and hence the Wiener solution is given by

$$\hat{\underline{\mathbf{g}}}_l^* = (\hat{\underline{\mathbf{h}}}_l^H)^+ \implies \hat{\underline{\mathbf{g}}}_l = (\hat{\underline{\mathbf{h}}}_l^T)^+ \quad (7.19)$$

which is the same as (6.9). For FDA-MCEQ_{fast}, the update equation for the l th frequency bin is given by,

$$\hat{\underline{\mathbf{g}}}_l(1) = \hat{\underline{\mathbf{g}}}_l(0) - \mu'(0, l) \nabla \underline{\mathbf{J}}_l(0). \quad (7.20)$$

Substituting $\mu'(0, l) = \underline{\mathbf{e}}(0, l) / [\hat{\underline{\mathbf{h}}}_l^T \nabla \underline{\mathbf{J}}_l(0)]$ in (7.20), the Wiener solution

$$\begin{aligned} \hat{\underline{\mathbf{g}}}_l(1) &= \hat{\underline{\mathbf{g}}}_l(0) + \underline{\mathbf{e}}(0, l) (\hat{\underline{\mathbf{h}}}_l^T)^+ \\ &= \hat{\underline{\mathbf{g}}}_l(0) + [1 - \hat{\underline{\mathbf{h}}}_l^T \hat{\underline{\mathbf{g}}}_l(0)] (\hat{\underline{\mathbf{h}}}_l^T)^+ \\ &= (\hat{\underline{\mathbf{h}}}_l^T)^+, \end{aligned} \quad (7.21)$$

Fig. 7.2: Convergence performance of FDA-MCEQ.

is obtained if the algorithm is initialized using $\hat{\underline{\mathbf{g}}}_l(0) = \mathbf{0}_{M \times 1}$. For $\hat{\underline{\mathbf{g}}}_l(0) \neq \mathbf{0}_{M \times 1}$, the algorithm still satisfies $\hat{\underline{\mathbf{h}}}_l^T \hat{\underline{\mathbf{g}}}_l(1) = 1$ and hence equalization is achieved.

As explained in Chapter 6, equalization in the frequency domain allows convolution to be implemented as element-wise complex multiplications and it does not require inversion of convolution matrices. This results in faster equalization. Compared to MCEQ-F, FDA-MCEQ_{fast} estimates $\hat{\underline{\mathbf{g}}}_l, l = 1, \dots, L_F$, in a single step using approximately $7ML_F$ complex multiplications and additions.

With $\hat{\underline{\mathbf{g}}}_m$, frame-by-frame equalization is achieved using

$$\begin{aligned} \hat{\underline{\mathbf{s}}}(k) &= \sum_{m=1}^M \hat{\underline{\mathbf{g}}}_m \odot \underline{\mathbf{x}}_m(k) \\ &= [\hat{\underline{\mathbf{s}}}(k, 1), \dots, \hat{\underline{\mathbf{s}}}(k, L)]^T. \end{aligned} \quad (7.22)$$

The estimated frames $\hat{\underline{\mathbf{s}}}(k)$ are obtained by taking the inverse DFT of $\hat{\underline{\mathbf{s}}}(k)$. Similar to other proposed algorithms which perform frame-by-frame processing, N coefficients $[\hat{\underline{\mathbf{s}}}(kN + \varrho L_x), \dots, \hat{\underline{\mathbf{s}}}(kN + \varrho L_x + N - 1)]^T$ from each frame are used for reconstruction in FDA-MCEQ, where $\varrho = (L_x - L_h)/(2L_x)$. To prevent information loss due to truncation, ϱL_x zeros are appended to the beginning and end of $\underline{\mathbf{x}}_m$ prior to partitioning. It is worth noting that a larger L_x with $N \leq L_x/2$ may yield better equalization performance at the cost of modest increase in the computational load.

Fig. 7.3: Convergence performance of FDA-MCEQ_{fast}.

7.3 Simulation results

Convergence performance of the algorithms is first illustrated. Since the normalized misalignment

$$\eta(n) = 20 \log_{10} \left(\frac{\|\mathbf{1} - \hat{\mathbf{d}}\|_2}{\|\mathbf{1}\|_2} \right) \text{ dB} \quad (7.23)$$

tends to $-\infty$ as the algorithms converge, the convergence performance is evaluated using

$$\eta'(n) = 20 \log_{10} \left(1 - \frac{\|\mathbf{1} - \hat{\mathbf{d}}\|_2}{\|\mathbf{1}\|_2} \right) \text{ dB}. \quad (7.24)$$

For perfect equalization, $\eta' = 0$ dB. A simulation was conducted using different sets of $M = 5$ synthetic AIRs generated from the method of images [215] for a room of dimension 8 m $\times$ 6 m $\times$ 5 m. The microphone array and source were positioned at a height of 1.6 m while the microphone spacing was 0.2 m. The AIRs of length $L_h = 4000, 8000, 12000$, and 16000 samples were generated with reverberation time $T_{60} = 0.25, 0.5, 0.75$ and 1 s, respectively, at a sampling frequency $f_s = 16$ kHz. FDA-MCEQ with $\mu = 0.5$ and FDA-MCEQ_{fast} were then employed to equalize $\hat{\mathbf{h}}_m = \mathbf{h}_m$. It can be noted that $\hat{\mathbf{g}}_m(0) = \mathbf{0}_{L_F \times 1}$ results $\eta'(0) = -\infty$. To avoid this $\hat{\mathbf{g}}_m(0) = \mathbf{1}_{L_F \times 1}$ is chosen for initialization. Two sets of eight recorded AIRs were also equalized. These AIRs were recorded inside a room of dimension 4.68 m $\times$ 11.68 m $\times$ 2.56 m at Nanyang Technological University, Singapore, with $f_s = 48$ kHz. The AIRs were 43200 samples long with an averaged estimated $T_{60} = 650$ ms. The convergence performance achieved with both synthetic and recorded AIRs are shown in Figs. 7.2

Fig. 7.4: Average SRR_{seg} for SB-MCEQ, FDA-MCEQ, MCEQ-F and FDA-MCEQ_{fast}.

and 7.3, respectively, for FDA-MCEQ and FDA-MCEQ_{fast}. For all cases FDA-MCEQ_{fast} converges in a single step achieving $\eta' = 0$ as expected. On the other hand, FDA-MCEQ required a few iterations to achieve $\eta' = 0$ dB.

The FDA-MCEQ_{fast} is now compared with MCEQ-F, FDA-MCEQ with $\mu = 0.5$, and SB-MCEQ in terms of equalization performance and the average processing time required to achieve dereverberation. Here, FDA-MCEQ_{fast} is initialized with $\hat{\mathbf{g}}_m(0) = \mathbf{0}_{L_F \times 1}$ so that it converges to the Wiener solution. This simulation was performed on an Intel Core i5-2400 3.1-GHz CPU with 8 GB of RAM and 64-bit operating system. A set of eight synthetic AIRs were generated at ten different positions for different L_h . The inverse filters were then estimated from $\hat{\mathbf{h}}_m$ with $\mathcal{M}_m = -30$ dB and equalization was subsequently performed on $x_m(n)$ at an SNR of 20 dB. The averaged SRR_{seg} and processing time across the ten sets of AIRs are presented in Figs. 7.4 and 7.5, respectively. MINT and A-MINT were not included in this simulation, respectively due to high complexity and slow convergence. It may also be noted that the average processing time is not a linear function of L_h . This is because, for each L_h , the Matlab heuristically selects a suitable FFT algorithm from its library (which may or may not be the optimal). Hence, it is recommended to compare the processing time with respect to an L_h . It can be seen from Fig. 7.4 that the performance of SB-MCEQ is poor in the presence of noise and AIR estimation error whereas FDA-MCEQ_{fast} and FDA-MCEQ, which achieve an average improvement of 26.8 dB and 25.8 dB, respectively, in SRR_{seg} over SB-MCEQ, yield better equalization performance.

For SB-MCEQ and MCEQ-F, the processing time increases significantly with L_h as can be seen

Fig. 7.5: Average processing time of (a) FDA-MCEQ, and MCEQ-F, (b) FDA-MCEQ_{fast}, and FDA-MCEQ, (c) FDA-MCEQ_{fast} and SB-MCEQ.

from Figs. 7.5(a) and (c); they require an average processing time of approximately 62.3 s and 4.2 s, respectively, when $L_h = 16000$. In Fig. 7.5(b), although FDA-MCEQ requires 0.36 s (corresponding to fifty-four iterations) to achieve an $\text{SRR}_{\text{seg}} = 16.6$ dB when $L_h = 16000$, FDA-MCEQ_{fast} only requires 0.14 s to achieve $\text{SRR}_{\text{seg}} = 17.8$ dB in a single step. This translates to an approximate 61% reduction in the processing time.

Lastly, $s(n)$ was convolved with eight recorded AIRs which have been downsampled to 16 kHz and truncated to $L_h = 1000$. White Gaussian noise was then added to achieve an $\text{SNR} = 50$ dB.

The AIRs were then estimated from $x_m(n)$ using the normalized multichannel fast least-mean-square (NMCFLMS) algorithm [153]. Dereverberation was achieved with inverse filters computed from these estimated AIRs using A-MINT with 1×10^4 iterations, SB-MCEQ and FDA-MCEQ_{fast}. The system mismatch for each channel is computed using the normalized projection misalignment (NPM) [224]. For this simulation, the average NPM across all channels was -12.3 dB. For the reverberant signal, $\text{SRR}_{\text{seg}} = -3$ dB was computed by averaging over the eight channels. The FDA-MCEQ_{fast} algorithm achieved $\text{SRR}_{\text{seg}} = 17.8$ dB while A-MINT and SB-MCEQ achieved 3.2 dB and 7.54 dB, respectively. This implies that FDA-MCEQ_{fast} achieves better speech dereverberation.

7.4 Conclusion

The MCEQ-F algorithm proposed in Chapter 6 achieves MCEQ with significant reduction in complexity. In this chapter, the FDA-MCEQ algorithm was proposed to further reduce the complexity involved in the frequency domain MCEQ. The algorithm computes the equalization filters in a least-square sense minimizing a cost function. With the aim of achieving further reduction in the processing time, the FDA-MCEQ_{fast} algorithm was also proposed. This algorithm computes an optimal step-size leading to the algorithm converging in a single iteration. It has been shown that FDA-MCEQ_{fast} converges to the Wiener solution derived for MCEQ-F in a single step when initialized with a null vector. Simulation results show that the proposed algorithms not only achieve significant reduction in computational complexity, but also exhibit, similar to MCEQ-F, higher robustness to channel estimation error.

Chapter 8

Blind Channel Identification for Speech Dereverberation

8.1 Introduction

Speech dereverberation via blind channel identification and MCEQ has been popular in the recent years. As described in Chapter 2, the performance of several BCI algorithms is largely affected by noise and developing a noise robust BCI algorithm is still a challenging problem. Although it is well-known that, under noisy conditions, LMS-based BCI algorithms such as NMCFLMS misconverge, the reason for this behavior has not been examined in detail. The effect of misconvergence due to noise on the estimated impulse response has been analyzed in [226]. This analysis shows that the AIR estimated after misconvergence will be the weighted sum of the eigenvectors of clean output data correlation matrix. It will be of high interest to understand why BCI algorithms such as NMCFLMS first converge towards the true solution after which they then misconverge.

In Section 8.2 of this chapter the cross-relation among channels is described and the popular NM-CFLMS algorithm that relies on this cross-relation is reviewed. A frequency-domain BCI algorithm that also exploits the interchannel cross-relation is proposed in Section 8.3. Unlike conventional MCFLMS algorithms, partitioned convolution in the frequency domain is applied here to establish a relationship for each frequency bin. A cost function is then derived for each frequency bin from the cross-relation. For the proposed partitioned multichannel frequency-domain LMS (PMCFLMS) algorithm the use of a variable step-size is proposed to achieve fast convergence. The proposed algorithm shows higher robustness to noise and allows one to partition the impulses responses into a number of segments and estimate them. Analysis on the convergence performance of PMCFLMS is also presented. More specifically, analysis

on the convergence behavior of the proposed PMCFLMS algorithm in the presence of noise presented in Section 8.4 explains why LMS-based algorithms misconverge under low SNR conditions. The analysis result then leads to the introduction of a sparseness constraint to the PMCFLMS algorithm. Simulation results presented in Section 8.6 demonstrate that the sparseness-constraint PMCFLMS (PMCFLMS-SC) algorithm proposed in Section 8.5 achieves better estimates of the AIRs. The PMCFLMS-SC algorithm shows improved convergence performance even under very low SNR conditions by the virtue of the sparseness constraint. The summary of this chapter is presented in Section 8.7.

8.2 Problem formulation

For the SIMO model depicted in Fig. 3.1, the m th channel received signal $x_m(n)$ is given by

$$\begin{aligned} x_m(n) &= h_m * s(n) + v_m(n) \\ &= y_m(n) + v_m(n), \quad m = 1, \dots, M, \end{aligned} \quad (8.1)$$

where $s(n)$ is the source signal, h_m is the m th channel AIR, $y_m(n) = h_m * s(n)$, and $v_m(n)$ is the additive noise at the m th channel. The signal model in (8.1) can be expressed in vector/matrix form as

$$\begin{aligned} \mathbf{x}_m(n) &= \mathbf{y}_m(n) + \mathbf{v}_m(n) \\ &= \mathbf{H}_m \mathbf{s}(n) + \mathbf{v}_m(n), \end{aligned} \quad (8.2)$$

where

$$\begin{aligned} \mathbf{x}_m(n) &= [x_m(n), x_m(n-1), \dots, x_m(n-L_h+1)]^T, \\ \mathbf{s}(n) &= [s(n), s(n-1), \dots, s(n-2L_h+2)]^T, \\ \mathbf{y}_m(n) &= [y_m(n), y_m(n-1), \dots, y_m(n-L_h+1)]^T, \\ \mathbf{v}_m(n) &= [v_m(n), v_m(n-1), \dots, v_m(n-L_h+1)]^T, \end{aligned} \quad (8.3)$$

and

$$\mathbf{H}_m = \begin{bmatrix} h_{m,0} & \dots & h_{m,L_h-1} & 0 & 0 & \dots & 0 \\ 0 & h_{m,0} & \dots & h_{m,L_h-1} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & h_{m,0} & \dots & h_{m,L_h-1} \end{bmatrix} \quad (8.4)$$

is an $L_h \times (2L_h - 1)$ filtering matrix constructed from the $L_h \times 1$ i th channel AIR

$$\mathbf{h}_m = [h_{m,0}, h_{m,1}, \dots, h_{m,L_h-1}]^T. \quad (8.5)$$

Under noiseless conditions,

$$\begin{aligned} \mathbf{x}_m(n) &= \mathbf{y}_m(n) \\ &= \mathbf{H}_m \mathbf{s}(n). \end{aligned} \quad (8.6)$$

The objective of a BCI algorithm is to estimate $\mathbf{h}_m$ from the received signals.

8.2.1 Review of NMCFLMS

The NMCFLMS algorithm [153] exploits the interchannel cross-relation of the received signals. Under noiseless condition, the cross-relation is given by [131]

$$\begin{aligned} x_m * h_q &= s * h_m * h_q \\ &= x_q * h_m, \quad m, q = 1, \dots, M, m \neq q, \\ \mathbf{x}_m^T(n) \mathbf{h}_q &= \mathbf{x}_q^T(n) \mathbf{h}_m. \end{aligned} \quad (8.7)$$

From this cross-relation an error signal is derived as

$$e_{mq}(n) = \begin{cases} \mathbf{x}_m^T(n) \hat{\mathbf{h}}_q(n) - \mathbf{x}_q^T(n) \hat{\mathbf{h}}_m(n) & m \neq q, \\ 0 & m = q. \end{cases} \quad (8.8)$$

For blind estimation of AIRs associated with a SIMO system, the following two conditions should be satisfied [153]:

- 1) The AIRs are coprime, i.e., they do not share any common zeros. If the channels are not coprime, the common factors across the channels may belong to either the input signal or AIRs.
- 2) The autocorrelation matrix $\mathbf{R}_{ss} = E[\mathbf{s}(n)\mathbf{s}^T(n)]$ of the source signal is a full rank matrix.

In addition, it is assumed that the $v_m(n), m = 1, \dots, M$, of variance σ_v^2 are uncorrelated and have a normal distribution $\mathcal{N}(0, \sigma_v^2)$, i.e., $E[v_m(n)v_q(n)] = 0, m \neq q$, and $E[v_m(n)v_m(n - n')] = 0, n \neq n'$. It is also assumed that $E[v_m(n)s(n)] = 0$.

The NMCFLMS algorithm estimates the AIRs by minimizing the cross-relation error in the frequency domain. The frequency-domain cross-relation error function is given by

$$\underline{\mathbf{e}}_{qm}(k) = \underline{\mathbf{D}}_{x_q}(k)\hat{\underline{\mathbf{h}}}_m(k) - \underline{\mathbf{D}}_{x_m}(k)\hat{\underline{\mathbf{h}}}_q(k), \quad (8.9)$$

where k is the frame index, the $2L \times 2L$ diagonal matrix

$$\underline{\mathbf{D}}_{x_q}(k) = \text{diag} [\mathbf{F}_{2L_h \times 2L_h}^H \mathbf{x}_q(k)] \quad (8.10)$$

while $\mathbf{x}_q(k) = [x_q(kL_h - L_h), \dots, x_q(kL_h + L_h - 1)]^T$. The unconstrained NMCFLMS update equation is summarized by

$$\hat{\underline{\mathbf{h}}}_m^{10}(k+1) = \hat{\underline{\mathbf{h}}}_m^{10}(k) - \mu[\mathcal{P}_m(k) + \delta \mathbf{I}_{2L_h \times 2L_h}]^{-1} \times \sum_{q=1}^M \underline{\mathbf{D}}_{x_q}^*(k) \underline{\mathbf{e}}_{qm}^{01}(k), \quad (8.11)$$

where μ is the step-size, δ is the regularization parameter and

$$\hat{\underline{\mathbf{h}}}_m^{10}(k) = \mathbf{F}_{2L_h \times 2L_h}^H \hat{\underline{\mathbf{h}}}_m^{10}(k) \quad (8.12)$$

$$\underline{\mathbf{e}}_{qm}^{01}(k) = \mathbf{F}_{2L_h \times 2L_h}^H \left\{ \mathbf{0}_{1 \times L_h}, [\mathbf{F}_{L_h \times L_h} \underline{\mathbf{e}}_{qm}(k)]^T \right\}^T, \quad (8.13)$$

$$\mathcal{P}_m(k) = \varsigma \mathcal{P}_m(k-1) + (1 - \varsigma) \sum_{q=1, q \neq m}^M \underline{\mathbf{D}}_{x_q}^*(k) \underline{\mathbf{D}}_{x_q}(k), \quad (8.14)$$

while

$$\begin{aligned}\widehat{\mathbf{h}}_m^{10}(k) &= [\widehat{\mathbf{h}}_m^T(k), \mathbf{0}_{1 \times L_h}]^T \\ &= \mathbf{W}_{2L_h \times L_h}^{10} \widehat{\mathbf{h}}_m(k),\end{aligned}\tag{8.15}$$

$$\mathbf{W}_{2L_h \times L_h}^{10} = [\mathbf{I}_{L_h \times L_h}, \mathbf{0}_{L_h \times L'_h}]^T\tag{8.16}$$

and $\varsigma = [1 - 1/(3L_h)]^{L_h}$ is the forgetting factor. As with any adaptive algorithm, an initialization is required for $\widehat{\mathbf{h}}_m(0)$. Since $\mathbf{h}_m(n)$ is often unknown, a common method for initializing the adaptive filters is to set the filter coefficients such that

$$\widehat{\mathbf{h}}_m(0) = [1 \ \mathbf{0}_{1 \times (L-1)}]^T,\tag{8.17}$$

for $m = 1, \dots, M$ [153]. The overall concatenated filter coefficients $\mathbf{h}(0) = [\mathbf{h}_1^T(0), \dots, \mathbf{h}_M^T(0)]^T$ are then normalized by $\sqrt{M}$.

