NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

An Investigation of the Utility of Monaural Sound Source Separation via Nonnegative Matrix Factorization applied to Acoustic Echo and Reverberation Mitigation for Hands-Free Telephony

By

Niall M. Cahill

B.Eng., M.Eng.Sc.

A thesis submitted to the

National University of Ireland

for the degree of

Doctor of Philosophy

February 2012

Callan Institute

Department of Electronic Engineering

Faculty of Science and Engineering

Head of Department: Dr. Sean McLoone

Supervisor: Dr. Bob Lawlor

ABSTRACT

In this thesis we investigate the applicability and utility of Monaural Sound Source Separation (MSSS) via Nonnegative Matrix Factorization (NMF) for various problems related to audio for hands-free telephony. We first investigate MSSS via NMF as an alternative acoustic echo reduction approach to existing approaches such as Acoustic Echo Cancellation (AEC). To this end, we present the single-channel acoustic echo problem as an MSSS problem, in which the objective is to extract the users signal from a mixture also containing acoustic echo and noise. To perform separation, NMF is used to decompose the near-end microphone signal onto the union of two nonnegative bases in the magnitude Short Time Fourier Transform domain. One of these bases is for the spectral energy of the acoustic echo signal, and is formed from the incoming far-end user's speech, while the other basis is for the spectral energy of the near-end speaker, and is trained with speech data a priori. In comparison to AEC, the speaker extraction approach obviates Double-Talk Detection (DTD), and is demonstrated to attain its maximal echo mitigation performance immediately upon initiation and to maintain that performance during and after room changes for similar computational requirements. Speaker extraction is also shown to introduce distortion of the near-end speech signal during double-talk, which is quantified by means of a speech distortion measure and compared to that of AEC.

Subsequently, we address Double-Talk Detection (DTD) for block-based AEC algorithms. We propose a novel block-based DTD algorithm that uses the available signals and the estimate of the echo signal that is produced by NMF-based speaker extraction to compute a suitably normalized correlation-based decision variable, which is compared to a fixed threshold to decide on doubletalk. Using a standard evaluation technique, the proposed algorithm is shown to have comparable detection performance to an existing conventional block-based DTD algorithm. It is also demonstrated to inherit the room change insensitivity of speaker extraction, with the proposed DTD algorithm generating minimal false doubletalk indications upon initiation and in response to room changes in comparison to the existing conventional DTD. We also show that this property allows its paired AEC to converge at a rate close to the optimum.

Another focus of this thesis is the problem of inverting a single measurement of a non-minimum phase Room Impulse Response (RIR). We describe the process by which perceptually detrimental all-pass phase distortion arises in reverberant speech filtered by the inverse of the minimum phase component of the RIR; in short, such distortion arises from inverting the magnitude response of the high-Q maximum phase zeros of the RIR. We then propose two novel partial inversion schemes that precisely mitigate this distortion. One of these schemes employs NMF-based MSSS to separate the all-pass phase distortion from the target speech in the magnitude STFT domain, while the other approach modifies the inverse minimum phase filter such that the magnitude response of the maximum phase zeros of the RIR is not fully compensated. Subjective listening tests reveal that the proposed schemes generally produce better quality output speech than a comparable inversion technique.

DECLARATION

I hereby declare that this thesis is my own work and has not been submitted in any form for another award at any other university or other institution of tertiary education. Information derived from the unpublished or published works of others has been acknowledged in the text and a list of references is given.

Signed:_____ Date:_____

ACKNOWLEDGEMENTS

I would like to thank Bob Lawlor for giving me the opportunity to pursue a Ph.D., and for his advice, encouragement, and support throughout its duration. My appreciation also goes out to Enterprise Ireland, and by extension the exchequer, for providing the initial funding for this project.

I would like to thank Ken Humphreys for his advice, support and encouragement throughout the Ph.D. process, in particular for his extensive help and advice during its later years. Thanks are also due Giorgio Bacelli for his support during the later years.

I wish to thank all the academic, technical and administrative staff at the Department of Electronic Engineering, and at the Callan institute, here at NUIM for providing an excellent working environment. In addition, I wish to sincerely thank all my fellow post-graduate students, both past and present, for their company and support throughout the years.

I would like to thank my parents, Marion and Michael, for their unconditional support, and encouragement over the many years of study.

I would like to thank Nicola for her constant support and encouragement during the Ph.D process. In particular, for her patience and understanding during the long drawn out final stages of the work.

Finally, I wish to especially thank my aunt Amelia for her kind support during the Ph.D. It is in her memory that I dedicate this work.

TABLE OF CONTENTS

NOTATION

n	Sample Index,
$x(n)$	Far-end user(s) Signal,
$d(n)$	Echo Signal,
$v(n)$	Near-end Speakers Signal (Reverberated),
$e(n)$	Error Signal
$u(n)$	Near-end users Speech signal (non-Reverberated)
$y(n)$	Near-end Microphone Signal
$w(n)$	Noise Signal
$h(n)$	Loudspeaker-Enclosure-Microphone Impulse Response
$\mathbf{h}^{\mathrm{T}}$	$h(n)$ in vector format, where subscript T denotes transposed
$g(n)$	Near-end speaker's lips-Microphone Impulse Response
$\mathbf{g}^{\mathrm{T}}$	$g(n)$ in vector format
L_h, L_g,	Truncated lengths of $h(n)$ and $g(n)$
T_{60}	Room reverberation time
f	Discrete frequency bin Index for short-time processing
k, k_x	Frame indices for short time processing.
N	STFT analysis window size
m, m_x	STFT analysis window step sizes.
$X(f,k)$, $Y(f,k)$,	Short-Time Fourier Transforms (STFT) of $x(n)$ and $y(n)$
$V(f,k)$, $D(f,k)$	Short-Time Fourier Transforms (STFT) of $v(n)$ and $d(n)$
$\mathbf{B}_d(k)$	Echo Basis of size $N/2+1 \times R_d$
$\mathbf{B}_v$	Near-End Speaker Basis of size $N/2+1 \times R_v$
R_d, R_{v}	Number of Columns, or rank, of $\mathbf{B}_d(k)$ and $\mathbf{B}_v$
$\mathbf{B}$	Microphone Basis, i.e. $\mathbf{B} = [\mathbf{B}_d(k), \mathbf{B}_v]$
$\mathbf{g}(k)$, $\mathbf{g}_d(k)$, $\mathbf{g}_v(k)$	Gain vectors for $\mathbf{B}$, $\mathbf{B}_d(k)$, and $\mathbf{B}_v$, for frame k.
ϕ	Number of iterations of restricted NMF procedure
ψ	Number of iterations of unrestricted NMF procedure
C	NMF cost function
$\mathbf{y}(k)$, $\mathbf{d}(k)$, $\mathbf{v}(k)$	$N/2+1$ vector of unique values of $Y(f,k)$, $D(f,k)$, and $V(f,k)$,
$\mathbf{e}(k)$	Residual error vector
$\hat{\mathbf{v}}(k)$, $\hat{\mathbf{d}}(k)$	Estimate of $\mathbf{v}(k)$, $\mathbf{d}(k)$
$\hat{v}(n)$	Estimate of $v(n)$
$\xi(k)$	NMF-DTD decision variable for frame k
$\|\mathbf{v}(k)\|_2$	Euclidean, or 2-norm, norm of $\mathbf{v}(k)$.
$E_{\hat{\mathbf{d}}\mathbf{y}}(k)$	Smoothed Inner product of $\mathbf{y}(k)$ and $\hat{\mathbf{d}}(k)$
$E_{\mathbf{y}}(k)$, $E_{\hat{\mathbf{d}}}(k)$	Temporally smoothed energy of $\mathbf{y}(k)$, $\hat{\mathbf{d}}(k)$
λ	Forgetting Factor
$I(k)$	Binary DT indicator function

T	DT Threshold
$\delta(n)$	Unit sample sequence
$\overline{g}(n)$	Inverse of $g(n)$
κ	Inversion scaling factor
t_g	Inversion delay factor
$\overline{g}_{\mathrm{LS}}(n)$	Length $L_{\overline{g}_{\mathrm{LS}}}$ Least Squares estimate of $\overline{g}(n)$
$g_{\mathrm{mp}}(n)$	Minimum phase component of Room Impulse Response
$g_{\mathrm{ap}}(n)$	All-pass component of Room Impulse Response filter
$\overline{g}_{\mathrm{mp}}(n)$	Inverse of Minimum phase filter
$G(z), \overline{G}(z)$	Transfer functions of $g(n)$ and $\overline{g}(n)$
$G_{\mathrm{mp}}(z)$, $G_{\mathrm{ap}}(z)$	Transfer functions of $g_{\mathrm{mp}}(n)$ and $g_{\mathrm{ap}}(n)$
L_{ap}	Length of $g_{\mathrm{mp}}(n)$, $g_{\mathrm{ap}}(n)$ and $\overline{g}_{\mathrm{mp}}(n)$
l	Frequency bin index
$\tilde{g}_{\mathrm{mp}}(n)$, $\tilde{g}(n)$	Real cepstrum of $g_{\mathrm{mp}}(n)$ and $g(n)$
φ	Positive Integer
$G_{\mathrm{eq}}(l)$	Equalized Response
$\overline{g}_{\mathrm{ap}}(n)$	Inverse of $g_{\mathrm{ap}}(n)$
$\tau(l)$, $\tau_{\mathrm{mp}}(l)$,$\tau_{\mathrm{ap}}(l)$	Group delays functions of $g(n)$, $g_{\mathrm{mp}}(n)$ and $g_{\mathrm{ap}}(n)$
$\tau_{\mathrm{M}}(l)$	Modified group delay function
$v_{\mathrm{ap}}(n)$	Minimum phase inverted Speech Signal
$V_{\mathrm{ap}}(f,k)$	Short-Time Fourier Transforms (STFT) of $v_{\mathrm{ap}}(n)$
τ_{apmin}	Minimum group delay function peak
a_p, k_p	radius and frequency of p^{th} pole,
P	number of poles,
$\hat{a}_p$	Replacement pole radius
$G_s^{(p)}(z)$	Selective filter
γ	Pole radius offset
α_p	Selective Filter gain
$\hat{\overline{G}}_{\mathrm{mp}}(z)$	Modified minimum phase inverse transfer function
$g_s^{(P)}(n)$, $\hat{\overline{g}}_{mp}(n)$	Impulse responses of $G_s^{(p)}(z)$ and $\hat{\overline{G}}_{mp}(z)$
t_{d}	Direct path delay of $g(n)$
R_v, R_d, R_{interf}, R_{tar}	Rank of $\mathbf{B}_v$, $\mathbf{B}_d(k)$, $\mathbf{B}_{\mathrm{interf}}$, and $\mathbf{B}_{\mathrm{tar}}$
$\lvert U_{\mathrm{tar}}(f,k)\rvert$,$\lvert U_{\mathrm{interf}}(f,k)\rvert$	Target and Interference spectral components
$\lvert U_t(f,k)\rvert$	STFT of direct path delayed $u(n)$ signal i.e. $u(n - t_{\mathrm{d}})$,
$\mathbf{v}_{\mathrm{ap}}(k)$,$\mathbf{u}_{\mathrm{tar}}(k)$, $\mathbf{u}_{\mathrm{interf}}(k)$	$N/2+1$ vector for $\lvert V_{\mathrm{ap}}(f,k)\rvert$, $\lvert U_{\mathrm{tar}}(f,k)\rvert$, $\lvert U_{\mathrm{interf}}(f,k)\rvert$
$\hat{\mathbf{u}}_{\mathrm{tar}}(k)$	NMF-APDS estimate of $\mathbf{u}_{\mathrm{tar}}(k)$
$\hat{u}_{\mathrm{tar}}(n)$	NMF-APDS output speech.

ACRONYMS

RIR	Room Impulse Response
AE	Acoustic Echo
AEC	Acoustic Echo Cancellation
AES	Acoustic Echo Suppression
STFT	Short Time Fourier Transform
NMF	Nonnegative Matrix Factorization
DT	DoubleTalk
DTD	DoubleTalk Detection
NSE	Near-end Speaker Extraction
LS	Least Squares
RLS	Recursive Least Squares
LMS	Least Mean Squares
NLMS	Normalized Least Mean Squares
APA	Affine Projection Algorithm
FDAF	Frequency Domain Adaptive filtering
FIR	Finite Impulse Response
IIR	Infinite Impulse Response
MSSS	Monaural Sound Source Separation
ICA	Independent Component Analysis
BSS	Blind Source Separation
SBSS	Semi-Blind Source Separation
APDS	All-pass Phase Distortion Suppression

1 INTRODUCTION

Hands-free telephony allows telephone users to converse, per conventional hand-held telephony, while simultaneously occupying their hands with another task, a convenience that has fueled the growing popularity of this form of telephony. Moreover, of late a number of factors have conflated to increase the prevalence of hands-free telephony in society. One of these has been the prohibition in certain jurisdictions of conventional hand-held telephony while driving because of safety concerns, which has led to an increase in hands-free operation of mobile telephones while driving, and another has been the dual factor of the high and rising cost of travel and improving telecommunications infrastructure, which has encouraged the use of audio conferencing tabletop products (also known as speakerphones) for conducting business and for collaboration in general. These developments serve to motivate the topic of this thesis, audio for hands-free telephony.

Providing high quality, full-duplex, audio for hands-free telephony is a considerable technical challenge, consisting of solving a number of problems that, to varying degrees, have an adverse effect on the quality of a telephone conversation [1]. One such problem is reverberation [2], where the (near-end) speaker's speech signal is received at the telephone microphone with strong reflections of this signal off the enclosure boundaries, and another is background noise [3, 4], which becomes problematic due to the typically weaker and more reverberant speaker signal received at the microphone. The resulting signal transmitted to the far-end user(s) is therefore less intelligible and of lower quality than would typically be the case during conventional hand-held telephony. Perhaps the most significant and well-known problem associated with hands-free telephony however is acoustic echo [3-5]. Acoustic echo occurs when the received (far-end) signal is broadcast by the loudspeaker in the near-end enclosure to be picked up by the corresponding microphone due to direct and indirect (echoic) acoustic coupling of these two devices. With sufficient latency and gain, the far-end user(s)

perceives this feedback as echo, which can have a deleterious effect on a telephone conversation. Note that, following convention, we regard the near-end user(s) as operating a hands-free telephone in an (near-end) enclosure, and the far-end or opposing user(s), as the receiver(s) of echo.

With the increasing hardware resources available in telephones, matched by the increasing expectations of telephone users, addressing these problems has been and still is an active field of research in both academia and industry. In this thesis, we expand on this effort by proposing, and investigating the utility of, a novel acoustic echo mitigation approach in which the acoustic echo problem is viewed as a Monaural Sound Source Separation (MSSS) problem. This approach considers the near-end microphone signal as a mixture comprised of three source signals; the near-end speakers speech signal, the acoustic echo signal, and noise; with acoustic echo mitigation achieved by extracting the near-end speakers speech signal from the mixture, allowing this signal alone to be transmitted to the far-end user. To achieve extraction, we present an algorithm that customizes the prevalent model-based, spectral pattern recognition approach to MSSS for the acoustic echo problem. Specifically, ignoring noise for the present, and operating in the magnitude STFT domain, the echo and near-end speaker(s) signal are separated by using Nonnegative Matrix Factorization (NMF) [6] to match the spectral features of the echo signal to the spectral features of the far-end signal, and those of the near-end speaker signal to a general speaker independent model of spectral features trained a priori. The spectral data assigned to the speaker model by NMF is then used to synthesize an estimate of the near-end speakers speech signal.

This novel technique, which we name NMF Near-end Speaker Extraction (NMF-NSE), has a number of distinct advantages over the prevalent approach to echo mitigation: Acoustic Echo Cancellation (AEC) [5]. AEC uses the far-end signal and the near-end microphone signals as the input and reference signals respectively to an adaptive filter that is tasked with estimating and tracking the acoustic paths between the near-end loudspeaker and microphone; from which, an estimate of the echo signal is produced and is subtracted from the near-end microphone signal. When the adaptive filter has converged, AEC removes the echo to leave the near-end speakers signal (and noise), thereby cancelling the echo; however, the adaptive filter is generally unable to adapt immediately to sudden changes to the echo path(s), resulting in echo after such events, and is also generally incapably of adapting during Double-Talk (DT), during which time both acoustic echo and the near-end speakers signal arrive simultaneously at the near-end microphone, a scenario which can cause rapid divergence of the adaptive filter. NMF-NSE, in contrast, may be characterized as using the spectral features of the far-end signal and those of a speaker independent spectral model to track the spectral features of the echo and the near-end speaker signals respectively to estimate the near-end users speech signal. By this way the near-end speaker is considered a source which may be either inactive or active, thereby innately addressing DT, and by performing a complete separation of the near-end microphone signal in each frame, it will be demonstrated that NMF-NSE produces a constant level of echo mitigation that is uninterrupted by room change

and/or doubletalk. It will however also be demonstrated that NMF-NSE invariably admits some distortion of the near-end users speech signal during DT.

For a sequel, we leverage the functionality of NMF-NSE for Double-Talk Detection (DTD) for block-based AEC algorithms. As mentioned, DT can cause rapid divergence of an adaptive filter, thereby diminishing the performance of AEC. A common strategy to mitigate this divergence is to pause adaptation during DT as indicated by a DT detector. Conventional DT detectors may be described generically as computing a decision variable from the available signals and the signals generated by the AEC. Adaptation is suspended if the variable breaches a preset DT threshold, with echo cancellation continuing using the model of the echo path(s) learned up until that time. By computing the DT decision variable from the signals generated by the AEC however, conventional DTD is sensitive to room change, which manifests as false positive DT indications that stall adaptation precisely when it should be occurring. To address this problem, we propose a novel DTD algorithm that uses the estimate of the echo signal that is produced by NMF-NSE to compute a suitable normalized DT decision variable, an approach we call NMF-DTD. In contrast with existing conventional DTD, by computing the DT variable from the NMF-NSE echo estimate, NMF-DTD will be shown to be insensitive to echo path change, allowing largely uninterrupted adaptation of its paired acoustic echo canceller during and after echo path changes, enabling optimum convergence after such events, which in turn results in optimum acoustic echo mitigation performance.

For a finale, we address the near-end speaker reverberation problem. A common approach to dereverberation is inverse filtering or deconvolution, which consists of two main tasks, namely, measuring or estimating the Room Impulse Response (RIR) from the user's lips to the microphone, and inverting this RIR. We address the latter problem, specifically in the case of a single microphone estimate or measurement of an RIR. To this end, we first explore existing approaches to RIR inversion in the field of audio equalization, in which the solution criteria differ slightly to those of dereverberation. Then, we turn our focus to those techniques that seek to compensate principly for the magnitude response of an RIR, and which employ an all-pass/minimum phase decomposition of the RIR. In this context, we elucidate the effect of all-pass phase distortion on the processed speech, and describe how it arises from inverting the magnitude response of the maximum phase zeros. Based on this description, we propose two low-delay single channel inversion techniques that mitigate perceivable all-pass phase distortion, and which, by virtue of their low-delay, are tailored for the requirements of single microphone inverse filtering for dereverberation applications. For one of these techniques, we again employ MSSS via NMF to remove all-pass phase distortion artifacts directly from the reverberated speech in the magnitude STFT domain, while the other technique obtains a partial inversion of the RIR such that perceptually detrimental all-pass phase distortion is mitigated in the resulting processed speech, an approach that represents a departure from the NMF-based MSSS theme of this thesis. Through comparative listening tests against an existing comparable approach, we assess the subjective performance of these schemes, the results of which will be shown to be encouraging.

1.1 Background

This section provides background for the problems that are addressed in this thesis, namely, the acoustic echo and reverberation problems; noise is not explicitly addressed in this thesis. The standard formulation of the acoustic echo problem follows from that of the earlier related problem of network echo; moreover, in the main, the prevalent acoustic echo mitigation techniques where originally proposed for network echo mitigation. As such, for contextual reasons, we first describe the network echo problem, and aspects of human auditory perception related to echo, before describing and formulating the acoustic echo problem. Subsequently, the near-end speaker reverberation problem is described, including a general description of reverberation. Note that, in this thesis we address the single microphone case of each of these problems, and we address each of these problems separately.

1.1.1 The Network and Acoustic Echo Problems

Network echo, also known as Line or Hybrid echo, is a source of echo in telephony that first became problematic in the 1960s [3, 5]. Network echoes are electrical reflections of a users send signal that are returned to this user to be acoustically transduced by his/her telephone loudspeaker. These reflections are generated as the users send signal propagates to the opposing user's telephone through the Public Telephone Switched Network (PTSN), which is basically an interconnection of exchanges, that function to route information between two users. The most common origin of network echo is the telephone hybrid coil, one of which is housed at each users local exchange. The hybrid is used to convert a users send signal between the two-wire circuit of the users local loop and the four-wire circuit that is used by the PTSN to connect exchanges, and vice versa. A well-known problem with the hybrid is signal leakage from the conversion process due to an impedance mismatch, by which a percentage of the users send signal is leaked back to the sender to be received as a scaled and delayed copy of his/her send signal, viz. network echo.

From the perspective of the human auditory system, an echo that arrives at a listeners ear before approximately 80-100 ms after the arrival of the original signal (sometimes known as the integration time of the ear [7]) is fused with the original signal (known as the integration effect [8]) and as such, is not perceived as a distinct auditory event; a somewhat related effect, known as perceptual masking, is where the presence of a louder sound stimulus can render inaudible a weaker sound that is nearby in either frequency or time [9]. Instead, in the context of network echo, an echo arriving within the integration time of the ear is perceived as a sidetone, which is not objectionable to the user, and can in fact have a beneficial effect, adding a sense of presence to a conversation [5]. It follows that network echoes are required to have a certain delay and gain before the user perceives them as such, upon which they can severely disturb conversation. Such echoes became more prevalent in the 1960s when long distance communications involving the use of geostationary satellite link ups, which entail a significant delay, became more widespread [3, 5]. Of late however, the advent of fiber optical cables has obviated satellite delay [5].

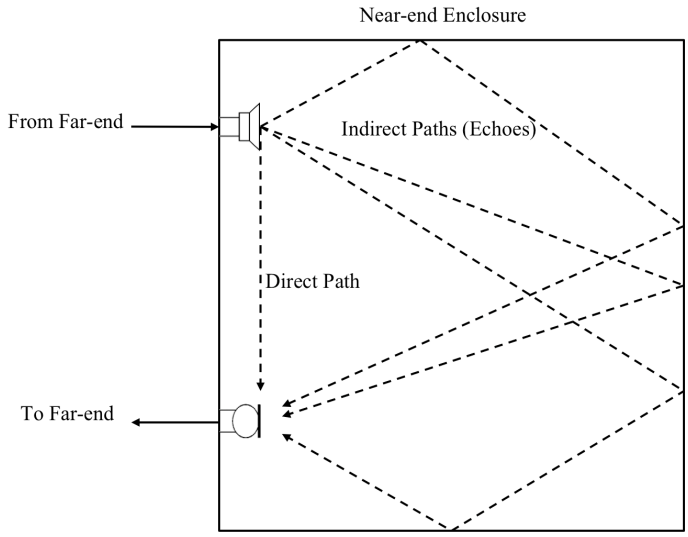

Figure 1.1: Schematic illustration of Acoustic Echo. The acoustic paths typically include a direct path between loudspeaker and microphone, and numerous indirect paths that arise due to reflections off the boundaries of the enclosure, and which result in numerous scaled and delayed copies of the far-end signal being returned to the far-end.

With the increasing popularity of hands-free operation of telephones, another source of echo has become problematic in telephony, namely, acoustic echo. For hand-held telephony, the loudspeaker of the handset is coupled to the near-end users ear such that the far-end users signal, radiated by the loudspeaker, is typically not incident on the telephone microphone. For hands-free telephony however, the user(s) is located some distance away from the loudspeaker in the near-end enclosure, which requires that the loudspeaker radiate the far-end signal at a level such that this signal is audible to this arbitrarily positioned near-end user(s). In this scenario, any acoustical paths that may exist between the near-end loudspeaker and near-end microphone, as illustrated in Figure 1.1, will result in the far-end signal propagating to the near-end microphone, to be electrically transduced and transmitted to the far-end user, giving rise to acoustic echo. Similar to network echo, acoustic echo is disturbing if it arrives at the far-end with sufficient delay (and gain), and if so, similarly impedes fluid conversation. Owing to the near-end room acoustics, described by Figure 1.1 and to be discussed further in section 1.1.2, the disturbing acoustic echo is usually perceived as a reverberated version of the far-end users speech signal rather than a simple echo as it is most commonly for network echo.