Under noiseless and high SNR conditions NMCFLMS yields AIR estimates that are close to the actual ones. However, its estimation performance is largely affected by noise. Under low SNR conditions NMCFLMS fails to converge to the desired solution. Since equalization using erroneous AIR estimates may result in an undesired output, noise robustness is an essential attribute for a BCI algorithm. This is the motivation behind the formulation of a noise robust BCI algorithm in this chapter.

8.3 The proposed partitioned MCFLMS (PMCFLMS) algorithm

Before deriving the algorithms, the fundamental idea behind the proposed algorithms is first introduced. Similar to that described in Section 6.3, efficient convolution of very long sequences can be achieved using the partitioned overlap-save method [227], [228]. Following the same notation as Section 6.3, let $\mathbf{h}_i$ be partitioned into $P \geq 1$ non-overlapping segments

$$\mathbf{h}_m(p) = [h_m(pL'_h + 1), \dots, h_m(pL'_h + L'_h)]^T, p = 0, \dots, P - 1,\tag{8.18}$$

each of length L'_h , and the $L_x \times 1$ vector

$$\mathbf{x}_m(k) = [x_m(kL'_h), \dots, x_m(kL'_h + L_x - 1)]^T, k = 0, \dots, K - 1, \quad (8.19)$$

be the k th frame of $\mathbf{x}_m(n)$ with a frame shift of L'_h samples where $L_x \geq 2L'_h$. Defining the $L' \times 1$ vector

$$\mathbf{s}(k) = [s(kL'_h - L'_h), \dots, s(kL'_h + L_x - 1)]^T, \quad (8.20)$$

as the k th frame of $\mathbf{s}^{01}(n) = [\mathbf{0}_{1 \times L'_h}, \mathbf{s}(n)^T]^T$, where $L' = L_x + L'_h$, it can be written, using partitioned overlap-save method, that

$$\mathbf{x}_m(k) = \sum_{p=0}^{P-1} \mathbf{W}_{L_x \times L'}^{01} \mathbf{C}_{s(k-p)} \mathbf{h}_m^{10}(p), \quad (8.21)$$

where

$$\mathbf{W}_{L_x \times L'}^{01} = [\mathbf{0}_{L_x \times L'_h}, \mathbf{I}_{L_x \times L_x}], \quad (8.22)$$

$$\begin{aligned} \mathbf{C}_{s(k-p)} &= \mathbf{F}_{L' \times L'}^{-1} \mathbf{D}_{s(k-p)} \mathbf{F}_{L' \times L'} \\ &= \begin{bmatrix} s[(k-p)L'_h - L'_h] & s[(k-p)L_h + L_x - 1] & \dots & s[(k-p)L'_h - L'_h + 1] \\ s[(k-p)L'_h - L'_h + 1] & s[(k-p)L'_h - L'_h] & \dots & s[(k-p)L'_h - L'_h + 2] \\ \vdots & \vdots & \ddots & \vdots \\ s[(k-p)L_h + L_x - 1] & s[(k-p)L_h + L_x - 2] & \dots & s[(k-p)L'_h - L'_h] \end{bmatrix} \end{aligned} \quad (8.23)$$

is the $L' \times L'$ circulant matrix constructed using $\mathbf{s}(k-p)$ and $\mathbf{D}_{s(k-p)}$ is, similar to (8.10), a $L' \times L'$ diagonal matrix where its diagonal elements are the coefficients of an L' -point DFT sequence,

$$\underline{\mathbf{s}}(k-p) = [\underline{s}(k-p, 1), \dots, \underline{s}(k-p, L')]^T. \quad (8.24)$$

This DFT sequence is obtained from $\mathbf{s}(k-p)$ and with reference to (8.21), the $L' \times 1$ vector

$$\mathbf{h}_m^{10} = \mathbf{W}_{L' \times L'_h}^{10} \mathbf{h}_m(p) \quad (8.25)$$

and

$$\mathbf{W}_{L' \times L'_h}^{10} = [\mathbf{I}_{L'_h \times L'_h}, \mathbf{0}_{L'_h \times L' - L'_h}]^T. \quad (8.26)$$

The frequency-domain representation of (8.21) is achieved as

$$\begin{aligned} \underline{\mathbf{x}}_m(k) &= \mathbf{F}_{L' \times L'}^H \mathbf{W}_{L' \times L_x}^{10} \mathbf{x}_m(k) \\ &= \sum_{p=0}^{P-1} \mathbf{F}_{L' \times L'}^H \mathbf{W}_{L' \times L_x}^{10} \mathbf{W}_{L_x \times L'}^{01} \mathbf{F}_{L' \times L'} \underline{\mathbf{D}}_{s(k-p)} \mathbf{F}_{L' \times L'}^H \mathbf{h}_m^{10}(p) \\ &= \underline{\mathbf{W}}_{L' \times L'}^{010} \sum_{p=0}^{P-1} \underline{\mathbf{D}}_{s(k-p)} \underline{\mathbf{h}}_m^{10}(p) \\ &= \underline{\mathbf{W}}_{L' \times L'}^{010} \underline{\dot{\mathbf{x}}}_m(k) \\ &= [\underline{x}_m(k, 1), \dots, \underline{x}_m(k, L')]^T, \end{aligned} \quad (8.27)$$

where

$$\underline{\mathbf{W}}_{L' \times L'}^{010} = \mathbf{F}_{L' \times L'}^H \mathbf{W}_{L' \times L_x}^{10} \mathbf{W}_{L_x \times L'}^{01} \mathbf{F}_{L' \times L'}, \quad (8.28)$$

$$\begin{aligned} \underline{\mathbf{h}}_m^{10}(p) &= \mathbf{F}_{L' \times L'}^H \mathbf{h}_m^{10}(p) \\ &= [\underline{h}_m^{10}(p, 1), \dots, \underline{h}_m^{10}(p, L')]^T, \end{aligned} \quad (8.29)$$

$$\begin{aligned} \underline{\dot{\mathbf{x}}}_m(k) &= \sum_{p=0}^{P-1} \underline{\mathbf{D}}_{s(k-p)} \underline{\mathbf{h}}_m^{10}(p) \\ &= [\underline{\dot{x}}_m(k, 1), \dots, \underline{\dot{x}}_m(k, L')]^T, \end{aligned} \quad (8.30)$$

$$\underline{\dot{x}}_m(k, l) = \sum_{p=0}^{P-1} \underline{h}_m^{10}(p, l) \underline{s}(k - p, l), l = 1, \dots, L'. \quad (8.31)$$

Alternatively, (8.31) can be expressed using the convolution operator giving

$$\begin{aligned} \underline{\dot{x}}_m(l) &= \underline{h}_m^{10}(l) * \underline{s}(l), \\ \underline{\dot{\mathbf{x}}}_m &= [\underline{\mathbf{H}}_{m,l}]^T \underline{\mathbf{s}}_l, \end{aligned} \quad (8.32)$$

where $\underline{\mathbf{s}}_l = [\underline{s}(0, l), \dots, \underline{s}(K - 1, l)]^T$, $\underline{\dot{\mathbf{x}}}_{m,l} = [\underline{\dot{x}}_m(0, l), \dots, \underline{\dot{x}}_m(K - 1, l)]^T$, and the $K \times K$ matrix $\underline{\mathbf{H}}_{m,l}$ is the filtering matrix obtained from $\underline{\mathbf{h}}_{m,l}^{10} = [\underline{h}_m^{10}(0, l), \dots, \underline{h}_m^{10}(P - 1, l)]^T$. Using the convolution

relation given in (8.32), a BCI algorithm will be derived in the next section.

8.3.1 The PMCFLMS algorithm

The PMCFLMS algorithm partitions $\mathbf{h}_m$ into $P \geq 1$ non-overlapping segments $\mathbf{h}_m(p)$, $p = 0, \dots, P-1$, each of length L'_h , and estimates these segments by employing the cross-relation between channels. To realize linear convolution using overlap-save method an $L \times 1$ data block

$$\mathbf{x}_m(n) = [x_m(nL_B), \dots, x_m(nL_B + L - 1)]^T \quad (8.33)$$

is first divided into $K' \geq 2P$ segments

$$\mathbf{x}_m(n, k) = [x_m(nL_B + kL'_h), \dots, x_m(nL_B + kL'_h + L' - 1)]^T, k = 0, \dots, K' - 1, \quad (8.34)$$

where $L_B \geq 1$ is the data block shift, $L = (2P - 1)L'_h + L' + \rho L'_h$, $\rho \geq 0$, $L' \geq 2L'_h$ is the length of $\mathbf{x}_m(n, k)$ and n is the block time index. The frequency-domain counterpart of $\mathbf{x}_m(n, k)$ is then obtained as

$$\underline{\mathbf{x}}_m(n, k) = \mathbf{F}_{L' \times L'}^H \mathbf{x}_m(n, k), \quad (8.35)$$

from which $\underline{\mathbf{x}}_{m,l}(n) = [\underline{x}_m(n, 0, l), \dots, \underline{x}_m(n, K' - 1, l)]^T$ is constructed.

Using (8.32) and the cross-relation between the channel outputs it can be written that

$$\begin{aligned} \underline{x}_m(l) * \underline{h}_q^{10}(l) &= \underline{s}(l) * \underline{h}_m^{10}(l) * \underline{h}_q^{10}(l) \\ &= \underline{x}_q(l) * \underline{h}_m^{10}(l), \quad m = 1, \dots, M, m \neq q. \end{aligned} \quad (8.36)$$

Let $\underline{z}_{mq}(n, l) = \underline{x}_m(n, l) * \hat{\underline{h}}_q^{10}(n, l)$ where $\hat{\underline{h}}_q^{10}(n, l)$ is an estimate of $\underline{h}_q^{10}(l)$. Using circular convolution,

$$\underline{z}_{mq,l}(n) = \mathbf{W}_{K \times K'}^{01} \mathbf{C}_{\underline{x}_m,l}(n) \hat{\underline{h}}_{q,l}^{100}(n), \quad (8.37)$$

where $\underline{\mathbf{z}}_{mq,l}(n) = [\underline{z}_{mq}(n, 0, l), \dots, \underline{z}_{mq}(n, K - 1, l)]^T$ is a $K \times 1$ vector, $K = K' - P$ is the difference between the number of segments of $\mathbf{h}_m$ and $\mathbf{x}_m$, $\mathbf{C}_{\underline{x}_m,l}(n)$ is the $K' \times K'$ circulant matrix constructed

using $\underline{\mathbf{x}}_{m,l}(n)$ and $\hat{\underline{\mathbf{h}}}_{m,l}^{100}(n) = \mathbf{W}_{K' \times P}^{10} \hat{\underline{\mathbf{h}}}_{m,l}^{10}(n)$.

Using (8.37), the cross-relation error, for the l th frequency bin, can be now defined as

$$\begin{aligned} \underline{\mathbf{e}}_{mq,l}(n) &= \underline{\mathbf{z}}_{mq,l}(n) - \underline{\mathbf{z}}_{qm,l}(n) \\ &= \mathbf{W}_{K \times K'}^{01} \left[\mathbf{C}_{\underline{\mathbf{x}}_{m,l}}(n) \hat{\underline{\mathbf{h}}}_{q,l}^{100}(n) - \mathbf{C}_{\underline{\mathbf{x}}_{q,l}} \hat{\underline{\mathbf{h}}}_{m,l}^{100}(n) \right] \\ &= [\underline{e}_{mq}(n, 0, l), \dots, \underline{e}_{mq}(n, K-1, l)]^T. \end{aligned} \quad (8.38)$$

To transform the cross-relation error into frequency domain, (8.38) is multiplied with $\mathbf{F}_{K' \times K'}^H$ giving

$$\begin{aligned} \underline{\underline{\mathbf{e}}}_{mq,l}(n) &= \mathbf{F}_{K' \times K'}^H \mathbf{W}_{K' \times K}^{10} \underline{\mathbf{e}}_{mq,l}(n) \\ &= \mathbf{F}_{K' \times K'}^H \mathbf{W}_{K' \times K}^{10} \mathbf{W}_{K \times K'}^{01} \left[\mathbf{C}_{\underline{\mathbf{x}}_{m,l}}(n) \hat{\underline{\mathbf{h}}}_{q,l}^{100}(n) - \mathbf{C}_{\underline{\mathbf{x}}_{q,l}}(n) \hat{\underline{\mathbf{h}}}_{m,l}^{100}(n) \right] \\ &= \underline{\mathbf{W}}_{K' \times K'}^{010} \left[\underline{\mathbf{D}}_{\underline{\mathbf{x}}_{m,l}}(n) \hat{\underline{\mathbf{h}}}_{q,l}^{100}(n) - \underline{\mathbf{D}}_{\underline{\mathbf{x}}_{q,l}}(n) \hat{\underline{\mathbf{h}}}_{m,l}^{100}(n) \right], \end{aligned} \quad (8.39)$$

where

$$\begin{aligned} \hat{\underline{\mathbf{h}}}_{m,l}^{100}(n) &= \mathbf{F}_{K' \times K'}^H \underline{\mathbf{h}}_{m,l}^{100}(n) \\ &= [\underline{h}_m^{100}(n, 1), \dots, \underline{h}_m^{100}(n, K')]^T. \end{aligned} \quad (8.40)$$

The frequency-domain squared error for the l th frequency bin at the n th iteration is then given by

$$\underline{\underline{J}}_l(n) = \sum_{m=1}^{M-1} \sum_{q=m+1}^M \underline{\underline{\mathbf{e}}}_{mq,l}^H(n) \underline{\underline{\mathbf{e}}}_{mq,l}(n). \quad (8.41)$$

Having defined $\underline{\underline{J}}_l(n)$, the gradient vector of PMCFLMS is now achieved by differentiating $\underline{\underline{J}}_l(n)$

with respect to $\left[\hat{\underline{\mathbf{h}}}_{q,l}^{100}(n)\right]^*$, $q = 1, \dots, M$, i.e.,

$$\begin{aligned}
\nabla_{q,l}(n) &= \frac{\partial J_l(n)}{\partial \left[\hat{\underline{\mathbf{h}}}_{q,l}^{100}(n)\right]^*} \\
&= \frac{\partial}{\partial \left[\hat{\underline{\mathbf{h}}}_{q,l}^{100}(n)\right]^*} \left\{ \sum_{m=1}^{q-1} \mathbf{e}_{mq,l}^H(n) \underline{\mathbf{e}}_{mq,l}(n) + \sum_{j=q+1}^M \mathbf{e}_{qj,l}^H(n) \underline{\mathbf{e}}_{qj,l}(n) \right\} \\
&= \sum_{m=1}^M \left[\mathbf{W}_{K' \times K'}^{010} \underline{\mathbf{D}}_{x_{m,l}}(n) \right]^H \underline{\mathbf{e}}_{mq,l}(n) \\
&= \sum_{m=1}^M \underline{\mathbf{D}}_{x_{m,l}}^*(n) \left[\mathbf{W}_{K' \times K'}^{010} \right]^H \underline{\mathbf{e}}_{mq,l}(n) \\
&= [\nabla_{q,l}(n, 1), \dots, \nabla_{q,l}(n, K')]^T.
\end{aligned} \tag{8.42}$$

Employing the gradient, the update equation of the PMCFLMS algorithm, given by

$$\hat{\underline{\mathbf{h}}}_{q,l}^{100}(n+1) = \hat{\underline{\mathbf{h}}}_{q,l}^{100}(n) - \mu \nabla_{q,l}(n), \tag{8.43}$$

is obtained where $\mu > 0$ is the step-size. The update equation for the entire L' frequency bins is represented in matrix form as

$$\hat{\underline{\mathbf{H}}}_q(n+1) = \hat{\underline{\mathbf{H}}}_q(n) - \mu \nabla_q(n) \tag{8.44}$$

where $\hat{\underline{\mathbf{H}}}_q(n) = [\hat{\underline{\mathbf{h}}}_{q,1}^{100}(n), \dots, \hat{\underline{\mathbf{h}}}_{q,L'}^{100}(n)]$ and $\nabla_q(n) = [\nabla_{q,1}(n), \dots, \nabla_{q,L'}(n)]$. The update equation can alternatively be written for the k' th subspace, which corresponds to the k' th row vector of $\hat{\underline{\mathbf{H}}}_q(n)$, as

$$\hat{\underline{\mathbf{h}}}_q^{100}(n+1, k') = \hat{\underline{\mathbf{h}}}_q^{100}(n, k') - \mu \nabla_q(n, k'), k' = 1, \dots, K', \tag{8.45}$$

where $\hat{\underline{\mathbf{h}}}_q^{100}(n, k') = [\hat{h}_{q,1}^{100}(n, k'), \dots, \hat{h}_{q,L'}^{100}(n, k')]^T$ and $\nabla_q(n, k') = [\nabla_{q,1}(n, k'), \dots, \nabla_{q,L'}(n, k')]^T$.

As opposed to NMCFLMS which computes a $2L_h$ -point DFT sequence, PMCFLMS involves two DFT computations; a L' -point DFT of $\mathbf{x}_m(n, k)$ and a K' -point DFT of $\underline{\mathbf{x}}_{m,l}(n)$ as shown in (8.35) and (8.39), respectively. Since PMCFLMS can estimate $\mathbf{h}_m(p)$ with $L' = 2L'_h$, it is possible to achieve complexity reduction in the computation of DFT for large L_h by selecting small values for L'_h . Another advantage of the above formulation is that PMCFLM allows one to use a single block of data $\mathbf{x}_m(n) =$

$[x_m(n), \dots, x_m(n + L - 1)]^T$ repeatedly for updating instead of applying a new data block for each iteration. Hence, it is possible to estimate the AIRs even from a single data block of only $(2P - 1)L'_h + L'$ samples. Therefore, estimation of a highly dynamic system can be accomplished by reusing a single data block of short duration over which the system is stationary, similar to the data-reusing LMS (DR-LMS) algorithm [229].

8.3.2 Convergence analysis of PMCFLMS

In this Section it will be shown that, similar to NMCFLMS, PMCFLMS converges to a mean-square solution in a noiseless and stationary environment. Let the $MK' \times 1$ concatenated error vector associated with the q th channel, constructed from $\underline{\mathbf{e}}_{mq,l}(n)$, be defined as

$$\underline{\mathbf{e}}_{q,l}(n) = \left[\underline{\mathbf{e}}_{1q,l}^H(n), \dots, \underline{\mathbf{e}}_{Mq,l}^H(n) \right]^H. \quad (8.46)$$

Defining the $K' \times K'$ matrix $\tilde{\underline{\mathbf{D}}}_{x_{q,l}}(n) = \underline{\mathbf{W}}_{K' \times K'}^{010} \underline{\mathbf{D}}_{x_{q,l}}(n)$,

$$\nabla_{q,l} = \tilde{\underline{\mathbf{D}}}_{x_l}^H(n) \underline{\mathbf{e}}_{q,l}(n), \quad (8.47)$$

where $\tilde{\underline{\mathbf{D}}}_{x_l}(n) = \left[\tilde{\underline{\mathbf{D}}}_{x_{1,l}}^H(n), \dots, \underline{\mathbf{D}}_{x_{M,l}}^H(n) \right]^H$ is a $MK' \times K'$ matrix. With

$$\underline{\mathbf{e}}_{mq,l}(n) = \tilde{\underline{\mathbf{D}}}_{x_{m,l}}(n) \hat{\underline{\mathbf{h}}}_{q,l}^{100}(n) - \tilde{\underline{\mathbf{D}}}_{x_{q,l}}(n) \hat{\underline{\mathbf{h}}}_{m,l}^{100}(n), \quad (8.48)$$

(8.46) can be decomposed as

$$\underline{\mathbf{e}}_{q,l}(n) = \left[\underline{\mathbf{A}}_{x_{q,l}}(n) - \underline{\mathbf{B}}_{x_{q,l}}(n) \right] \hat{\underline{\mathbf{h}}}_l^{100}(n), \quad (8.49)$$

where $\hat{\underline{\mathbf{h}}}_l^{100}(n) = \left\{ \left[\hat{\underline{\mathbf{h}}}_{1,l}^{100}(n) \right]^H, \dots, \left[\hat{\underline{\mathbf{h}}}_{M,l}^{100}(n) \right]^H \right\}^H$ is a $MK' \times 1$ concatenated vector, and the $MK' \times MK'$ matrices

$$\begin{aligned} \underline{\mathbf{A}}_{x_{q,l}}(n) &= [\mathbf{0}_{MK' \times (q-1)K'}, \tilde{\underline{\mathbf{D}}}_{x_l}(n), \mathbf{0}_{MK' \times (M-q)K'}] \\ \underline{\mathbf{B}}_{x_{q,l}}(n) &= \begin{bmatrix} \tilde{\underline{\mathbf{D}}}_{x_{q,l}}(n) & \mathbf{0}_{K' \times K'} & \cdots & \mathbf{0}_{K' \times K'} \\ \mathbf{0}_{K' \times K'} & \tilde{\underline{\mathbf{D}}}_{x_{q,l}}(n) & \cdots & \mathbf{0}_{K' \times K'} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{K' \times K'} & \mathbf{0}_{K' \times K'} & \cdots & \tilde{\underline{\mathbf{D}}}_{x_{q,l}}(n) \end{bmatrix}. \end{aligned} \quad (8.50)$$

Substituting (8.49) in (8.47) and using (8.50),

$$\nabla_{q,l}(n) = \left[-\underline{\mathbf{R}}_{1q,l}(n), -\underline{\mathbf{R}}_{2q,l}(n), \dots, \sum_{m \neq q} \underline{\mathbf{R}}_{mm,l}(n), \dots, -\underline{\mathbf{R}}_{Mq,l}(n) \right] \hat{\underline{\mathbf{h}}}_l^{100}(n), \quad (8.51)$$

where $\underline{\mathbf{R}}_{mq,l}(n) = \tilde{\underline{\mathbf{D}}}_{x_{m,l}}^H(n) \tilde{\underline{\mathbf{D}}}_{x_{q,l}}(n)$. The update equation in (8.43) is rewritten for the concatenated vector $\hat{\underline{\mathbf{h}}}_l^{100}$ as

$$\hat{\underline{\mathbf{h}}}_l^{100}(n+1) = \hat{\underline{\mathbf{h}}}_l^{100}(n) - \mu \underline{\mathbf{R}}_l(n) \hat{\underline{\mathbf{h}}}_l^{100}(n), \quad (8.52)$$

where the $MK' \times MK'$ matrix

$$\underline{\mathbf{R}}_l(n) = \begin{bmatrix} \sum_{m \neq 1} \underline{\mathbf{R}}_{mm,l}(n) & -\underline{\mathbf{R}}_{21,l}(n) & \cdots & -\underline{\mathbf{R}}_{M1,l}(n) \\ -\underline{\mathbf{R}}_{12,l}(n) & \sum_{m \neq 2} \underline{\mathbf{R}}_{mm,l}(n) & \cdots & -\underline{\mathbf{R}}_{M2,l}(n) \\ \vdots & \vdots & \ddots & \vdots \\ -\underline{\mathbf{R}}_{1M,l}(n) & -\underline{\mathbf{R}}_{2M,l}(n) & \cdots & \sum_{m \neq M} \underline{\mathbf{R}}_{mm,l}(n) \end{bmatrix}. \quad (8.53)$$

Defining the $MK'L' \times 1$ vector $\hat{\underline{\mathbf{h}}}^{100}(n) = \left\{ \left[\hat{\underline{\mathbf{h}}}_0^{100}(n) \right]^H, \dots, \left[\hat{\underline{\mathbf{h}}}_{L'-1}^{100}(n) \right]^H \right\}^H$ it can be written that

$$\hat{\underline{\mathbf{h}}}^{100}(n+1) = \hat{\underline{\mathbf{h}}}^{100}(n) - \mu \underline{\mathbf{R}}(n) \hat{\underline{\mathbf{h}}}^{100}(n), \quad (8.54)$$

where

$$\underline{\underline{\mathbf{R}}}(n) = \begin{bmatrix} \underline{\underline{\mathbf{R}}}_0(n) & \mathbf{0}_{MK' \times MK'} & \cdots & \mathbf{0}_{MK' \times MK'} \\ \mathbf{0}_{MK' \times MK'} & \underline{\underline{\mathbf{R}}}_1(n) & \cdots & \mathbf{0}_{MK' \times MK'} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{MK' \times MK'} & \mathbf{0}_{MK' \times MK'} & \cdots & \underline{\underline{\mathbf{R}}}_{L'-1}(n) \end{bmatrix} \quad (8.55)$$

is a $MK'L' \times MK'L'$ matrix. Subtracting the desired solution

$$\underline{\underline{\mathbf{h}}}^{100} = \left\{ \left[\underline{\underline{\mathbf{h}}}_0^{100} \right]^H, \dots, \left[\underline{\underline{\mathbf{h}}}_{L'-1}^{100} \right]^H \right\}^H \quad (8.56)$$

from (8.54) gives

$$\Delta \underline{\underline{\mathbf{h}}}^{100}(n+1) = \Delta \underline{\underline{\mathbf{h}}}^{100}(n) - \mu \underline{\underline{\mathbf{R}}}(n) \hat{\underline{\underline{\mathbf{h}}}}^{100}(n), \quad (8.57)$$

where $\Delta \underline{\underline{\mathbf{h}}}^{100}(n) = \hat{\underline{\underline{\mathbf{h}}}}^{100}(n) - \underline{\underline{\mathbf{h}}}^{100}$. Assuming statistical independence between $x_m(n)$ and h_m ,

$$E [\Delta \underline{\underline{\mathbf{h}}}^{100}(n+1)] = E [\Delta \underline{\underline{\mathbf{h}}}^{100}(n)] - \mu \underline{\underline{\mathbf{R}}} E [\underline{\underline{\mathbf{h}}}^{100} + \Delta \underline{\underline{\mathbf{h}}}^{100}(n)], \quad (8.58)$$

where $\underline{\underline{\mathbf{R}}} = E[\underline{\underline{\mathbf{R}}}(n)]$. Based on the principle of orthogonality [35] where $\underline{\underline{\mathbf{R}}} \underline{\underline{\mathbf{h}}}^{100} = 0$, (8.58) results in

$$E [\Delta \underline{\underline{\mathbf{h}}}^{100}(n+1)] = [\mathbf{I}_{MK'L' \times MK'L'} - \mu \underline{\underline{\mathbf{R}}}] E [\Delta \underline{\underline{\mathbf{h}}}^{100}(n)]. \quad (8.59)$$

It can be noted from (8.59) that the algorithm converges if $|1 - \mu \lambda_k| < 1$ where λ_k is the k th eigenvalue of $\underline{\underline{\mathbf{R}}}$. Hence, the algorithm converges in the mean to $\underline{\underline{\mathbf{h}}}^{100}$ if the step-size satisfies

$$0 < \mu < \frac{2}{\lambda_{\max}}, \quad (8.60)$$

where $\lambda_{\max}$ is the largest eigenvalue.

The step-size and hence the rate of convergence is dependent on the input signal. In order to achieve

fast convergence a $K' \times K'$ step-size matrix

$$\mathcal{S}(n) = \begin{bmatrix} \mu(n, 1) & 0 & \dots & 0 \\ 0 & \mu(n, 2) & \dots & 0 \\ \vdots & \dots & \ddots & \vdots \\ 0 & \dots & 0 & \mu(n, K') \end{bmatrix} \quad (8.61)$$

is computed, similar to [230], for each iteration. This computation uses

$$\mu(n, k') = \beta \left| \frac{\hat{\underline{\mathbf{h}}}^{100H}(n, k') \nabla(n, k')}{\|\nabla(n, k')\|_2^2 + \delta} \right|, \quad (8.62)$$

where

$$\begin{aligned} \hat{\underline{\mathbf{h}}}^{100}(n, k') &= \left\{ \left[\hat{\underline{\mathbf{h}}}_1^{100}(n, k') \right]^H, \dots, \left[\hat{\underline{\mathbf{h}}}_M^{100}(n, k') \right]^H \right\}^H \\ \nabla(n, k') &= \left[\nabla_1^H(n, k'), \dots, \nabla_M^H(n, k') \right]^H, \end{aligned} \quad (8.63)$$

$\beta > 0$ is a scale factor and δ is a small positive number introduced to avoid division by zero. The rate of convergence can be controlled using β . A larger β results in faster convergence and vice versa. However, if β is significantly large, the algorithm will misconverge. In order to avoid any possible misconvergence, in this work, the maximum value of $\mu(n, k')$ is limited to $\left| \frac{\hat{\underline{\mathbf{h}}}^{100H}(n, k') \nabla(n, k')}{\|\nabla(n, k')\|_2^2 + \delta} \right|$ by selecting $0 < \beta \leq 1$. It is noted that $\beta > 1$ does not improve the convergence rate significantly. In this work $\delta = 10^{-6}$. The update equation for PMCFLMS, given in (8.44), is subsequently modified as

$$\hat{\underline{\mathbf{H}}}_q(n+1) = \hat{\underline{\mathbf{H}}}_q(n) - \mathcal{S}(n) \nabla_q(n). \quad (8.64)$$

8.4 Misconvergence of LMS-based BCI algorithms in the presence of noise

As mentioned in the introduction, NMCFLMS misconverges in noisy environments. The convergence performance of NMCFLMS is shown in Fig. 8.1 using the normalized projection misalignment (NPM) [224]

Fig. 8.1: Convergence performance of NMCFLMS for SNR = 50 and 20 dB. At SNR = 20 dB the algorithm misconverges.

in dB. The NPM between $\mathbf{h}$ and $\hat{\mathbf{h}}$ is given by

$$\chi(n) = \frac{\left\| \mathbf{h} - \frac{\mathbf{h}^T \hat{\mathbf{h}}(n)}{\hat{\mathbf{h}}^T(n) \hat{\mathbf{h}}(n)} \hat{\mathbf{h}}(n) \right\|_2^2}{\|\mathbf{h}\|_2^2}. \quad (8.65)$$

A smaller $\chi(n)$ implies better AIR estimation performance. It can be seen from Fig. 8.1 that when SNR = 50 dB NMCFLMS converges achieving a low NPM. When SNR is reduced to 20 dB the algorithm fails to converge. In this section the misconvergence problem is analyzed for the above proposed PMCFLMS algorithm in a way that is different from [226]. This analysis provides further insights on the convergence behavior of LMS-based BCI algorithms.

The above formulation of the PMCFLMS algorithm assumes no additive noise. The effect of noise on the convergence behavior of LMS-based adaptive BCI algorithm has been analyzed in the time-domain in [155]. That analysis shows that the noise prevents the cost function of the algorithm from approaching zero resulting in non-zero gradients. This results in the misconvergence of the algorithm. It has also been shown that the misconvergence results in spectral attenuation of the estimated AIRs. However, the reason for this spectral attenuation has not been examined in the literature. This analysis provides further insight into the effect of noise on convergence performance explaining the reason behind the spectral attenuation.

To begin the analysis, the cross-relation error given in (8.39) can also be expressed as

$$\underline{\mathbf{e}}_{mq,l}(n) = \tilde{\underline{\mathbf{D}}}_{x_{mq,l}}(n) \hat{\underline{\mathbf{h}}}_l^{100}(n), \quad m < q \quad (8.66)$$

where the $K' \times MK'$ matrix $\tilde{\underline{\mathbf{D}}}_{x_{mq,l}}(n)$ is defined as

$$\tilde{\underline{\mathbf{D}}}_{x_{mq,l}}(n) = \underline{\mathbf{W}}_{K' \times K'}^{010} \left[\mathbf{0}_{K' \times (m-1)K'}, -\underline{\mathbf{D}}_{x_{q,l}}(n), \mathbf{0}_{K' \times (q-m-1)K'}, \underline{\mathbf{D}}_{x_{m,l}}(n), \mathbf{0}_{K' \times (M-q)K'} \right]. \quad (8.67)$$

When additive noise is present (8.2) is applicable and hence the cross-relation error given in (8.39) becomes

$$\begin{aligned} \underline{\mathbf{e}}_{mq,l}(n) &= \underline{\mathbf{W}}_{K' \times K'}^{010} \left\{ \left[\underline{\mathbf{D}}_{y_{m,l}}(n) + \underline{\mathbf{D}}_{v_{m,l}}(n) \right] \hat{\underline{\mathbf{h}}}_{q,l}^{100}(n) - \left[\underline{\mathbf{D}}_{y_{q,l}}(n) + \underline{\mathbf{D}}_{v_{q,l}}(n) \right] \hat{\underline{\mathbf{h}}}_{m,l}^{100}(n) \right\}, \\ &= \left[\tilde{\underline{\mathbf{D}}}_{y_{mq,l}}(n) + \tilde{\underline{\mathbf{D}}}_{v_{mq,l}}(n) \right] \hat{\underline{\mathbf{h}}}_l^{100}(n), \\ &= \underline{\mathbf{e}}_{mq,l}^y(n) + \underline{\mathbf{e}}_{mq,l}^v(n) \end{aligned} \quad (8.68)$$

where $\underline{\mathbf{D}}_{y_{m,l}}(n)$ and $\underline{\mathbf{D}}_{v_{m,l}}(n)$ correspond to $\underline{\mathbf{y}}_{m,l}(n) = [\underline{y}_{m,l}(n, 0, l), \dots, \underline{y}_{m,l}(n, K' - 1, l)]^T$ and $\underline{\mathbf{v}}_{m,l}(n) = [\underline{v}_{m,l}(n, 0), \dots, \underline{v}_{m,l}(n, K' - 1)]^T$, respectively, $\underline{\mathbf{e}}_{mq,l}^y(n) = \tilde{\underline{\mathbf{D}}}_{y_{mq,l}}(n) \hat{\underline{\mathbf{h}}}_l^{100}(n)$ and $\underline{\mathbf{e}}_{mq,l}^v(m) = \tilde{\underline{\mathbf{D}}}_{v_{mq,l}}(n) \hat{\underline{\mathbf{h}}}_l^{100}(n)$. Equation (8.68) implies that the cross-relation error comprises of error due to signal and error due to noise.

With the $2MK' \times 1$ vector

$$\underline{\mathbf{e}}_l(n) = [\underline{\mathbf{e}}_{12,l}^H(n), \dots, \underline{\mathbf{e}}_{1M,l}^H(n), \underline{\mathbf{e}}_{23,l}^H(n), \dots, \underline{\mathbf{e}}_{(M-1)M,l}^H(n)]^H \quad (8.69)$$

the $2MK'L' \times 1$ concatenated error vectors are defined as

$$\begin{aligned}
\underline{\underline{\mathbf{e}}}(n) &= \left[\underline{\underline{\mathbf{e}}}_0^H(n), \dots, \underline{\underline{\mathbf{e}}}_{L'-1}^H(n) \right]^H \\
&= \underline{\underline{\mathbf{e}}}^y(n) + \underline{\underline{\mathbf{e}}}^v(n), \\
\underline{\underline{\mathbf{e}}}^y(n) &= \left[\underline{\underline{\mathbf{e}}}_0^{yH}(n), \dots, \underline{\underline{\mathbf{e}}}_{L'-1}^{yH}(n) \right]^H \\
&= \underline{\underline{\mathbf{D}}}_y(n) \underline{\underline{\mathbf{h}}}^{100}(n), \\
\underline{\underline{\mathbf{e}}}^v(n) &= \left[\underline{\underline{\mathbf{e}}}_0^{vH}(n), \dots, \underline{\underline{\mathbf{e}}}_{L'-1}^{vH}(n) \right]^H \\
&= \underline{\underline{\mathbf{D}}}_v(n) \underline{\underline{\mathbf{h}}}^{100}(n),
\end{aligned} \tag{8.70}$$

where

$$\underline{\underline{\mathbf{D}}}_y(n) = \begin{bmatrix} \underline{\underline{\mathbf{D}}}_{y_0}(n) & \mathbf{0}_{2MK' \times MK'} & \cdots & \mathbf{0}_{2MK' \times MK'} \\ \mathbf{0}_{2MK' \times MK'} & \underline{\underline{\mathbf{D}}}_{y_1}(n) & \cdots & \mathbf{0}_{2MK' \times MK'} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{2MK' \times MK'} & \mathbf{0}_{2MK' \times MK'} & \cdots & \underline{\underline{\mathbf{D}}}_{y_{L'-1}}(n) \end{bmatrix}, \tag{8.71}$$

$$\underline{\underline{\mathbf{D}}}_{y_l}(n) = \left[\underline{\underline{\mathbf{D}}}_{y_{12,l}}^H(n), \underline{\underline{\mathbf{D}}}_{y_{13,l}}^H(n), \dots, \underline{\underline{\mathbf{D}}}_{y_{1M,l}}^H(n), \underline{\underline{\mathbf{D}}}_{y_{23,l}}^H(n), \dots, \underline{\underline{\mathbf{D}}}_{y_{(M-1)M,l}}^H(n) \right]^H. \tag{8.72}$$