The process of generating either network or acoustic echo can be cast in a linear framework, where the far-end users signal, $x(n)$, where n is a sample index, excites a linear system with an impulse response, $h(n)$, comprised of a superposition of impulses each corresponding to an echo; in the network echo case these impulses arise electrically in the PTSN, in the acoustic echo case they arise due to acoustical echoes (and a direct acoustical path component) between the loudspeaker and microphone in the near-end enclosure. Both of these scenarios produce an echo signal, $d(n)$, which is received at the far-end. Mathematically $d(n)$ is generated by the linear convolution of $x(n)$, with the impulse response of the echo path(s), $h(n)$, which is expressed as,

$$d(n) = \sum_{i=0}^{L_h-1} h(i)x(n-i), \tag{1.1}$$

where i denotes sample index, and L_h is the truncated length of the echo path(s) impulse response (necessary for a Finite Impulse Response (FIR) digital realization), with L_h being chosen such that h has sufficiently decayed to zero. The discrete linear convolution in (1.1) can also be expressed in vector format as $d(n) = \mathbf{h}^T\mathbf{x}(n)$, where $\mathbf{x}(n) = [x(n), x(n-1)\ldots x(n-L_h+1)]^T$, $n \geq L_h$, and $\mathbf{h} = [h(0), h(1)\ldots, h(L_h-1)]^T$, and the superscript T represents the matrix transpose operation. This model ignores non-linear effects that may be introduced by certain components of the hands-free communication chain; we will visit this issue briefly in chapter 2.

Apart from the echo signal $d(n)$, the far-end user will also receive the desired near-end users speech signal, $v(n)$, typically reverberated (to be discussed in section 1.1.2), and a noise source, $w(n)$, which represents all extraneous noise sources, including any background acoustical noise generated in the near-end enclosure. The far-end users receive signal, $y(n)$, or the near-end microphone signal (if network echo is ignored), can be expressed as,

$$y(n) = d(n) + v(n) + w(n). \tag{1.2}$$

The ultimate aim of single channel echo mitigation, both network and acoustic, is to present the near-end users speech signal, $v(n)$, to the far-end user without echo i.e. $d(n) = 0$, with or without $w(n)$; noise reduction is generally considered a separate problem, which is the purview of speech enhancement, a review of which is available in [10].

While both the network echo and acoustic echo share the same model, the network echo impulse response and the acoustic impulse response, also known as a Room Impulse Response (RIR), differ considerably in their complexity. After emanating from the loudspeaker, the far-end signal, $x(n)$, excites the near-end enclosure for a relatively long period of time owing to the relatively slow speed of sound. This results in numerous impulses of decaying amplitude over a relatively long period of time, giving RIRs their characteristic exponentially decaying profile; an example RIR is displayed in chapter 6, Figure 6.2. Consequently, the acoustic echo impulse response is typically longer and contains a higher density of impulses than the network echo impulse response, which is typically sparse, comprised of one or more well-spaced impulses. Moreover, while the network echo path is typically stationary during a conversation, though differs for different paths through the PTSN, the acoustic echo paths often change during a conversation due to the movement of people, objects, of the loudspeaker or microphone in the near-end enclosure during the conversation, which often leads to sudden and drastic changes in the RIR. Acoustic echo is also often accompanied by higher levels of background noise due to the typically weaker and more reverberant near-end speaker signal received at the microphone. These issues serve to make the acoustic echo problem a more difficult problem to address than the network echo problem.

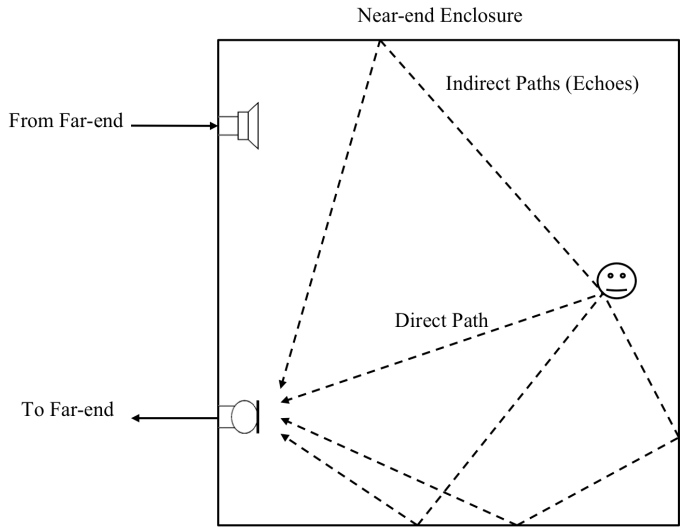

Figure 1.2: Schematic illustration of Reverberation.

1.1.2 The Near-end Speaker Reverberation Problem

Hands-free telephony also gives rise to another problem that is mostly irrelevant during conventional hand-held telephony, namely, reverberation of the near-end users speech. Similar to the composition of the acoustic echo signal $d(n)$, during both hands-free and hand-held telephony the near-end speakers speech signal, $v(n)$, is comprised of a direct path component, the speech signal of the near-end user after propagating the direct acoustic path between the users lips and the microphone, and numerous indirect acoustic path components or echoes, arising from indirect acoustic paths between the users lips and the microphone due to reflections in the near-end enclosure. For hands-free telephony, the relatively large displacement between the near-end users lips and the telephone microphone, as illustrated in Figure 1.2, means that the direct acoustic path is longer than during hand-held telephony, and therefore, the corresponding signal is weaker; for this discussion the indirect paths may be considered to be the same in both cases. The relative strength of the direct path signal during hand-held usage is such that the indirect components or echoes typically negligibly reverberate the resulting near-end users speech signal, $v(n)$; whereas, during hands-free telephony the weaker direct path signal can be significantly reverberated by these echoes, so much so that the intelligibility of $v(n)$ at the far-end can be significantly impaired [2]; to be discussed further below.

To model this undesired reverberation, linearity is assumed (similar to the network and acoustic echo problems) such that the effect of a room is characterized by the Room Impulse Response (RIR) sequence from the users lips to the near-end microphone, which also incorporates the effect of direct path attenuation and delay. It follows that the process of creating the near-end users speech signal, $v(n)$, is represented by the discrete linear convolution of the speech signal emanating from the users lips, $u(n)$, and the RIR measured between the users lips and microphone, $g(n)$, (assumed stationary for simplicity for the present), given by,

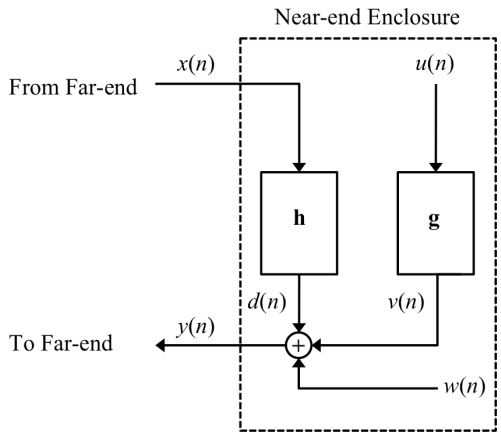

Figure 1.3: Block diagram of handsfree telephony.

$$v(n) = \sum_{i=0}^{L_g-1} g(i)u(n-i), \tag{1.3}$$

where L_g is the truncated length of $g(n)$. As for (1.1), (1.3) can be expressed in vector format as $v(n) = \mathbf{g}^{\mathrm{T}}\mathbf{u}(n)$, where $\mathbf{u}(n) = [u(n), u(n-1)\ldots u(n-L_g+1)]^{\mathrm{T}}$, $n \geq L_g$, $\mathbf{g} = [g(0), g(1)\ldots g(L_g-1)]^{\mathrm{T}}$. The complete hands-free problem, incorporating echo, (1.1), reverberation, (1.3), and noise can now be expressed as,

$$y(n) = d(n) + v(n) + w(n) = \sum_{i=0}^{L_h-1} h(i)x(n-i) + \sum_{j=0}^{L_g-1} g(j)u(n-j) + w(n), \tag{1.4}$$

where j denotes an index variable, or in vector format as,

$$y(n) = \mathbf{h}^{\mathrm{T}}\mathbf{x}(n) + \mathbf{g}^{\mathrm{T}}\mathbf{u}(n) + w(n). \tag{1.5}$$

This model is also displayed in block diagram form in Figure 1.3. It is apparent that the ideal objective for hands-free audio processing is for the far-end user to receive only $u(n)$.

As described for network echo, the reason reverberation is perceived as a single auditory event rather than as a succession of distinct echoes is because the delay between successive echoes is such that these echoes are integrated by the human auditory system, resulting in a single but prolonged auditory event. The effects of reverberation on the original speech signal, $u(n)$, are often judged with reference to the RIR measured between the speech source and the microphone. RIRs are often characterized as being comprised of early reflections, which are individually resolvable as impulses in the early part of the RIR, and late reflections that are irresolvable in the tail of the RIR owing to their density of arrival at the microphone. Early reflections impart coloration on the speech spectrum, and late echoes smear the spectral content of speech imparting it with a distinctive echoic sound. By smearing the frequency content of a speech signal, the late echoes cause the spectral content of successive speech phones or phonemes to overlap, resulting in what is known as overlap masking [11, 12], which severely reduces the intelligibility of the overlapped phones or phonemes, and thus, that of the overall speech signal. By this way, late echoes are considered to impart the greatest adverse effect on speech [8], while early echoes, in contrast, are often beneficial as they can reinforce the direct path signal thereby increasing the signal to noise ratio of the original speech [8].

Although no standard measure of reverberation has emerged [2], an oft-cited parameter of a room is its reverberation time, T_{60}, which is measured as the time required for the energy in a room to decay by 60 dB, with higher values for T_{60} indicating that the room produces more reverberated speech signals. Reverberation time varies depending on the physical characteristics of a room, with the degree of attenuation or delay of a particular echo depending on the characteristics of its acoustic path; its length relative to the direct path, and the frequency dependent absorption characteristics of its reflecting boundaries. For example, in a standard office sized room T_{60} can be expected to vary between 0.1-1 s [2], which implies thousands of taps for a FIR realization of an associated RIR at standard sampling rates. Another measure of reverberation is the ratio of the energy of the direct path signal to the accumulated energy of the echoes, or reverberant energy, known as the Direct to Reverberant energy Ratio (DRR) [2], which is usually computed from the RIR as is typical for reverberation measures. DRR, per the description of near-end speaker reverberation above, is typically high for conventional hand-held usage of telephones, while hands-free gives rise to lower values of this measure, implying more highly reverberated speech signals.

1.2 Outline of thesis

This thesis is composed of six subsequent chapters, the subjects and structure of which are outlined here.

In chapter 2, we review existing techniques for acoustic echo mitigation including relevant network echo mitigation techniques. This chapter, like each chapter of this thesis, begins with an introduction in which the subject matter of the chapter is broadly set out; in chapter 2, this involves discussing some early acoustic and network echo mitigation techniques and introducing AEC. AEC is then the primary focus of the remainder of chapter 2, in which we first describe the various classes of adaptive algorithms that have been deployed for AEC, and which are central to its performance, and then comprehensively review step-size control and DTD algorithms, which are also central to AEC performance. Throughout chapter 2, the issues of the Acoustic Echo problem, such as DT and echo path variations, i.e. room changes, and those of DTD, are elucidated to motivate the contributions in chapters 4 and 5. Chapter 2 ends with a discussion concerning existing acoustic echo mitigation approaches with a view to motivating an alternative perspective of this problem, one that is offered by chapter 4.

In chapter 3, we review existing approaches to model-based MSSS, and we describe NMF. The purpose of this chapter is to survey the model-based MSSS and NMF literatures, and to describe and motivate the techniques that will be subsequently employed in chapters 4, 5 and 6. After the chapter introduction, we describe NMF, which includes a description of the various cost functions and optimization techniques that have been deployed to compute NMF, with a particular focus on those that been employed for model-based MSSS. The remainder of the chapter then focuses on the model-based MSSS paradigm, in which information concerning the sources in a mixture is known, and where separation is performed using either

probabilistic inferential or matrix factorization techniques. At the end of the chapter, we motivate the application of NMF to the AE problem by demonstrating the utility of NMF for modeling the spectrogram of reverberant speech.

In chapter 4, the first contribution chapter of this thesis, we present Nonnegative Matrix Factorization Near-end Speaker Extraction (NMF-NSE), which addresses the acoustic echo problem based on some of the techniques outlined in chapter 3. NMF-NSE is described and formulated in detail and is then evaluated experimentally. The evaluation is comprised of two parts: a parameter study, which elucidates the influence of the various parameters of NMF-NSE; and a comparative study with a existing AEC-DTD approach, in which the performance of NMF-NSE is positioned with respect to AEC-DTD, and the room change robustness of NMF-NSE is demonstrated.

In chapter 5, we present a Double-Talk Detection (DTD) algorithm based on NMF-NSE, which is called NMF-DTD. The new detector is formulated and is then evaluated by comparing its classification performance to that of conventional DTD in a standard DTD detector evaluation framework, both stable near-end enclosures and enclosures that vary are employed. The benefits of NMF-DTD for conventional AEC are also demonstrated via the relative performance of its paired acoustic echo canceller.

In chapter 6, we present two new single channel RIR inversion techniques for dereverberation applications. This final contributory chapter is more self-contained than the two previous contributory chapters in that we begin with a review of existing approaches to single channel RIR inversion, mainly from the perspective of audio equalization, and then we elucidate open problems, particularly in relation to phase distortion and processing delay. Subsequently, we describe in detail how audible artifacts related to phase distortion arise during RIR inversion, and explore some of the properties of this distortion. We then propose two separate inversion approaches that mitigate phase distortion that are tailored for dereverberation. Continuing the central theme of this thesis, the first approach applies NMF in a supervised MSSS framework to the inverted speech signal directly; whereas, the second approach modifies the RIR prior to inversion to mitigate phase distortion.

In chapter 7, we discuss the contributions of the thesis and offer directions for future work. Chapter 8 contains the references of the thesis.

2 ACOUSTIC ECHO MITIGATION

In this chapter we describe and review acoustic echo mitigation, including relevant network echo mitigation algorithms. Section 2.1 establishes the required background for the review, which includes; briefly describing echo suppression; presenting the standard linear formulation underlying Network/Acoustic Echo Cancellation; and introducing the major themes associated with this approach, namely, adaptive filter type and step-size control, especially Doubletalk. In sections 2.2 and 2.3 the topics, adaptive filtering for AEC, and step-size control, are respectively addressed in greater detail. In these sections existing approaches are surveyed. After describing alternative acoustic echo mitigation techniques in section 2.4, the chapter ends in section 2.5 with a discussion concerning the conflicting requirements of AEC, that is, robustness to Doubletalk and fast convergence after room changes, which motivates the contributions in chapters 4 and 5.

2.1 Introduction and Background

The echo suppressor is an early device for network echo mitigation [3, 5, 13, 14]. Echo suppressors basically attenuate, or block completely, the near-end microphone signal, $y(n)$, if it is deemed to contain only echo, i.e. if $y(n) = d(n)$ (assuming $w(n) = 0$), thereby preventing echo from reaching the far-end user; an identical device is placed symmetrically for the near-end user. To implement such an approach, a detection algorithm is required to discriminate between three states; $d(n) > 0,\ v(n) = 0$; $d(n) = 0,\ v(n) > 0$; and $d(n) > 0,\ v(n) > 0$; with suppression only applied for the first state such that the near-end users speech $v(n)$ is not explicitly suppressed; consequently for the state $d(n) > 0,\ v(n) > 0$; $v(n)$ is received with echo. Apart from the obvious drawback of unattenuated echo during the state $d(n) > 0,\ v(n) > 0$ (double-talk), the state detector is liable to generate errors, due to noise for example, which may subsequently lead to suppression, i.e. clipping, of $v(n)$ [5]. A further drawback of echo

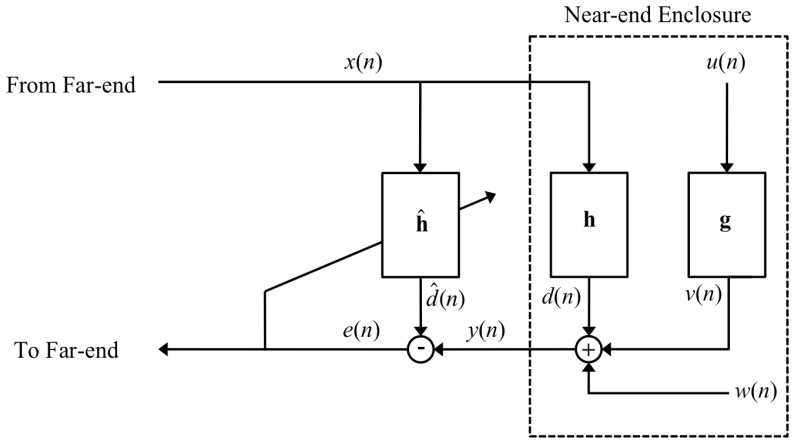

Figure 2.1: Block diagram of AEC.

suppressors is that the delay inherent in long distance telecommunications changes normal conversation patterns making detection error even more likely [5], and which can effectively result in half duplex discipline being enforced on users [5].

Another early approach to the network echo problem, which has since become the predominant approach to both the network and acoustic echo problems, and is, accordingly, the main topic of this chapter, is Echo Cancellation (EC) [13], or Acoustic Echo Cancellation (AEC) as it is referred to in the Acoustic Echo case [3-5]. Echo cancellers, in both the network and acoustic contexts, seek to estimate the echo signal, $d(n)$, so that the estimate may be subtracted from $y(n)$ in an attempt to cancel $d(n)$, thereby mitigating the echo disturbance for the far-end user. EC/AEC overcomes the problems of echo suppression by seeking to remove $d(n)$ only, thereby avoiding suppression of $v(n)$, and thus enabling better quality full-duplex telecommunication. With reference to Figure 2.1, echo cancellers are essentially an application of system identification, where the far-end speech signal $x(n)$ is considered the input/reference signal that is passed to both the system to be identified, denoted by $\mathbf{h}$, and to the estimate of this system, $\hat{\mathbf{h}}$, which generates an estimate of the echo signal, $\hat{d}(n)$, given by,

$$\hat{d}(n) = \sum_{i=0}^{L_h-1} \hat{h}(i)x(n-i), \tag{2.1}$$

or in vector format by,

$$\hat{d}(n) = \hat{\mathbf{h}}^{\mathrm{T}}\mathbf{x}(n), \tag{2.2}$$

where it is assumed that $\mathbf{h}$ and $\hat{\mathbf{h}} = [\hat{h}(0), \hat{h}(1)..., \hat{h}(L_h - 1)]^{\mathrm{T}}$ have identical FIR structures. The echo estimate is then subtracted from $y(n)$ to yield the error signal, $e(n)$,

$$e(n) = y(n) - \hat{d}(n) = d(n) - \hat{\mathbf{h}}^{\mathrm{T}}\mathbf{x}(n) + v(n) + w(n). \tag{2.3}$$

Assuming no post-processing, the output from the echo/acoustic canceller, $e(n)$, can be considered the signal received by the far-end user; $e(n)$ is also employed to guide the identification of $\mathbf{h}$, which will be discussed momentarily. If $\mathbf{h}$ has been identified or is known, i.e. $\hat{\mathbf{h}} = \mathbf{h}$, then $[d(n) - \mathbf{h}^{\mathrm{T}}\mathbf{x}(n)] = 0$ i.e. cancellation of the echo is achieved, and therefore, $e(n)$ contains only the desired near-end users speech signal and the noise signal i.e. $e(n) = v(n) + w(n)$. However, if $\mathbf{h}$ is only partly identified, $\hat{\mathbf{h}} \neq \mathbf{h}$, then the residual $[d(n) - \mathbf{h}^{\mathrm{T}}\mathbf{x}(n)]$

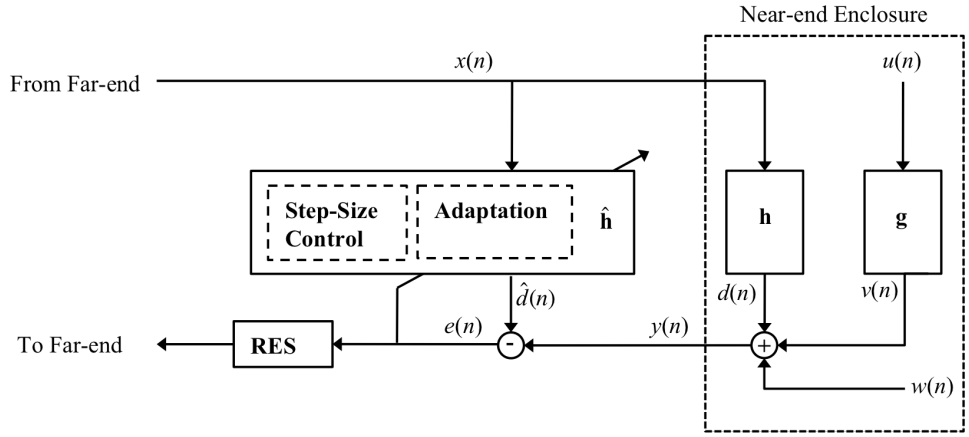

Figure 2.2: Block diagram of a generic AEC based Acoustic Echo Mitigation system.

represents echo that is sent to the far-end user along with the $v(n)$ and $w(n)$ signals, i.e. $e(n) = [d(n) - \mathbf{h}^T\mathbf{x}(n)] + v(n) + w(n)$. It is apparent therefore that the level of echo received by the far-end user, or equivalently, the efficacy of an (acoustic) echo canceller, depends on how close $\hat{\mathbf{h}}$ is to $\mathbf{h}$.

As described in chapter 1, $\mathbf{h}$ may vary significantly over the course of a conversation due to changes in the near-end enclosure, or between calls in the network echo context. Therefore, to identify $\mathbf{h}$ (acoustic) echo cancellers conventionally employ an adaptive algorithm to update the coefficients of $\hat{\mathbf{h}}$. Adaptive algorithms work by periodically minimizing some function of the error signal $e(n)$ that represents the closeness of $\hat{\mathbf{h}}$ and $\mathbf{h}$; collectively, the adaptive algorithm and the associated time-varying filter $\hat{\mathbf{h}}$ are often referred to as an adaptive filter. The field of adaptive system identification is well established, as is its application to the acoustic/network echo problem, and the associated literature contains a wealth of algorithms for obtaining the optimal $\hat{\mathbf{h}}$ in an adaptive fashion for AEC/EC. Of the many desirable attributes of an adaptive algorithm for this application, some of the most important are; fast convergence upon initialization, to minimize echo at the start of conversations; good tracking performance, to minimize the echo ensuing a change to $\mathbf{h}$; a low level of residual echo at steady state; robustness to noise and $v(n)$, i.e. doubletalk; a small hardware resource requirement, i.e. low computational and memory requirements; low delay; and numerical robustness. We devote section 2.2 to describing the main approaches to adaptive filtering for AEC, and to discussing their various properties with reference to the above attributes. Section 2.2 is an exposition of the major themes of adaptive filtering relating to AEC, rather than an exhaustive review of the wider topic of adaptive system identification, and its outline broadly follows that of previous reviews of adaptive filtering for AEC/EC that are available in [1, 3-5, 15].

The broad range of attributes desired of an adaptive filter for AEC, and various operational conditions of the network and acoustic echo problems, require that the adaptation rate or step size of the adaptive filter be controlled in some way in order to achieve satisfactory performance. This topic is the focus of Section 2.3. The most problematic such condition is Double-Talk (DT), that is, contemporaneous echo and near-end speaker activity

in the near-end enclosure, $v(n) \neq 0$ and $d(n) \neq 0$; as mentioned above, echo suppressors are ineffectual for this situation. DT occurs approximately 20 % of the time over the course of a normal conversation [16], a common example being when the near-end speaker politely interrupts the far-end speaker. During (DT), the near-end signal $v(n)$ acts as a strong interference source that can perturb adaptive algorithms and can consequently cause the coefficients of $\hat{\mathbf{h}}$ to diverge, often rapidly, from optimality. A common and effective strategy to prevent such divergence is to fix $\hat{\mathbf{h}}$ at the onset of DT, until DT ceases [3, 5]. Such a strategy necessitates a mechanism for DT Detection, for which it is desirable to correctly classify DT for a minimum of miss-classifications. DT detection is reviewed as part of the more general topic of step-size control in Section 2.3.

The many attributes required of AEC algorithms has also prompted alternative methods of system identification, which will be discussed in Section 2.4. Section 2.4 will also describe Residual Echo Suppressors (RES), which are tasked with suppressing the echo that is not canceled by the AEC. The relationships between the various components of a generic acoustic echo mitigation system are illustrated in Figure 2.2, in which AEC is considered to be an adaptive filter allied with a step-size control technique, the output of which is passed to the RES before being sent to the far-end user.

2.2 Adaptive System Identification for Acoustic Echo Cancellation

In this section we describe the main families of adaptive algorithms that have been deployed for EC/AEC, and we describe the various customizations to these algorithms that have been engineered for the EC/AEC application. Unless stated otherwise, it is assumed in this section that the near-end speaker is silent, $v(n) = 0$, i.e. no DT, such that $y(n) = d(n) + w(n)$.