The NPM between $\underline{\underline{\mathbf{h}}}^{100}$ and $\underline{\underline{\mathbf{h}}}^{100}(n)$ is given by

$$\underline{\underline{\chi}}(n) = \frac{\|\underline{\underline{\chi}}'(n)\|_2^2}{\|\underline{\underline{\mathbf{h}}}^{100}\|_2^2}, \tag{8.73}$$

$$\underline{\underline{\chi}}'(n) = \underline{\underline{\mathbf{h}}}^{100} - \alpha \underline{\underline{\mathbf{h}}}^{100}(n), \tag{8.74}$$

where

$$\alpha = \frac{[\underline{\underline{\mathbf{h}}}^{100}]^H \underline{\underline{\mathbf{h}}}^{100}(n)}{[\underline{\underline{\mathbf{h}}}^{100}(n)]^H \underline{\underline{\mathbf{h}}}^{100}(n)} \tag{8.75}$$

is the scale factor and $\underline{\underline{\chi}}'$ is the misalignment vector. To establish the relationship between $\underline{\underline{\chi}}'(n)$ and

$\underline{\underline{\mathbf{e}}}^y(n)$, (8.74) is multiplied with $\tilde{\underline{\underline{\mathbf{D}}}}_y(n)$ giving

$$\begin{aligned}\tilde{\underline{\underline{\mathbf{D}}}}_y(n)\underline{\underline{\mathbf{x}}}'(n) &= \tilde{\underline{\underline{\mathbf{D}}}}_y(n)\underline{\underline{\mathbf{h}}}^{100} - \alpha\tilde{\underline{\underline{\mathbf{D}}}}_y(n)\hat{\underline{\underline{\mathbf{h}}}}^{100}(n) \\ &= -\alpha\tilde{\underline{\underline{\mathbf{D}}}}_y(n)\hat{\underline{\underline{\mathbf{h}}}}^{100}(n) = -\alpha\underline{\underline{\mathbf{e}}}^y(n)\end{aligned}\quad (8.76)$$

since $\tilde{\underline{\underline{\mathbf{D}}}}_y(n)\underline{\underline{\mathbf{h}}}^{100} = \mathbf{0}_{2MK'L' \times 1}$. From (8.76), it can be deduced that

$$\underline{\underline{\mathbf{e}}}^y(n) = -\frac{\tilde{\underline{\underline{\mathbf{D}}}}_y(n)\underline{\underline{\mathbf{x}}}'(n)}{\alpha}. \quad (8.77)$$

Similarly, the relationship between $\underline{\underline{\mathbf{x}}}'(n)$ and $\underline{\underline{\mathbf{e}}}^y(n)$ is given by

$$\begin{aligned}\tilde{\underline{\underline{\mathbf{D}}}}_v(n)\underline{\underline{\mathbf{x}}}'(n) &= \tilde{\underline{\underline{\mathbf{D}}}}_v(n)\underline{\underline{\mathbf{h}}}^{100} - \alpha\tilde{\underline{\underline{\mathbf{D}}}}_v(n)\hat{\underline{\underline{\mathbf{h}}}}^{100}(n), \\ \tilde{\underline{\underline{\mathbf{D}}}}_v(n)\left[\underline{\underline{\mathbf{x}}}'(n) - \underline{\underline{\mathbf{h}}}^{100}\right] &= -\alpha\tilde{\underline{\underline{\mathbf{D}}}}_v(n)\hat{\underline{\underline{\mathbf{h}}}}^{100}(n) \\ &= -\alpha\underline{\underline{\mathbf{e}}}^v(n).\end{aligned}\quad (8.78)$$

From (8.78),

$$\underline{\underline{\mathbf{e}}}^v(n) = -\frac{\tilde{\underline{\underline{\mathbf{D}}}}_v(n)\left[\underline{\underline{\mathbf{x}}}'(n) - \underline{\underline{\mathbf{h}}}^{100}\right]}{\alpha}. \quad (8.79)$$

The expected value of the aggregate cost function $\underline{\underline{J}} = \underline{\underline{\mathbf{e}}}^H(n)\underline{\underline{\mathbf{e}}}(n)$ can be expressed as

$$\begin{aligned}E[\underline{\underline{J}}(n)] &= E\left[\underline{\underline{\mathbf{e}}}^H(n)\underline{\underline{\mathbf{e}}}(n)\right] \\ &= E\left\{\underline{\underline{\mathbf{e}}}^{yH}(n)\underline{\underline{\mathbf{e}}}^y(n)\right\} + E\left\{\underline{\underline{\mathbf{e}}}^{vH}(n)\underline{\underline{\mathbf{e}}}^v(n)\right\} + 2E\left\{\underline{\underline{\mathbf{e}}}^{yH}(n)\underline{\underline{\mathbf{e}}}^v(n)\right\}.\end{aligned}\quad (8.80)$$

Since $y_m(n)$ and $v_m(n)$ are uncorrelated and independent $E\left\{\underline{\underline{\mathbf{e}}}^{yH}(n)\underline{\underline{\mathbf{e}}}^v(n)\right\} = 0$. Hence,

$$E[\underline{\underline{J}}(n)] = E[\underline{\underline{J}}^y(n)] + E[\underline{\underline{J}}^v(n)], \quad (8.81)$$

where

$$\begin{aligned}\underline{\underline{J}}^y(n) &= \underline{\underline{\mathbf{e}}}^{yH}(n)\underline{\underline{\mathbf{e}}}^y(n) \\ &= \boldsymbol{\eta}'^H(n)\underline{\underline{\mathbf{R}}}_{yy}(n)\boldsymbol{\eta}'\end{aligned}\quad (8.82)$$

is the cost function corresponding to signal and

$$\begin{aligned}\underline{\underline{J}}^v(n) &= \underline{\underline{\mathbf{e}}}^{vH}(n)\underline{\underline{\mathbf{e}}}^v(n) \\ &= \left[\underline{\underline{\boldsymbol{\chi}}}'(n) - \underline{\underline{\mathbf{h}}}^{100}\right]^H \underline{\underline{\mathbf{R}}}_{vv}(n) \left[\underline{\underline{\boldsymbol{\chi}}}'(n) - \underline{\underline{\mathbf{h}}}^{100}\right]\end{aligned}\quad (8.83)$$

is the cost function corresponding to noise while

$$\underline{\underline{\mathbf{R}}}_{yy}(n) = \alpha^{-2}\tilde{\underline{\underline{\mathbf{D}}}}_y^H(n)\tilde{\underline{\underline{\mathbf{D}}}}_y(n), \quad (8.84)$$

$$\underline{\underline{\mathbf{R}}}_{vv}(n) = \alpha^{-2}\tilde{\underline{\underline{\mathbf{D}}}}_v^H(n)\tilde{\underline{\underline{\mathbf{D}}}}_v(n). \quad (8.85)$$

The effect of misalignment on $\underline{\underline{J}}^y(n)$ and $\underline{\underline{J}}^v(n)$ can be analyzed by differentiating (8.82) and (8.83) with respect to $\underline{\underline{\boldsymbol{\chi}}}'^H(n)$ which yields

$$\frac{d \underline{\underline{J}}^y(n)}{d \underline{\underline{\boldsymbol{\chi}}}'^H(n)} = \underline{\underline{\boldsymbol{\chi}}}'^T(n)\underline{\underline{\mathbf{R}}}_{yy}(n), \quad (8.86)$$

$$\frac{d \underline{\underline{J}}^v(n)}{d \underline{\underline{\boldsymbol{\chi}}}'^H(n)} = \left[\underline{\underline{\boldsymbol{\chi}}}'(n) - \underline{\underline{\mathbf{h}}}^{100}\right]^T \underline{\underline{\mathbf{R}}}_{vv}(n). \quad (8.87)$$

Consequently, it can be interpreted from (8.86) and (8.87) that, since $\underline{\underline{\mathbf{R}}}_{yy}$ and $\underline{\underline{\mathbf{R}}}_{vv}$ are not null matrices,

$$\begin{aligned}\frac{d \underline{\underline{J}}^y(n)}{d \underline{\underline{\boldsymbol{\chi}}}'^H(n)} &\rightarrow 0, \underline{\underline{\boldsymbol{\chi}}}'(n) \rightarrow \mathbf{0}_{MK'L' \times 1} \\ &\Rightarrow \hat{\underline{\underline{\mathbf{h}}}}^{100}(n) \rightarrow \underline{\underline{\mathbf{h}}}^{100},\end{aligned}\quad (8.88)$$

$$\begin{aligned}\frac{d \underline{\underline{J}}^v(n)}{d \underline{\underline{\boldsymbol{\chi}}}'^H(n)} &\rightarrow 0, \underline{\underline{\boldsymbol{\chi}}}'(n) \rightarrow \underline{\underline{\mathbf{h}}}^{100} \\ &\Rightarrow \hat{\underline{\underline{\mathbf{h}}}}^{100}(n) \rightarrow \mathbf{0}_{MK'L' \times 1}.\end{aligned}\quad (8.89)$$

It can be noted from (8.88) that as $\underline{\underline{J}}^y$ decreases with reducing $\underline{\underline{\chi}}'(n)$, $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n)$ approaches $\underline{\underline{\mathbf{h}}}^{100}$. On the contrary, (8.89) requires $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n)$ to converge to $\mathbf{0}_{MK'L' \times 1}$ for $\underline{\underline{J}}^v(n)$ to decrease. Since both $\underline{\underline{J}}^y(n)$ and $\underline{\underline{J}}^v(n)$ are simultaneously reduced adaptively, the combined effect of (8.88) and (8.89) is reflected on the misalignment vector $\underline{\underline{\chi}}'$. As a result if $d \underline{\underline{J}}^y(n)/d \underline{\underline{\chi}}'^H(n) \gg d \underline{\underline{J}}^v(n)/d \underline{\underline{\chi}}'^H(n)$, (8.88) dominates resulting in a fast convergence of the algorithm towards the true solution. On the other hand, in the presence of noise, as the algorithm converges towards the true solution $d \underline{\underline{J}}^y(n)/d \underline{\underline{\chi}}'^H(n)$ will be reduced at each iteration and once $d \underline{\underline{J}}^y(n)/d \underline{\underline{\chi}}'^H(n) < d \underline{\underline{J}}^v(n)/d \underline{\underline{\chi}}'^H(n)$, (8.89) will become dominant. Consequently, when $d \underline{\underline{J}}^v(n)/d \underline{\underline{\chi}}'^H(n) \gg d \underline{\underline{J}}^y(n)/d \underline{\underline{\chi}}'^H(n)$ the algorithm tends to misconverge to $\mathbf{0}_{MK'L' \times 1}$ and the misalignment is determined by the combined effect of (8.88) and (8.89). The algorithm continues to converge to the desired solution or misconverges depending on $\underline{\underline{J}}^y(n)$ and $\underline{\underline{J}}^v(n)$ until it reaches a steady state.

To illustrate the effect of $\underline{\underline{J}}^y(n)$ and $\underline{\underline{J}}^v(n)$ on $\underline{\underline{\chi}}(n)$ a set of five synthetically generated AIRs, each of length $L_h = 2048$, corresponding to a reverberation time $T_{60} = 128$ ms is simulated. The AIRs were generated inside a $5 \text{ m} \times 4 \text{ m} \times 3 \text{ m}$ room using the method of images [215] with an inter-microphone spacing of 10 cm and sampling frequency $f_s = 16$ kHz. For this simulation $L'_h = 1024$ giving $P = 2$, $L' = 2L'_h$, $L = (2P - 1)L'_h + L'$ resulting in $K' = 4$ and $L_B = L$. A speech source was convolved with the AIRs and an additive noise of zero-mean Gaussian distribution was introduced to achieve a desired SNR. It may be noted from (8.62) that the variable step-size $\mu(n, k') \rightarrow 0$ as $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n) \rightarrow \mathbf{0}_{ML \times 1}$ when the algorithm misconverges. As a result, when the algorithm misconverges to a steady state, $\mu(n, k')$ becomes insignificant. To avoid this, a fixed step-size with normalization applied is used, similar to NMCFLMS, to show the effect of noise on the convergence performance. The update equation with normalization is given by

$$\hat{\underline{\underline{\mathbf{h}}}}_{q,l}^{100}(n+1) = \hat{\underline{\underline{\mathbf{h}}}}_{q,l}^{100}(n) - \mu \mathbf{R}_l^{-1}(n) \nabla_{q,l}(n), \quad (8.90)$$

Fig. 8.2: Convergence performance of PMCFMLS for SNR 40 and 5 dB.

where

$$\begin{aligned}\mathfrak{R}_l(n) &= \varsigma \mathfrak{R}_l(n-1) + (1-\varsigma) \sum_{m=1, m \neq q}^M \underline{\mathbf{D}}_{\underline{x}_{m,l}}(n) \underline{\mathbf{D}}_{\underline{x}_{m,l}}^*(n) + \delta \mathbf{I}_{K' \times K'}, q = 1, \dots, M, \\ \varsigma &= [1 - (1/3K)]^K.\end{aligned}$$

The AIRs were then estimated using PMCFMLS with $\mu = 0.2$ for SNR = 40 and 5 dB. The proposed algorithms was arbitrarily initialized using

$$\hat{\underline{\underline{\mathbf{H}}}}_q(0) = \mathbf{1}_{K' \times L'} / \sqrt{P}, \quad (8.91)$$

where $\mathbf{1}_{K' \times L'}$ is a $K' \times L'$ unit matrix (matrix of ones).

The NPM curves achieved are shown in Fig. 8.2. It can be noted that the convergence performance of the proposed algorithm is affected by noise. The algorithm converges when SNR = 40 dB. On the other hand, for SNR = 5 dB the algorithm starts misconverging approximately after 250 iterations. In Fig. 8.3, $\underline{\underline{J}}^y(n)$ and $\underline{\underline{J}}^v(n)$ (computed in dB after applying a smoothing filter) are shown. It can be noted that, for SNR = 40 dB, since the source power is significantly higher than the noise power $\underline{\underline{J}}^y(n) > \underline{\underline{J}}^v(n)$. Hence (8.88) dominates and the algorithm converges towards the desired solution resulting in a low NPM. For SNR = 5 dB $\underline{\underline{J}}^y(n)$ initially dominates, which results in a reduction of NPM with reducing $\underline{\underline{J}}^y(n)$. When $\underline{\underline{J}}^y(n) < \underline{\underline{J}}^v(n)$ approximately after 250 iterations (8.89) becomes dominant. This explains why PMCFMLS starts misconverging after 250 iterations. These results show that, under low SNR conditions, although $\underline{\underline{J}}^y(n)$ and $\underline{\underline{J}}^v(n)$ are minimized, the algorithm tends to misconverge when $\underline{\underline{J}}^v(n) \gg \underline{\underline{J}}^y(n)$.

Fig. 8.3: $\underline{\underline{J}}^y(m)$ and $\underline{\underline{J}}^v(m)$ for (a) SNR = 40 and (b) SNR = 5 dB.

Having noted that PMCFLMS is not robust to noise, its convergence performance is improved by imposing a constraint on the cost function as will be described in the next section.

8.5 Noise robust PMCFLMS-SC algorithm

It has been shown in Section 8.4 that in the presence of significant amount of noise $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n) \rightarrow \mathbf{0}_{MK'L' \times 1}$, i.e., $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k') \rightarrow \mathbf{0}_{ML' \times 1}$. However, this attenuation is not uniform. As the algorithm misconverges $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')$ becomes more sparse due to the nonuniform attenuation of its coefficients. In Fig. 8.4, $|\hat{h}_{1,l}^{100}(n, 1)|$, $l = 1, \dots, L$ corresponding to the original AIR $\mathbf{h}_1$ generated in Section 8.4 and $|\hat{\underline{\underline{h}}}_{1,l}^{100}(n, 1)|$ obtained after the misconvergence of the algorithm at SNR = 5 dB are shown. It can be noted that $\hat{\underline{\underline{\mathbf{h}}}}_1^{100}(n, 1)$ is more sparse than $\underline{\underline{\mathbf{h}}}_1^{100}(n, 1)$ and its coefficients $\hat{\underline{\underline{h}}}_{1,l}^{100}(n, 1)$ tend to zero non-uniformly as the algorithm misconverges. In order to analyze the effect of noise on the sparseness of $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')$ sparseness measure [231], defined by

$$\xi(n, k') = \frac{ML'}{ML' - \sqrt{ML'}} \left(1 - \frac{\|\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')\|_1}{\sqrt{ML'} \|\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')\|_2} \right), \quad (8.92)$$

Fig. 8.4: Plot of (a) $|h_{1,l}^{100}(n, 1)|$, and (b) $|\hat{h}_{1,l}^{100}(n, 1)|$.

Fig. 8.5: Sparseness measure comparison for SNR = 40, 10 and 5 dB.

is employed to quantify the degree of sparseness. In Fig. 8.5 the averaged sparseness measure $\xi(n) = \sum_{k'=0}^{K'-1} \xi(n, k') / K'$ computed for SNR = 40, 10 and 5 dB is shown. It can be noted that when the algorithm converges at SNR = 40 dB a low $\xi(n, k')$, closer to that of the actual AIR, is achieved for the estimated AIR. However, for SNR = 10 and 5 dB the algorithm misconverges resulting in larger $\xi(n, k')$. This implies that as the algorithm misconverges the sparseness measure of $\hat{\underline{\mathbf{h}}}_1^{100}(n, k')$ increases.

The above observation leads to the development of the noise robust PMCFLMS-SC algorithm. A penalty function exploiting the sparseness measure for the PMCFLMS algorithm is derived to improve its robustness to noise. Figure 8.4(a) shows that the energy of $\hat{\underline{\mathbf{h}}}_1^{100}(n, k')$ is not uniformly distributed. Hence, unlike [232] which imposes a uniform energy constraint, a penalty function that ensures a lower

sparseness of $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')$ given by

$$\begin{aligned} \underline{\underline{J}}^p(n, k') &= \xi^{-1}(n, k') \\ &= \frac{ML' - \sqrt{ML'}}{ML'} \left(1 - \frac{\|\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')\|_1}{\sqrt{ML'} \|\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')\|_2} \right)^{-1} \end{aligned} \quad (8.93)$$

is maximized. The above penalty cost function can further be simplified as

$$\underline{\underline{J}}^p(n, k') = \frac{ML' - \sqrt{ML'}}{ML'} \left(1 - \|\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')\|_1 \right)^{-1}, \quad (8.94)$$

where an energy constraint,

$$\left\| \hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k') \right\|_2 = 1/\sqrt{ML'}, \quad (8.95)$$

have been imposed on $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')$. By differentiating (8.94) with respect to $\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')$ the gradient of the penalty function

$$\begin{aligned} \nabla^p(n, k') &= \left[\nabla_1^{pH}(n, k'), \dots, \nabla_M^{pH}(n, k') \right]^H \\ &= \frac{ML' - \sqrt{ML'}}{2ML' \left(1 - \|\hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k')\|_1 \right)^2} \mathbf{D}_{\hat{\underline{\underline{\mathbf{h}}}}}^{100}(n, k') \hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k') \end{aligned} \quad (8.96)$$

is obtained, where $\hat{\underline{\underline{\mathbf{h}}}}_a^{100}(n, k') = \left[|h_{1,1}^{100}(n, k')|^{-1}, \dots, |h_{1,L'}^{100}(n, k')|^{-1}, |h_{2,1}^{100}(n, k')|^{-1}, \dots, |h_{M,L'}^{100}(n, k')|^{-1} \right]^T$.