2.2.1 Stochastic Gradient Descent

A commonly employed error criteria for adaptive filtering in the (acoustic) echo cancellation context is the Mean Squared Error (MSE), denoted by $J_{\mathrm{MSE}}(\hat{\mathbf{h}})$, and which is defined as,

$$J_{\mathrm{MSE}}(\hat{\mathbf{h}}) = E\{e(n)^2\} = E\{[y(n) - \hat{\mathbf{h}}^{\mathrm{T}}\mathbf{x}(n)]^2\}, \tag{2.4}$$

where $E\{\cdot\}$ denotes the expected value, and where we assume, for the present, that $\mathbf{h}$ is time-invariant. This cost function is minimized by setting the derivative of (2.4) with respect to the coefficients of $\hat{\mathbf{h}}$ to zero and then solving for $\hat{\mathbf{h}}$, which yields the following,

$$\mathbf{R}_{xx}\hat{\mathbf{h}} = \mathbf{r}_{yx}, \tag{2.5}$$

where the autocorrelation matrix of $x(n)$, $\mathbf{R}_{xx}$, of dimensions $L_h \times L_h$, is defined as,

$$\mathbf{R}_{xx} = E\{\mathbf{x}(n)\mathbf{x}(n)^{\mathrm{T}}\}, \tag{2.6}$$

and the length L_h cross correlation vector, $\mathbf{r}_{yx}$, is defined as,

$$\mathbf{r}_{yx} = E\{y(n)\mathbf{x}(n)\}. \tag{2.7}$$

Assuming that the inverse of $\mathbf{R}_{xx}$ exists,

$$\mathbf{h}^* = \mathbf{R}_{xx}^{-1}\mathbf{r}_{yx}. \tag{2.8}$$

The $L_h \times 1$ vector $\mathbf{h}^*$ is known as the Wiener optimal solution.

The Wiener optimal solution can also be obtained in a more adaptive fashion using the method of Gradient Descent (GD). The GD method begins by initializing the coefficients of $\hat{\mathbf{h}}$ with an arbitrary set of values; typically, $\hat{\mathbf{h}}(n_{\text{NG}}) = \mathbf{0}, n_{\text{NG}} = 0$, where n_{NG} is the GD iteration index. Then $\hat{\mathbf{h}}(n_{\text{NG}})$ is updated iteratively, in the direction of steepest descent at each iteration, until it converges to $\mathbf{h}^*$. The direction of steepest *ascent*, at a point $\hat{\mathbf{h}}(n_{\text{NG}})$, is the direction of the gradient of the cost function (2.4) at that point, which is defined as,

$$\nabla J(\hat{\mathbf{h}}(n_{\text{NG}})) = 2[\mathbf{R}_{xx}\hat{\mathbf{h}}(n_{\text{NG}}) - \mathbf{r}_{yx}] = 2E\{\mathbf{x}(n_{\text{NG}})e(n_{\text{NG}})\}. \tag{2.9}$$

At each n_{GD}, the coefficients of $\hat{\mathbf{h}}(n_{\text{NG}})$ are updated in the direction of steepest *descent* as,

$$\hat{\mathbf{h}}(n_{\text{GD}}+1) = \hat{\mathbf{h}}(n_{\text{GD}}) - \mu\nabla J(\hat{\mathbf{h}}(n_{\text{GD}})), \tag{2.10}$$

where μ is the step size parameter, and controls how far in the direction of the negative of the gradient the coefficients of $\hat{\mathbf{h}}$ are updated at each n_{GD}. By this way, the stepsize parameter μ controls if, and at what the rate, $\hat{\mathbf{h}}(n_{\text{NG}})$ converges to $\mathbf{h}^*$. Note that for notational convenience μ, $\hat{\mathbf{h}}$ and $e(n)$ are used to denote stepsize, adaptive filter impulse response, and error signal, respectively, for each of the adaptive algorithms to be described in this chapter; the particular adaptive algorithm referred to should be clear from the context.

The convergence of $\hat{\mathbf{h}}(n_{\text{NG}})$ to $\mathbf{h}^*$ by the GD method may be analyzed by expressing successive updates of (2.10) recursively, and by incorporating the initial estimate, $\hat{\mathbf{h}}(0)$, and the desired estimate $\mathbf{h}^*$ to give,

$$\hat{\mathbf{h}}(n_{\text{GD}}) = \mathbf{h}^* + [\mathbf{I} - 2\mu\mathbf{R}_{xx}]^{n_{\text{GD}}}[\hat{\mathbf{h}}(0) - \mathbf{h}^*]. \tag{2.11}$$

It is apparent from this expression that $\hat{\mathbf{h}}(n_{\text{GD}}) - \mathbf{h}^*$ tends to $\mathbf{0}$ as the term $[\mathrm{I} - 2\mu\mathbf{R}_{xx}]$ tends to $\mathbf{0}$. By diagonalizing $\mathbf{R}_{xx}$, this may be expressed as,

$$\lim_{n_{\text{GD}} \to \infty} [1 - 2\mu\vartheta_{xx,i}]^{n_{\text{GD}}} = 0, \quad \text{for } i = 1, \ldots, L_h, \tag{2.12}$$

where $\vartheta_{xx,i}$ is the i^{th} eigenvalue of $\mathbf{R}_{xx}$. GD is therefore guaranteed to converge $\hat{\mathbf{h}}(n_{\text{NG}})$ to $\mathbf{h}^*$ if,

$$0 < \mu < 1/\vartheta_{\max}, \tag{2.13}$$

where $\vartheta_{\max}$ is the largest eigenvalue of $\mathbf{R}_{xx}$. It follows from (2.12) that the rate at which $\hat{\mathbf{h}}(n_{\text{NG}})$ converges to $\mathbf{h}^*$ by GD is directly proportional to the spread of the eigenvalues of $\mathbf{R}_{xx}$, which is quantified by the condition number of $\mathbf{R}_{xx}$, given by the ratio of $\vartheta_{\max}$ to $\vartheta_{\min}$. For an ill-conditioned $\mathbf{R}_{xx}$, i.e. $\vartheta_{\max}/\vartheta_{\min} >> 1$, μ must be set very small to ensure that all the eigenmodes in (2.12) converge, which results in modes with large eigenvalues converging slowly. The convergence rate of the GD algorithm therefore is fastest when $\vartheta_{\max}/\vartheta_{\min} = 1$, which implies $\mathbf{R}_{xx} = \mathbf{I}_{Lh \times Lh}$, with white noise being an example of a signal with such autocorrelation characteristics. Unfortunately, speech signals exhibit high autocorrelation, and therefore, in contrast to white noise, the $\mathbf{R}_{xx}$ of speech signals typically differ considerably from the identity matrix [15], and thus, speech signals are relatively poor excitation signals for

GD. This analysis serves to demonstrate that the convergence of GD algorithms, and stochastic GD algorithms (to follow), is relatively slow for speech excitation [3, 5, 15].

To apply GD in practice, the expectation operation in (2.9) must be approximated; moreover, considering now the inherent time-variation of $\mathbf{h}$, $\hat{\mathbf{h}}$ must be updated periodically; for example, a batch estimate of the gradient in (2.9) for each n can be attained by computing $E\{\mathbf{x}(n)e(n)\}$ as the arithmetic mean over a number of previous samples. Of particular interest is the minimal estimate of the gradient, that is, using only the current input vector $\mathbf{x}(n)$ and current error signal sample, $e(n)$, to compute the gradient for each sample index n, which leads to the following update rule,

$$\hat{\mathbf{h}}(n+1) = \hat{\mathbf{h}}(n) + 2\mu\mathbf{x}(n)e(n). \tag{2.14}$$

This update rule modifies $\hat{\mathbf{h}}(n)$, dependency on n added, at each sample index using the instantaneous estimate of the gradient of $J_{\mathrm{MSE}}(\hat{\mathbf{h}}(n))$, which is known as the stochastic approximation of the gradient, and (2.14) is the update rule for the well-known Least Mean Squares (LMS) adaptive algorithm, which is an example of a stochastic gradient descent algorithm [17].

The LMS updates converge $\hat{\mathbf{h}}(n)$ to $\mathbf{h}^*$ in the mean, that is, the expectation of the squared difference of $\hat{\mathbf{h}}(n)$ and $\mathbf{h}^*$ converges to zero as the number of iterations tends to infinity [17]. Assuming a stable configuration of LMS, upon achieving a steady state value for $J_{\mathrm{MSE}}(\hat{\mathbf{h}}(n))$, $\hat{\mathbf{h}}(n)$ wanders randomly in a region around $\mathbf{h}^*$ known as the minimal capture zone, in which the expected mean squared difference between $\hat{\mathbf{h}}(n)$ and $\mathbf{h}^*$ is known as the misadjustment [17], i.e. $E\{[\hat{\mathbf{h}}(n) - \mathbf{h}^*]^2\}$. As $\mathbf{R}_{xx}$ is unlikely to be known when using LMS, the stepwise bounds to ensure convergence are usually given as [3, 5],

$$0 < \mu < \frac{1}{L_h\sigma_x^2}, \tag{2.15}$$

where σ_x^2 is the variance of $x(n)$. The denominator term in (2.15) is equivalent to the sum of the eigenvalues of $\mathbf{R}_{xx}$, which is always greater than $\vartheta_{\max}$, since $\mathbf{R}_{xx}$ is a positive definite matrix and thus all its eigenvalues are positive. The stepsize bound in (2.15) therefore, is a conservative estimate of the range of stable step sizes that can be employed for LMS. Within this range, the misadjustment of $\hat{\mathbf{h}}(n)$ is directly proportional to μ, with large values of μ resulting in larger random deviations about the optimum, and therefore, a larger minimal capture zone and larger misadjustment; small values of μ have the opposite effect.

The normalized LMS (NLMS) algorithm is a variant of the LMS algorithm that normalizes the instantaneous gradient estimate to remove the dependency on the scaling of $\mathbf{x}(n)$, and its update rule is given by,

$$\hat{\mathbf{h}}(n+1) = \hat{\mathbf{h}}(n) + 2\mu\frac{e(n)\mathbf{x}(n)}{\mathbf{x}(n)\mathbf{x}(n)^{\mathrm{T}} + \delta}, \tag{2.16}$$

where δ is typically a small positive regularization parameter. Applying the standard convergence analysis to this algorithm results in the stability bound, $0 < \mu < 2$. From this bound it can be seen that, unlike LMS, the convergence rate of NLMS is independent of the

gain of the input signal; however, the convergence rate of NLMS is still dependent on the excitation properties of $x(n)$. Within this step-size bound, the choice of μ is a trade-off between misadjustment and convergence rate, with μ being directly proportional to both. NLMS is a popular adaptive algorithm for AEC. This stems from its simplicity, robustness with finite precision arithmetic, and low cost, with its computational load being on the order of the length of $\hat{\mathbf{h}}(n)$, i.e. L_h; or in big O notation, $O(L_h)$ computations per sample. However, because the convergence rate of NLMS is relatively slow in the EC/AEC application, due to the speech excitation signal, it produces rather modest echo cancellation upon initialization and after enclosure changes [3, 5, 15].

The many desirable attributes of NLMS have meant that numerous auxiliary schemes have been devised to improve its convergence rate and tracking performance for the EC/AEC application [18, 19]. One effective and relatively straightforward approach, is to artificially enhance the excitation properties of the far-end speech signal, $x(n)$, before supplying it to $\hat{\mathbf{h}}(n)$, such that faster convergence and better tracking performance is attained [18-22]. To this end, error predictive filters, both adaptive and fixed, have been employed to decorrelate $x(n)$ before it is passed to the adaptive algorithm such that the ensuing optimization problem is better conditioned. The resulting estimate of the echo signal is then passed to the inverse of the decorrelation filter before being subtracted from $y(n)$ to produce $e(n)$ [20, 22], or alternatively, this inverse is required to be learned by the adaptive filter [21].

If $\mathbf{h}$ is sparse, by sparse we mean a small number of non-zero coefficients, then another way to improve the convergence rate of NLMS is to assign each coefficient of $\hat{\mathbf{h}}(n)$ a separate weighting that is proportional to the its magnitude. This is the idea behind Proportionate NLMS (PNLMS) [23], which has the following update rule,

$$\hat{\mathbf{h}}(n+1) = \hat{\mathbf{h}}(n) + 2\mu \frac{e(n)\mathbf{Q}(n)\mathbf{x}(n)}{\mathbf{x}(n)^{\mathrm{T}}\mathbf{Q}(n)\mathbf{x}(n) + \delta}, \tag{2.17}$$

where $\mathbf{Q}(n)$ is an $L_h \times L_h$ diagonal matrix that weighs the adjustment applied to each coefficient of $\hat{\mathbf{h}}(n)$ at each update. The L_h diagonal elements of $\mathbf{Q}(n)$, denoted by $q_\upsilon(n)$, $0 \le \upsilon \le L_h\text{-}1$, are updated as,

$$\varpi_\upsilon(n) = \max\left\{\rho, \max\left(\delta_p, \left|\hat{\mathbf{h}}_0(n)\right|, \ldots, \left|\hat{\mathbf{h}}_{L\text{-}1}(n)\right|\right), \left|\hat{\mathbf{h}}_\upsilon(n)\right|\right\}, \tag{2.18}$$

$$q_\upsilon(n) = \frac{\varpi_\upsilon(n)}{\sum_{i=0}^{L-1}\varpi_i(n)}, \tag{2.19}$$

where the parameter δ_p is a small positive value that ensures the diagonal elements of $\mathbf{Q}(n)$ are not initialized to zero, and ρ is another small positive value used to prevent stalling of the adaptation of coefficients that are have a magnitude much smaller than that of the largest coefficient. By assigning large weights to the dominant non-zero coefficients of a sparse $\hat{\mathbf{h}}(n)$, as expressed in (2.17)(2.18)(2.19), the convergence rate of these coefficients is increased [24].

PNLMS has originally proposed for the network echo problem, which has an inherently sparse impulse response, and in that context, was shown to converge faster than

NLMS [24]. However, PNLMS is less useful for AEC, because RIRs are generally dispersive; save for the start of the response, which is dominated by early reflections, which was specifically exploited in [25]. Indeed, the crude step size rule in (2.18)(2.19), which is tailored for sparse impulse responses, means PNLMS converges slower than NLMS for dispersive impulse responses [26]. This problem is ameliorated by PNLMS++ [27] which employs alternating NLMS and PNLMS updates. To extend the idea of proportionate stepwise more generally, the Improved PNLMS (IPNLMS) algorithm was proposed in [26]. IPNLMS allows the level of proportionality incorporated into $\mathbf{Q}(n)$ to vary between that of NLMS i.e. $\mathbf{Q}(n) = \mathbf{I}$, and PNLMS. In the acoustic echo context, IPNLMS can exploit the structure of the RIR, assuming an appropriate choice of proportionality, to attain a faster convergence rate over NLMS; this is similar to an earlier idea [28], where each coefficient of $\hat{\mathbf{h}}(n)$ was assigned a static weight, with the weight profile imitating that of typical RIRs. It has also been proposed [25, 29] to more specifically employ impulse response sparseness for adaptive filtering, i.e. sparseness controlled NLMS (SC-NLMS), where $\mathbf{Q}(n)$ is dependent on the sparseness of $\mathbf{h}$, as measured by the relationship between the L_1 norm and L_2 norm of $\hat{\mathbf{h}}(n)$. More information on adaptive filtering for identifying sparse impulse responses may be found at [30].

Another approach to optimize the performance of NLMS is to employ a time-varying step size, $\mu(n)$, such that the conflicting attributes of fast convergence after changes to $\mathbf{h}$ and a low misadjustment are attained. In general, Variable Step-Size (VSS) adaptive algorithms adjust μ so that it becomes larger whenever fast convergence is desired, and smaller whenever a small misadjustment is desired. As these traits are desirable for adaptive filtering in general, numerous applicable techniques for varying $\mu(n)$ for NLMS, and for stochastic gradient descent algorithms in general, have been proposed [31-35], some of which are compared in [36]. VSS adaptive algorithms that are more specific to the AEC/EC application, i.e. attempt to also contend with the DT condition, have also been proposed [37-39]. As this approach is a form of step-size control, these algorithms will be described in section 2.3.

The utility of the regularization parameter δ (in (2.16)) to improve NLMS performance has also been explored [17]. This parameter, which is often nominally described as preventing division by zero, can also be used to improve or control the performance of NLMS in a manner similar to the step-size parameter, i.e. for NLMS, $\mu = 0$ corresponds to $\delta = \infty$. This role for the regularization parameter is explored in [17], where it is compared to the step-size parameter. Regularization of NLMS and related algorithms, such as PNLMS, is examined in [40], in which an optimal time-varying expression for the regularization parameter was developed. This time-varying regularization parameter was demonstrated to allow for lower misadjustments and consistent convergence rates for varying levels of $w(n)$ compared to conventional NLMS. The topic of regularization will be revisited in section 2.2.4, where it will be discussed in relation to the more general problem of inverting ill-conditioned matrices for adaptive filtering.

2.2.2 Least Squares Adaptive filtering

Least Squares (LS) is another approach to adaptive filtering [41]. The error criterion of Least Squares for the time index n may be expressed as

$$J_{LS}(\hat{\mathbf{h}}(n)) = \sum_{i=0}^{M-1} e^2(n-i), \tag{2.20}$$

which can be interpreted as the arithmetic mean over the current and M-1 previous samples of $e(n)$, where it is assumed that $M > L_h$. By substituting for $e(n)$ in terms of $\hat{\mathbf{h}}(n)$ and $\mathbf{x}(n)$, this cost function can be expressed as,

$$J_{LS}(\hat{\mathbf{h}}(n)) = \hat{\mathbf{h}}(n)^{\mathrm{T}}\mathbf{X}(n)\mathbf{X}(n)^{\mathrm{T}}\hat{\mathbf{h}}(n) - 2\mathbf{r}_{yx}(n)^{\mathrm{T}}\hat{\mathbf{h}}(n) + \sum_{i=0}^{M-1} y^2(n-i), \tag{2.21}$$

where $\mathbf{X}(n) = [\mathbf{x}(n), \mathbf{x}(n\text{-}1), \ldots, \mathbf{x}(n\text{-}M\text{+}1)]$, and where the $L_h \times 1$ vector $\mathbf{r}_{yx}(n)$ is given by,

$$\mathbf{r}_{yx}(n) = \sum_{i=0}^{M-1} y(n-i)\mathbf{x}(n-i). \tag{2.22}$$

Setting the gradient of (2.21) to zero yields,

$$\mathbf{r}_{yx}(n) = \mathbf{X}(n)\mathbf{X}(n)^{\mathrm{T}}\hat{\mathbf{h}}(n). \tag{2.23}$$

The $\hat{\mathbf{h}}(n)$ that minimizes this is computed by,

$$\hat{\mathbf{h}}(n) = [\mathbf{X}(n)\mathbf{X}(n)^{\mathrm{T}}]^{-1}\mathbf{r}_{yx}(n). \tag{2.24}$$

If the $L_h \times L_h$ batch auto-correlation matrix $\mathbf{X}(n)\mathbf{X}(n)^{\mathrm{T}}$ is ill-conditioned, (which is often the case in AEC given the excitation signal, as discussed in section 2.2.1), the matrix may be regularized to improve its conditioning, which is further discussed in section 2.2.4. In terms of computations, inverting $\mathbf{X}(n)\mathbf{X}(n)^{\mathrm{T}}$ requires $O(L_h^3)$ computations per update, which if performed per sample is a prohibitively high computational load for AEC. Note that, assuming $x(n)$ and $w(n)$ are Gaussian random processes, the LS solution in (2.24) and the Wiener optimum solution in (2.8) are related, whereby as M tends to infinity, the LS solution approaches $\mathbf{h}^*$, asymptotically [3]. Moreover, under the same assumptions, the LS solution can be interpreted as that which maximizes the likelihood of the output vector, $[y(n), y(n\text{-}1), \ldots, y(n\text{-}M\text{+}1)]^{\mathrm{T}}$, given the input vectors in $\mathbf{X}(n)$.

Recursive Least Squares (RLS) is a computationally efficient variant of the LS algorithm just described. The RLS cost function is defined as,

$$J_{\mathrm{RLS}}(\hat{\mathbf{h}}(n)) = \sum_{i=0}^{\infty} \lambda^i e^2(n-i), \tag{2.25}$$

where λ is chosen in the range $0 < \lambda < 1$. The parameter λ, known as a forgetting factor, serves to weight present and past samples of the $e(n)$ such that earlier error samples have less influence on the current update; as such, λ is said to apply an exponential window to the data.

The RLS cost function can be expressed in terms of $\hat{\mathbf{h}}(n)$ and $\mathbf{x}(n)$ as,

$$J_{\mathrm{RLS}}(\hat{\mathbf{h}}(n)) = \hat{\mathbf{h}}(n)^{\mathrm{T}}\mathbf{R}_{xx}^{\lambda}(n)\hat{\mathbf{h}}(n) - 2\mathbf{r}_{yx}^{\lambda}(n)^{\mathrm{T}}\hat{\mathbf{h}}(n) + \sum_{i=0}^{\infty} \lambda^i y^2(n-i), \tag{2.26}$$

where the exponentially weighted auto-correlation $L_h \times L_h$ matrix, $\mathbf{R}_{xx}^{\lambda}(n)$, is given as

$$\mathbf{R}_{xx}^{\lambda}(n) = \sum_{i=0}^{\infty} \lambda^i \mathbf{x}(n-i)\mathbf{x}(n-i)^{\mathrm{T}}, \tag{2.27}$$

and the exponentially weighted cross-correlation vector, $L_h \times 1$, vector $\mathbf{r}_{yx}^{\lambda}(n)$ is given by,

$$\mathbf{r}_{yx}^{\lambda}(n) = \sum_{i=0}^{\infty} \lambda^i y(n-i)\mathbf{x}(n-i). \tag{2.28}$$

Setting the derivative of (2.26) to zero, the optimum $\hat{\mathbf{h}}(n)$ at sample n is the solution of,

$$\hat{\mathbf{h}}(n) = \mathbf{R}_{xx}^{\lambda}(n)^{-1}\mathbf{r}_{yx}^{\lambda}(n), \tag{2.29}$$

which, as for Least Squares, incurs a prohibitively high computational load. However, in this case, the terms $\mathbf{R}_{xx}^{\lambda}(n)^{-1}$ and $\mathbf{r}_{yx}^{\lambda}(n)$ can be estimated recursively at each sample as,

$$\mathbf{r}_{yx}^{\lambda}(n) = \lambda \mathbf{r}_{yx}^{\lambda}(n-1) + y(n)\mathbf{x}(n-i), \tag{2.30}$$

$$\begin{aligned}\mathbf{R}_{xx}^{\lambda}(n)^{-1} = {} & \lambda^{-1}\mathbf{R}_{xx}^{\lambda}(n-1)^{-1} \\ & - \lambda^{-2}\mathbf{R}_{xx}^{\lambda}(n-1)^{-1}\mathbf{x}(n)[1+\lambda^{-1}\mathbf{x}(n)^{\mathrm{T}}\mathbf{R}_{xx}^{\lambda}(n-1)^{-1}\mathbf{x}(n)]^{-1}\mathbf{x}(n)^{\mathrm{T}}\mathbf{R}_{xx}^{\lambda}(n-1)^{-1}\end{aligned} \tag{2.31}$$

which is a computationally efficient way of estimating these terms.

RLS may also be expressed in the form of an update rule as,

$$\hat{\mathbf{h}}(n+1) = \hat{\mathbf{h}}(n) + \mathbf{R}_{xx}^{\lambda}(n)^{-1}\mathbf{x}(n)e(n), \tag{2.32}$$

which is similar to the LMS update rule in (2.14) except that the term 2μ has been subsumed into $\mathbf{R}_{xx}^{\lambda}(n)^{-1}$. Indeed, assuming $x(n)$ is stationary, such that,

$$\mathbf{R}_{xx}^{\lambda}(n) = \frac{1}{1-\lambda}\mathbf{R}_{xx}, \tag{2.33}$$

the convergence of RLS may be contrasted with that of GD by substituting $\mathbf{R}_{xx}^{\lambda}(n)^{-1}$ for 2μ in (2.11), which gives the following expression for the convergence of RLS,

$$\hat{\mathbf{h}}(n) = \mathbf{h}^* + [\mathbf{I} - (1-\lambda)\mathbf{R}_{xx}^{-1}\mathbf{R}_{xx}]^n[\hat{\mathbf{h}}(0) - \mathbf{h}^*], \tag{2.34}$$

such that,

$$\hat{\mathbf{h}}(n) = \mathbf{h}^* + \lambda^n[\hat{\mathbf{h}}(0) - \mathbf{h}^*]. \tag{2.35}$$

From this analysis, it apparent that the convergence of RLS is independent of $\mathbf{R}_{xx}$ such that the convergence of the eigenmodes of (2.34) is independent of the conditioning of $\mathbf{R}_{xx}$, and therefore converges at a rate of λ^n, which is a considerable improvement over GD-based adaptive algorithms. These convergence characteristics suggest that RLS can be used effectively with autocorrelated excitation signals such as speech. In terms of hardware resources, the RLS algorithm requires $O(L_h^2)$ computations per iteration, making it more computationally efficient than LS but less efficient than NLMS algorithm. This computational load can be reduced to $O(L_h)$ per iteration by using Fast RLS (FRLS) algorithms [42].