Incorporating (8.96) into (8.45) the update equation for the proposed PMCFLMS-SC algorithm is given by

$$\hat{\underline{\underline{\mathbf{h}}}}_q^{100}(n+1, k') = \frac{\hat{\underline{\underline{\mathbf{h}}}}_q^{100}(n, k') - \mu(n, k') \left[\nabla_q(n, k') - \gamma(n, k) \nabla_q^p(n, k') \right]}{\sqrt{ML'} \left\| \hat{\underline{\underline{\mathbf{h}}}}^{100}(n, k') \right\|_2}, \quad (8.97)$$

where $\gamma(n, k)$ is the weighting for $\nabla_q^p(n, k)$. The negative sign preceding the penalty function indicates that as $J^p(n, k')$ is maximized the total cost function is minimized. In the steady-state condition $\nabla(n, k) \approx \gamma(n, k) \nabla^p(n, k')$. Therefore, $\nabla^{pH}(n, k') \nabla(n, k) \approx \gamma(n, k') \parallel \nabla^p(n, k') \parallel_2^2$ from which

Fig. 8.6: Convergence performance of PMCFLMS as a function of β at SNR = 40 dB.

an expression for $\gamma(n, k')$ is derived as

$$\gamma(n, k') = \kappa \left| \frac{\nabla^{pH}(n, k') \nabla(n, k')}{\|\nabla^p(n, k')\|_2^2 + \delta} \right|, \quad (8.98)$$

where κ is a coupling factor. The performance improvement achieved by the introduction of the penalty function will be shown in the next section.

8.6 Simulation results

8.6.1 Convergence performance

The convergence performance of the proposed algorithms is now examined using a set of five AIRS, each of length $L_h = 4096$, generated using the same setup as described in Section 8.4. A white Gaussian sequence was used as the source which was convolved with the AIRs. In this section the step-size matrix defined in (8.61) is used for the proposed algorithms.

PMCFLMS

The performance of PMCFLMS is first assessed at SNR = 40 dB for $0 < \beta \leq 1$. A relatively high SNR was chosen in order to avoid the misconvergence of PMCFLMS. When $\beta = 1$ the step-size μ attains its maximum value. As stated in Section 8.3.2, the rate of convergence does not vary significantly for $\beta > 1$. From the NPM curves shown in Fig. 8.6 it can be noted that, for all β , PMCFLMS converges towards the desired solution. A larger β results in faster convergence since it results in a larger step-size. Moreover,

Fig. 8.7: Convergence performance of PMCFLMS as a function of P at SNR = 40 dB.

Fig. 8.8: Convergence performance of PMCFLMS using a single block of data.

it can be seen from Fig. 8.6 that the rate of convergence varies only modestly as β approaches 1.

The convergence performance is also analyzed for different P . For this simulation $L = 3L_h$ so that L is independent of L'_h . The NPM curves achieved for $L'_h = L_h/P$, $P = 1, 2, 32, 128$, with $L' = 2L'_h$ and $\beta = 1$ after 200 iterations are shown in Fig. 8.7. It can be noted that PMCFLMS achieves almost same convergence performance for $L' = 8192, 4096, 256$ and 64 . This implies that the algorithm is relatively less sensitive to L'_h .

It will be shown next that PMCFLMS can estimate AIRs by using a single data block as mentioned in the last paragraph of Section 8.3.1. The same simulation setup with $\beta = 1$ is used in this simulation. Here, instead of using a new block of data for each iteration the same block was used for all iterations. The AIRs were estimated using a data block of $3L_h$ samples for $L'_h = L_h/P$, $P = 1, 2, 8, 32, 64$, with $L' = 2L'_h$. Results presented in Fig. 8.8 show that PMCFLMS with the same data block for all iterations can achieve good performance. This implies that data reusing can be employed in situations where data length is insufficient to be divided into several frames. It can be also noted that the convergence rate increases as P increases.

Fig. 8.9: Convergence performance of PMCFLMS and NMCFLMS at different SNRs.

The performance of PMCFLMS is next compared with NMCFLMS. Similar to NMCFLMS, $L = 2L_h$ and $L_B = L$ for PMCFLMS while $P = 2$ giving $L'_h = 2048$ and $L' = 4096$. The scale factor $\beta = 0.6$ was chosen for PMCFLMS so that both the algorithms achieved approximately equal initial convergence rate. The NPM curves obtained for SNR = 40, 20, 10 and 0 dB are shown in Fig. 8.9. It can be noted that at SNR = 40 dB, both the algorithms converge with PMCFLMS achieving a performance improvement of approximately 1.6 dB over NMCFLMS. However, at SNR = 20 dB NMCFLMS misconverges whereas the proposed PMCFLMS algorithm still converges albeit with modest performance degradation. For SNR = 10 and 0 dB, both the algorithms misconverge as (8.89) significantly dominates over (8.89). Nevertheless, it may be noted that NPM for PMCFLMS does not reach 0 dB since the variable step-sizes become insignificant after some iterations during misconvergence.

PMCFLMS-SC

The performance of the noise robust PMCFLMS-SC algorithm derived in Section 8.5 is assessed using the same set of AIRs described in Section 8.6.1 where $\kappa = 1$. Results achieved with the robust variable step-size MCFLMS (RVSS-MCFLMS) [232] are also shown for comparison. The RVSS-MCFLMS algorithm is one of the recent algorithms developed to address the problem of misconvergence. The AIRs were estimated at SNR = 40, 20, 10 and 0 dB with $\beta = 0.2$ and 0.4 for PMCFLMS-SC. A scale factor $\beta = 0.4$ was used such that PMCFLMS-SC achieved approximately the same initial convergence rate as RVSS-MCFLMS.

Figures 8.9 and 8.10 show the NPM curves obtained for PMCFLMS-SC and PMCFLMS-SC, respec-

Fig. 8.10: Convergence performance of PMCFLMS-SC at different SNRs.

tively. It can be noted that, both PMCFLMS and PMCFLMS-SC exhibit similar convergence behavior at SNR = 40 dB since, with $J^y(n) > J^v(n)$, the penalty function is not significant. When SNR = 20 dB although both the proposed algorithms converge, PMCFLMS-SC exhibits better performance due to the sparseness constraint imposed by the penalty function given in (8.93). However, for SNR = 10 and 0 dB, PMCFLMS starts to misconverge and reaches its steady state after some iterations. On the other hand, the penalty function of PMCFLMS-SC is utilized effectively as the algorithm starts misconverging; it aims to prevent spectral attenuation by enforcing dispersiveness to $\hat{\underline{\mathbf{h}}}_i^{100}(n, k)$. This prevents the algorithm from misconverging resulting in a lower NPM as can be observed from Fig. 8.10. Comparing Figs. 8.10 and 8.11, although RVSS-MCFLMS does not misconverge, it can be seen that PMCFLMS-SC achieves a performance improvement of approximately 2 dB over RVSS-MCFLMS. Moreover, it can be observed that a lower β results in a smaller NPM at the cost of a reduction in the convergence rate. Figure 8.12 shows $\xi(n)$ obtained for PMCFLMS-SC with $\beta = 0.4$. This result shows that PMCFLMS-SC ensures dispersiveness of $\hat{\underline{\mathbf{h}}}^{100}(n)$ thereby preventing $\hat{\underline{\mathbf{h}}}^{100}(n) \rightarrow \mathbf{0}_{MK'L' \times 1}$.

8.6.2 Estimation of time-varying system

The above simulations assume stationary channels. However, acoustic channels are time-varying in practice. Even a slight shift in the source position may significantly change the AIRs. In this simulation the ability of the proposed algorithms to track time-varying AIRs is examined using WGN as the source. A set of five synthetic AIRs with $T_{60} = 256$ ms is used where $L_h = 4096$. The source position was made to change after every 4 seconds such that the source bearing was altered by one degree. In order to utilize

Fig. 8.11: Convergence performance of RVSS-MCFLMS at different SNRs.

Fig. 8.12: Sparseness measure comparison of PMCFLMS-SC for SNR = 40, 20, 10 and 0 dB.

more number of data blocks for adaptation $L_B = L_h/4$ is chosen in this simulation while $L = 3L_h$, $P = 128$ and $\beta = 1$. SNR = 10 dB is selected for PMCFLMS-SC while a higher SNR = 20 dB is selected for PMCFLMS to avoid its misconvergence.

Figures 8.13(a) and (b) shows the convergence performance of PMCFLMS and PMCFLMS-SC, respectively. It can be noted that the algorithms continue to converge until the source position is shifted and thereafter, NPM increases significantly due to the mismatch between the estimated AIRs and the new AIRs. However, the algorithms successfully track the change in system and start converging towards the new AIRs as can be seen from their convergence profile. In Fig. 8.14 the actual and the estimated AIRs achieved after 1693 iterations (corresponding to the fifth source shift) are also shown with normalized magnitude. It can be noted that the proposed algorithms are able to approximate the AIRs with good accuracy even under low SNR conditions.

To further examine the estimation accuracy, the original AIRs shown in Fig. 8.14 were equalized using inverse filters computed using the corresponding estimated AIRs and the FDA-MCEQ_{fast} algorithm

Fig. 8.13: The convergence performance of (a) PMCFLMS (SNR = 20 dB) and (b) PMCFLMS-SC (SNR = 10 dB) for time-varying AIRs.

proposed in Chapter 7. The desired response after equalization is a Kronecker delta function. The equalized responses which are shown in Fig. 8.15 are very close to the desired response. The PMCFLMS and PMCFLMS-SC algorithms achieve, respectively, $\xi = 0.75$ and $\xi = 0.6$ for $\hat{\mathbf{d}}$. This implies that channel equalization using the AIRs estimated by the proposed algorithms results in significant suppression of early as well as late reflections even for low SNRs, which can be translated into a good equalization performance.

8.6.3 Speech dereverberation

Finally, the proposed algorithms are applied to achieve speech dereverberation. A set of five recorded AIRs obtained from the MARDY database are used in this simulation. The AIRs were re-sampled to 8 kHz and truncated to $L_h = 4096$. The reverberant signals were generated by convolving a clean female speech signal, sampled at 8 kHz, with the AIRs. Channel estimation was then performed using PMCFLMS, PMCFLMS-SC and RVSS-MCFLMS and equalization was subsequently achieved via inverse filtering using FDA-MCEQ_{fast} for SNR = 40 and 10 dB. To evaluate the quality of the equalized speech the Bark spectral distortion (BSD) [206], and perceptual evaluation of speech quality (PESQ) [208] are employed with the clean speech signal as the reference signal. For the reverberant data the evaluation

Fig. 8.14: (a) Actual AIRs. (b) AIRs estimated by PMCFLMS at SNR = 20 dB. (c) AIRs estimated by PMCFLMS-SC at SNR = 10 dB.

Fig. 8.15: Equalized response. (a) PMCFLMS at SNR = 20 dB. (b) PMCFLMS-SC at SNR = 10 dB.

Table 8.1: Dereverberation performance analysis metrics.

SNR dB	Case	BSD	PESQ
40	Reverberant data	0.147	2.3
	PMCFLMS	0.026	3.9
	PMCFLMS-SC	0.020	3.9
	RVSS-MCFLMS	0.050	3.5
10	Reverberant data	0.501	1.6
	PMCFLMS	0.245	2.1
	PMCFLMS-SC	0.151	2.2
	RVSS-MCFLMS	0.208	2.1

metrics are averaged over the five channels. Results obtained are tabulated in Table 8.1, where a larger value of PESQ implies better perceptual quality, i.e., better equalization performance. On the contrary, the lower the BSD the better the quality of the equalized speech signal.

The results given in Table 8.1 show that equalization of the reverberant data using the AIRs estimated by the algorithms improves the speech quality. The PESQ and BSD measures show that the proposed PMCFLMS and PMCFLMS-SC outperform the RVSS-MCFLMS at $\text{SNR} = 40$ dB. However, when $\text{SNR} = 10$ dB both PMCFLMS and RVSS-MCFLMS display similar performance and PMCFLMS-SC achieves comparatively better speech dereverberation yielding the lowest BSD and highest PESQ measures.

8.7 Conclusion

The performance of a multichannel equalization system is largely dependent on the estimated AIRs. Although existing BCI algorithms such as NMCFLMS can achieve good AIR estimates, most of them fail to converge in the presence of noise. In this chapter a noise robust BCI algorithm was formulated. An adaptive BCI algorithm based on the partitioned convolution method was first proposed. The algorithm divides the AIRs into segments and estimates them. An analysis on the misconvergence of LMS-based adaptive BCI algorithms in the presence of noise was also presented in order to provide further insights on the effect of noise on convergence performance. Based on the analysis, the PMCFLMS-SC algorithm, which is robust to noise, was proposed. Noise robustness of the proposed PMCFLMS-SC algorithm was achieved by imposing a sparseness constraint to its cost function. This sparseness constraint prevents

the algorithm from misconverging in a noisy environment. Simulation results show that the proposed algorithms can provide good estimates of AIRs. The proposed PMCFLMS-SC algorithm achieves good dereverberation performance even at low SNRs.

Chapter 9

Conclusion and Recommendations

9.1 Conclusion

Speech dereverberation has been a challenging problem for many years. Dereverberation has many applications such as speech and speaker recognition, internet telephony, teleconferencing, home automation and mobile multimedia systems. This thesis deals with speech dereverberation via blind channel identification and multichannel equalization. In Chapters 3–7 various MCEQ algorithms were proposed and in Chapter 8 two blind channel equalization algorithms were proposed.

In Chapter 2 of this thesis, the reverberation problem was briefly explained and some of the existing dereverberation techniques were reviewed. Following this, inverse filtering problem and existing popular inverse filtering algorithms were discussed. In the last section of this chapter some of the commonly used speech quality measures were presented.

Existing well-known MCEQ algorithms achieve equalization in the time domain. Equalization in other transform domains is also possible. However, a general framework to achieve equalization in various transform domains is not available yet. This forms the motivation to develop a generalized MCEQ framework in Chapter 3. The proposed generalized MCEQ framework can be applied in different transform domains to achieve multichannel equalization. It has been shown with examples that the proposed framework can be used to develop new MCEQ algorithms. To exemplify this, three subspace based algorithms (two in the time domain and one in the KLT domain) were developed in that chapter by applying the framework. Compared to MINT, which is very sensitive to channel estimation errors and modeling delay, the proposed time-domain MCEQ-T and MCEQ-T_{opt} algorithms exhibit better equalization performance. The MCEQ-T algorithm employs a dedicated set of filters for each subspace and achieves a lower mean squared equalization error than MINT. On the other hand, MCEQ-T_{opt}

employs a single set of filters that achieves the lowest error among L sets of filters. It has also been shown that the proposed KLT-domain MCEQ-KL algorithm can be applied in conjunction with a KL-domain denoising algorithm to achieve equalization in noisy environments. Simulation results showed that the joint application of MCEQ-KL and Wiener filter improves the equalization quality significantly.

MINT as well as the time-domain algorithms presented in Chapter 3 involve matrix inversion which is computationally expensive for high-order AIRs. Hence it is challenging to equalize high-order AIRs which are common in practice. Motivated by this an algorithm to achieve subspace based time-domain MCEQ with reduced complexity was presented in Chapter 4. The proposed TDA-MCEQ algorithm employs adaptive filtering techniques to accomplish MCEQ. Since TDA-MCEQ does not require matrix inversion the complexity involved in MCEQ is reduced. In order to enhance the performance of TDA-MCEQ further a sparseness-based constraint was introduced to its cost function and the resultant algorithm is termed as TDA-MCEQ-SP. The TDA-MCEQ-SP algorithm achieves good equalization with improved convergence performance.

The cost function of existing MINT-based adaptive algorithms are formed similar to the MINT equation. In contrast to this, the adaptive MCEQ algorithms proposed in Chapter 5 achieve equalization by minimizing a newly derived cost function. Simulation results showed that the proposed A-MCEQ and A-MCEQ-N algorithms could achieve faster convergence and better equalization performance. In order to improve the rate of convergence of adaptive MCEQ algorithms like A-MINT and A-MCEQ, computation of a variable step-size for each iteration based on the sparseness measure of the equalized response was also proposed. It was shown that the proposed variable step-size approach improved the convergence rate of the algorithms significantly while yielding better equalization performance.

Although the proposed adaptive algorithms reduce the computational complexity, their deployment in practical applications is limited by slow convergence. This motivated the development of the frequency-domain MCEQ algorithms in Chapter 6. The MCEQ framework was applied in the frequency domain and consequently two frequency-domain MCEQ algorithms were derived. The proposed MCEQ-F algorithm is a computationally efficient MCEQ algorithm, which can be applied for the fast equalization of high-order AIRs in practical scenarios. This algorithm also exhibits higher robustness to AIR inaccuracies. The proposed PMCEQ-F, allows one to partition data as well as the AIRs into frames and achieve

equalization. This significantly reduce the complexity further.

Complexity reduction achieved by adaptive filtering techniques is exploited to further reduce the complexity of MCEQ-F proposed in Chapter 6. The FDA-MCEQ algorithm proposed in Chapter 7 is an adaptive version of the MCEQ-F algorithm. In FDA-MCEQ, the frequency-domain inverse filters are iteratively estimated by minimizing a cost function. Furthermore, an optimal step-size for the MCEQ-F algorithm was derived. The proposed FDA-MCEQ_{fast} algorithm therefore computes an optimal step-size and converges to the desired solution in a single iteration. Application of the optimal step-size helps FDA-MCEQ_{fast} to significantly reduce the processing time compared to MCEQ-F.

In order to achieve good dereverberation quality, high AIR estimation accuracy is required. Accurate AIR estimation in a noise environment is still a challenging problem. Least-means-squares based adaptive BCI algorithms such as NMCFLMS exhibit misconvergence in the presence of noise. However, the reason behind the misconvergence has not been analyzed comprehensively in existing works. Motivated by this a new BCI algorithm called PMCFLMS was first derived in Chapter 6, using which the misconvergence behavior was analyzed. Based on the analysis, a noise robust BCI algorithm known as PMCFLMS-SC was formulated. The noise robustness was achieved by employing a sparseness based constraint to the cost function of PMCFLMS. Simulation results depict a higher AIR estimation performance of the proposed algorithms and their ability to estimate the AIRs in non-stationary environments.

9.2 Recommendations for further research

The main objective of this research work is to develop speech dereverberation and equalization algorithms. In this thesis a two-stage approach was adopted for speech dereverberation. In the first stage, the AIRs are blindly estimated. The second stage achieves equalization via inverse filtering. Speech dereverberation will continue to be an important area of research in the near future. Some of future research goals are given below.

Performance of existing MCEQ algorithms depends largely on modeling delay. These algorithms employ a desired response with an arbitrary modeling delay and compute the inverse filters in a least-square sense. The MCEQ-T_{opt} algorithm proposed in Chapter 3 achieves equalization using an optimal

set of inverse filters corresponding to an optimal modeling delay. However, this technique involves estimation of inverse filters for all possible delays and is therefore, computationally not efficient. One idea is to devise an algorithm which can adaptively select a target response with an appropriate delay, instead of using a target response with an arbitrary delay. The algorithm will choose a suitable modeling delay for the equalized response such that it can achieve a minimum-norm solution. This can ensure a better equalization performance irrespective of the modeling delay.

Another future research direction is to achieve speech dereverberation via inverse filtering in a single step. As opposed to existing two-stage techniques which compute the equalization filters using the AIRs estimated using BCI, the idea here is to estimate the equalization filters directly from the received signals thereby avoiding the need of estimation and inversion of the AIRs. This can not only reduce the computational load significantly but also address the challenges associated with the inversion of non-minimum-phase AIRs. Hence, research in this direction is envisaged to have some potential applications.

Most of the current dereverberation algorithms perform dereverberation for listening purpose. Some of the preliminary results obtained show that speech dereverberation sometimes adversely affects the performance of ASRs which are trained with clean speech data. Speech feature vectors such as MFCC are affected by reverberation due to which the performance of ASRs is degraded. In order to achieve a good ASR accuracy, it is desirable that the feature vectors derived from the received speech signals closely match with that derived from the corresponding clean speech. However, it is hard to achieve this in a reverberant environment. An improved speech recognition performance can be expected if the effect of reverberation on the feature vectors is minimized or eliminated substantially so that the feature vectors derived from the enhanced signal become close to that of the clean speech signal. Hence, one of the future goals is to develop an algorithm to mitigate the reverberation effects on the speech feature vectors with an aim of improving the speech recognition accuracy of AIRs under reverberant conditions.

Appendix A

Approximate diagonalization of filtering matrix by the Fourier matrix

A.1 diagonalization of filtering matrix

Let $\mathbf{H}^c$ be an $L \times L$ circulant matrix

$$\mathbf{H}^c = \begin{bmatrix} h(0) & h(1) & \dots & h(L-2) & h(L-1) \\ h(L-1) & h(0) & \dots & h(L-3) & h(L-2) \\ \vdots & \vdots & \dots & \vdots & \vdots \\ h(1) & h(2) & \dots & h(L-1) & h(0) \end{bmatrix}, \quad (\text{A.1})$$

which is fully described by the vector $\mathbf{h} = [h(0), h(1), \dots, h(L-1)]^T$. The eigenvectors of a circulant matrix of a given size are the column vectors of the unitary frequency-domain transform matrix $\mathbf{F}_{L \times L}$ of the same size, and the eigenvalues are the Fourier coefficients of $\mathbf{h}$ with the same size [222], i.e.,

$$\mathbf{F}^H \mathbf{H}^c \mathbf{F} = \mathbf{\Lambda}, \quad (\text{A.2})$$

$$\mathbf{H}^c \mathbf{f}_l = \lambda(l) \mathbf{f}_l, \quad l = 1, \dots, L, \quad (\text{A.3})$$

where $\mathbf{\Lambda} = \text{diag}\{\lambda(1), \dots, \lambda(L)\}$ is the corresponding $L \times L$ diagonal eigenvalue matrix with eigenvalues $\lambda(l)$, and $\mathbf{F}$, defined in (6.1), is an $L \times L$ matrix whose columns are the eigenvectors of $\mathbf{H}^c$. Given $\mathbf{f}_l = [f_{l,1}, f_{l,2}, \dots, f_{l,L}]^T$, the Fourier coefficients of $\mathbf{h}$ can be obtained as $\mathbf{f}_l^H \mathbf{h}$, $l = 1, \dots, L$, which is equivalent to $\lambda^*(l)$.

Let $\mathbf{H}_l^c$ denote the l th row of $\mathbf{H}^c$ and define the $L \times L$ matrix $\tilde{\mathbf{H}}^c = \left[(\mathbf{H}_1^c)^T, \dots, (\mathbf{H}_L^c)^T, \mathbf{0}_{L \times (L-R)} \right]^T$,

consisting of the first R rows of $\mathbf{H}^c$ appended by $L - R$ rows of zeros, and define the matrix

$$\tilde{\mathbf{\Lambda}} = \mathbf{F}^H \tilde{\mathbf{H}}^c \mathbf{F}. \quad (\text{A.4})$$

can be defined. Next, it will be proved that $\tilde{\mathbf{\Lambda}}_{l,l} = \mathbf{\Lambda}_{l,l} R/L$, $l = 1, \dots, L$, and $\tilde{\mathbf{\Lambda}}_{l,q} \approx 0$, $l \neq q$ with $l, q = 1, \dots, L$.

The diagonal elements in $\tilde{\mathbf{\Lambda}}$ defined in (A.4) can be calculated by first post-multiplying $\mathbf{f}_l^H$ by a $1 \times L$ vector, expressed as

$$\tilde{\mathbf{\Lambda}}_{l,l} = \mathbf{f}_l^H \left[\mathbf{H}_1^c \mathbf{f}_l, \dots, \mathbf{H}_R^c \mathbf{f}_l, \mathbf{0}_{1 \times (L-R)} \right]^T, \quad l = 1, \dots, L. \quad (\text{A.5})$$

Based on (A.3), $\mathbf{H}_r^c \mathbf{f}_l = \lambda(l) f_{l,r}$, $r = 1, \dots, R$ and hence,

$$\begin{aligned} \tilde{\mathbf{\Lambda}}_{l,l} &= [f_{l,1}^*, \dots, f_{l,L}^*] \left[\lambda(l) f_{l,1}, \dots, \lambda(l) f_{l,R}, \mathbf{0}_{1 \times (L-R)} \right]^T \\ &= \frac{\lambda(l)}{L} \sum_{r=1}^R \exp \left[(j - j) \frac{2\pi(l-1)}{L} (r-1) \right] \\ &= \lambda(l) \frac{R}{L} \\ &= \mathbf{\Lambda}_{l,l} \frac{R}{L}. \end{aligned} \quad (\text{A.6})$$

Similarly, the non-diagonal elements in $\tilde{\mathbf{\Lambda}}$ defined in (A.4) are given by

$$\begin{aligned} \tilde{\mathbf{\Lambda}}_{l,q} &= \mathbf{f}_l^H \left[\mathbf{H}_1^c \mathbf{f}_q, \dots, \mathbf{H}_R^c \mathbf{f}_q, \mathbf{0}_{1 \times (L-R)} \right]^T \\ &= [f_{l,1}^*, \dots, f_{l,L}^*] \left[\lambda(q) f_{q,1}, \dots, \lambda(q) f_{q,R}, \mathbf{0}_{1 \times (L-R)} \right]^T \\ &= \frac{\lambda(q)}{L} \sum_{r=1}^R \exp \left[j \frac{2\pi(q-l)}{L} (r-1) \right]. \end{aligned} \quad (\text{A.7})$$

Since $\sum_{r=1}^R \exp[j\theta(r-1)] = [1 - \exp(jR\theta)]/[1 - \exp(j\theta)]$, (A.7) can be rewritten as

$$\tilde{\mathbf{\Lambda}}_{l,q} = \frac{\lambda(q)}{L} \times \frac{1 - \exp[j2\pi(q-l)\frac{R}{L}]}{1 - \exp[j2\pi(q-l)\frac{1}{L}]}. \quad (\text{A.8})$$

From (A.6), $\frac{\lambda(q)}{L} = \frac{\tilde{\Lambda}_{q,q}}{R}$ and therefore when $R \gg \tilde{\Lambda}_{q,q}$,

$$\frac{\lambda(q)}{L} = \frac{\tilde{\Lambda}_{q,q}}{R} \approx 0. \quad (\text{A.9})$$

Hence, for large L , the number of columns R in $\mathbf{H}^c$ becomes large, giving small values of $\tilde{\Lambda}_{l,q}$ in (A.8).

It can therefore be concluded that $\tilde{\Lambda}$ in (A.4) can be well approximated by a diagonal matrix.

Author's Publications

Journal papers

1. R. S. Rashobh, A. W. H. Khong and D. Liu, "Multichannel equalization in the KLT and frequency domains with application to speech dereverberation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 3, pp. 634–646, Jan. 2014.
2. R. S. Rashobh and A. W. H. Khong, "A fast frequency-domain algorithm for equalizing acoustic impulse responses," *IEEE Signal Process. Lett.*, Vol. 19, no. 12, pp. 797–800, Dec. 2012.

Conference papers

1. R. S. Rashobh and A. W. H. Khong, "Adaptive Multichannel Equalization Applied to Room Acoustics Exploiting Sparsity of Target Response," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, May 2014, pp. 5162–5166.
2. R. S. Rashobh and A. W. H. Khong, "A multichannel time-domain subspace approach exploiting multiple time-delays for acoustic channel equalization," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Jul. 2013, pp. 418–422.
3. R. S. Rashobh and A. W. H. Khong, "A variable step-size multichannel equalization algorithm exploiting sparseness measure for room acoustics," in *Proc. IEEE Int. Symp. Circuits Syst.*, Dec. 2012, pp. 797–800.
4. R. S. Rashobh, D. Liu and A. W. H. Khong, "An adaptive subspace based multichannel equalization algorithm for room acoustics equalization with sparseness constraint," in *Proc. IEEE Int. Conf. Info., Comm., Signal Process.*, Dec. 2011, pp. 1–5.
5. D. Liu, R. S. Rashobh, A. W. H. Khong and M. Yukawa, "A subspace-based adaptive approach for multichannel equalization of room acoustics," in *Proc. Asia-Pacific Signal and Info. Process. Assoc. Annual Summit and Conf.*, Oct. 2011.
6. R. S. Rashobh, A. W. H. Khong and P. A. Naylor, "Adaptive blind system identification for speech dereverberation using a priori estimates," in *Proc. IEEE Asia Pacific Conference on Circuits and Syst.*, Dec. 2010, pp. 632–635.

References

- [1] *Methods for subjective determination of transmission quality*, International Telecommunications Union (ITU-T) Std. Recommendation P.800, Feb. 1996.
- [2] J. Benesty, M. M. Sondhi, and Y. A. Huang, *Springer Handbook of Speech Processing*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.
- [3] E. Hänsler and G. Schmidt, *Speech and Audio Processing in Adverse Environments*. Springer Publishing Company Inc., 2008.
- [4] P. A. Naylor and N. D. Gaubitch, *Speech Dereverberation*. Springer Publishing Company Inc., 2010.
- [5] S. T. Neely and J. B. Allen, “Invertibility of a room impulse response,” *J. Acoust. Soc. Amer.*, vol. 66, pp. 165–169, Jul. 1979.
- [6] T. Hikichi, M. Delcroix, and M. Miyoshi, “Inverse filtering for speech dereverberation less sensitive to noise and room transfer function fluctuations,” *EURASIP J. Adv. Signal Process.*, vol. 2007, no. 1, pp. 1–12, Apr. 2007.
- [7] J. Benesty, J. Chen, and Y. Huang, “Noise reduction algorithms in a generalized transform domain,” *IEEE Trans. Audio, Speech, Language Process.*, vol. 17, no. 6, pp. 1109–1123, Aug. 2009.
- [8] L. Tong and S. Perreau, “Multichannel blind identification: From sub-space to maximum likelihood methods,” in *Proc. IEEE*, vol. 86, no. 10, Oct. 1998, pp. 1951–1968.
- [9] N. D. Gaubitch, T. Hasan, and P. A. Naylor, “Noise robust adaptive blind channel identification using spectral constraints,” in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, May 2006.
- [10] R. S. Rashobh and A. W. H. . Khong, “A multichannel time-domain subspace approach exploiting multiple time-delays for acoustic channel equalization,” in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Jul. 2013, pp. 418–422.
- [11] R. S. Rashobh, A. W. H. . Khong, and D. Liu, “Multichannel equalization in the KLT and frequency domains with application to speech dereverberation,” *IEEE Trans. Audio, Speech, Lang.*

Process., submitted.

- [12] D. Liu, R. S. Rashobh, A. W. H. Khong, and M. Yukawa, "A subspace-based adaptive approach for multichannel equalization of room acoustics," in *Proc. Asia Pac. Sig. Info. Process. Assoc. Ann. Summ. Conf. (APSIPA ASC 2011)*, Oct. 2011.
- [13] R. S. Rashobh, D. Liu, and A. W. H. Khong, "An adaptive subspace based multichannel equalization algorithm for room acoustics equalization with sparseness constraint," in *Proc. IEEE Int. Conf. Info., Comm., Signal Process.*, Dec. 2011, pp. 1–5.
- [14] R. S. Rashobh and A. W. H. Khong, "A variable step-size multichannel equalization algorithm exploiting sparseness measure for room acoustics," in *Proc. IEEE Int. Symp. Circuits and Sys.*, May. 2012, pp. 2753–2756.
- [15] R. S. Rashobh, D. Liu, and A. W. H. Khong, "Adaptive multichannel equalization of room acoustics in the frequency domain," in *Proc. Int. Conf. Signal Process. Sys.*, Aug. 2011.
- [16] R. S. Rashobh and A. W. H. . Khong, "A fast frequency-domain algorithm for equalizing acoustic impulse responses," *EEE Signal Process. Lett.*, vol. 19, no. 12, pp. 797–800, Dec. 2012.
- [17] G. Fant, *Acoustic Theory of Speech Production*. The Hague: Mouton, 1960.
- [18] R. Y. Litovsky, H. S. Colburn, W. A. Yost, and S. J. Guzman, "The precedence effect," *J. Acoust. Soc. Am.*, vol. 106, no. 4, pp. 1633–1654, 1999.
- [19] J. S. Bradley, H. Sato, and M. Picard, "On the importance of early reflections for speech in rooms," *J. Acoust. Soc. Am.*, vol. 113, no. 6, pp. 3233–3244, 2003.
- [20] H. Kuttruff, *Room acoustics*, 4th ed. Taylor & Francis, 2000.
- [21] G. Souloudre, N. Popplewell, and J. Bradley, "Combined effects of early reflections and background noise on speech intelligibility," *J. Sound and Vibration*, vol. 135, no. 1, pp. 123–133, 1989.
- [22] M. Omologo, P. Svaizer, and M. Matassoni, "Environmental conditions and acoustic transduction in hands-free speech recognition," *Speech Communication*, vol. 25, pp. 75–95, 1998.
- [23] J. Lochner and J. Burger, "The influence of reflections on auditorium acoustics," *J. Sound and Vibration*, vol. 1, no. 4, pp. 426 – 454, 1964.
- [24] R. H. Bolt and A. D. MacDonald, "Theory of speech masking by reverberation," *J. Acoust. Soc. Am.*, vol. 21, no. 6, pp. 577–580, 1949.

- [25] A. K. Nábělek, T. R. Letowski, and F. M. Tucker, "Reverberant overlap- and self-masking in consonant identification," *J. Acoust. Soc. Am.*, vol. 86, no. 4, pp. 1259–65, 1989.
- [26] W. C. Sabine, *Collected Papers on Acoustics*. Cambridge : Harvard University Press, 1922.
- [27] A. Chesnokov and L. SooHoo, "Influence of early to late energy ratios on subjective estimates of small room acoustics," in *Audio Engineering Society Convention 105*, Sep. 1998, pp. 1–18.
- [28] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing: Concepts and Techniques*, facsimile ed. Prentice Hall, 1993.
- [29] M. Brandstein and D. Ward, Eds., *Microphone Arrays: Signal Processing Techniques and Applications*, 1st ed. Springer, Jun. 2001.
- [30] B. D. Van Veen and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE Signal Process. Mag.*, vol. 5, no. 2, pp. 4–24, 1988.
- [31] G. Elko, "Microphone array systems for hands-free telecommunication," *Speech Commun.*, vol. 20, no. 3-4, pp. 229–240, Dec. 1996.
- [32] J. B. Allen, D. A. Berkley, and J. Blauert, "Multimicrophone signal-processing technique to remove room reverberation from speech signals," *J. Acoust. Soc. Am.*, vol. 62, no. 4, pp. 912–915, Oct. 1977.
- [33] G. DeMuth, "Frequency domain beamforming techniques," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, vol. 2, May 1977, pp. 713–715.
- [34] E. A. P. Habets and J. Benesty, "A perspective on frequency-domain beamformers in room acoustics," *IEEE Trans. Audio, Speech and Language Process.*, vol. 20, no. 3, pp. 947–960, Mar. 2012.
- [35] S. Haykin, *Adaptive filter theory*. 4th ed. Upper Saddle River, NJ: Prentice-Hall, 2002.
- [36] L. J. Griffiths and C. W. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas, Propag.*, vol. 30, no. 1, pp. 27–34, Jan. 1982.
- [37] H. Cox, R. Zeskind, and M. Owen, "Robust adaptive beamforming," *IEEE Trans. Acoust. Speech, Signal Process.*, vol. 35, no. 10, pp. 1365–1376, Oct. 1987.
- [38] H. L. V. Trees, *Optimum Array Processing (Detection, Estimation, and Modulation Theory, Part IV)*, 1st ed., 2002.

- [39] S. Affes and Y. Grenier, "A signal subspace tracking algorithm for microphone array processing of speech," *IEEE Trans. Speech, Audio, Process.*, vol. 5, no. 5, pp. 425–437, Sep. 1997.
- [40] O. Hoshuyama, A. Sugiyama, and A. Hirano, "A robust adaptive beamformer for microphone arrays with a blocking matrix using constrained adaptive filters," *IEEE Trans. Signal Process.*, vol. 47, no. 10, pp. 2677–2684, Oct. 1999.
- [41] S. Gannot, D. Burshtein, and E. Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Trans. Signal Process.*, vol. 49, no. 8, pp. 1614–1626, Aug. 2001.
- [42] E. Warsitz and R. Hae-Umbach, "Acoustic filter-and-sum beamforming by adaptive principal component analysis," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, vol. 4, Mar. 2005, pp. 797–800.
- [43] J. Bitzer, K. Simmer, and K. D. Kammeyer, "Theoretical noise reduction limits of the generalized sidelobe canceller (GSC) for speech enhancement," in *IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 5, 1999, pp. 2965–2968.
- [44] E. A. P. Habets, J. Benesty, and P. A. Naylor, "A speech distortion and interference rejection constraint beamformer," *IEEE Trans. Audio, Speech and Language Process.*, vol. 20, no. 3, pp. 854–867, Mar. 2012.
- [45] J. Capon, "High resolution frequency-wavenumber spectrum analysis," *Proc. IEEE*, vol. 57, no. 8, pp. 1408–1418, Aug. 1969.
- [46] E. A. P. Habets, J. Benesty, I. Cohen, and S. Gannot, "On a tradeoff between dereverberation and noise reduction using the MVDR beamformer," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Apr. 2009, pp. 3741–3744.
- [47] E. A. P. Habets, J. Benesty, I. Cohen, S. Gannot, and J. Dmochowski, "New insights into the MVDR beamformer in room acoustics," *IEEE Trans. Audio, Speech and Language Process.*, vol. 18, no. 1, pp. 158–170, Jan. 2010.
- [48] C. Marro, Y. Mahieux, and K. U. Simmer, "Analysis of noise reduction and dereverberation techniques based on microphone arrays with postfiltering," *IEEE Trans. Speech Audio Porcess.*, vol. 6, no. 3, pp. 240–259, May 1998.