A disadvantage of the RLS algorithm, particularly for EC/AEC applications, is that it is susceptible to numerical instability in finite precision arithmetic due to its inherent recursive nature; FRLS algorithms are even more susceptible to this instability [15, 42]. This implies that in practice λ must be chosen carefully, and so to must the initial values for $\mathbf{R}_{xx}^{\lambda}(n)^{-1}$; indeed, in practice these parameters are periodically reset for certain FRLS

implementations to ensure numeric stability [3, 5]. Another significant issue with RLS is tracking. After a change in **h**, the ensuing re-convergence period can be prolonged with RLS because the current updates continue to be influenced by past samples of $e(n)$, which occurred before the change, and which serve to slow convergence to the new optimum [15].

2.2.3 Affine Projection Algorithm

The NLMS algorithm computes the estimate of the gradient for each n using only the current input signal vector i.e. $\mathbf{x}(n)$. The Affine Projection Algorithm (APA) generalizes the NLMS algorithm by computing the current estimate of the gradient using M input vectors, where M can be less then L_h [43]. The update rule for the APA algorithm is [43],

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n\text{-}1) + \mu\mathbf{X}(n)[\mathbf{X}(n)^{\mathrm{T}}\mathbf{X}(n) + \delta\mathbf{I}]^{-1}\mathbf{e}(n), \tag{2.36}$$

where $\mathbf{X}(n)$ is defined as above except the constraint $M > L_h$ is lifted, $\mathbf{e}(n) = [e(n), e(n\text{-}1),\ldots e(n\text{-}M\text{+}1)]^{\mathrm{T}}$, and the matrix $\delta\mathbf{I}$, regularizes $\mathbf{X}(n)^{\mathrm{T}}\mathbf{X}(n)$. Note that for $M = 1$ the NLMS update rule is obtained, and the stability stepsize bound for APA in general is the same as that of NLMS, i.e. $0 < \mu < 2$ [15].

APA obtains its name from the interpretation of its update as a projection of $\hat{\mathbf{h}}(n)$ onto an affine subspace, the dimensionality of which depends on the order of the APA algorithm i.e. M [43]. The convergence rate achievable by APA can be appreciated by expressing (2.36) in the following form [3],

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n\text{-}1) + [\mathbf{X}(n)\mathbf{X}(n)^{\mathrm{T}} + \delta\mathbf{I}]^{-1}\mathbf{x}(n)e(n). \tag{2.37}$$

It can be seen that the APA update is similar to the RLS update in (2.32), except the inverse term of the APA update is a regularized, rank deficient version of the autocorrelation matrix of RLS, or more precisely the LS autocorrelation matrix. The convergence of APA therefore can be considered to range between that of NLMS for $M = 1$, and that of RLS as M approaches infinity, implemented using an exponentially weighted recursive update of the inverse. The computational load of APA varies accordingly, with $O(ML_h)$ computations required per update, with Fast APA (FAPA) [44] implementations reducing the computational load to approximately $O(2L_h + 14M)$ [15].

In terms of AEC/EC, the APA algorithm can be tailored for speech excitation by varying M, which results in faster converge over NLMS [15, 44]. Furthermore, the tracking performance of APA is better than that of RLS as fewer previous input vectors are used to compute the gradient at each update. For fast APA algorithms however, which are implemented similarly to FRLS algorithms, numerical issues arise for $M > 1$ [44]. Moreover, the tracking performance of APA algorithms degrades if too many previous samples are employed when computing the update. Like for NLMS, Proportionate APA (PAPA) was proposed in [45] for sparse system identification.

2.2.4 Regularization for AEC

As described above, the autocorrelation matrix of $x(n)$, $\mathbf{R}_{xx}$, and similar matrices such as $\mathbf{X}(n)\mathbf{X}(n)^{\mathrm{T}}$, are typically ill-conditioned, or in some cases even singular, owing to the highly auto-correlated nature of speech signals; in the case of singular matrices the pseudo-inverse may be employed. Moreover, for RLS, an ill-conditioned $\mathbf{R}_{xx}^{\lambda}(n)$ term implies a greater variance on the estimated impulse response [46]. Regularization is a technique for improving the conditioning of such matrices.

The most common form of regularization is known as Tikonov regularization [47] whereby a diagonal matrix, i.e. $\delta\mathbf{I}$, is added to the matrix before it is inverted; this was explicitly included in equation (2.37) for APA. In [46], a more general, non-identical diagonal matrix regularization term was presented specifically for adaptive filtering for AEC. For this approach, the available input signal $x(n)$ and a priori knowledge of the characteristics of the RIR were incorporated into selecting the diagonal entries of the regularization term such that it is optimized to improve the performance of the adaptive filter for AEC. The resulting regularization approach, applied to the RLS algorithm, was demonstrated to result in a faster convergence rate and a lower misadjustment than both Tikonov and Levenberg-Marquardt regularized RLS. Also in [46], by considering NLMS as an underdetermined recursive least squares algorithm, a link was established between NLMS, regularized by this term, and the proportionate style NLMS algorithms that where discussed in section 2.2.1.

In [48] regularization for AEC is achieved by assuming that $\mathbf{h}$ is both sparse and non-negative. Incorporating these prior assumptions into the LS cost function, for which sparsity was introduced by way of a L_1 constraint in the form of a penalty term on the non-negative $\hat{\mathbf{h}}(n)$; it is shown that the resulting optimization problem requires a nonnegative quadratic programming problem to be solved, which the authors solve using multiplicative updates that where proposed in [49]. This system identification approach was evaluated on RIRs generated using the mirror image method [50], which produces non-negative impulse responses, and speech data. In the absence of noise, it was shown to have a comparable convergence rate and misadjustment to NLMS. For decreasing SNR, the proposed approach was demonstrated to have a faster convergence rate and a lower misadjustment than NLMS.

2.2.5 Frequency Domain Adaptive filtering

Stochastic Gradient descent can be performed in a block fashion, whereby $\hat{\mathbf{h}}$ is updated less frequently, once per block instead of once per sample, but still converges to $\mathbf{h}^*$ [51]. This approach is known as Batch Stochastic Gradient Descent, and involves buffering N input vectors to form a block, computing an estimate of the gradient from this block, using this estimate to update the adaptive filter, to eventually produce an output vector of length N. Formally, the batch echo estimate is computed as,

$$\hat{\mathbf{d}}(m) = \hat{\mathbf{h}}(m)^{\mathrm{T}}\mathbf{X}(m), \tag{2.38}$$

where m is the block index, corresponding to blocks of length N, $\mathbf{X}(m) = [\mathbf{x}(Nm), \mathbf{x}(Nm\text{-}1), \ldots, \mathbf{x}(Nm\text{-}N\text{+}1)]$ and $\hat{\mathbf{d}}(m) = [\hat{d}(Nm), \hat{d}(Nm-1), .., \hat{d}(Nm-N+1)]^{\mathrm{T}}$; these terms correspond to the

same terms in sections 2.2.2 and 2.2.3 except for the substitution of the time index n with the block index m, where $n = Nm$. The block error vector, $\mathbf{e}(m)$, is computed as,

$$\mathbf{e}(m) = \mathbf{y}(m) - \mathbf{d}(m), \tag{2.39}$$

where $\mathbf{e}(m) = [e(Nm), e(Nm\text{-}1),\ldots e(Nm\text{-}N\text{+}1)]^{\mathrm{T}}$. The update rule for the Block exact version of LMS (BLMS) is defined as [51],

$$\hat{\mathbf{h}}(m+1) = \hat{\mathbf{h}}(m) + 2\mu\mathbf{X}(m)\mathbf{e}(m), \tag{2.40}$$

The main significance of batch-based adaptive filtering is that both $\hat{\mathbf{d}}(m)$ and the batch gradient are the result of a linear convolution, which can be computed efficiently using the Fast Fourier Transform (FFT). This serves as one of the main motivations for Frequency Domain Adaptive Filtering (FDAF) [51-53]. FDAF employs the FFT in conjunction with either the Overlap-Save or Over-Add method to perform the linear convolutions in (2.38) and (2.40) [51]. Another factor that motivates FDAF is that the FFT approximately orthogonalises a signal [51], such that the values produced at each frequency bin over time can be treated separately. FDAF algorithms exploit this by assigning each frequency bin of $x(n)$ a separate time varying step size, which can be inversely proportional to the average energy at that frequency bin, thereby compensating for spectral non-uniformity; spectral non-uniformity being the frequency–domain manifest of auto-correlated signals. FDAF algorithms therefore offer fast convergence for low computational load, making them attractive for AEC [51].

A drawback of FDAF however, and BLMS, is the buffering delay associated with batch processing. Early FDAF algorithms employed a block size that was the same length as the adaptive filter, i.e. $N = L_h$ [51, 54], which is the optimal setting from a computational efficiency standpoint. For applications such as AEC however, where $\mathbf{h}$ is typically very long, such block sizes incur a prohibitively long buffering delay. This can be remedied simply by reducing N; however, such block sizes are not computationally optimal [51]. To address this problem in a computationally optimal fashion, a FDAF scheme for block sizes $N < L_h$ was proposed in [55], in which the overlap-save paradigm was extended by partitioning both the input signal and the impulse response $\mathbf{h}$ into blocks; this will be described in detail shortly. This approach, which is known as the Multi-Delay adaptive Filter (MDF), performs FDAF efficiently using block sizes shorter than L_h; therefore, it retains the advantages of FDAF, while also enabling the block size, and therefore the delay, to be controlled independently of the length of the adaptive filter. These attributes mean that MDF is highly suited for AEC [56]. The Generalized Multi-Delay adaptive Filter with overlap (α) (GMDFα) generalizes the MDF filter by allowing for the blocks to overlap by more than 50% [57].

We now seek to formally describe FDAF generally. To this end, we first describe how linear convolution is performed in a block-based manner using the overlap-save method by considering the convolution in (2.38). Then, we describe how the GMDFα algorithm, which is equivalent to MDF and conventional FDAF for certain configurations of its parameters, performs adaptation. The following description of GMDFα is adapted from that in [57]. Underlined symbols denote frequency domain variables in the following description. Note

that in Chapters 4 and 5 of this thesis the GMDFα algorithm is employed as an experimental benchmark.

To compute the convolution in (2.38) in a block fashion in the frequency domain, the adaptive filter impulse response, $\hat{\mathbf{h}}$, dependency on m removed for the present, is partitioned into K contiguous blocks (non-overlapping) of length N.

$$\hat{\mathbf{h}}_k = [\hat{h}(kN), \hat{h}(kN+1), \ldots \hat{h}(kN+N-1)], \quad 0 \le k \le K-1 \tag{2.41}$$

where k is the block index and where it is assumed that $L_h/KN = 1$, i.e. K is a non-negative integer. Each block is then padded with N zeros, and the resultant vector is transformed into the frequency domain using the DFT to give the following frequency domain vector $\underline{\hat{\mathbf{h}}}_k$,

$$\underline{\hat{\mathbf{h}}}_k = \mathbf{W}\begin{bmatrix}\hat{\mathbf{h}}_k \\ \mathbf{0}_{N\times 1}\end{bmatrix}, \tag{2.42}$$

where $\mathbf{W}$ is the $2N \times 2N$ DFT matrix:

$$\mathbf{W} = \exp\left(-i\frac{2\pi\chi\nu}{2N}\right), 0 \le \chi, \nu \le 2N\text{-}1. \tag{2.43}$$

This matrix representation for the DFT enables later operations to be expressed neatly using matrix algebra; in practice, the FFT algorithm is used to perform the DFT. The N zeros appended to the impulse response block $\hat{\mathbf{h}}_k$ before computing the DFT are necessary to perform linear convolution using the overlap-save method. The resulting K frequency domain filter vectors are stacked vertically to yield the $K{\cdot}2N$ vector $\underline{\hat{\mathbf{h}}}$,

$$\underline{\hat{\mathbf{h}}} = \begin{bmatrix}\underline{\hat{\mathbf{h}}}_k \\ \vdots \\ \underline{\hat{\mathbf{h}}}_{k\text{-}1}\end{bmatrix}. \tag{2.44}$$

Turning now to the input signal, we define $\underline{\mathbf{X}}(m)$ as a $2N \times 2N$ diagonal matrix, whose entries are equal to the DFT coefficients of a input vector of size $2N$,

$$\underline{\mathbf{X}}(m) = \mathrm{diag}_{2N}\left[\mathbf{W}\begin{bmatrix}x(Qm-N) \\ x(Qm-N+1) \\ \vdots \\ x(Qm+N-1)\end{bmatrix}\right], \tag{2.45}$$

where $\mathrm{diag}_N[\cdot]$ is the diagonal matrix of size N operator. The nonnegative integer Q denotes the step-size of the input vector, and as such denotes the number of new input samples that are processed per block m. We also define a nonnegative integer α to denote the overlap factor, i.e. $Q = N/\alpha$. Like for $\underline{\hat{\mathbf{h}}}_k$, it is necessary to take the DFT over $2N$ samples of $x(n)$ instead of N so that linear convolution can be performed by the overlap–save method. Note that for $\alpha = 1$, MDF is attained i.e. the overlap between successive input blocks is N; and for $\alpha = 1$, $K = 1$ regular FDAF is attained i.e. $N = L_h$, and $\hat{\mathbf{h}}$ is not partitioned into blocks.

Having defined the relevant terms, the frequency domain estimate of the echo signal, $\underline{\hat{\mathbf{d}}}(m)$, is expressed as,

$$\underline{\hat{\mathbf{d}}}(m) = [\underline{\mathbf{X}}(m), \ldots \underline{\mathbf{X}}(m\text{-}\alpha(K\text{-}1))]\underline{\mathbf{h}}(m) = \underline{\tilde{\mathbf{X}}}(m)\underline{\mathbf{h}}(m). \tag{2.46}$$

Taking the IFFT of $\hat{\underline{\mathbf{d}}}(m)$ i.e. $\mathbf{W}^{-1}\hat{\underline{\mathbf{d}}}(m)$, yields a $2N \times 1$ time domain vector of which the upper or earlier N samples correspond to circular convolution in which time aliasing has occurred, and the lower or more recent N samples correspond to the filtered output samples for the current block m. For $\alpha = 1$, i.e. MDF, these lower N samples correspond to $\hat{\mathbf{d}}(m)$, completing the convolution in (2.38). For $\alpha > 1$ successive input blocks overlap by more that N samples; therefore, to form the output vector each output section, i.e. each batch of N linearly convolved samples, are joined to the preceding batch using Weighted Overlap and Add (WOLA), which is described in [57].

We now proceed to describe the GMDFα update rule. Defining the frequency domain microphone signal vector $\underline{\mathbf{y}}(m)$, of dimensions $2N \times 1$, as,

$$\underline{\mathbf{y}}(m) = \mathbf{W}\begin{bmatrix}\mathbf{0}_{N\times 1}\\ \mathbf{y}(m)\end{bmatrix}, \tag{2.47}$$

$\hat{\underline{\mathbf{h}}}$ is updated for each new block as,

$$\hat{\underline{\mathbf{h}}}(m+1) = \hat{\underline{\mathbf{h}}}(m) + \frac{\mu}{2N}\mathbf{C}\mathbf{T}(m)\tilde{\underline{\mathbf{X}}}^{\mathrm{H}}(m)[\underline{\mathbf{y}}(m)-\mathbf{S}\hat{\underline{\mathbf{d}}}(m))], \tag{2.48}$$

where the superscript H denotes Hermitian transpose, and the dependence of $\hat{\underline{\mathbf{h}}}$ on m has been restored. This expression may be construed as a generic FDAF update rule, and will be explained by analogy with the time domain BLMS update rule in (2.40).

The $(2N \times 2N)$ sectioning matrix $\mathbf{S}$, which is defined as,

$$\mathbf{S} = \mathbf{W}\begin{bmatrix}\mathbf{0}_{N\times N} & \mathbf{0}_{N\times N}\\ \mathbf{0}_{N\times N} & \mathbf{I}_{N}\end{bmatrix}\mathbf{W}^{-1}, \tag{2.49}$$

is used to select the useful part of the estimated desired signal, $\hat{\underline{\mathbf{d}}}(m)$. $\mathbf{S}$ does this by transforming $\hat{\underline{\mathbf{d}}}(m)$ into the time domain, zeroing the upper N time aliased samples, and transforming the resultant vector back to the frequency domain. The resultant frequency domain vector is then subtracted from $\underline{\mathbf{y}}(m)$ to give the GMDFα analogue of the error term of BLMS. The GMDFα error term is then multiplied by the input signal block matrix term, $\tilde{\underline{\mathbf{X}}}^{\mathrm{H}}(m)$, to compute the frequency domain estimate of the gradient of each block; analogous to the convolution in (2.38).

The diagonal block Toeplitz matrix, $\mathbf{T}(m)$, of dimensions $K{\cdot}2N \times K{\cdot}2N$, is known as the normalization matrix, and its purpose is to exploit the orthogonalising properties of the FFT for a faster convergence rate. This matrix applies a separate, block wise varying, weight to each frequency bin of the K frequency responses in $\tilde{\underline{\mathbf{X}}}^{\mathrm{H}}(m)$, such that spectral coloration, which is indicative of a correlated input signal, is mitigated. A relevant choice is to vary the weights according to the inverse of the signal power [57], which can be computed recursively,

$$\mathbf{T}(m) = \lambda\mathbf{T}(m) + (\lambda-1)[\tilde{\underline{\mathbf{X}}}(m)^{\mathrm{H}}\tilde{\underline{\mathbf{X}}}(m)]^{-1}, \tag{2.50}$$

where λ is a forgetting factor; many variations on this theme exist [58]. This implementation is often referred to as the self-orthogonalisation implementation.

Before updating $\hat{\underline{\mathbf{h}}}$ it is necessary to multiply the K gradient estimates by a constraint matrix $\mathbf{C}$, of dimension $K{\cdot}2N \times K{\cdot}2N$, which is defined as

$$\mathbf{C} = \mathrm{diag}_K\left[\mathbf{W}\begin{bmatrix}\mathbf{1}_N & \mathbf{0}_N \\ \mathbf{0}_N & \mathbf{0}_N\end{bmatrix}\mathbf{W}^{-1}, \cdots, \mathbf{W}\begin{bmatrix}\mathbf{1}_N & \mathbf{0}_N \\ \mathbf{0}_N & \mathbf{0}_N\end{bmatrix}\mathbf{W}^{-1}\right]. \tag{2.51}$$

Similar to **S**, the role of **C** is to apply data constraints to each block gradient estimate such that the N zeros that where appended to the block $\hat{\mathbf{h}}_k$ in (2.42) are reflected in the corresponding gradient estimate so that the structure in (2.42) is preserved for subsequent updates. As is evident, it does this by transforming each block gradient estimate into the time domain using the IFFT, zeroing the lower N samples of the time domain vectors, and transforming the resulting vectors back to the frequency domain using the FFT. After scaling by $\mu/2N$ the update is then ready to be applied to $\hat{\underline{\mathbf{h}}}(m)$. Note that for, $\mathbf{C} = \mathbf{I}$ the unconstrained version of GMDF is obtained, which is computationally more efficient than the constrained version described above, but the estimate of the desired response is no longer obtained by linear convolution [52]. GMDF without any data constraints, i.e. no zero-padding or overlap add, results in a relatively high misadjustment as the solution does not converge to the Weiner optimum [51]. This special case is related to system identification in the STFT domain, which may achieve a lower misadjustment by employing cross-band filters as will be discussed in section 2.2.7.

The stepsize bound for GMDF is given as [57],

$$0 < \mu < \frac{4}{1+K}. \tag{2.52}$$

It can be seen that the upper bound for μ decreases as the number of blocks K increases. By way of different assumptions, a less restrictive step-size bound was proposed in [59].

The main advantage of the GMDF algorithm over MDF is that it allows overlap greater than N between successive input blocks. This option allows the adaptive filter to be updated more often, which was demonstrated in [60], in the AEC context, to produce faster convergence and better tracking; albeit, for a somewhat higher misadjustment and N/Q extra computations per block [57]. This option has also engendered GMDFα for use in conjunction with noise reduction algorithms, which typically employ overlapping frames; for a review of combined AEC and noise reduction applications see [61, 62].

Given its desirable attributes, FDAF has received a significant amount of attention in the AEC literature. In [5], a generalized derivation of adaptive filtering in the frequency domain was presented, which encompasses the RLS cost function. Also in [5], a link was established between the MDF algorithm and a frequency domain version of APA. In [38], the MDF algorithm was extended, by which the correlations between successive blocks were exploited to further improve the convergence rate. The resulting algorithm was shown to have faster convergence than MDF, albeit for an increased computational load [38].

Data-reuse [63, 64] is another option for improving the convergence rate and tracking performance of adaptive filtering in general. Data-reuse implies that the adaptive filter is updated more than once per input sample (time-domain), or block of input data (frequency domain). For sample-by-sample based time-domain algorithms data-reuse adds significant computational complexity [65], precluding its use in this domain. It was similarly shown in

[65] that data-reuse is a rather computationally intensive for FDAF; although, it was reported in [63] [64] to improve the convergence rate of MDF, for moderate increases over baseline implementations.

2.2.6 Subband Adaptive filtering

Another established adaptive filtering approach that can be employed for AEC is subband adaptive filtering [66]. For this form of adaptive filtering, the input signal $x(n)$ and the near-end microphone signal $y(n)$ are each passed to an identical subband filter, which is comprised of an analysis filterbank that serves to decompose a signal into a number of subband signals. Each pair of resulting input and output subband signals is then supplied to a separate subband adaptive filter, which is tasked with adaptively identifying that subband's contribution to the overall impulse response. In general, the resulting subband error signals are combined using a synthesis filter bank to create the full band error signal, $e(n)$.

While the total number of subband and fullband adaptive filter coefficients is usually the same, subband adaptive filtering can be rendered computationally efficient by down-sampling each subband signal before performing adaptive filtering [51]. The resulting decimated signals contain fewer samples resulting in fewer computations compared to fullband adaptive filtering. Moreover, the subband signals are approximately decorrelated, similar to FDAF, such that the convergence rate can be optimized for each subband adaptive filter separately, with each subband signal typically having a more uniform distribution of spectral energy. As for FDAF, these two attributes have made subband adaptive filtering, which can be performed using an array of different adaptive algorithms, an attractive choice for AEC [5].

There are however some performance trade-offs associated with subband adaptive filtering, mostly related to the design of the analysis filterbank [66]. For example, the analysis filterbank typically impart the subband signals with a delay, the length of which is directly proportional to the order of its constituent bandpass filters. However, for optimum subband adaptive filtering, it is desirable to minimize the spectral overlap between the bandpass filters of the filterbank. Relatedly, it is desirable to have high stop band attenuation so that the decimation rate can be increased without introducing excessive spectral aliasing. Both of these attributes entail bandpass filters of higher orders and therefore a longer delay. It is apparent therefore that the performance of subband adaptive filtering is limited by the attributes of the filterbank. Further information on subband adaptive filtering can be found in [66].

2.2.7 STFT based adaptive filtering

System identification can also be performed in the STFT domain. As mentioned in section 2.2.5, this approach can be considered a special case of FDAF adaptive filter. However, it most often considered a special case of subband adaptive filtering, with the FFT/IFFT corresponding respectively to analysis/synthesis filterbanks, with each frequency bin coefficient with respect to time corresponding to a subband signal. The use of the FFT as a

ALGORITHM	DOMAIN	DELAY	COST FUNCTION	RECURSIVE	COMPLEXITY	CONVERGENCE PROPERTIES
LMS/NLMS	Time	Sample	MSE	No	$O(L_h)$	Dependent on Condition number of excitation signal
RLS	Time	Sample	LS	Yes	$O(L_h^2)$	Independent of Condition number of excitation signal
APA	Time	Sample	MSE	No, (FAPA Yes)	$O(ML_h)$	Adjustable, can be independent of Condition number of excitation signal
FDAF	Frequency	Block length	MSE	No	Less than NLMS. See section 4.2.2	Ill-conditioning mitigated by spectral normalisation.
Subband	Subband	Filterbank delay	MSE	Depends on adaptive filter algorithm, see [5].	Depends on adaptive filter algorithm, see [5].	Ill-conditioning mitigated in subbands
STFT	STFT	STFT frame length.	MSE	Depends on adaptive filter algorithm, see [67, 68].	Depends on adaptive filter algorithm, see [67, 68].	Potential for STFT bin-specific stepsizes.

Table 2.1 : Summary table listing some important properties of the different Adaptive filtering paradigms.

filter bank implies a large degree of spectral overlap between the subband signals [51]; therefore, to adequately identify a system in the STFT domain it is necessary to employ cross-band filters in each subband [69]. The performance of STFT system identification using cross-band filters adapted using the NLMS algorithm was analyzed for the task of AEC in [67, 68]. It was shown that in comparison to no cross-band filters, cross-band filters slow the convergence of NLMS but they enable a lower misadjustment to be achieved at steady state. It was also shown that in comparison with fullband NLMS, cross-band STFT-based NLMS has a lower computational load but has a higher misadjustment. Further information on STFT adaptive filtering using cross-band filters is available in [67].

2.3 AEC Control

This section reviews step-size control schemes for optimizing AEC performance, including those schemes that were originally proposed for EC and which are relevant to AEC. As described in the introduction, of most importance are those schemes that mitigate divergence of the adaptive filter during DT, for which DT detection is the most prevalent. Generically speaking, DT detectors compute the value of a decision variable, denoted as ξ unless otherwise stated, for each adaptive filter update (sample/frame) and compare this value to a prescribed fixed threshold, denoted as T. This results in a binary control variable, which multiplies the step-size to control adaptation. By this way, the step-size can be considered as varying between two values, 0 and its prescribed value. A selection of AEC control techniques that produce a more variable step size will also be described in this section.

2.3.1 The Geigel DTD Algorithm

The Geigel DTD algorithm [70] performs a waveform level comparison between $x(n)$ and $y(n)$, given as,

$$\xi(n) = \frac{\max\{|x(n)|, \ldots |x(n-L_h+1)|\}}{|y(n)|}. \quad (2.53)$$

The comparison is performed over L_h samples of $x(n)$, i.e. $\mathbf{x}(n)$, to compensate for the delay between $y(n)$ and $x(n)$. The detection variable $\xi(n)$ is then compared to a user prescribed threshold, with DT indicated for the current sample if $\xi(n) < T$. Because of noise, indications of DT are typically held for a hold time of between 20-30 ms.

In the presence of DT, this approach relies on a relative increase in the magnitude of $y(n)$ relative to the maximum magnitude of $x(n)$ over the last L_h samples. This works well for network echo cancellation, where the echo level is typically 6 dB below that of $x(n)$ such that the difference between $y(n) = d(n)$, and $y(n) = d(n) + v(n)$, can be discerned by $T = 2$. For AEC however, such a salient level difference relationship does not exist, with RIRs producing $d(n)$ signals with various level relationships to $x(n)$. As such, the Geigel algorithm's utility is typically limited to the network echo application.

2.3.2 Similarity based DTD

The similarity between various signals of the AEC problem in both the time and frequency domain has been used to detect DT. In [71], a DTD algorithm was proposed that recursively computes a sample estimate of the envelope of $x(n)$, and $y(n)$, and then performs an energy based comparison. DT is declared if the envelope of $y(n)$ exceeds that of $x(n)$ by a certain amount. A time-varying threshold, $T(n)$, was also proposed, which is dependent on the ratio of the envelope of $\hat{d}(n)$ to that of $y(n)$. This algorithm was shown to exhibit equivalent performance to the Normalized Cross Correlation (NCC) DTD algorithm (to be described in section 2.3.4) in terms of classification of DT, but generated more false classifications of DT. The proposed approach was also shown to be more computationally efficient than NCC DTD.

In [60] a DTD was presented that periodically measures the distance between the spectra of $y(n)$ and $x(n)$. The signal spectra were attained from two auto-regressive models, one fitted separately to each of these speech signals; the squared distance between these two models was then compared to a fixed threshold to decide on DT. Similarly in [37], the cepstral distance between the $x(n)$ and $y(n)$ signals over short time windows was proposed as a DT decision variable. This approach was found to be accurate classifier of DT and somewhat robust to echo path changes. In [72] various techniques were investigated that incorporate pitch estimators to solve the DT problem. It was found that pitch information, extracted from $y(n)$ can improve the discrimination of DT.

Another comparison based DTD that exploits psychoacoustic principles was proposed in [73]. This DTD inserts spectral gaps into the spectrum of $x(n)$ before it is emitted by the loudspeaker. The spectral gaps are chosen to exploit the masking properties of the human auditory system such that the corresponding distortion is inaudible to the near-end user. DT detection is then performed by monitoring the energy levels in the same frequency bands of the near-end microphone signal $y(n)$, whereby if the band energy exceeds a threshold DT is indicated. This technique is shown to achieve consistent performance for most conditions, but choosing the threshold is problematic as the spectral gaps in $y(n)$ will typically contain

smeared spectral energy resulting from reverberation in the absence of DT. A similar DTD, based on watermarking, is proposed in [74].

2.3.3 Cross-Correlation Step-Size control

The cross-correlation between various AEC signals has been proposed as a means of controlling the adaptation rate of AEC. In [75], a cross-correlation detection value between $e(n)$ and $\mathbf{x}(n)$ samples was proposed to control the step-size of stochastic gradient descent algorithms. The decision variable of the algorithm is defined as,

$$\begin{aligned}\mathbf{c}_{xe}^{(1)}(n) &= \frac{E\{\mathbf{x}(n)e(n)\}}{\sqrt{E\{x^2(n)\}E\{e^2(n)\}}}, \\ &= \left[c_{xe,0}^{(1)}, c_{xe,1}^{(1)}, \ldots c_{xe,L_h-1}^{(1)}\right]^{\mathrm{T}},\end{aligned} \tag{2.54}$$

where $c_{xe,0}^{(1)}$ is the cross-correlation coefficient between $x(n\text{-}0)$ and $y(n)$; as for the Geigel algorithm, the cross-correlation vector is computed over the length of $\mathbf{x}(n)$ to ensure coverage over L_h previous samples. A suitable norm, such as the infinity norm, is then applied to the resulting vector $\mathbf{c}_{xe}^{(1)}(n)$ to produce a scalar decision variable, for example

$$\xi_{xe}(n) = \left\|\mathbf{c}_{xe}^{(1)}(n)\right\|_{\infty}. \tag{2.55}$$

The decision variable $\xi_{xe}(n)$ is then compared to a preset threshold; if $\xi_{ex}(n) \geq T$, further adaptation of the AEC is permitted, and if $\xi_{ex}(n) < T$ adaptation of the AEC is halted. This decision logic is based the assumption that the signals $v(n)$ and $x(n)$ are uncorrelated, such that if $\hat{\mathbf{h}}$ has converged or if there is DT, there will be little correlation between $e(n)$ and $\mathbf{x}(n)$; conversely if $\hat{\mathbf{h}}$ was not converged then the correlation will be high. This approach therefore indicates when further adaptation is warranted, and does not explicitly detect DT.

The cross-correlation between $\mathbf{x}(n)$ and $y(n)$ was also proposed as a means of controlling the adaptation of an AEC [76], defined as,

$$\xi_{xy}(n) = \left\|\frac{E\{\mathbf{x}(n)y(n)\}}{\sqrt{E\{x^2(n)\}E\{y^2(n)\}}}\right\|_{\infty}. \tag{2.56}$$

Unlike $\xi_{xy}(n)$, this decision variable explicitly indicates DT to control adaptation, with a relatively low cross-correlation between $\mathbf{x}(n)$ and $y(n)$, i.e. $\xi_{xy}(n) < T$, indicating $v(n) > 0$ and therefore adaptation is suspended, and high cross-correlation, i.e. $\xi_{xy}(n) \geq T$ indicating $v(n) = 0$ and therefore adaptation continues.

Both of these cross-correlation based step-size controllers provide significantly better DT detection for AEC then energy based DTDs such as the Geigel DTD in section 2.3.1, with $\xi_{xy}(n)$ being considered more robust and reliable than $\xi_{xe}(n)$ [3]. However, a criticism of these decision variables is that they are poorly normalized [77], that is, for different circumstances they will produce different ranges of correlation values. This in turn means that the task of setting a value for T that is applicable in all contexts is difficult, with a T optimized for AEC in a certain room may be inadequate for another. This issue motivates the normalized decision variables that are described in the following section.

ALGORITHM	DECISION VARIABLE DESCRIPTION	COMMENTS
Energy-based DTD	Maximum value of Waveform-level Comparison	Suited to Network Echo, Level comparisons unreliable for Acoustic Echo
Cross-Correlation based DTD	Maximum value of Cross-correlation function	Range of maximum correlation values may vary, may not explicitly detect DT.
Normalised Cross-Correlation based DTD	Normalised Cross-correlation value, Coherence, Normalised inner product.	NCC value constrained to be less than or equal to one, with less than one idicative of DT, explicitly detects DT.

Table 2.2 : Summary table comparing key properties of Energy based, Cross-correlation based, and Normalised Cross-correlation based DTD algorithms.

Note that the statistical quantities introduced in this section, and to be introduced in the next, typically involve expectation(s) that have to be approximated in practice, and the underlying speech signals are non-stationary such that the estimates must be computed periodically. Two common methods for estimating these quantities in a time varying fashion are running window estimation and recursive estimation. For example, a running window estimate of the quantity, $E\{\mathbf{x}(n)e(n)\}$, may be computed as,

$$\mathbf{r}_{ex}(n) = \sum_{i=0}^{I-1} e(n-i)\mathbf{x}(n-i), \tag{2.57}$$

where I is the length of the window employed. The parameter I controls the trade-off between the response time of the estimate, and the reliability of the estimate, with I being inversely proportional to response time and directly proportional to smoothing.

A recursive estimate may be computed as,

$$\mathbf{r}_{ex}(n) = \lambda\mathbf{r}_{ex}(n-1) + (1-\lambda)\mathbf{x}(n)e(n), \tag{2.58}$$

where $0 < \lambda < 1$ is a forgetting factor. Similar to I, λ strongly influences the estimate, and it is inversely proportional to time response and directly proportional to smoothing/reliability. In the context of step-size control, a fast response time is desirable to mitigate divergence of the adaptive filter at the onset of DT, but a sufficiently accurate estimate is also desirable to mitigate false positive indications of DT that slow adaptation.

2.3.4 Normalized Cross-Correlation DTD

The normalized cross-correlation algorithm DTD (NCC) [77] explicitly detects DT, and has a suitably normalized decision variable, in the sense that for $v(n) = 0$, $\xi(n) = 1$ and for $v(n) > 0$, $\xi(n) < 1$. The NCC decision variable is explained through the following derivation, taken from [77], where the relevant sources are assumed stationary.

Suppose that $v(n) = 0$ therefore,

$$\sigma_y^2 = E\{y^2(n)\} = \mathbf{h}^{\mathrm{T}}\mathbf{R}_{xx}\mathbf{h}. \tag{2.59}$$

Since $y(n) = \mathbf{h}^{\mathrm{T}}\mathbf{x}(n)$, and $\mathbf{r}_{xy} = \mathbf{R}_{xx}\mathbf{h}$, (2.59) can be re-written as,

$$\sigma_y^2 = \mathbf{r}_{xy}^{\mathrm{T}}\mathbf{R}_{xx}^{-1}\mathbf{r}_{xy}. \tag{2.60}$$

In general, for $v(n) \neq 0$, (2.56) becomes,

$$\sigma_y^2 = \mathbf{r}_{xy}^{\mathrm{T}}\mathbf{R}_{xx}^{-1}\mathbf{r}_{xy} + \sigma_v^2 . \tag{2.61}$$

Dividing (2.60) by (2.61) and taking the square root, the following normalized cross-correlation detection variable is obtained,

$$\xi = \sqrt{\mathbf{r}_{xy}^{\mathrm{T}}\left(\sigma_y^2\mathbf{R}_{xx}\right)^{-1}\mathbf{r}_{xy}} . \tag{2.62}$$

By substituting (2.59) and (2.61) into (2.62), the detection variable may be expressed as

$$\xi = \frac{\sqrt{\mathbf{h}^{\mathrm{T}}\mathbf{R}_{xx}\mathbf{h}}}{\sqrt{\mathbf{h}^{\mathrm{T}}\mathbf{R}_{xx}\mathbf{h} + \sigma_v^2}} . \tag{2.63}$$

It can be deduced form (2.63) that for $v(n) = 0$, $\xi = 1$ and for $v(n) > 0$, $\xi < 1$, demonstrating that the NCC decision variable is properly normalized. Note that, theoretically, this decision variable is also immune to echo path changes.

The definition of the NCC detection value in (2.62) requires the inverse of $\mathbf{R}_{xx}$, which is a computationally intensive operation, and $\mathbf{R}_{xx}$ is ill conditioned, making this version of the NCC decision variable impractical. A fast version of the NCC decision variable in (2.62) was implemented in conjunction with the FRLS algorithm in [78]. However, NCC is most commonly implemented by noticing that (2.62)(2.63) can be simplified by assuming that $\mathbf{h} \approx \hat{\mathbf{h}}$, and by incorporating the expression $\mathbf{R}_{xx}^{-1}\mathbf{r}_{xy} = \mathbf{h}$, to give a computationally efficient variant of the NCC decision variable [79],

$$\xi_{NCC} = \frac{\sqrt{\mathbf{r}_{\mathbf{xy}}^{T}\mathbf{h}}}{\sigma_y} \approx \frac{\sqrt{\mathbf{r}_{\mathbf{xy}}^{T}\hat{\mathbf{h}}}}{\sigma_y} . \tag{2.64}$$

The NCC-DTD algorithm, incorporating this decision variable, is a popular DTD algorithm. In general, the NCC-DTD has been shown to provide superior DT detection performance relative to cross-correlation techniques, such as those described in section 2.3.3; the properties of these algorithms are contrasted in Table 2.1. The NLMS-NCC pairing is used as an experimental comparison in Chapter 4.

While the detection value in (2.64) is computationally much less intensive than that of (2.62)(2.63) and the fast NCC in [78], the accuracy of this approximation will depend on the misadjustment of $\hat{\mathbf{h}}$. In general, the misadjustment will be large upon initialization, and during and after room changes, during which time NCC-DTD will consequently produce inaccurate detection values, which can lead to false positive indications of DT [80]. False positives in turn lead to pauses in adaptation precisely when it should be taking place. A particularly deleterious consequence of this sensitivity is that should adaptation be paused due to a change in $\mathbf{h}$ being mislabeled as DT, the DTD may consequently mislabel DT in subsequent frames, indefinitely [37]. This scenario can severely reduce the convergence rate, and can lead to a self-perpetuating loop, effectively freezing the AEC system [37], which has been referred to as deadlock in the literature [81, 82].

To control the trade-off between missed and false DT detection; those induced either by noise, room changes or estimation error; and to reduce the risk of deadlock, a threshold is prescribed for NCC with adaptation suspended if $\xi(n) < T$, where $0 < T \leq 1$ is the allowed range of the threshold, with T typically being close to 1. Similar to the influence the step size

has on the convergence rate of an adaptive filter, T influences the sensitivity of NCC-DTD (and DTD algorithms generally), with the choice of T being a trade-off between higher sensitivity to DT (T closer to 1), entailing less missed detection of DT but more false detections, and lower sensitivity (T further away from 1), entailing more missed detections of DT and less false detections. This in turn implies that, analogous to the step-size for the adaptive filter, T influences the convergence rate of the adaptive filter, with greater sensitivity to DT entailing slower convergence due to false positives, while lower sensitivity filter implying greater instances of divergence of the adaptive due to DT.

The robustness of NCC-DTD to noise was improved by the introduction of a noise offset term into the NCC decision variable definition [83]. This noise offset was computed recursively from $y(n)$ during detected pauses in both echo or near-end speaker speech. The robustness of NCC to noise was further improved in [83] by a variable DT threshold, $T(n)$, that is dependent on an estimate of the echo to noise ratio. In [84], the NCC decision variable was analyzed in a statistical context for the purposes of calibration and performance evaluation. Based on this analysis a signal dependent variable decision threshold, $T(n)$, was introduced, which, compared to a fixed DT threshold, was closer to a theoretically defined maximum detection of DT probability for the same probability of false detection of DT.

A normalized DTD based on the cross-correlation between $y(n)$ and $e(n)$ was proposed in [85]; named Microphone signal, Error signal, Cross-Correlation (MECC). The MECC decision variable is,

$$\xi(n) = 1 - \frac{\mathbf{r}_{ey}(n)}{\sigma_y^2(n)}. \tag{2.65}$$

where $\mathbf{r}_{ey}(n)$ and $\sigma_y^2(n)$ are estimated recursively. The similarities between this decision variable and that of NCC in (2.63) are apparent by expressing (2.65) equivalently as,

$$\xi = \frac{\hat{\mathbf{h}}^{\mathrm{T}}\mathbf{R}_{xx}\mathbf{h}}{\hat{\mathbf{h}}^{\mathrm{T}}\mathbf{R}_{xx}\mathbf{h} + \sigma_v^2}, \tag{2.66}$$

where the time dependency has been removed. It is apparent that the MECC value differs only in the square root and substitution of $\hat{\mathbf{h}}$ for $\mathbf{h}$, which, given that $\hat{\mathbf{h}}$ is routinely used as an approximation for $\mathbf{h}$ for NCC, implies that the decision variables of these DTDs are very similar, a fact also borne out by the results of a comparative study, also presented in [85]. In terms of computation load, since $e(n)$ and $y(n)$ are time aligned only the zeroth lag cross-correlation value is computed; in contrast, for NCC-DTD L_h lags are required to account for the length of $\mathbf{x}(n)$, thus MECC-DTD is computational more efficient. However, both room changes and DT induce large increases in the magnitude of the error signal $e(n)$; therefore, similar to NCC, the substitution of $\hat{\mathbf{h}}$ for $\mathbf{h}$ renders MECC sensitive to room changes, with concomitant slower tracking performance of its paired AEC.

In [86], a DTD was presented with a normalized decision variable that can be interpreted as a measure of the angle between $\mathbf{y}(n)$ and $\hat{\mathbf{d}}(n)$, which is defined as,

$$\xi(n) = \frac{\hat{\mathbf{d}}^{\mathrm{T}}(n)\mathbf{y}(n)}{\|\hat{\mathbf{d}}(n)\|_2 \|\mathbf{y}(n)\|_2}. \tag{2.67}$$

This decision variable is then compared to a threshold to decide on DT. This approach was demonstrated to be an effective DTD; although the behavior of this detection value was not analyzed for room changes, for which we contend the increase in the magnitude of $e(n)$ after a room change will impact negatively on its performance. The decision variable and decision logic of this algorithm were analyzed in [87] and were improved, resulting in a reduced decision lag.

In [88] a DTD detection variable was introduced based on the estimated squared coherence between $\mathbf{x}(n)$ and $y(n)$, whereby the coherence is equal to 1 for $v(n) = 0$ and less than 1 when $v(n) > 0$, i.e. a normalized decision variable. The squared coherence, $S(k)$, is defined as,

$$S(k) = \frac{|\mathbf{S}_{xy}(k)|^2}{\mathbf{S}_{xx}(k)\mathbf{S}_{yy}(k)}, \tag{2.68}$$

where $\mathbf{S}_{xy}(k)$ is the DFT based cross-power density spectrum, $\mathbf{S}_{xx}(k)$ and $\mathbf{S}_{yy}(k)$ are the power spectral densities of $x(n)$ and $y(n)$ respectively, for the k^{th} frequency index. The DT detection value was formed by taking the average over a number of frequency bins, which were chosen for their propensity to have higher signal to noise ratios [88]. A connection between the coherence DTD and the NCC-DTD was demonstrated in [77].

The coherence function is particularly useful for batch based AEC control, including FDAF [89], for which step-size control/DT detection is also necessary (and subband techniques). Moreover, adaptive algorithms with fast convergence typically diverge fast at the onset of DT, making DTD for such algorithms particularly important [38]. In [90], a Frequency domain version of the NCC-DTD algorithm was presented. This variant of NCC was then generalized for MDF in [56]. Based on the notation introduced in section 2.2.5, the MDF-DTD decision variable is given by

$$\xi(m) = \frac{\sqrt{E\{\tilde{\underline{\mathbf{X}}}^{\mathrm{H}}(m)\mathbf{y}(m)\}^{\mathrm{H}}\hat{\underline{\mathbf{h}}}(m)}}{\sigma_y^2}. \tag{2.69}$$

This frequency domain decision variable is essentially analogous to that in (2.64). Through experiments it was demonstrated that the benefits of NCC translate into the frequency domain; though, so to the sensitivity to echo path changes, which, as described, elicit false positive indications of DT. The frequency domain version of MECC was presented in [91], with similar performance to the time-domain version. Note that MDF-DTD is used as an experimental comparison in Chapters 4 and 5 of this thesis.

Assuming a time-invariant and linear loudspeaker transfer function, it was shown in [92], that knowledge of the loudspeaker impulse response can be used to reduce the computational load of practical AEC-DTD algorithms by reducing the number of adaptive filter coefficients i.e. obviating the estimation of the coefficients attributable to the loudspeaker. This approach was demonstrated for the AEC-DTD pairing of NLMS and NCC DTD.

2.3.5 Two Path Model

The two-path model is another AEC adaptation control approach, which can be used as an alternative to DTD. The original two-path model approach [93] was proposed for network echo cancellation, and employed two identical filters to model **h**; a foreground filter and a background filter, with the foreground filter echo estimate used to cancel the echo. The foreground filter is not adaptive per AEC/EC, and instead obtains it coefficients from the background filter, which is adapted per AEC/EC. The coefficients of the background are transferred over to the foreground filter only when they are deemed to be performing better than the foreground filter's current coefficients [93]. By this way, during DT the foreground coefficients are fixed while the background coefficients diverge, such that echo cancellation performance is not interrupted. As no DTD is employed, the original two-path model approach overcomes the deadlock problem. However, echo path changes still affect the two-path model performance while the background filter is converging.

The decision to transfer the background filter coefficients to the foreground filter was originally based on a number of conditions that were tailored to the network echo problem, such as energy-based comparisons similar to that used by the Geigel DTD algorithm, which are unsuitable for AEC. To remedy this, two-path model transfer logic structures tailored for AEC have been proposed [81, 82, 94]. Of particular interest, is the decision logic introduced in [95], which does not require the tuning of any threshold parameters.

The main incarnation of the two-path model in AEC however, is as a means of preventing deadlock [78]. For this, both the background and foreground adaptive filter are adapted, and as before, only the adaptation of the foreground AEC is stalled when DT is detected by a DTD. However, to compute its decision variable, the DTD uses the estimate of the enclosure from the background filter. This ensures that the DTD never freezes foreground adaptation; although, since room changes affect both filters equally, the DTD may still generate false positives in response to room changes. Another benefit of this approach is that the characteristics of the background filter can be customized for the DT problem, i.e. the background filter can be endowed with a faster convergence rate such that it diverges faster than the foreground filter in the face of DT, enabling faster detection of DT, which in turn allows the foreground adaptive filter to be fixed before it diverges. Examples of AEC DTD pairings that have adopted this approach include those presented in [38, 56, 78, 90]. A downside to this approach is the extra computational load required to run an independent background adaptive filter. Note that this two-path model structure for AEC-DTD is employed in Chapter 5, wherein a novel DTD algorithm is proposed.

2.3.6 Echo Path Change Detection

Some schemes have been devised to differentiate between echo path change and DT, which, as described, are often confused by DTD detectors, and which demand the opposite response from the AEC in terms of adaptation. By contrasting the decisions from an echo path change detector with those of a DT detector, the sensitivity of DT detector can be increased without the increase in false positives following room changes.

It was noted in [37] that echo path changes mainly effect the higher frequencies of $e(n)$, with DT influencing lower and higher frequencies equally. This knowledge was used to devise an echo path change detector that computes the ratio of the short time power spectra of $e(n)$ to that of $y(n)$, separately for a higher and lower frequency band respectively; computes a decision variable by computing the ratio of the high frequency and low frequency values, summed over previous such values to remove noise; and then compares the resulting detection variable to a threshold to flag echo path change. This approach was demonstrated to detect echo path change reliably, i.e. without erroneously flagging DT, but after a small delay [37].

Also in [37], the two-path model was customized to perform echo path change detection for AEC. For this approach both the foreground and background filters are adapted per AEC, with DTD used to control only the foreground adaptation. However, the background filter is given less filter coefficients and its step size is dependent only on the input signal $x(n)$, i.e. for NLMS $\mu = 1$, such that it has a faster convergence rate than that of the foreground adaptive filter; though the foreground filter achieves a lower misadjustment. It follows then that after a room change the level of the error signal of the background filter will drop below that of the foreground adaptive filter for a period, which is used to indicate an echo path change. In comparison with the previously described approach, this echo path change detector flagged echo path changes with similar accuracy though with less delay.

Echo path change was tackled by a two stage DTD proposed in [96]. The proposed algorithm first detects both DT and echo path change, and then, assuming that $\hat{\mathbf{h}}$ changes more rapidly for an echo path change than for DT, DT and echo change are distinguished by comparing the derivative of the gain of $\hat{\mathbf{h}}$ to a threshold. It was shown experimentally that this approach can accurately distinguish between DT and echo path change, and the accompanying DTD performance has a lower probability of missed detection of DT than NCC DTD.

An echo path detector based on the smoothed short time squared coherence between $e(n)$ and the echo estimate $\hat{\mathbf{d}}(n)$ was proposed in [97]. This technique uses the increase in the squared coherence between these two signals that follows a room change as a signifier of echo path change. In [98], and following a similar derivation to the MECC DTD, a normalized echo path change detector was proposed with a decision variable that may be expressed as:

$$\xi_{rc} = \left| \frac{(\hat{\mathbf{h}} - \mathbf{h})^{\mathrm{T}} \mathbf{R}_{xx} \hat{\mathbf{h}}}{\mathbf{h}^{\mathrm{T}} \mathbf{R}_{xx} \hat{\mathbf{h}}} \right|, \tag{2.70}$$

where ξ_{rc} denotes the room change decision variable. It can be seen that for $\hat{\mathbf{h}} = \mathbf{h}$ this variable is equal to 0 and for $\hat{\mathbf{h}} \neq \mathbf{h}$ is greater than zero. This detector was combined with the MECC DTD, which was shown to provide comprehensive control of an AEC in terms of echo path and DT [99].