- [49] I. Cohen, S. Gannot, and B. Berdugo, "An integrated real-time beamforming and postfiltering system for non-stationary noise environments," *EURASIP J. Appl. Signal Process.*, vol. 2003, pp. 1064–1073, Jan. 2003.
- [50] H. Dudley, "The vocoder," *Bell Labs Record*, vol. 17, no. 2, pp. 122–126, 1939.
- [51] H. Attias, J. C. Platt, A. Acero, and L. Deng, "Speech Denoising and Dereverberation Using Probabilistic Models," in *Advances in Neural Info. Process. Sys.*, Dec. 1999.
- [52] J. Hardwick, C. D. Yoo, and J. S. Lim, "Speech enhancement using the dual excitation speech model," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, no. 367-370, 1993.
- [53] C. D. Yoo and J. S. Lim, "Speech enhancement based on the generalized dual excitation model with adaptive analysis window," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 1, 1995, pp. 832–835.
- [54] M. S. Brandstein, "On the use of explicit speech modeling in microphone array applications," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 6, 1998, pp. 3613–3616.
- [55] M. Brandstein and S. Griebel, "Explicit speech modeling for microphone array applications," in *Microphone Arrays: Signal Processing Techniques and Applications*. Springer, 2001, ch. 7, pp. 133–153.
- [56] L. Rabiner and R. Schafer, *Digital Processing of Speech Signals*. Englewood Cliffs: Prentice Hall, 1978.
- [57] M. Brandstein and S. Griebel, "Nonlinear, modelbased microphone array speech enhancement," in *Acoust. Signal Process. for Telecommun.*, S. L. Gay and J. Benesty, Eds. Kluwer Academic, Boston, 2000, ch. 12, pp. 261–279.
- [58] N. D. Gaubitch, D. B. Ward, and P. A. Naylor, "Statistical analysis of the autoregressive modeling of reverberant speech," *J. Acoust. Soc. Am.*, vol. 120, no. 6, pp. 4031–4039, 2006.
- [59] J. B. Allen, "Synthesis of pure speech from a reverberant signal," U. S Patent 3 786 188, 1974.
- [60] S. M. Griebel and M. S. Brandstein, "Wavelet transform extrema clustering for multi-channel speech dereverberation," in *Proc. Int. Workshop Acoust. Echo Noise Control*, 1999.
- [61] S. M. Griebel, "A microphone array system for speech source localization, denoising and dereverberation," Ph.D. thesis, Harvard University, Cambridge, Massachusetts, 2002.

- [62] S. M. Griebel and M. Brandstein, "Microphone array speech dereverberation using coarse channel modeling," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 1, Nov. 2001, pp. 201–204.
- [63] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech and Signal Process.*, vol. 24, no. 4, pp. 320–327, Aug. 1976.
- [64] B. Yegnanarayana, P. S. Murthy, C. Avendano, and H. Hermansky, "Enhancement of reverberant speech using LP residual," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 1, 1998, pp. 405–408.
- [65] B. Yegnanarayana and P. Satyanarayana, "Enhancement of reverberant speech using LP residual signals," *IEEE. Trans. Speech, Siganl Process.*, vol. 8, no. 3, pp. 267–281, May 2000.
- [66] B. Yegnanarayana, S. R. M. Prasanna, and K. S. Rao, "Speech enhancement using excitation source information," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, 2002, pp. 541–544.
- [67] B. Yegnanarayana, S. R. M. Prasanna, R. Duraiswami, and D. Zotkin, "Processing of reverberant speech for time-delay estimation," *IEEE. Trans. Speech, Audio, Process.*, vol. 13, no. 2, pp. 1110–1118, Feb. 2005.
- [68] N. D. Gaubitch, P. A. Naylor, and D. Ward, "On the use of linear prediction for dereverberation of speech," in *Proc. Int. Workshop Acoust. Echo Noise Control*, 2003, pp. 99–102.
- [69] B. Gillespie, H. Malvar, and D. Florêncio, "Speech dereverberation via maximum-kurtosis subband adaptive filtering," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, vol. 6, 2001, pp. 3701–3074.
- [70] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. Wiley-Interscience, 2001.
- [71] M. Tonelli and N. Mitianoudis, "A maximum likelihood approach to blind audio de-reverberation," in *Proc. Int. Conf. Digital Audio Effects*, Oct. 2004, pp. 1–6.
- [72] M. Tonelli, M. G. Jafari, and M. E. Davies, "A multi-channel maximum likelihood approach to de-reverberation," in *Proc. European Signal Process. Conf.*, Sep. 2006.
- [73] N. D. Gaubitch, D. B. Ward, and P. A. Naylor, "Statistical analysis of the autoregressive modeling

- of reverberant speech,” *J. Acoust. Soc. Am.*, vol. 120, no. 6, pp. 4031–4039, Dec. 2006.
- [74] N. D. Gaubitch and P. A. Naylor, “Spatiotemporal averaging method for enhancement of reverberant speech,” in *Proc. Int. Conf. Digital Signal Process.*, Jul. 2007, pp. 607–610.
- [75] M. R. P. Thomas, N. D. Gaubitch, J. Gudnason, and P. A. Naylor, “A practical multichannel dereverberation algorithm using multichannel DYPSA and spatiotemporal averaging,” in *Proc. Int. Workshop Applications of Signal Process. to Audio, Acoust.*, Oct. 2007.
- [76] M. Triki and D. T. M. Slock, “Blind dereverberation of quasi-periodic sources based on multi-channel linear prediction,” in *Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2005.
- [77] —, “Delay and predict equalization for blind speech dereverberation,” in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, vol. 5, May 2006, p. V.
- [78] M. Miyoshi, “Estimating AR parameter-sets for linear-recurrent signals in convolutive mixtures,” in *Proc. Int. Symp. ICA and BSS*, 2003, pp. 585–589.
- [79] M. Delcroix, T. Hikichi, and M. Miyoshi, “Blind dereverberation algorithm for speech signals based on multi-channel linear prediction,” *Acoust. Sci. Tech.*, vol. 26, no. 5, pp. 432–439, 2005.
- [80] —, “On the use of lime dereverberation algorithm in an acoustic environment with a noise source,” in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, vol. 1, May 2006, p. I.
- [81] M. Miyoshi, M. Delcroix, K. Kinoshita, T. Yoshioka, T. Nakatani, and T. Hikichi, “Inverse filtering for speech dereverberation without the use of room acoustics information,” in *Speech Dereverberation*, ser. Signals and Communication Technology, P. A. Naylor and N. D. Gaubitch, Eds. Springer London, 2010, pp. 271–310.
- [82] T. Nakatani, T. Hikichi, K. Kinoshita, T. Yoshioka, and M. Delcroix, “Robust blind dereverberation of speech signals based on characteristics of short-time speech segments,” in *Proc. IEEE Int. Symp. Circuits and Sys.*, May 2007, pp. 2986–2989.
- [83] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B. Juang, “Blind speech dereverberation with multi-channel linear prediction based on short time fourier transform representation,” in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, 2008, pp. 85–88.
- [84] M. Delcroix, T. Hikichi, and M. Miyoshi, “Dereverberation and denoising using multichannel linear prediction,” *IEEE Trans. Audio, Speech and Language Process.*, vol. 15, no. 6, pp. 1791–

1801, Aug. 2007.

- [85] K. Kinoshita, M. Delcroix, T. Nakatani, and M. Miyoshi, "Suppression of late reverberation effect on speech signal using long-term multiple-step linear prediction," *IEEE Trans. Audio, Speech and Language Process.*, vol. 17, no. 4, pp. 1–12, May 2009.
- [86] P. Krishnamoorthy and S. R. M. Prasanna, "Reverberant speech enhancement by temporal and spectral processing," *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 17, no. 2, pp. 253–266, Feb. 2009.
- [87] T. Nakatani and M. Miyoshi, "Blind dereverberation of single channel speech signal based on harmonic structure," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, vol. 1, 2003, pp. 92–95.
- [88] T. Nakatani, K. Kinoshita, and M. Miyoshi, "Harmonicity-based blind dereverberation for single-channel speech signals," *IEEE Trans. Audio, Speech and Language Process.*, vol. 15, no. 1, pp. 80–95, Jan. 2007.
- [89] T. Nakatani, B. Juang, K. Kinoshita, and M. Miyoshi, "Harmonicity based dereverberation with maximum a posteriori estimation," in *Proc. IEEE Int. Workshop Applications of Signal Process. to Audio, Acoust.*, Oct. 2005, pp. 16–19.
- [90] K. Kinoshita, T. Nakatani, and M. Miyoshi, "Harmonicity based dereverberation for improving automatic speech recognition performance and speech intelligibility," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E88-A, no. 7, pp. 1724–1731, 2005.
- [91] —, "Fast estimation of a precise dereverberation filter based on speech harmonicity," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, vol. 1, Mar. 2005, pp. 1073–1076.
- [92] T. Iida, Y. Nomura, J. Lu, H. Sekiya, and T. Yahagi, "Blind dereverberation using correlation coefficients considering periodicity of voiced speech," in *Proc. IEEE Int. Symp. Circuits and Sys.*, May 2006.
- [93] K. Lebart, J. Boucher, and P. Denbigh, "A new method based on spectral subtraction for speech dereverberation," *Acta Acoustica*, vol. 87, no. 3, pp. 359–366, 2001.
- [94] E. A. P. Habets, "Single-channel speech dereverberation based on spectral subtraction," in *Proc. 15th Annual Workshop on Circuits, Systems and Signal Process.*, Nov. 2004, pp. 250–254.

- [95] E. A. P. Habets, S. Gannot, and I. Cohen, "Dual-microphone speech dereverberation in a noisy environment," in *Proc. IEEE Int. Symp. Signal Process., Info. Tech. (ISSPIT)*, Aug. 2006, pp. 651–655.
- [96] E. A. P. Habets, "Single- and multi-microphone speech dereverberation using spectral enhancement," Ph.D. thesis, Technische Universiteit Eindhoven, 2007.
- [97] K. Lebart, J. M. Boucher, and P. N. Denbigh, "A new method based on spectral subtraction for the suppression of late reverberation from speech signals," in *Proc. Audio Engineering Soc. Convention 105*, 1998.
- [98] J. D. Polack, "La transmission de l'énergie sonore dans les salles," Ph.D. Thesis, Université du Maine, La mans, 1988.
- [99] I. Tashev and D. Allred, "Reverberation reduction for improved speech recognition," in *Proc. the Joint Workshop on Hands-Free Speech Commun., Mic. Arrays (HSCMA'05)*, Mar. 2005.
- [100] E. A. P. Habets, S. Gannot, I. Cohen, and P. Sommen, "Joint dereverberation and residual echo suppression of speech signals in noisy environments," *IEEE. Trans. Audio, Speech, Language Process.*, vol. 16, no. 8, pp. 1433–1451, Nov. 2008.
- [101] E. A. P. Habets, N. D. Gaubitch, and P. A. Naylor, "Temporal selective dereverberation of noisy speech using one microphone," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, Sep. 2008, pp. 4577–4580.
- [102] E. A. P. Habets, S. Gannot, and I. Cohen, "Speech dereverberation using backward estimation of the late reverberant spectral variance," in *Proc. IEEE 25th Convention of Electric. Electron. Engineers*, Dec. 2008, pp. 384–388.
- [103] E. A. P. Habets, S. Gannot, and I. Cohen, "Late reverberant spectral variance estimation based on a statistical model," *IEEE Signal Process. Lett.*, vol. 16, no. 9, pp. 770–773, Sep. 2009.
- [104] J. S. Erkelens and R. Heusdens, "Single-microphone late-reverberation suppression in noisy speech by exploiting long-term correlation in the DFT domain," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Apr. 2009, pp. 3997–4000.
- [105] M. Wu and D. Wang, "A one-microphone algorithm for reverberant speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 1, Apr. 2003, pp. I–892–I–895.

- [106] —, “A two-stage algorithm for enhancement of reverberant speech,” in *IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 1, Mar. 2005, pp. 1085–1088.
- [107] —, “A two-stage algorithm for one-microphone reverberant speech enhancement,” *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 14, no. 3, pp. 774–784, May 2006.
- [108] N. D. Gaubitch, “Multimicrophone speech dereverberation using spatiotemporal and spectral processing,” in *Proc. IEEE Int. Symp. Circuits and Sys.*, May 2008, pp. 3222–3225.
- [109] K. Furuya and A. Kataoka, “Robust speech dereverberation using multichannel blind deconvolution with spectral subtraction,” *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 15, no. 5, pp. 1579–1591, Jul. 2007.
- [110] T. Houtgast and H. Steeneken, “A review of the MTF concept in room acoustics and its use for estimating speech intelligibility in auditoria,” *J. Acoust. Soc. Am.*, vol. 77, no. 3, pp. 1069–1077, 1985.
- [111] T. Langhans and W. Strube, “Speech enhancement by nonlinear multiband envelope filtering,” in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 7, May 1982, pp. 156–159.
- [112] H. G. Hirsch, “Automatic speech recognition in rooms,” in *Signal Process. IV: Theories and Applications. Proc. of EUSIPCO-88. 4th European Signal Process. Conf.* Elsevier Science Publishers B.V, 1988, pp. 1177–1180.
- [113] J. M. J. Hammond, “Modelling and enhancement of reverberant speech using an envelope convolution method,” in *IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 8, Apr. 1983, pp. 1144–1147.
- [114] S. Hirobayashi, H. Nomura, T. Koike, and M. Tohyama, “Speech waveform recovery from a reverberant speech signal using inverse filtering of power envelope transfer function,” *IEEE Trans. Inst. Electron. Info. Commun. Engineers*, vol. J81-A, no. 10, pp. 1323–1330, 1998.
- [115] H. Wang and F. Itakura, “An approach to dereverberation using multi-microphone sub-band envelope estimation,” in *IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 2, 1991, pp. 953–956.
- [116] H. Hermansky, E. A. Wan, and C. Avendano, “Speech enhancement based on temporal processing,” in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 1, May 1995, pp. 405–408.

- [117] C. Avendano and H. Hermansky, "Study on the dereverberation of speech based on temporal envelope filtering," in *Proc. Fourth Int. Conf. Spoken Language (ICSLP 96)*, vol. 2, Oct. 1996, pp. 889–892.
- [118] 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 Int. Conf. Acoust., Speech, and Signal Process.*, vol. 1, Apr. 2003, pp. I–888–I–891.
- [119] K. Sakata, M. Unoki, M. Furukawa, and M. Akagi, "An improved method based on the MTF concept for restoring the power envelope from a reverberant signal," *Acoustical Science and Technology*, vol. 25, no. 4, pp. 232–242, 2004.
- [120] —, "A speech dereverberation method based on MTF concept in power envelope restoration," *Acoustical Science and Technology*, vol. 25, no. 4, pp. 243–254, 2004.
- [121] M. Unoki and X. Lu, "Unified denoising and dereverberation method used in restoration of MTF-based power envelope," in *Proc. Int. Symp. Chinese Spoken Language Process.*, Dec. 2012, pp. 215–219.
- [122] A. V. Oppenheim and R. W. Schaffer, *Digital Signal Processing*. Prentice Hall, Englewood Cliffs, NJ, 1975.
- [123] D. Bees, M. Blostein, and P. Kabal, "Reverberant speech enhancement using cepstral processing," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, May 1991, pp. 977–980.
- [124] M. Tohyama, R. H. Lyon, and T. Koike, "Source waveform recovery in a reverberant space by cepstrum dereverberation," in *IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 1, Apr. 1993, pp. 157–160.
- [125] Q. Liu, B. Champagne, and P. Kabal, "Room speech dereverberation via minimum-phase and all-pass component processing of multi-microphone signals," in *Proc. IEEE Pacific Rim Conf. Commun., Computers, Signal Process. (Victoria, BC)*, May 1995, pp. 571–574.
- [126] S. Subramaniam, A. P. Petropulu, and C. Wendt, "Cepstrum-based deconvolution for speech dereverberation," *IEEE Trans. Speech, Audio Porcess.*, vol. 4, no. 5, pp. 392–396, Sep. 1996.
- [127] R. A. Kennedy and B. D. Radlovic, "Iterative cepstrum-based approach for speech dereverberation," in *Proc. Int. Symposium Signal process. and its Applications*. Signal Process. Research

Centre, Brisbane, Australia, August 1999, pp. 55–58.

- [128] C. Breithaupt, T. Gerkmann, and R. Martin, “Cepstral smoothing of spectral filter gains for speech enhancement without musical noise,” *IEEE Signal Process. Lett.*, vol. 14, no. 12, pp. 1036–1039, Dec. 2007.
- [129] R. S. Rashobh, A. W. H. Khong, and P. A. Naylor, “Speech dereverberation employing adaptive blind multichannel identification algorithm using a priori estimates of room impulse responses,” in *Proc. IEEE Asia Pacific Conf. Circuits and Systems*, 2010, pp. 632–635.
- [130] K. Kumar and R. M. Stern, “Maximum likelihood based cepstral inverse filtering for blind speech dereverberation,” in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Mar. 2010, pp. 4282–4285.
- [131] Y. Huang and J. Benesty, “Adaptive multi-channel least mean square and Newton algorithms for blind channel identification,” *Signal Process.*, vol. 82, pp. 1127–1138, Aug. 2002.
- [132] S. Gannot and M. Moonen, “Subspace methods for multimicrophone speech dereverberation,” *EURASIP J. Appl. Signal Process.*, vol. 11, pp. 1074–1090, Oct. 2003.
- [133] S. Haykin, *Blind Deconvolution (Prentice Hall Information and System Sciences)*, 4th ed. Prentice-Hall, 1994.
- [134] Y. Sato, “A method of self-recovering equalization for multilevel amplitude-modulation,” *IEEE Trans. Commun.*, vol. COM-23, pp. 679–682, Jun. 1975.
- [135] J. A. Cadzow, “Blind deconvolution via cumulant extrema,” *IEEE Signal Process. Mag.*, vol. 13, no. 3, pp. 24–42, May 1996.
- [136] T. Yoshioka, T. Hikichi, M. Miyoshi, and H. G. Okuno, “Robust decomposition of inverse filter of channel and prediction error filter of speech signal for dereverberation,” in *Proc. European Signal Process. Conf.*, Sep. 2006.
- [137] L. Tong, G. Xu, and T. Kailath, “A new approach to blind identification and equalization of multipath channels,” in *Proc. 25th Asilomar Conf. Signals, Systems, Computers*, vol. 2, Nov. 1991, pp. 856–860.
- [138] H. Liu, G. Xu, and L. Tong, “A deterministic approach to blind identification of multi-channel FIR systems,” in *Proc. IEEE Int. Conf. on Acoust., Speech, Signal Process.*, vol. 4, Apr. 1994, pp.

IV/581–IV/584.