2.3.7 Combined AEC control

It is evident that some AEC step-size control techniques require a number of decision variables to configure the AEC for the various operational conditions of the Acoustic Echo

problem. It has been tacitly assumed thus far that the various detectors, such as DT and echo path change detectors produce binary control signals that can be combined with Boolean logic to control the stepsize, which entails the tedious task of manually tuning various detector thresholds. Accordingly, some work has been performed on systematically combining the control information fed to a specialized AEC control algorithm. A general framework was described in [15] for which information from various sources or detectors can be combined to classify the current operational condition, and to vary the step-size accordingly. For this approach, the various AE operational conditions are classified as states, such as DT and misadjustment. A representative set of feature vectors containing values of the various decision variables are then collected during each state. A Vector Quantization technique, or other similar techniques, can then be used to partition the feature space, with each state being assigned a suitable step-size. Based on this approach, in [100], a fuzzy rule-based control approach was proposed that uses information about the misadjustment of the adaptive filter and the decision variable from an echo path change detector, [37] described in section 2.3.6, to classify the current state of the system and to vary the step-size of the adaptive filter accordingly. This approach was shown to produce a considerable improvement in performance for most states or operational conditions.

In [101] an approach to DTD was presented that uses a real-time recurrent learning classifier to combine information from various independent speech detectors. For this approach, three speech detectors are employed; a far-end speech detector, to detect far-end speaker activity; a near-end speaker detector, to detect near-end activity, which can be from the near-end speaker or echo; and a third detector which is used to compare the far-end and near-end signals. In a frame wise manner, the various time-frequency features produced by each detector are fed to a single layer neural network with recurrent feedback, which outputs a value between 0 and 1 that is used as a decision variable for DT. This approach is shown to produce comparable performance to the NCC DTD for less computational load, and was stated to work independently of the RIR.

2.3.8 Doubletalk Robust Adaptive Filtering

As described at the end of section 2.3.3, a delay is required to estimate the statistical terms of the various detection variables that have been described in this section. This implies that at the onset of DT, a short interval typically ensues until the DT detector flags DT, during which time the adaptive filter may have diverged significantly. Moreover, DTD is prone to detection errors, such as failing to detect a short period of DT, which may also cause divergence of the adaptive filter. Under the interpretation of adaptive algorithms that minimize the mean squared error as being optimized for a noise signal, $w(n)$, drawn from a Gaussian PDF, samples of $v(n)$ manifest as large errors in the tails of this PDF that in turn, because the first derivative of the log-likelihood function of a Gaussian PDF is monotonically increasing, cause commensurately large gradients that diverge the adaptive filter coefficients rapidly. As described above, the rate of this divergence may be curtailed by setting a small step-size, or by decreasing the DTD threshold T, or both, at the expense of a reduction in convergence rate.

Alternatively, researchers have proposed robust forms of many of the adaptive algorithms used for AEC [102]. Robustness here implies an ability to withstand a certain amount of DT without diverging; for example, during the period between the onset and detection of DT. To derive a robust adaptive algorithm, a cost function is employed that has the property that its gradient is bounded. This means that values, either negative or positive, of the error signal $e(n)$ above a certain threshold induce a gradient that is bounded. This strategy therefore reduces divergences due to DT but also reduces the convergence rate after room changes; though, varying the level of robustness can mitigate this.

To exemplify the robust adaptive filtering paradigm, the update rule for the robust form of the NLMS algorithm derived in [102], is given as,

$$\hat{\mathbf{h}}(n+1) = \hat{\mathbf{h}}(n) + \frac{\mu \mathbf{x}(n)}{\mathbf{x}(n)^{\mathrm{T}}\mathbf{x}(n) + \delta}\psi\left[\frac{|e(n)|}{s(n)}\right]\mathrm{sign}\{e(n)\}s(n), \tag{2.71}$$

where $s(n)$ is the adaptive scale factor. The function $\psi(\cdot)$ in (2.71) was chosen in [102] to be the following limiter function:

$$\psi\left[\frac{|e(n)|}{s(n)}\right] = \min\{\frac{|e(n)|}{s(n)}, \zeta\}, \tag{2.72}$$

where ζ is a minimum offset. It is apparent from (2.72) that for $s(n) = 1$ and $\zeta = 0$, the NLMS gradient is attained, while $s(n) > 1$ results in a scaled or robust form of the NLMS gradient. The factor $s(n)$ therefore controls the level of robustness of this algorithm, and may be controlled in some way to maximize robustness to DT while allowing for a fast convergence rate in the absence of DT [102]. Robust adaptive algorithms such as this are typically accompanied by a DTD, and as for conventional AEC, adaptation is stalled by setting the $\mu = 0$. Robust versions of other adaptive algorithms have also been derived, such as FRLS in [103], extended MDF in [38], PAFA in [104], PNLMS in [105] and Variable Step Size NLMS (VSS NLMS) in [106].

Other DT robust acoustic echo mitigation approaches outside the preceding paradigm, include those presented in [107, 108], which additionally allow for adaptation during DT. For these approaches a pseudo-random noise sequence is added to $x(n)$ such that both signals are radiated by the near-end loudspeaker. Cross-correlating the pseudo random sequence with the near-end microphone signal produces a time-varying estimate of $\mathbf{h}$, which can be estimated both during DT and in the absence of DT, thereby obviating DTD. Based on the same principles, a DT detector that differentiates between echo path change and DT was proposed in [109].

Inspired by similar techniques used for Acoustic Echo Feedback, a prediction-error-approach was applied to the AEC problem in [110], whereby both $\mathbf{h}$ and an auto-regressive model of $v(n) + w(n)$ are identified jointly. For this purpose, two adaptive algorithms were derived, one based on a stochastic gradient descent update and another based on a Gauss-Newton update, both of which seek to optimize a prediction error based error criterion for the joint estimation of the coefficients of $\mathbf{h}$ and those of the AR model. The learned $\mathbf{h}$ is used to create an echo estimate for AEC. These approaches were demonstrated to be very robust to

DT, without the a fall off in convergence rate after changes to the echo path, with the Gauss-Newton based adaptive approach performing worse then stochastic gradient descent after echo path changes.

2.3.9 Variable step size

The hard decision of DTD and the resulting two-state step size variable, which deals with the DT operational condition only, contrasts with the Variable Step Size (VSS) approach, which provides a varying stepwise that attempts to obtain the optimum adaptation rate for all AE operational conditions, i.e. echo path change, DT, and noise. AEC specific VSS approaches therefore, aim ultimately to mitigate the divergence of the coefficients of an adaptive filter in response to DT, and, as discussed in 2.2.1, to improve adaptive filter performance by satisfying the competing requirements of fast recovery after changes to **h** and a small misadjustment. Intuitively, this requires a zero or low stepsize during DT; and in the absence of DT, a step size in proportionate to the adaptive filter misadjustment, and the background noise level.

Under the assumption that $x(n)$, $w(n)$ and $v(n)$, are white, short-time stationary, mutually uncorrelated signals, the following optimum step-size, $\mu^*(n)$, for NLMS, was proposed in [111],

$$\mu^*(n) = \frac{E\{(d(n) - \hat{d}(n))^2\}}{E\{e^2(n)\}}. \tag{2.73}$$

This expression follows from the preceding discussion, with $\mu^*(n)$ becoming small whenever $E\{e^2(n)\}$ becomes large due to DT or high background noise, and with $\mu^*(n)$ tending towards zero as the numerator tends to zero. In practice, this VSS rule is problematic however, as speech signals do not satisfy the underlying assumptions of (2.73), and the numerator of (2.73) is not observable and must be estimated. Techniques for estimating the numerator and the associated issues are discussed in [37].

Recently, another variable step size approach was proposed for the NLMS algorithm [112], which, as above, considers both the misadjustment and near-end speaker when varying the step size, but unlike above, no assumptions are made regarding the signals. It is argued in [112] that the optimal stepsize should be set such that,

$$E\{[y(n) - \hat{d}(n)]^2\} = E\{[v(n) + w(n)]^2\}, \tag{2.74}$$

which entails the following expression for $\mu^*(n)$,

$$\mu^*(n) = 1 - \frac{E\{[v(n) + w(n)]^2\}}{E\{e(n)^2\}}. \tag{2.75}$$

For $v(n) = 0$, it can be seen that for a large misadjustment, such as upon initiation or after a room change, $\mu^*(n)$ approaches 1 enabling fast convergence, while $\mu^*(n)$ tends to zero as the adaptive filter converges, enabling a lower misadjustment. Moreover, during DT $E\{(v(n) + w(n))^2\} = E\{e(n)^2\}$, and thus, $\mu^*(n) = 0$, assuming the adaptive filter has converged. Another benefit of this VSS rule is the absence of thresholds, which can be difficult to tune in practice; although, as for (2.73), the VSS rule requires the terms in (2.75) to be estimated. In

[112], it is assumed $v(n) = 0$ and $w(n)$ is stationary, such that $E\{[v(n) + w(n)]^2\}=E\{w(n)^2\}$ which is estimated during pauses in speech, and the resulting adaptive algorithm, which is termed Non-Parametric Variable Step Size NLMS (NPVSS-NLMS), was demonstrated to provide a lower misadjustment for a similar convergence rate to conventional NLMS.

In the presence of DT, the inherent non-stationarity of $v(n)$, entails that $E\{[v(n) + w(n)]^2\}$ must be estimated periodically. To this end, a Near-End Signal Energy Estimator (NESEE) was proposed in [39] that accurately estimates $E\{[v(n) + w(n)]^2\}$ using,

$$\begin{aligned}\gamma(n) &= E\{e(n)^2\} - \frac{1}{E\{x(n)^2\}}\mathbf{r}_{ex}(n)^{\mathrm{T}}\mathbf{r}_{ex}(n),\\ &= E\{v(n)^2\} + E\{w(n)^2\},\\ &= E\{[v(n)+w(n)]^2\},\end{aligned} \tag{2.76}$$

where $\gamma(n)$ denotes the NESEE estimate, $v(n)$ and $w(n)$ are assumed uncorrelated, and all terms are observable and estimated recursively in practice. NPVSS-NLMS in conjunction with NESEE was demonstrated to be very robust to DT [39]; a similar novel algorithm, named NEW-NPVSS-NLMS, was also proposed in [39] and shown to have comparable performance to NPVSS-NLMS. However, both NEW-NPVSS-NLMS and NPVSS-NLMS exhibited a slower convergence rate, compared to fixed step size NLMS, after a change to **h**, which was ascribed to a trade-off between robustness to DT and convergence rate. These adaptive algorithms were not tested for a room change during DT.

The preceding VSS paradigm and the proportionate step-size paradigm were melded for NLMS in [23]; and in [113], VSS for the APA algorithm was presented. VSS has also been investigated for MDF in [114], where an optimal step size for NLMS was derived and then applied to the MDF algorithm.

2.4 Alternative Acoustic Echo Mitigation techniques and Residual Echo Suppression

This section describes alternative adaptive system identification approaches to acoustic echo mitigation and Residual Echo Suppression (RES) algorithms.

2.4.1 Blind Source Separation AEC

Conventional AEC adaptive algorithms almost exclusively employ first and second order statistics of the available AEC signals, i.e. mean, variance, and correlation of $x(n)$, $y(n)$ and $e(n)$, to identify **h** in an adaptive fashion; likewise for DT detectors with respect to DTD. In an bid to obviate DTD and its complications, and to perform adaptation during DT, techniques developed for Blind Source Separation (BSS) [115], for which higher order statistics are often employed, have been considered to adaptively identify **h**. BSS is applicable here because the AE problem can be viewed as a BSS problem, in which $y(n)$ and $x(n)$ are two mixtures, with $y(n)$ containing the sources $v(n)$ and a reverberated version of $x(n)$; because of the availability of $x(n)$ this has been referred to as Semi-Blind Source Separation (SBSS). The aim in this case therefore is the separate $v(n)$ and $d(n)$. Ignoring noise, the associated mixing model may be expressed as [116],

$$\begin{bmatrix} y(n) \\ \mathbf{x}(n) \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{h}^{\mathrm{T}} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} v(n) \\ \mathbf{x}(n) \end{bmatrix}. \tag{2.77}$$

The main benefit of this model of AEC is that $v(n)$ is incorporated into the model from the start, and therefore, DTD is not required and adaptation can occur during DT. The demixing model for (2.77) is,

$$\begin{bmatrix} e(n) \\ \mathbf{x}(n) \end{bmatrix} = \begin{bmatrix} -1 & \hat{\mathbf{h}}^{\mathrm{T}} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} y(n) \\ \mathbf{x}(n) \end{bmatrix}, \tag{2.78}$$

where, as in the AEC case, $e(n)$ represents the signal sent to the far-end user, and contains any residual echo, $v(n)$ and noise. Like conventional adaptive filtering, it is apparent from the demixing model that $\hat{\mathbf{h}}$ must be estimated to achieve separation; this entails a convergence period upon initiation and after room changes similar to conventional adaptive filtering. Note that for this discussion, we ignore the scaling and permutation ambiguities inherent to BSS.

System identification algorithms based on well-known BSS cost functions have been developed to estimate the demixing system in (2.78) to achieve AEC. In [116], a system identification approach based on Independent Component Analysis was presented, whereby a cost function measuring the mutual information between $y(n)$ and $x(n)$ was employed. A sample-by-sample based stochastic gradient descent algorithm was presented to minimize this cost function, where the required signal PDFs were estimated recursively using a Parzen window over previous samples of the signals. The performance of the resulting technique was shown to be the comparable to NLMS in terms of initial convergence, tracking and computational load, but the proposed algorithm worked well during DT.

It was shown [117] that by inserting different noise suppressing, memory-less, non-linearity's into the error fed back loop of a conventional AEC, different cost functions are attained, some of which reduce to cost functions based on higher order statistics [117], such as those used for ICA and BSS in general. This approach is also similar to Robust Adaptive filtering discussed in section 2.3.8 in that a non-linearity is used to compress $e(n)$ to limit large increases in misadjustment. Several different non-linearity's were investigated in [117] by placing them in the error feedback loop of NLMS, and separately, in the error feedback loop of a block based FDAF algorithm. It was shown the introducing non-linearity's provides robustness to DT, with passable convergence and misadjustment performance. It was also shown that by allowing the FDAF with the non-linearity to update more often than once per frame (data-reuse), improved convergence and misadjustment performance are obtained relative to once per frame. This improvement was ascribed to the fact that iterative algorithms based on higher order statistics are more suited to processing batch data recursively in an offline fashion, rather than sample-by-sample data in an online fashion.

In [118], the correlation functions of the AEC signals was exploited to perform AEC in the context of convolutive BSS, which was implemented efficiently in the frequency domain. In [119], an AEC approach based on BSS was presented that, using a gradient descent approach, minimizes the Kullback-Leibler divergence between the PDFs of $x(n)$ and $y(n)$, which are estimated periodically using a histogram method. It was reported that the proposed

algorithm achieves good results on simulated data and promising results on real recorded data. More recently, the same author proposed a BSS system identification approach for both EC and AEC that exploits the non-stationary, or the distinct time domain correlation of the signals, to produce separation [120]. It was shown that the resulting approach was efficacious at learning **h**; although convergence is relatively slow as much speech data is required to obtain enough information to converge. In [121], the non-gaussianity of the near-end speakers speech signal is used to effect separation of the echo and the near-end speaker. More specifically, super Gaussian priors for the echo and near-end users speech signals are employed in a maximum likelihood framework to perform separation. The efficacy this approach was then demonstrated empirically. Semi-BSS techniques have also been researched for multichannel AEC applications [122].

2.4.2 Non-Linear AEC

In this thesis, we assume linearity of the various components in the hands-free telecommunications chain. While this is an acceptable and routinely invoked assumption, efforts have been undertaken to address a source of non-linearity for this problem, that is, the loudspeaker [123]. Consumer loudspeakers operating at the highest signal level are known to introduce a significant amount of harmonic distortion into the broadcast far-end users speech signal [123-125], which may in turn be propagated to the far-end user. To cancel these non-linear effects, which are typically perceived as echo by the far-end user, non-linear AEC algorithms have been proposed. A typical approach for non-linear AEC is to use a separate model for the harmonic distortion; Volterra filters are commonly employed for this purpose, in combination with a conventional linear AEC model. A suitably tailored adaptive algorithm is used to updates the coefficients of both models [124, 125]. In general, non-linear AEC based on Volterra filters has been shown to be an effective approach to mitigating the harmonic distortion induced by consumer loudspeakers [124, 125]. However, Volterra filters have a high computational cost, which is approximately $O(L^3)$ per sample, and have slow convergence [124, 125]. Non-linear AEC filtering has also been addressed using higher order statistics in [126]; Volterra filtering in the STFT domain in investigated in [127]; and the performance of Volterra filters in the frequency domain using the MDF algorithm was examined in [128].

A novel approach to non-linear AEC was described in [129]. For this approach, the signal acquired from an accelerometer mounted on the frame or magnet of the loudspeaker is fed with $e(n)$ to the adaptive filter. This approach was demonstrated to induce a 15 dBs increase in echo reduction over conventional, linear, AEC.

2.4.3 Acoustic Echo Suppression

The goal of AEC is to cancel $d(n)$ at the waveform level; in contrast, the algorithms in this section attempt to estimate the echo such that it can be suppressed via noise reduction/speech enhancement techniques. In [130], acoustic echo mitigation is performed in the magnitude

STFT domain. This approach, known as Acoustic Echo Suppression (AES), considers the following model of the acoustic echo problem,

$$|Y(f,k)| = |D(f,k)| + |V(f,k)| + |W(f,k)|, \tag{2.79}$$

where $Y(f,k)$ is the STFT of $y(n)$, f denotes discrete frequency, k is the frame index, and $|\cdot|$ is the magnitude of a complex value; the STFT is defined in Chapter 3, in which the use of such approximate spectral models is also discussed. $|V(f,k)|$ and $|D(f,k)|$ represent the $v(n)$ and $d(n)$ components of the mixture in the magnitude STFT domain. The aim of AES is to estimate the echo component $|D(f,k)|$ such that it may be suppressed.

To estimate $|D(f,k)|$, $D(f,k)$ is first estimated using adaptive system identification in the STFT domain. Regular FIR subband filters are employed instead of cross-band filters, and the RLS algorithm is used to adapt the filter coefficients. To mitigate the large misadjustment error, attributable to system identification in the STFT domain without cross-band filters, an estimate of the magnitude response of $|D(f,k)|$, denoted by $\left|\hat{D}(f,k)\right|$, is computed and is applied to $|Y(f,k)|$ using the following spectral subtraction rule,

$$\left|\hat{E}(f,k)\right| = \left[|Y(f,k)|^{\alpha} - \beta\left|\hat{D}(f,k)\right|^{\alpha}\right]^{\frac{1}{\alpha}}, \tag{2.80}$$

where $\left|\hat{E}(f,k)\right|$ is the modified error term. The terms, α and β control the trade-off between distortion of the near-end users speech during DT, and suppression of the echo interference, with β being directly proportional to distortion and to echo suppression. The modified time domain error signal, to be sent to the far-end user, is then synthesized from $\left|\hat{E}(f,k)\right|$ and the phase response of $Y(f,k)$, with a simple overlap and add scheme. It was shown that, in comparison with AEC, this AES algorithm attains higher echo reduction, and offers robustness to changes in $\mathbf{h}$, but also introduces distortion into the resulting near-end users speech during DT. It was described that the room change robustness demonstrated by AES is proportional to β; therefore, AES may be construed as sacrificing some magnitude distortion (and phase distortion) in return for some robustness to room changes.

Another AES technique was presented in [131, 132], in which the subbands are partitioned according to the bark scale, with a subband filter in each critical band. The reduced number of subbands signals significantly reduces the computational load of this algorithm in comparison to the previous approach. The estimate of $|D(f,k)|$ is computed by a linear interpolation of the gains of each of the subband filters. This interpolated estimate was then used to suppress the echo using the following spectral modification rule,

$$\left|\hat{E}(f,k)\right| = \left[\frac{|Y(f,k)|^{\alpha} - \beta\left|\hat{D}(f,k)\right|^{\alpha}}{|Y(f,k)|^{\alpha}}\right]^{\frac{1}{\alpha}}, \tag{2.81}$$

with the phase response of $Y(f,k)$ and a simple overlap and add scheme used to synthesize the output. This AES approach was compared to AEC, with both AES and AEC adapted using NLMS. Both approaches were shown to exhibit a comparable misadjustment and initial convergence rate. Furthermore, similar to the previous AES algorithm, because of its

magnitude STFT formulation and the option to sacrifice magnitude distortion for robustness to changes in **h** by increasing β, AES was shown to be robust to room changes that affect the phase response of **h** and the fine detail of its magnitude response. It was also reported that musical noise was less prevalent in the processed speech in comparison to the previously described AES algorithm. This was ascribed to applying the smoother modifications to $|Y(f, k)|$.

Like AEC, AES algorithms require a DTD. In [133] the previously described AES technique was paired with a frequency domain DTD. This DTD employs numerous different cross-correlations between the various signals of the AES problem (similar to the time domain decision variables that were presented in [134]), which were chosen for their ability to discriminate DTD. Training data consisting of feature vectors comprised of values for these cross-correlations variables collected during separate periods of double-talk and echo, where used to train a separate Gaussian Mixture Model (GMM) for these events. These GMMs were subsequently used, given an observation of $y(n)$, in a likelihood ratio test to classify DT and single talk to control the AES algorithm. This approach was shown to be effective at discriminating DT but was not tested for room change.

A hybrid AEC and AES technique was presented in [135]. This approach employs AEC to cancel the early reflections of the RIR and AES to suppress the late reflections of the RIR. It is shown that this results in a better compromise between distortion and robustness to room change. This hybrid approach also serves as a link between AES, which substitutes for AEC, and some similar Residual Echo Suppression (RES) techniques, which compliment AEC, and are described in the following section.

2.4.4 Residual Echo Suppression

Residual Echo Suppression (RES) algorithms are commonly employed to reduce residual echo that is not cancelled by the AEC; the residual echo may be comprised of non-linear artifacts, as discussed above, or may arise from a poorly adjusted adaptive filter. Early RES techniques evolved from the echo suppressor, introduced at the start of this chapter, and were typically nonlinear processors (NLP) that serve to clip or suppress $e(n)$ in the absence of DT [14]. Like echo suppressors however, NLP can significantly suppress $v(n)$ due to detection errors, which has motivated more sophisticated RES algorithms.

RES algorithms based on speech enhancement have been developed [136]. Similar to AES, these RES algorithms operate in the magnitude spectral domain and consider $|E(f, k)|$, as containing a near-end speaker component, $|V(f, k)|$ and a residual echo, component, denoted by $|D_R(f, k)|$; the aim being to suppress $|D_R(f, k)|$ for minimum distortion of $|V(f, k)|$. Noise is also often considered by RES algorithms to give the following model,

$$|E(f, k)| = |V(f, k)| + |D_R(f, k)| + |W(f, k)|. \tag{2.82}$$

The phases of $E(f, k)$ are used to synthesize an output signal. In [136], three techniques for suppressing $|D_R(f, k)|$ based on post-filtering $|E(f, k)|$ were compared. Each technique employed smoothed recursive estimates of the power spectral densities of the respective

signals, but differed in the type of post-filter; generally speaking, these post-filters were Weiner-based and assigned time-frequency points with predominately residual echo a low gain and those with predominately near-end user speech a relatively high gain. It was shown that post filtering can significantly suppress the residual echo in a typical $e(n)$ signal. As expected however, objective measures indicated that each algorithm produced some distortion of the near-end users speech signal, but the authors contend that subjectively this distortion has a modest effect on the speech quality.

In [137], a RES algorithm was presented that incorporated psychoacoustic knowledge to improve the subjective quality of the processed speech, and to reduce speech enhancement artifacts. Specifically, a perceptually weighted frequency domain filter was constructed in a frame-wise manner to shape both $|D_R(f, k)|$ and $|W(f, k)|$ so that they are, as much as possible, masked by $|V(f, k)|$ during DT, and such that suppression artifacts are not introduced in the absence of DT. This approach was shown to result in better quality speech with less over-suppression artifacts such as musical noise (a significant issue for RES algorithms) but requires a significant computational expense to compute the masking threshold of the near-end speech. A statistical RES approach was presented in [138], where a frequency domain Wiener filter is applied to $|E(f, k)|$ to suppress $|D_R(f, k)|$. This filter is updated depending on four different states: $x(n) > 0;\ x(n) > 0,\ v(n) > 0;\ x(n) > 0,\ [d(n) - \mathbf{h}^T\mathbf{x}_n] > 0;$ and $v(n) > 0,\ x(n) > 0,\ [d(n) - \mathbf{h}^T\mathbf{x}_n] > 0$. Each state is discerned using hypothesis testing based on the estimated PSD of the sources, which are then used to update the Weiner filter. This approach is shown to produce better performance in terms of echo suppression than a comparative algorithm.

Operating in the same domain and again similar to AES, adaptive RES algorithms have recently been developed. In [139] a regression approach is adopted whereby each frequency bin of $|D_R(f, k)|$ is modeled as a linear combination of the previous bins of $|X(f, k)|$. This model is learned adaptively using stochastic gradient descent. This approach was evaluated and shown to offer an additional 7 dB of echo mitigation. Using the same model, $|D_R(f, k)|$ is related to the previous frames of $|X(f, k)|$ using the expectation maximization algorithm in [140]. This approach performs achieves significantly increased echo mitigation, relative to baseline AEC, in terms of objective measures of performance, and subjectively in terms of mean opinion scoring. An adaptive RES algorithm is presented in [141], in which each bin of $|D_R(f, k)|$ is modeled as a linear combination of previous bins of the magnitude STFT of $\hat{d}(n)$. It is shown that by applying a time varying gain to each frequency bin of $|D_R(f, k)|$ suppression of echo is achieved, albeit with some distortion of $|V(f, k)|$ at high frequencies. Note that adaptive RES algorithms must be controlled using a DTD to prevent divergence of their model.

Finally, jointly optimizing AEC and RES for acoustic echo mitigation has also been explored [142-145]. For the approach described in [142] the acoustic echo-path(s) is modeled as a linear statistical model in an attempt to explicitly model its time-invariance. The corresponding MMSE solution is described as a generalized Wiener solution, which consists

of a deterministic component and a statistical component that are addressed by AEC and a Wiener post-filter [146] respectively. A state space model-based controller for the adaptation of both filters was presented in [142], and, using the FFT algorithm and a number of approximations, a computationally efficient implementation of the Kalman filter was derived to estimate the state from the available signals. It was shown that this approach achieved good performance for a slowly varying echo path model and does not require any tuning of parameters.

2.5 Discussion

It is apparent that the performance of conventional AEC is constrained primarily by echo path change and DT. Considering these conditions in the context of adaptive filtering alone, while adaptive algorithms such as APA, RLS and GMDFα have a fast convergence rate, which can be employed to mitigate the effects of room changes, faster convergence generally implies faster divergence during DT. It follows therefore that room changes dictate a fast convergence rate, while DT in contrast, dictates a slow convergence rate; setting a static compromise step-size to address these constraints is inadequate, as adaptive filters can diverge quite rapidly at the onset of DT even for small stepsizes, while also having an impaired convergence rate and tracking capability. Now considering these conditions in the context of adaptive filtering paired with DTD, by detecting DT, thereby enabling adaptation to be stalled, DT detectors mitigate the deleterious divergence of the adaptive filter during DT, which, ostensibly at least, allows for a faster convergence rate. However, since DT detectors tend to erroneously flag echo path changes as DT, stalling convergence after such events, the benefits of a faster convergence rate are effectively diminished. Moreover, if the sensitivity of the DTD is reduced, by adjusting its threshold variable, such that it is less likely to erroneously flag echo path changes, then there is greater chance of missed DT, leading the greater divergence of the adaptive filter, which in turn entails a lowering of its step-size. This circular dependency between DTD and AEC serves to highlight the difficulty in dealing with DT and echo path change using this approach, and the resulting inconstancy of AEC-DTD performance due to these events.

As described in this Chapter, to alleviate this dependency, alternative AEC control approaches incorporating echo path change detectors or the two-path model have been deployed. However, both room change and DT detectors typically require a lead time before the relevant statistical quantities register a change, which implies that the adaptive filter will be somewhat diverged, curtailing the convergence rate of the adaptive filter or necessitating greater detector sensitivity; both of which, as described, having effectively the same effect on performance. Adaptive algorithms robust to DT and variable step-size adaptive algorithms have also been proposed, but like AEC-DTD, they too exhibit a trade-off between convergence rate and robustness to DT; perhaps confirming a fundamental trade-off of AEC. Furthermore, even if perfect DTD were available or perfect robustness to DT, adaptive algorithms cannot generally immediately adapt to a room change, and as such some echo is

usually sent to the far-end user after a room change irrespective of the adaptive algorithm. This is complicated further still by the possibility of a room change occurring during DT, which is an operational condition not commonly addressed by the conventional AEC literature. These problems have motivated distinctly new approaches to AEC, such as BSS based system identification, AES, prediction error filtering, and more elaborate RES approaches. However, these approaches are still rooted in the adaptive system identification approach of AEC and are consequently sensitive to echo path change; albeit AES, and AEC with RES are less so; with AES also requiring a DTD, and so too some RES techniques.

Given the complications with the predominant adaptive system identification approach to AE, and the apparent maturity of acoustic echo mitigation as a research topic within this framework (beginning as it does with the earlier problem of network echo), we contend that it may now be fruitful to experiment with some new frameworks for acoustic echo mitigation that may offer the flexibility required to more aptly address the AE problem. Accordingly, in Chapter 4 we eschew the conventional adaptive system identification framework and the concomitant complications and propose a new framework for acoustic echo mitigation. This framework casts the AE problem as a monaural sound source separation problem, in which the near-end microphone signal is considered a mixture containing an echo and a near-end speaker signal, with the aim of separating the echo and the near-end speaker. By treating $v(n)$ as a source that may be inactive or inactive, akin to BSS system identification; and by acknowledging that complete separation is not possible, manifesting as distortion of the near-end users speech signal, akin to AES/AEC-RES; we will show that this framework allows for an invariable level of acoustic echo mitigation, during all the operational conditions of the AE problem, without the requirement of a control algorithm or without the trade-offs associated with adaptive system identification. This approach does however give rise to a new trade-off, which can also be controlled, namely, between distortion of $v(n)$ and suppression of $d(n)$, which we contend is more attractive trade-off than those of AEC.

Additionally, since this approach treats $v(n)$ as a source that may be inactive or inactive, it may also be employed, independently, to address the DT detection problem for conventional AEC. Specifically, the separated near-end signal can be used to form a DT decision variable, which can be compared to a threshold to control the step-size of a conventional acoustic echo canceller. Because many of the desired attributes of an acoustic echo mitigation system have a counterpart in AEC control, such as sensitivity to DT and insensitivity to room change; the benefits of framing acoustic echo mitigation as an MSSS problem are duly inherited by this DTD, and therefore it compliments conventional AEC. This novel DTD is described and evaluated in Chapter 5.

The next chapter reviews the MSSS techniques that inspired, and enabled, this alternative interpretation of the acoustic echo mitigation problem and of the Doubletalk Detection problem to be realized.

3 MODEL-BASED MONAURAL SOUND SOURCE SEPARATION AND NONNEGATIVE MATRIX FACTORIZATION

This chapter reviews Monaural Sound Source Separation (MSSS) and Nonnegative Matrix Factorisation (NMF). This review begins in section 3.1 with a general introduction to MSSS and model-based MSSS, including some background on the more general source separation problem. Before tackling model-based MSSS in greater detail, we deem it necessary to describe NMF in section 3.2. Since NMF is a central theme of this thesis, we take the opportunity to describe it in considerable detail, beginning with a general description of its properties, followed by specific subsections in which common NMF cost functions, auxilary contraints and algorithms are discussed. In section 3.3, we return to model-based MSSS. Section 3.3 begins by formulating the MSSS problem, followed by an outline of CASA, including hybrid CASA/model-based techniques. Then, in section 3.3.2, we describe the various spectral representations in which model-based MSSS algorithm operate, and discuss some of the theortical and practical implications of using these representations. In the subsequent two sections; 3.3.3, and 3.3.4, we review model-based MSSS by considering two classes of algorithms; namely, those based on probabilistic inferential techniques and matrix factorisation techniques respectively. Both sections have a similar structure, whereby a generic algorithm, representative of this class, is first described, followed by a review of the specific techniques contained by the class. Chapter 3 ends in section 3.4 in which we discuss source modeling for reverberated speech signals. In this section, it is demonstrated how low-rank NMF bases can be used to represent both reverberated and non-reverberated speech signals in the spectral domain, motivating its application to the acoustic echo and doubletalk detection problems in subsequent chapters. Also in section 3.4, we introduce and motivate some of the techniques that are to be used in these chapters.

3.1 Introduction and Background

There exist numerous scenarios in which an arbitrary number of signals, emanating from independent sources of the same physical class, are observed contemporaneously by an arbitrary number of sensors. Some examples include; microphone recordings of people speaking in a room, or of a musical ensemble playing; non-invasive observation of neuronal brain activity using electroencephalography, magnetoencephalography, or functional magnetic resonance imaging; or multiple telecommunication signals received at an array of antennas. In each of these scenarios, the source-sensor geometry and the possibility for multi-path propagation means that each sensor observes a superposition of a filtered version of each source signal. The type of filtering or source mixing depends on the specific scenario, with source mixing often classified as either linear, where the filtering applied to a source in a mixture is modeled by a single gain, i.e. instantaneous mixing; or more generally, as convolutive, where it is modeled by an arbitrary number of delayed gains, accounting for multi-path propagation, i.e. reverberation. A special case of convolutive mixing is where a scaled and delayed version of the source signals is received, corresponding to direct path signals only, which has been referred to as anechoic mixing [115].

Given a mixture or set of mixtures it is often desirable to observe the original source signals in isolation, motivating source separation; it is also desirable for generality that a minimum number of source signal properties be invoked to achieve separation. These considerations are the remit of the active research topic of Blind Source Separation (BSS). Considering linear-mixing BSS problems where the number of sensors exceeds (over-determined) or matches (even-determined) the number of sources, BSS algorithms typically seek to separate the sources by identifying the inverse mixing system, which can be identified up to an indeterminate permutation and scaling. In this context, a commonly exploited property of source signals is their statistical independence [147]. This assumption underpins numerous BSS algorithms that identify an inverse mixing system by optimizing an objective/cost function that measures the independence of the set of mixtures, a review of which is available in [148]. This approach can be interpreted as decomposing a multivariate signal into its independent components, which gives rise to the term Independent Component Analysis (ICA) [149]. Besides ICA, numerous other BSS algorithms for over-determined/even-determined mixing have been devised that exploit alternative, and typically equally generic, properties of source signals to identify the inverse mixing system; see [150] for a general review of BSS. Many of these algorithms, including ICA, have been applied to the special BSS case of AEC, as discussed in Chapter 1 section 2.4.1.

Considering BSS problems with fewer mixtures than sources, the now under-determined mixing system does not permit an inverse mixing system, precluding separation by the techniques just described. Another property of signals that has been exploited for BSS, and which is applicable to under-determined problems, is sparsity [151-154]. A signal is said to have a sparse representation in a certain domain if its energy in this domain is concentrated in relatively few coefficients, such that most coefficients are zero. For a mixture of sources in

a sparse representation, this implies that most of the energy of the source signals reside in mutually disjoint sets of coefficients, allowing for each source's coefficients to be grouped using an appropriate grouping cue, effectively separating the sources. A formalism of this concept for speech signals in the STFT domain is provided in [153, 155] where it is termed Windowed Disjoint Orthogonality (WDO), and where it is empirically demonstrated that speech signals in the STFT domain approximately satisfy this property. Based on the WDO assumption, the signals of different speakers in two anechoic speech mixtures can be separated in the STFT domain by grouping the domain coefficients based on their inter-mixture spatial information [155, 156]. An approach for separation of stereo music signals (linear mixing) that uses sparsity is described in [157, 158]. Sparsity-based sound source separation is reviewed in [115].

Separating an arbitrary number of sound sources from a single mixture of those sources, which we refer to as Monaural Sound Source Separation (MSSS) [1], is particularly challenging because the use of spatial information is precluded; an exception being the MSSS algorithm outlined in [159], which leverages spatial information in the form of a head-related transfer function to perform MSSS. One approach to MSSS is to use the natural cues of the signal class to group the transform coefficients of the source signals in a sparse representation of their mixture. For speech signals, this approach has been applied in various spectral domains, where cues such as common onset/offset [160] and amplitude/frequency co-modulation [161] of a speech signal's time-frequency energy have been used to group localized segments of time-frequency energy, which are then pieced together using temporal cues such as pitch to achieve a separation [161, 162]. An overarching aim of these techniques is to emulate the inherent ability of humans to discern sound sources from mixtures using these cues, a topic known as Computational Auditory Scene Analysis (CASA) [163-165]. In this chapter, we are interested in a more recent approach to MSSS in which it is viewed broadly as a model-based statistical pattern recognition problem; first proposed in [152]. This viewpoint is driven by the regularity and distinctiveness of the patterns exhibited by the speech signal of different speakers, particularly in a spectral representation, with a view to applying powerful machine learning and matrix factorization techniques to identify and segregate these features. A central theme of MSSS techniques based on this perspective has been the use of prior information about the sources in a mixture to perform separation. In general, prior information is used to train a model of each source in the mixture a priori, giving rise to the term model-based MSSS. While the availability of such information is an assumption that deviates considerable from the ideal of BSS, it is often deemed necessary in light of the underdetermined nature of the MSSS problem.

To describe model-based MSSS, we broadly classify the algorithms that constitute this approach into two classes according to the type of source modeling and inference

37

[1] MSSS is also referred to variously as single/one sensor/channel/microphone/mixture sound source separation in the literature.

approach taken. In the first class of techniques, a class we refer to as probabilistic model-based MSSS, a probabilistic discrete-state model structure is used for the source modeling, an instance of which is trained for each source on their respective training data. Then, the probability of the sources is inferred given the trained source models and the mixture, afterwhich the sources are estimated using some criteria. A feature of the techniques of this class is that they generally conform to a Bayesian formalism of the MSSS problem [166, 167]. In the second class, termed matrix factorization model-based MSSS, the source models are typically low-rank factorizations of their respective source's training data, and the mixture is factorized by the union of these source models, such that each source model represents its source's likely contribution to the mixture, effectively separating the sources. Of the different matrix factorization techniques that have been employed for this task, Nonnegative Matrix Factorization (NMF) and its variants are the most prevalent. This is mainly because a low-rank NMF of the spectrogram of a source's training data provides a parts-based decomposition, with the parts typically expressing characteristic source-specific spectral features that can be matched to that sources contribution in the mixture during its decomposition. Note that these two classes of techiques are treated in greater detail in sections 3.3.3 and 3.3.4 respectively.

3.2 Nonnegative Matrix Factorization

Nonnegative Matrix Factorization (NMF) is an unsupervised linear basis decomposition technique for nonnegative data [6, 168]. NMF expresses a nonnegative $F \times K$ matrix of data $\mathbf{A}$, as the product of two nonnegative matrices $\mathbf{B}$ and $\mathbf{G}$ plus an $F \times K$ residual matrix $\mathbf{E}$ that is not constrained to be nonnegative, such that

$$\mathbf{A} = \mathbf{BG} + \mathbf{E} \approx \mathbf{BG}. \tag{3.1}$$

The dimensions of $\mathbf{B}$ and $\mathbf{G}$ are determined by the user specified rank of the factorization R, such that $\mathbf{B}$ is $F \times R$ and $\mathbf{G}$ is $R \times K$, with R typically chosen to be less than K. The columns of $\mathbf{A}$ and $\mathbf{G}$ are in one-to-one correspondence, such that when $\mathbf{A}$ has a single column, $\mathbf{a} = \mathbf{Bg}$ expresses $\mathbf{a}$ as a linear combination of the columns of $\mathbf{B}$, and as such $\mathbf{B}$ is commonly referred to as a rank-R NMF basis, though neither *rank* nor *basis* carries its standard linear algebra meaning. As reflected by the estimation error term $\mathbf{E}$, in most cases $\mathbf{BG}$ approximates rather than factorizes $\mathbf{A}$, and as such, what is referred to here as NMF is also referred to as non-negative matrix approximation or approximate NMF, amongst others terms, in the literature.

The process of estimating $\mathbf{B}$ and $\mathbf{G}$ is an optimization problem, the goal of which is to minimize some cost function, $C(\cdot)$, with respect to $\mathbf{G}$ and $\mathbf{B}$, subject to a nonnegativity constraint on $\mathbf{G}$ and $\mathbf{B}$, which can be expressed generally as,

$$\{\mathbf{B},\mathbf{G}\} = \arg\min_{\mathbf{B},\,\mathbf{G}} C(\mathbf{A};\mathbf{B},\mathbf{G}) \quad \text{subject to } \mathbf{B},\mathbf{G} \geq \mathbf{0}. \tag{3.2}$$

The process of estimating $\mathbf{B}$ and $\mathbf{G}$ can also be viewed in a maximum likelihood framework in which some probability distribution for the elements in $\mathbf{E}$ is assumed. Cost functions for NMF are surveyed in section 3.2.1.

The NMF problem does not lend itself to a unique solution, a fact borne out by the following expression,

$$\mathbf{BG} = \mathbf{BSS}^{-1}\mathbf{G}\,. \tag{3.3}$$

where **S** is any nonnegative, monomial, $R \times R$ matrix; the uniqueness of the NMF solution is further discussed in [169] and [170].

In this thesis, we adopt the view that the columns of **B** each contain a feature learned from the data in **A**, and the rows of **G** contain the contribution of each basis vector to each column of **A**; under this interpretation, it can be useful to view the factorizations as the sum of R rank-one matrices, with each matrix corresponding to the outer product of a basis vector of **B** and its corresponding row in **G**. Alternatively, in the BSS literature [171] **A** has also been viewed as containing K observations of F mixtures, each containing a mix of nonnegative sources. In this case, the resulting **B** and **G** are considered to contain a demixing system and estimates of the original sources or components, respectively, up to the indeterminacy highlighted in (3.3).

The non-negativity constraint of NMF means that only additive and not subtractive combinations of the columns of **B** are permitted, which gives rise to a parts-based factorization/approximation of **A** [6]. This contrasts with earlier data decomposition techniques such as Principal Component Analysis (PCA) [172] and ICA [148], which can use cancellations between various components to factorize the original data. NMF is therefore suited to decomposing inherently non-negative data, such as pixel intensities, amplitude spectra or counts data, data for which factors bearing negative values do not have a corresponding physical meaning; indeed, this is what originally motivated NMF [168] (or Positive Matrix Factorization, as it was referred to initially). The decomposition of data by NMF also better agrees with the intuitive notion of building the whole by the sum of the parts, a point argued in the influential paper [6] with reference to the NMF of a database of aligned vectorized facial images, where each basis vector, reconstituted as an image, expressed a localized and physically meaningful facial feature; which contrasted with classical methods such as PCA which produced holistic features.

It was later shown [173], with reference to decompositions of non-aligned vectorized facial images, that NMF generally decomposes **A** into holistic features rather than localized features as reported in [6]; although, it was demonstrated that localized parts can be obtained in this context by imposing auxiliary constraints on **B** and/or **G**, such as sparseness, orthogonality, or smoothness constraints [173]. Imposing auxiliary constraints such as sparsity and smoothness on **B** and/or **G** is also recommended in [174], to generate basis vectors containing more interesting or physically meaningful features of **A**, and to constrain the solutions to the NMF problem and thus increase their uniqueness. Some typical auxiliary constraints for NMF are described in section 3.2.2.

Some links have been established between NMF and data clustering. It was shown in [175] that by imposing an orthogonality constraint on **G** i.e. $\mathbf{GG}^{T} = \mathbf{I}$ (in a least squares cost formulation of NMF, defined below), which, owing to the non-negative constraint, implies

that only one element in each column of **G** can be non-zero, results in a vector quantization of the columns of **A** equivalent to that obtained by *k*-means clustering. This relationship gives rise to an interpretation of NMF as performing soft clustering of data, or relaxed *k*-means clustering of data [176]. A similar link between a symmetric NMF decomposition of **A**, i.e. $\mathbf{A} \approx \mathbf{B}\mathbf{B}^T$, and spectral clustering has also been established [177]. NMF with a Kullback-Leibler divergence was shown in [178, 179] to optimize the same cost function as Probability Latent Semantics Analysis (PLSA) [180], which was devised to identify related terms amongst word count data to indicate the latent topics in text data.

Hybridized versions of NMF and some of the aforementioned algorithms have been proposed, such as non-negative PCA [181] and non-negative ICA [182]. Semi-NMF has also been proposed [183] where **B** is allowed to take on negative values, and convex NMF, where only convex combinations of the columns of **B** are allowed [183]. In addition, NMF has also been investigated when there is a nonlinear relationship between the data and the factorization i.e. $\mathbf{A} = h(\mathbf{BG})$, where h is some element wise non-linear function of its matrix argument [184].

NMF and its extensions have been applied to datasets in a wide range of application domains. In data analysis applications, a low rank NMF of the data is typically computed such that some underlying features or structure is captured in the basis vectors. For instance, in envirometrics, NMF has been used to analyze the chemical concentrations found in various substances so that the underlying latent sources and their contribution to the substances can be identified [185]. In image processing, NMF has been used for facial recognition in [186] [173], and facial expression recognition in [187]. Also in image processing, 2-D variants of NMF have been devised to identify shift invariant features of natural images for object classification tasks [188]. Like PLSA, NMF has been applied to text processing or data mining to find sets of related terms from word count data to enable identification of the latent topics in a collection of documents [189, 190]. NMF has been applied to financial data to discover latent variables underlying fluctuations in stock prices [191, 192]. A fuller review of the applications of NMF is available in [167].

NMF and its many variants have found a multitude of applications in audio signal processing. As demonstrated in [193], NMF performs an automatic scene analysis of the spectrogram of an audio signal, with the basis vectors typically capturing interesting spectral features that are said to correspond to meaningful auditory *objects*, and the rows of **G** express their temporal extent. This property has engendered NMF to such applications as audio event classification [194, 195], automatic note transcription [196, 197], and musical instrument classification [198], tasks where the discriminative spectral features of each auditory object, i.e. event, note and instrument, can be embedded in a low-rank basis of labeled data. NMF also provides a perceptually relevant decomposition of audio [199]. Specifically, it was shown in [199] that the low rank NMF of the magnitude spectrogram of speech signals yields a filter-bank design with remarkable similarities to perceptually supported designs such as the mel-scale filter-bank. Some other audio applications of NMF include, automatic fundamental

frequency estimation in [200], reverberated speech recognition in [201], and robust speech dereverberation in [202].

As expressed earlier, NMF and variants of NMF have been employed extensively to perform MSSS within the model-based framework. This application stems from its ability to capture pertinent features of the audio spectrogram of individual sound sources in a dictionary or basis [203]. The various model-based MSSS algorithms based on NMF and its variants will be reviewed in section 3.3.4; model-based MSSS based on probalistic techniques are reviewed beforehand in section 3.3.3. In the next sections, we describe various aspects of NMF in more detail. Specifically, in section 3.2.1, we review cost functions that have been proposed for NMF; in section 3.2.2 auxiliary constraints are discussed, and in section 3.2.3 extensions to the basic NMF model are described. Finally, in section 3.2.4, we describe a number of iterative optimization schemes that have been proposed for computing NMF. The description of NMF in the following sections is biased towards it application to model-based MSSS. For a more general treatment of NMF and its applications see [167, 174].

3.2.1 Cost Functions for NMF

Before computing the NMF, it is necessary to select a cost function, $C(\cdot)$, to measure the quality of the approximation of $\mathbf{A}$ by $\mathbf{BG}$. In a maximum likelihood context, this is equivalent to selecting a distribution for the approximation error $\mathbf{E}$. A variety of cost functions have been proposed for the NMF problem, with each producing a different factorization of $\mathbf{A}$, each being optimal for a certain distribution of the elements in $\mathbf{E}$; the choice of cost function therefore is data dependent. In this section, we review a selection of such cost functions and discuss some of their respective properties; in particular, in relation to audio. To begin, we note that each of these cost functions share the following properties; continuosly differentiable (at least once) in $\mathbf{B}$ and $\mathbf{G}$ individually; convex in $\mathbf{B}$ and $\mathbf{G}$ individually i.e. for a fixed $\mathbf{B}$ the cost function is convex in $\mathbf{G}$ and visa versa; and are positive, being equal to $\mathbf{0}$ only when $\mathbf{A} = \mathbf{BG}$, such that they are indeed indicative of the quality of the approximation.

Perhaps the best-known cost function for NMF measures the Squared Euclidean Distance (SED) between $\mathbf{A}$ and $\mathbf{BG}$, which corresponds to the assumption that $\mathbf{E}$ is independently and identically distributed (i.i.d) additive Gaussian noise. This cost function, $C_{\mathrm{SED}}(\cdot)$, is given as [204],

$$C_{\mathrm{SED}}(\mathbf{A}, \mathbf{BG}) = \| \mathbf{A} - \mathbf{BG} \|_{fro}^2, \tag{3.4}$$

where the subscript *fro* denotes frobenius norm. $C_{\mathrm{SED}}(\cdot)$ is metrically symmetric [205], that is, $C_{\mathrm{LS}}(\mathbf{A}, \mathbf{BG}) = C_{\mathrm{LS}}(\mathbf{BG}, \mathbf{A})$, and thus, weighs under- and over-estimation of $\mathbf{A}$ by $\mathbf{BG}$ equally.