- [139] G. Xu, H. Liu, L. Tong, and T. Kailath, “A least-squares approach to blind channel identification,” *IEEE Trans. Signal Process.*, vol. 43, no. 12, pp. 2982–2993, Dec. 1995.
- [140] E. Moulines, P. Duhamel, J. F. Cardoso, and S. Mayrague, “Subspace methods for the blind identification of multichannel FIR filters,” *IEEE Trans. Signal Process.*, vol. 43, no. 2, pp. 516–525, Feb. 1995.
- [141] Y. Hua, “Fast maximum likelihood for blind identification of multiple FIR channels,” in *Proc. 28th Asilomar Conf. Signals, Systems, Computers*, vol. 1, Nov. 1994, pp. 415–419.
- [142] —, “Fast maximum likelihood for blind identification of multiple FIR channels,” *IEEE Trans. Signal Process.*, vol. 44, no. 3, pp. 661–672, Mar. 1996.
- [143] M. I. Gurelli and C. L. Nikias, “EVAM: An eigenvector-based algorithm for multichannel blind deconvolution of input colored signals,” *IEEE Trans. Signal Process.*, vol. 43, pp. 134–149, 1995.
- [144] S. Gannot and M. Moonen, “Subspace methods for multi-microphone speech dereverberation,” in *Proc. Int. Workshop Acoust. Echo Noise Control*, 2001.
- [145] K. Furuya and Y. Kaneda, “Two-channel blind deconvolution for non-minimum phase impulse responses,” in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 2, 1997, pp. 1315–1318.
- [146] T. Hikichi, M. Delcroix, and M. Miyoshi, “Speech dereverberation algorithm using transfer function estimates with overestimated order,” *Acoust. Sci. Tech.*, vol. 27, no. 1, pp. 28–35, 2006.
- [147] —, “Blind dereverberation based on estimates of signal transmission channels without precise information on channel order,” in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, vol. 1, Mar. 2005, pp. 1069–1072.
- [148] J. K. Tugnait, “Blind spatio-temporal equalization and impulse response estimation for MIMO channels using a Godard cost function,” *IEEE Trans. Signal Process.*, vol. 45, no. 1, pp. 268–271, Jan. 1997.
- [149] L. Couvreur, “Blind deconvolution for multi-microphone speech dereverberation: Application to ASR in reverberant environments,” in *Proc. 3rd IEEE Signal Process. Symp.*, Mar. 2002.
- [150] J. Hopgood, “Nonstationary signal processing with application to reverberation cancellation in

acoustic environments,” Ph.D. thesis, Cambridge University, 2001.

- [151] T. Yoshioka, T. Hikichi, and M. Miyoshi, “Second-order statistics based dereverberation by using nonstationarity of speech,” in *Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2006.
- [152] H. Buchner, R. Aichner, and W. Kellermann, “TRINICON: a versatile framework for multichannel blind signal processing,” in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 3, May 2004, pp. iii – 889–92.
- [153] Y. Huang and J. Benesty, “A class of frequency-domain adaptive approaches to blind multichannel identification,” *IEEE Trans. Signal Process.*, vol. 51, no. 1, pp. 11–24, Jan. 2003.
- [154] R. Ahmad, A. W. H. Khong, and P. A. Naylor, “Proportionate frequency domain adaptive algorithms for blind channel identification,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 5, May 2006.
- [155] M. K. Hasan and P. A. Naylor, “Analyzing effect of noise on LMS-type approaches to blind estimation of SIMO channels: robustness issue,” in *Proc. European Signal Process. Conf.*, Sep. 2006.
- [156] M. K. Hasan, J. Benesty, P. A. Naylor, and D. B. Ward, “Improving robustness of blind adaptive multichannel identification algorithms using constraints,” in *Proc. European Signal Process. Conf.*, ser. CD-ROM, Sep. 2005.
- [157] R. Ahmad, A. W. H. . Khong, and P. A. Naylor, “An extended normalized multichannel flms algorithm for blind channel identification,” in *Proc. European Signal Process. Conf.*, Sep. 2006.
- [158] M. A. Haque, T. Islam, and M. K. Hasan, “Robust speech dereverberation based on blind adaptive estimation of acoustic channels,” *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 4, pp. 775–787, May 2011.
- [159] L. Liao and A. W. H. Khong, “A noise robust multichannel algorithm for blind estimation of room impulse responses,” in *Proc. Int. Workshop Acoust., Echo, Noise Control*, Aug. 2010.
- [160] R. Ahmad, A. W. H. Khong, and P. A. Naylor, “A practical adaptive blind multichannel estimation algorithm with application to acoustic impulse responses,” in *Proc. IEEE Int. Conf. Digital Signal Process.*, Jul. 2007, pp. 31–34.
- [161] X. Lin, A. W. H. Khong, and P. A. Naylor, “A forced spectral diversity algorithm for speech

- dereverberation in the presence of near-common zeros,” *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 20, no. 3, pp. 888–899, Mar. 2012.
- [162] A. W. H. Khong, X. S. Lin, and P. A. Naylor, “Algorithms for identifying clusters of near-common zeros in multichannel blind system identification and equalization,” in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Mar. 2008, pp. 389–392.
- [163] P. A. Naylor, X. Lin, and A. W. H. Khong, “Near-common zeros in blind identification of SIMO acoustic systems,” in *Proc. Joint Workshop Hands-Free Speech Commun. Mic. Arrays (HSCMA)*, May 2008, pp. 21–24.
- [164] F. S. Pacheco and R. Seara, “A single-microphone approach for speech signal dereverberation,” in *Proc. European Signal Process. Conf.*, Sep. 2005.
- [165] S. Mosayyebpour, A. Sayyadiyan, M. Zareian, and A. Shahbazi, “Single channel inverse filtering of room impulse response by maximizing skewness of LP residual,” in *Proc. Int. Conf. Signal Acquisition Process. (ICSAP '10)*, Feb. 2010, pp. 130–134.
- [166] T. Yoshioka, T. Nakatani, M. Miyoshi, and H. G. Okuno, “Blind separation and dereverberation of speech mixtures by joint optimization,” *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 19, no. 1, pp. 69–84, Jan. 2011.
- [167] R. Rotili, S. Cifani, E. Principi, S. Squartini, and F. Piazza, “A robust iterative inverse filtering approach for speech dereverberation in presence of disturbances,” in *Proc. IEEE Asia Pacific Conf. Circuits and Systems*, Dec. 2008, pp. 434–437.
- [168] J. Jeong, S. Lee, Y. Park, and D. Youn, “Multi-channel dereverberation system using modified correlation-based blind deconvolution and multi-microphone spectral subtraction,” in *Audio Engineering Society Convention 124*, May 2008.
- [169] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B. Juang, “Speech dereverberation based on variance-normalized delayed linear prediction,” *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 18, no. 7, pp. 1717–1731, Sep. 2010.
- [170] A. Tsilfidis and J. Mourjopoulos, “Blind single-channel suppression of late reverberation based on perceptual reverberation modeling,” *J. Acoust. Soc. Am.*, vol. 129, no. 3, pp. 1439–1451, 2011.
- [171] S. K. Mitra, *Digital Signal Processing: A Computer-Based Approach*, 2nd ed. McGraw-Hill

Higher Education, 2000.

- [172] J. Mourjopoulos, P. Clarkson, and J. Hammond, "A comparative study of least-squares and homomorphic techniques for the inversion of mixed phase signals," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 7, May 1982, pp. 1858–1861.
- [173] M. Miyoshi and Y. Kaneda, "Inverse filtering of room acoustics," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 2, pp. 145–152, Feb. 1988.
- [174] W. Putnam, D. Rocchesso, and J. Smith, "A numerical investigation of the invertibility of room transfer functions," in *Proc. IEEE ASSP Workshop on Applications of Signal Process. to Audio and Acoust.*, Oct. 1995, pp. 249–252.
- [175] T. Hikichi, M. Delcroix, and M. Miyoshi, "Inverse filtering for speech dereverberation less sensitive to noise," in *Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2006, pp. 1–4.
- [176] T. Hikichi, M. Delcroix, and M. Moiyoshi, "On robust inverser filter design for room transfer function fluctuations," in *Proc. European Signal Process. Conf.*, Sep. 2006.
- [177] N. Nagata, Y. Tatekura, H. Saruwatari, and K. Shikano, "Iterative inverse filter relaxation algorithm for adaptation to acoustic fluctuation in sound reproduction system," *Electron. Commun. in Japan*, vol. 87, no. 7, 2004.
- [178] M. Kallinger and A. Mertins, "Multi-channel room impulse response shaping - a study," in *Proc. Int. Conf. Acoust., Speech, and Signal Process.*, vol. 5, May 2006.
- [179] —, "Room impulse response shortening by channel shortening concepts," in *Proc. Asilomar Conf. Signals, Systems, Computers*, Nov. 2005, pp. 898–902.
- [180] —, "Impulse response shortening for acoustic listening room compensation," in *Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2005, pp. 197–200.
- [181] W. Zhang, E. A. P. Habets, and P. A. Naylor, "On the use of channel shortening in multichannel acoustic system equalization," in *Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2010.
- [182] I. Kodrasi, S. Goetze, and S. Doclo, "Increasing the robustenss of acoustic multichannel equalaiza-tion by means of regularization," in *Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2012.
- [183] I. Kodrasi and S. Doclo, "Robust partial multichannel equalization techniques for speech derever-beration," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Mar. 2012, pp. 537–540.

- [184] F. Lim and P. A. Naylor, "Relaxed multichannel least squares with constrained initial taps for multichannel dereverberation," in *Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2012, pp. 1–4.
- [185] S. J. Elliot and P. A. Nelson, "Algorithm for multichannel LMS adaptive filtering," *Electron. Lett.*, vol. 21, no. 21, pp. 979–981, Oct. 1985.
- [186] S. J. Elliot, I. M. Stothers, and P. A. Nelson, "A multiple error LMS algorithm and its application to the active control of sound and vibration," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 10, pp. 1423–1434, Oct. 1987.
- [187] P. A. Nelson, F. Orduna-Bustamante, and H. Hamada, "Inverse filter design and equalization zones in multichannel sound reproduction," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 3, pp. 185–192, May 1995.
- [188] W. Zhang, A. W. H. Khong, and P. A. Naylor, "Adaptive inverse filtering of room acoustics," in *Proc. IEEE Asilomar Conf. Signals, Sys., and Computers*, Oct. 2008, pp. 788–792.
- [189] L. Liao and A. W. H. Khong, "Adaptive channel equalization of room acoustics exploiting sparseness constraint," *IEEE Signal Process. Letts.*, vol. 18, no. 4, pp. 275–278, Apr. 2011.
- [190] —, "Equalization of multichannel acoustic system using sub-systems for speech dereverberation," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, May 2011, pp. 313 – 316.
- [191] L. Liao and A. W. H. . Khong, "An adaptive sub-systems based algorithm for channel equalization in a SIMO system," *IEEE Trans. Circuits and Sys. I*, no. 99, Dec. 2012.
- [192] H. Yamada, H. Wang, and F. Itakura, "Recovering of broad band reverberant speech signal by sub-band MINT method," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, vol. 2, Apr. 1991, pp. 969–972.
- [193] H. Wang and F. Itakura, "Realization of acoustic inverse filtering through multi-microphone sub-band processing," *IEICE Trans. Fund.*, vol. E75-A, no. 11, pp. 1474–1483, Nov. 1992.
- [194] S. Weiss, G. W. Rice, and R. W. Stewart, "Multichannel equalization in subbands," in *Proc. IEEE Workshop on Applications of Signal Process. to Audio and Acoust.*, Oct. 1999, pp. 203–206.
- [195] N. D. Gaubitch and P. A. Naylor, "Equalization of multichannel acoustic systems in oversampled subbands," *IEEE Trans. Audio, Speech, Language Process.*, vol. 17, no. 6, pp. 1061–1070, Aug.

2009.

- [196] S. Weiss, R. W. Stewart, A. Stenger, and R. Rabenstein, "Performance limitations of subband adaptive filters," in *Proc. European Signal Process. Conf.*, Sep. 1998, pp. 1245–1248.
- [197] D. Schmid and G. Enzner, "A parametric least-squares approximation for multichannel equalization of room acoustics," in *Proc. Int. Workshop Acoust. Echo Noise Control*, 2010.
- [198] S. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang, "Multichannel blind deconvolution and equalization using the natural gradient," in *Proc. IEEE Signal Process. Workshop on Signal Process. Advances in Wireless Commun.*, Apr. 1997, pp. 101–104.
- [199] S. Amari and S. C. Douglas, "Why natural gradient?" in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 2, May 1998, pp. 1213–1216.
- [200] S. G. Norcross, M. Bouchard, and G. A. Soulodre, "Inverse filtering design using a minimal-phase target function from regularization," in *Audio Engineering Society Convention 121*, Oct. 2006.
[Online]. Available: <http://www.aes.org/e-lib/browse.cfm?elib=13763>
- [201] H. Luo and R. Liu, "Blind equalizers for multipath channels with best equalization delay," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 5, 1999, pp. 2511–2514.
- [202] A. Carini, S. Cecchi, and L. Romali, "Multipoint room response equalization with group delay compensation," in *Proc. Int. Workshop Acoust. Echo Noise Control*, 2010.
- [203] L. Fuster, M. Diego, M. Ferrer, A. Gonzales, and G. P. nero, "A biased multichannel adaptive algorithm for room equalization," in *Proc. European Signal Process. Conf.*, Aug. 2012.
- [204] W. Voiers, "Diagnostic acceptability measure for speech communication systems," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.* 2, 1977, pp. 204–207.
- [205] P. A. Naylor and N. D. Gaubitch, "Speech dereverberation," in *Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2005.
- [206] S. Wang, A. Skey, and A. Gersho, "An objective measure for predicting subjective quality of speech coders," *IEEE J. Select. Areas Commun.*, vol. 10, no. 5, pp. 819–829, Jun. 1992.
- [207] H. Wang and F. Itakura, "An implementation of multi-microphone dereverberation approach as a preprocessor to the word recognition system," *J. the Acoust. Soc. Japan*, vol. 13, no. 5, pp. 285–293, 1992.

- [208] *Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs.*, International Telecommunications Union (ITU-T) Std. Recommendation P.862, Feb. 2001.
- [209] Y. Huang, J. Benesty, and J. Chen, “A blind channel identification-based two-stage approach to separation and dereverberation of speech signals in a reverberant environment,” *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 882–895, Sep. 2005.
- [210] S. Treitel and E. Robinson, “The design of high resolution digital filters,” *IEEE Trans. Geosci. Electron.*, vol. VGE-4, pp. 25–38, Jun. 1966.
- [211] G. Harikumar and Y. Bresler, “FIR perfect signal reconstruction from multiple convolutions: minimum deconvolver orders,” *IEEE Trans. Signal Process.*, vol. 46, no. 1, pp. 215–218, Jan. 1998.
- [212] J. C. Claerbout and E. A. Robinson, “The error in least squares inverse filtering,” *Geophysics*, vol. 29, pp. 118–120, Feb. 1964.
- [213] T. E. Tuncer, “Deconvolution and preequalization with best delay LS inverse filters,” *J. Signal Process.*, vol. 84, pp. 2207–2219, Jun. 2004.
- [214] J. J. Shynk, “Frequency-domain and multirate adaptive filtering,” *IEEE Signal Process. Mag.*, vol. 9, no. 1, pp. 14–37, Jan. 1992.
- [215] J. B. Allen and D. A. Berkley, “Image method for efficiently simulating small room acoustics,” *J. Acoust. Soc. Am.*, vol. 65, no. 4, pp. 943–950, Apr. 1979.
- [216] J. Y. C. Wen, N. D. Gaubitch, E. A. P. Habets, T. Myatt, and P. A. Naylor, “Evaluation of speech dereverberation algorithms using the MARDY database,” in *Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2006, pp. 1–4.
- [217] J. D. Johnston, “Transform coding of audio signals using perceptual noise criteria,” *IEEE J. Select. Areas Commun.*, vol. 6, no. 2, pp. 314–323, Feb. 1988.
- [218] A. W. H. Khong and P. A. Naylor, “Efficient use of sparse adaptive filters,” in *Proc. IEEE Asilomar Conf. Signals, Sys., and Computers*, Oct. 2006, pp. 1375–1379.
- [219] W. Zhang, N. D. Gaubitch, and P. A. Naylor, “Computationally efficient equalization of room impulse responses robust to system estimation errors,” in *Proc. IEEE Int. Conf. Acoust., Speech,*

and *Signal Process.*, Mar. 2008, pp. 4025 – 4028.

- [220] F. Lim and P. A. Naylor, “Relaxed multichannel least squares with constrained initial taps for multichannel dereverberation,” in *Proc. Int. Workshop Acoust. Signal Enhancement*, Sep. 2012.
- [221] L. Liao and A. W. H. Khong, “An adaptive sub-systems based algorithm for channel equalization in a SIMO system,” *IEEE Trans. Circuits and Sys. I*, vol. 60, no. 6, pp. 1559–1569, May 2013.
- [222] B. R. Hunt, “A matrix theory proof of the discrete convolution theorem,” *IEEE Trans. Audio Electroacoust.*, vol. AU-19, no. 4, pp. 285–288, Dec. 1971.
- [223] O. Kirkeby, P. A. Nelson, H. Hamada, and F. Orduna-Bustamante, “Fast deconvolution of multichannel systems using regularization,” *IEEE Trans. Speech, Audio Process.*, vol. 6, no. 2, pp. 189–194, Mar. 1998.
- [224] D. R. Morgan, J. Benesty, and M. M. Sondhi, “On the evaluation of estimated impulse responses,” *IEEE Signal Process. Lett.*, vol. 5, pp. 174–176, Jul. 1998.
- [225] M. Stojanovic, J. Catipovic, and J. G. Proakis, “Adaptive multichannel combining and equalization for underwater acoustic communications,” *J. Acoust. Soc. Am.*, vol. 94, no. 3, pp. 1621–1631, Sep. 1993.
- [226] M. A. Haque, M. Bashar, P. A. Naylor, K. Hirose, and M. K. Hasan, “Energy constrained frequency-domain normalized LMS algorithm for blind channel identification,” *J. Signal, Image, Video Process.*, pp. 204–213, Apr. 2007.
- [227] J. S. Soo and K. K. Pang, “A multistep size (MSS) frequency domain adaptive filter,” *IEEE Trans. Signal Process.*, vol. 39, no. 1, pp. 115–121, Jan. 1991.
- [228] A. Torger and A. Farina, “Real-time partitioned convolution for ambiophonics surround sound,” in *Proc. IEEE Workshop Applications of Signal Process. to Audio, Acoust.*, Oct. 2001, pp. 195–198.
- [229] S. Shaffer and C. S. Williams, “Comparison of LMS, alpha LMS, and data reusing LMS algorithms,” in *Proc. 17th Asilomar Conf. Circuits, Sys., Comp.*, Nov. 1983, pp. 260–264.
- [230] M. A. Haque and M. K. Hasan, “Variable step-size multichannel frequency-domain LMS algorithm for blind identification of finite impulse response systems,” *IET Signal Process.*, vol. 1, no. 4, pp. 182–189, Dec. 2007.
- [231] P. O. Hoyer, “Non-negative matrix factorization with sparseness constraints,” *J. Machine Learning*

Res., vol. 5, pp. 1457–1469, Nov. 2004.

- [232] M. A. Haque and M. K. Hasan, “Noise robust multichannel frequency-domain LMS algorithms for blind channel identification,” *IEEE Signal Process. Lett.*, vol. 15, pp. 305–308, Feb. 2008.