Another well-known NMF cost function is a measure of the divergence of $\mathbf{BG}$ from $\mathbf{A}$, similar in its functional form to the Kullback-Leibler divergence [204]. This cost function, $C_{\mathrm{KL}}(\cdot)$, corresponds to ML estimation in Poisson noise [206], and is given by,

$$C_{\mathrm{KL}}(\mathbf{A} || \mathbf{BG}) = \| \mathbf{A} \circ \ln\left[\frac{\mathbf{A}}{\mathbf{BG}}\right] - \mathbf{A} + \mathbf{BG} \|, \tag{3.5}$$

where $\circ$ is the Hadamard product such that $(\mathbf{Q} \circ \mathbf{Z})_{ij} = \mathbf{Q}_{ij}\,\mathbf{Z}_{ij}$, and division is similarly element-wise. Unlike $C_{\mathrm{SED}}(\cdot)$, divergence measures such as $C_{\mathrm{KL}}(\cdot)$ are generally metrically asymmetric [205], that is, $C_{\mathrm{KL}}(\mathbf{A}, \mathbf{BG}) \neq C_{\mathrm{KL}}(\mathbf{BG}, \mathbf{A})$. The significance of this is that for $C_{\mathrm{KL}}(\cdot)$ a reference argument must be specified beforehand. Another consequence is that $C_{\mathrm{KL}}(\cdot)$ penalizes under-estimation of $\mathbf{A}$ by $\mathbf{BG}$ more than over-estimation, with concomitant effects for the resulting factorization.

Another cost function for NMF that measures the Itakura-Saito distance between $\mathbf{A}$ and $\mathbf{BG}$, corresponding to ML estimation in multiplicative gamma noise [206], is given by,

$$C_{\mathrm{IS}}(\mathbf{A} \,||\, \mathbf{BG}) = \sum \left(\frac{\mathbf{A}}{\mathbf{BG}} - \ln\left[\frac{\mathbf{A}}{\mathbf{BG}}\right] - 1 \right). \tag{3.6}$$

Like $C_{\mathrm{KL}}(\cdot)$, $C_{\mathrm{IS}}(\cdot)$ is metrically asymmetric and penalizes under-estimation of $\mathbf{A}$ by $\mathbf{BG}$ more than over-estimation. The Itakura-Saito distance measure is well known in the speech processing literature, where it is used as a perceptually relevant speech distortion measure between spectra, making $C_{\mathrm{IS}}(\cdot)$ appealing from a speech processing perspective. As pointed out in [206], $C_{\mathrm{IS}}(\cdot)$ also has the property of scale invariance, that is, low-energy regions of $\mathbf{A}$ bear the same relative importance as high energy regions. It was also shown in [206] that NMF with $C_{\mathrm{IS}}(\cdot)$ tends to discover more semantically relevant spectral features in music audio spectrograms than NMF with either the $C_{\mathrm{SED}}(\cdot)$ or $C_{\mathrm{KL}}(\cdot)$ objectives.

The preceding cost functions are all generalized by the β divergence [207], given as,

$$C_{\beta}(\mathbf{A} \,||\, \mathbf{BG}) = \sum_{i,j} \mathbf{A}_{i,j} \frac{\mathbf{A}_{i,j}^{\beta-1} - (\mathbf{BG})_{i,j}^{\beta-1}}{\beta(\beta-1)} - (\mathbf{BG})_{i,j}^{\beta-1} \frac{(\mathbf{BG})_{i,j} - \mathbf{A}_{i,j}}{\beta}\,, \tag{3.7}$$

where $\beta = 2$ gives $C_{\mathrm{SED}}(\cdot)$, $\beta \to 1$ tends to $C_{\mathrm{KL}}(\cdot)$, and for $\beta \to 0$ tends to $C_{\mathrm{IS}}(\cdot)$. By varying the parameter β a wide range of different divergences for NMF are attained.

The β divergence is in turn generalized by the Bregman divergences [208], which were proposed for NMF in [184]. The Bregman divergence for any continuosly-differentiable strictly convex function, $\Phi(\cdot)$, is,

$$C_{\mathrm{B}}(\mathbf{A} \,||\, \mathbf{BG}) = \sum_{i,j} \Phi(\mathbf{A}_{i,j}) - \Phi((\mathbf{BG})_{i,j}) - \nabla\Phi((\mathbf{BG})_{i,j})(\mathbf{A} - \mathbf{BG})_{i,j}\,, \tag{3.8}$$

which corresponds to the beta divergences for $\Phi(x) = x^{\beta\text{-}1}$. The Bregman divergences generalize a wide variety of cost functions; indeed, there is a one-to one correspondence between the Bregman divergences and the exponential family of probability distributions, which encompass a wide variety of possible distributions for the elements in $\mathbf{E}$ [167]. A host of other generalizing NMF cost functions were introduced in [171], [209], and [210].

In relation to audio applications, in which NMF is typically applied to spectrogram data, cost functions can be chosen based on their perceptual relevance. Since the human auditory system has asymmetric sensitivity to energy in the spectral domain due to the effects of masking [9], it is reasonable to assume, as suggested in [205], that metrically asymmetric cost functions, with similar error weighting characteristics to that of the human ear i.e. weighting over-estimation less than under-estimation, are more suitable for audio spectrogram

factorization than symmetric cost functions such as $C_{SED}(\cdot)$. This suggestion was explored in [205], in which the quality of the reconstructions of speech signals produced by NMF for a range of different cost functions, each corresponding to instances of the β-divergence (including $C_{SED}(\cdot)$, $C_{KL}(\cdot)$, and $C_{IS}(\cdot)$), were compared using a perceptually relevant objective measure of speech quality; the cost functions where optimized using their associated multiplicative updates, discussed in section 3.2.4. A novel cost function constrained by the masking patterns of the input signal was also proposed in [205]. In relation to the special cases of $C_{SED}(\cdot)$, $C_{KL}(\cdot)$, and $C_{IS}(\cdot)$; it was found that the $C_{KL}(\cdot)$ cost function produced superior speech reconstructions throughout the experiment, i.e. for different numbers of basis vectors; despite, for instance, the relevance of $C_{IS}(\cdot)$ to speech processing. Based on this result, it was concluded that $C_{KL}(\cdot)$ may be the most suitable candidate for audio processing applications. The error weighting of a generalizing NMF cost function and how this relates to perception is also discussed in [211], and in [212], where the $C_{KL}(\cdot)$ and $C_{SED}(\cdot)$ are compared qualitatively, and where $C_{KL}(\cdot)$ is also deemed more suitable.

3.2.2 Auxiliary Constraints for NMF

As expressed in the introduction to NMF, in certain applications it may be desirable to impose auxiliary constraints on the factors **B**, **G**, or both, to influence the NMF of **A**, or to steer **B** and **G** towards a more unique factorization. One common method is to extend the chosen cost function with penalty terms, expressed generally as,

$$C_{\ell}(\mathbf{A};\mathbf{B},\mathbf{G},\ell_{\mathbf{B}},\ell_{\mathbf{G}}) = C(\mathbf{A};\mathbf{B},\mathbf{G}) + \ell_{\mathbf{B}} l_{\mathbf{B}}(\mathbf{B}) + \ell_{\mathbf{G}} l_{\mathbf{G}}(\mathbf{G}), \tag{3.9}$$

where $\ell_{\mathbf{B}}$ and $\ell_{\mathbf{G}}$ are positive regularization parameters that control the trade-off between the influence of the constraint on the resulting factorization and estimation error, and where $l_{\mathbf{B}}$ and $l_{\mathbf{G}}$ are functions, assumed differentiable, that measure the desired feature of **G** and **B**. In probabilistic terms, these penalty terms correspond to some prior distribution for the factors **B** and **G**, and thus, finding the minimum of (3.9) corresponds to Maximum A Posteriori (MAP) estimation.

Sparseness is an oft-desired feature of data decompositions [213]. As described earlier, transforming a dataset into a representation in which its energy is sparsely distributed implies a compaction of its energy into a small number of domain coefficients. In terms of matrix factorization, sparseness corresponds to projecting **A** onto a basis such that the energy in the coefficient matrix is sparsely distributed. Although NMF already typically produces sparse activation patterns in **G**, in certain applications an explicit, controllable sparsity constraint is useful; for example, to prevent over-fitting when learning an over-complete basis from **A** [213, 214]. In [215], for what is referred to as Non-Negative Sparse Coding (NNSC), the L_1 norm, i.e. $l_{\mathbf{G}}(\cdot) = |\cdot|$, is used to measure sparsity, to regularize the $C_{SED}(\cdot)$ cost function; this corresponds to the prior assumption that the elements in **A** are i.i.d. one-sided exponential [167]. The NNSC cost function is given as,

$$C_{NNSC}(\mathbf{A}, \mathbf{BG}) = \| \mathbf{A} - \mathbf{BG} \|_{fro}^2 + \ell_{\mathbf{G}} \sum_{i,j} \left| \mathbf{G}_{i,j} \right| . \tag{3.10}$$

To prevent the trivial satisfaction of the penalty term, by up scaling **B** and down scaling **G**, the columns of **G** were constrained to be unit norm. A similar cost function was proposed in [216] for what is referred to as Sparse NMF (SNMF), where the scaling constraint, applied to the columns of **B**, was built into the regularized cost function. In [217], sparsity is measured based on a relation between the L_1 and L_2 norm of the columns of **G**. Different measures of sparsity, each regularizing the $C_{\text{SED}}(\cdot)$ cost function, are compared in [218]. Various desirable attributes of different measures of sparsity are explored in [219].

Another feature of data decomposition is smoothness, which can be desirable when the data is received with noise, or when the data is slowly time varying. Smoothness entails that the temporal information encoded in **G** is smooth. One measure of smoothness or temporal continuity for NMF that was proposed in [220] is,

$$l_{\mathbf{G}}(\mathbf{G}) = \sum_{i,j} \left| \mathbf{G}_{i,\,j-1} - \mathbf{G}_{i,\,j} \right|, \tag{3.11}$$

which requires scale constraints to prevent trivial minimization. Other proposed measures of smoothness include; the averaged squared difference between successive columns of **G** [212], and comparing **G** to a matrix containing low pass filtered versions of the rows of **G** [221]. Temporal constraints are discussed in more detail in [221].

An orthogonality constraint on **B** is another typical auxiliary constraint. In [173], the following orthogonality constraint was imposed on **B**,

$$l_{\mathbf{B}}(\mathbf{B}) = \sum_{i \neq j} (\mathbf{B}^{\mathrm{T}}\mathbf{B})_{i,\,j}\,. \tag{3.12}$$

Again, scale constraints are required to prevent trivial up scaling of **G** and down scaling of **B**. Relatedly, it was proposed in [222] to compare **B** to a fixed reference basis such that the factorization is steered away from this reference. Orthogonal NMF algorithms are discussed further in [223].

3.2.3 NMF Extensions

It was proposed, independently in [193, 224] and [225], to extend the expressive power of the nonnegative basis **B** by allowing its basis functions to express features of **A** that span $T > 1$ columns. This is achieved by considering the basis **B** as consisting of R, $F \times T$ matrices, referred to as basis functions in [193, 224], rather than R, $F \times 1$ basis vectors for NMF. In [224] the cost function for this approach, which is termed Convolutive NMF (CNMF), is formulated based on the $C_{\text{KL}}(\cdot)$ cost function, and is expressed as,

$$C_{\text{KL}}(\mathbf{A} || \hat{\mathbf{A}}) = \left\| \mathbf{A} \circ \ln\left[\frac{\mathbf{A}}{\hat{\mathbf{A}}}\right] - \mathbf{A} + \hat{\mathbf{A}} \right\|, \tag{3.13}$$

where

$$\hat{\mathbf{A}} = \sum_{t=0}^{T-1} \mathbf{B}(t) \circ \overset{t\rightarrow}{\mathbf{G}}, \tag{3.14}$$

and the $\overset{t\rightarrow}{(\cdot)}$ operator is a shift operator that moves the columns of its argument t columns to the right with vacated columns on the left filled with zeros. The shift operator incorporates the columns of each basis function into the cost function such that they are included in the

subsequent optimization scheme. CNMF was also formulated for the $C_{SED}(\cdot)$ cost function in [226]. CNMF with a sparsity constraint was proposed in [225] and in [227], in which an L_1 norm constraint on **G** is proposed. Two-dimensional CNMF was devised in [174, 228], where the basis functions of CNMF can shift horizontally which is performed using a row shift operator for $\mathbf{B}(t)$, allowing the basis functions to capture features that vary horizontally.

Nonnegative factorization has also been extended to tensors, which is termed as Nonnegative Tensor Factorization (NTF) [174]. The standard NFT model, which is the PARAFAC (Parallel Factor Analysis) model with non-negative constraints, is given as,

$$\mathbf{A}_q = \mathbf{B}\mathbf{D}_q\mathbf{G} + \mathbf{E}_q \quad \text{for } q = 1, 2, \ldots, Q, \tag{3.15}$$

where $\mathbf{A}_q = \underline{\mathbf{A}}_{:,:,q}$ are frontal slices of the nonnegative $F \times K \times Q$ tensor, $\underline{\mathbf{A}}$, Q is the number of frontal slices, $\mathbf{E}_q = \underline{\mathbf{E}}_{:,:,q}$ are frontal slices of the $F \times K \times Q$ residual tensor $\underline{\mathbf{E}}$, $\mathbf{D}_q$ is a diagonal matrix that holds the q^{th} row of the $Q \times R$ nonnegative matrix **D**, and as above **B** and **G** are the basis and coefficient matrices respectively. The aim of NTF is to estimate the factorization in (3.15) given the tensor $\underline{\mathbf{A}}$. Tensors and 2-D CNMF are combined in [229]. Nonnegative Tensor Factorization is thoroughly reviewed in [174].

3.2.4 Algorithms for NMF

Given a cost function, possible augmented with penalty terms, and a data matrix **A**, the NMF problem becomes one of estimating **B** and **G**. As demonstrated by (3.3), a global minimum solution is not obtainable for this problem, so a local minimum solution is sought instead. In this section, we review a selection of iterative optimization schemes that attempt to obtain this solution. Unless stated otherwise, these schemes have a generic algorithmic structure, whereby the factors **B** and **G** are initialized with non-negative random values, and at each iteration the factors are updated once by separate update rules; that is, for each iteration, a factor is fixed while the other is updated by a specific update rule and visa versa before the next iteration. Alternating updates are employed because NMF cost functions are convex in **B** and **G** individually, but not jointly [204], such that one factor may be optimized with respect to the other factor. The number of iterations for these schemes is usually prescribed, or the trajectory of the cost function can be monitored for convergence; systematic stopping criteria for NMF algorithms for the $C_{SED}(\cdot)$ cost function are outlined in [230] and [231]. It is recommended in [232] to run the NMF scheme a number of times for different initializations of **B** and **G** to obtain the best solution; especially for large-scale problems. A wide range of NMF algorithms have been implemented in Matlab and have been collated in the NMFLAB toolbox [233]. The test bench accompanying NMFLAB is often used for comparing various NMF algorithms with different cost functions and constraints, and consists of a nonnegative BSS problem comprised of mixtures of nonnegative sources, such that the algorithms are assessed under the BSS interpretation of **B** and **G** discussed above.

The best known scheme for finding **B** and **G** is via gradient descent by multiplicative updates [204]. The multiplicative update approach exploits the fact that multiplying any two nonnegative values produces another non-negative value. Therefore, by initializing the

elements of **G** and **B** to non-negative values and given a nonnegative **A**, the non-negativity constraint is imposed by applying multiplicative updates to **B** and **G**; this also implies however that if an element of either factor is assigned the value zero during the procedure it remains at zero. An appealing feature of this approach is that it is applicable to a wide variety of cost functions. For generality, we exemplify this approach for the β-divergence [207]; a similar treatment can be applied to the Bregman divergences [184]. The partial derivatives of the β-divergence with respect to **G** and **B** are, respectively,

$$\nabla_{\mathbf{G}} C_{\beta}(\mathbf{A} \,||\, \mathbf{BG}) = \mathbf{B}^{\mathrm{T}}(\mathbf{BG})^{\beta-1} - \mathbf{B}^{\mathrm{T}}((\mathbf{BG})^{\beta-2} \circ \mathbf{A}), \tag{3.16}$$

$$\nabla_{\mathbf{B}} C_{\beta}(\mathbf{A} \,||\, \mathbf{BG}) = (\mathbf{BG})^{\beta-1}\mathbf{G}^{\mathrm{T}} - ((\mathbf{BG})^{\beta-2} \circ \mathbf{A})\mathbf{G}^{\mathrm{T}}. \tag{3.17}$$

The multiplicative update rules for **G** and **B** can be constructed by the ratio of the negative and positive terms of the partial derivatives in (3.16) and (3.17) respectively, giving the following multiplicative update rules,

$$\mathbf{G} \leftarrow \mathbf{G} \circ \frac{\mathbf{B}^{\mathrm{T}}((\mathbf{BG})^{\beta-2} \circ \mathbf{A})}{\mathbf{B}^{\mathrm{T}}((\mathbf{BG})^{\beta-1})}, \tag{3.18}$$

$$\mathbf{B} \leftarrow \mathbf{B} \circ \frac{((\mathbf{BG})^{\beta-2} \circ \mathbf{A})\mathbf{G}^{\mathrm{T}}}{((\mathbf{BG})^{\beta-1})\mathbf{G}^{\mathrm{T}}}. \tag{3.19}$$

A small positive regularization value is usually added to the denominator of each update to prevent to prevent an element of either factor from being fixed at zero thereby preventing division by zero. Furthermore, in some treatments of NMF the columns of **B** are normalized to unit norm at each iteration, which is reported to increase the uniqueness of the resulting solution [234]; this, in general, is optional for all NMF algorithms.

A key benefit of the multiplicative updates is the absence of step size tuning; however, they can also be viewed as conventional (additive) gradient descent updates that have the following stepsizes,

$$\mu_{\mathbf{G}} = \frac{\mathbf{G}}{\mathbf{B}^{\mathrm{T}}((\mathbf{BG})^{\beta-1})}, \tag{3.20}$$

$$\mu_{\mathbf{B}} = \frac{\mathbf{B}}{((\mathbf{BG})^{\beta-1})\mathbf{G}^{\mathrm{T}}}, \tag{3.21}$$

where, $\mu_{\mathbf{G}}$, $\mu_{\mathbf{B}}$ are the step sizes for the **G** and **B** factors respectively.

The convergence of $C_{\beta}(\mathbf{A} \,||\, \mathbf{B}, \mathbf{G})$ for $1 \leq \beta \leq 2$ under these update rules was analyzed in [207], where it is proved that $C_{\beta}(\mathbf{A} \,||\, \mathbf{B}, \mathbf{G})$ for $1 \leq \beta \leq 2$ is non-increasing under these update rules. This convergence property does not imply convergence to a stationary point, which is a necessary condition for convergence to a local minimum; the convergence of the multiplicative updates is discussed further in [189]. For the specific case of $C_{\mathrm{SED}}(\cdot)$ cost function it was shown in [235] that by using the stepsizes defined in (3.20) and (3.21) in a conventional (additive) gradient descent algorithm with a small positive value explicitly added to the denominator of $\mu_{\mathbf{G}}$ and $\mu_{\mathbf{B}}$, convergence to a stationary point is guaranteed. For the case of $C_{\mathrm{KL}}(\cdot)$ i.e. $\beta \rightarrow 1$ a link between the multiplicative updates and the expectation maximization algorithm was developed and convergence analysis was analyzed in that

context in [236]. The convergence of the multiplicative updates of **G** for the special case of a fixed **B** was analyzed in [237] for $1 \leq \beta \leq 2$.

Owing to their applicability to a wide range of different cost functions and their intuitiveness, the multiplicative updates are routinely employed to perform NMF, and have been applied to a vast range of different datasets. Given this diversity, it is difficult to generalize their performance. However, it has been reported by some researchers that the multiplicative updates converge relatively slowly [189, 232], especially when performed on dense matrices, and that they tend to get stuck in poor local minima [232], particularly for large scale problems. It was reported in [230] however that their computational load requirement is low relative to other techniques.

Modifications have been proposed to the multiplicative approach. In [238], it was demonstrated that by scaling $\mu_{\mathbf{G}}$ and $\mu_{\mathbf{B}}$ by a value greater then one for the additive gradient descent version of the multiplicative updates, such that **G** and **B** are updated further along the direction of the negative gradient at each iteration, the convergence of the resultant algorithm is accelerated compared to the regular multiplicative updates. Similarly, it has been suggested in [237] to employ an exponent to each update to control the convergence rate, which can be similarly accelerated if the value of the exponent is greater than one. This approach was also proposed for the multiplicative updates in [239] to aid in the sparsification of the factors.

Multiplicative update rules were also proposed for CNMF with $C_{\mathrm{KL}}(\cdot)$ in [224] and with $C_{\mathrm{SED}}(\cdot)$ in [228], for each of which effectively T NMF sub-problems are solved. In [224], at each iteration of CNMF, each NMF sub-problem generates an update for $\mathbf{B}(t)$ and a **G** update. Since the **G** matrix is shared by each of the T columns of each basis function, the average of the T updates is applied to **G**. Similar updates were derived for 2-D CNMF in [240].

A number of NMF optimization schemes have been proposed specifically for the $C_{\mathrm{SED}}(\cdot)$ cost function. Alternating Nonnegative Least Squares (ANLS) [241] is one general approach. For ANLS, methods for finding the least squares solution to a system of linear equations under a non-negative constraint, known as Non-Negative Least Squares (NNLS), are employed to find the **G** and **B**. ANLS has the property of guaranteed convergence to a stationary point of $C_{\mathrm{SED}}(\cdot)$, which is a stronger convergence property than the multiplicative updates for $C_{\mathrm{SED}}(\cdot)$, but conventional NNLS methods are slow when straightforwardly applied to typical NMF problems [168, 242]. To improve the computational efficiency of ANLS while preserving its appealing convergence property, a number of NNLS methods have been customized for ANLS, for instance in [231], and in [242].

Traditional techniques for unconstrained optimization with modifications for nonnegative constraints have also been investigated for NMF with the $C_{\mathrm{SED}}(\cdot)$ cost function. One such technique is Projected Gradient Descent (PGD) [230]. Basic PGD updates, i.e. with a basic projection step to deal with unfeasible values of **B** and **G**, can be expressed as,

$$\mathbf{G} \leftarrow \max[\mathbf{0},\ \mathbf{G} - \mu_{\mathbf{G}}[\nabla_{\mathbf{G}} C_{\mathrm{SED}}(\mathbf{A} \,\|\, \mathbf{BG})]], \tag{3.22}$$

$$\mathbf{B} \leftarrow \max[\mathbf{0}, \mathbf{B} - \mu_{\mathbf{B}}[\nabla_{\mathbf{B}} C_{\text{SED}}(\mathbf{A} \,||\, \mathbf{BG})]]. \tag{3.23}$$

It is reported in [189], that for random initializations of **B** and **G** for a fixed stepsize, these updates converge to a factorization that is not very far from the original matrices; formal proof of convergence is difficult to prove given the nonlinear projection step. Their exist however, several strategies for choosing the optimal values for $\mu_{\mathbf{G}}$ and $\mu_{\mathbf{B}}$ at each iterative step such that the cost function is minimized along a computed negative gradient direction and the non-negativity of the factors is preserved; in many cases, convergence to a stationary point is guaranteed [230]. Specific approaches include the ARMJO method in [230], the interior point method [239], a fast Newton-type PGD in [243]; PGD for large scale NMF problems is addressed in [244].

Alternating Least Squares (ALS) is another method for estimating **B** and **G** for the $C_{\text{SED}}(\cdot)$ cost function [189]. The ALS updates rule is obtained by setting the partial derivatives of $C_{\text{SED}}(\cdot)$ with respect to **G** and **B** to zero, and then solving for **G** and **B**, with a basic projection step for negative values. The ALS updates are expressed as follows,

$$\mathbf{G} \leftarrow \max[\mathbf{0}, (\mathbf{B}^{\mathrm{T}}\mathbf{B})^{-1}(\mathbf{B}^{\mathrm{T}}\mathbf{A})], \tag{3.24}$$

$$\mathbf{B} \leftarrow \max[\mathbf{0}, (\mathbf{G}\mathbf{A}^{\mathrm{T}})(\mathbf{G}\mathbf{G}^{\mathrm{T}})^{-1}]. \tag{3.25}$$

Unlike ANLS, the projection step of ALS makes formal convergence analysis of this approach non-trivial, which is discussed further in [189], though the convergence of the ALS algorithm is observed in practice [189].

The matrix inversions of the ALS updates can be problematic; the underlying matrices may be ill conditioned, and inversion is a computational expensive operation. An alternative approach based on ALS that does not entail a matrix inversion is to minimize R local least square cost functions instead of the global cost function [245]. These cost functions are given as,

$$C^{r}_{\text{SED}}(\mathbf{A}^{(r)}, \mathbf{b}_r\mathbf{g}_r) = \frac{1}{2}\left\|\mathbf{A}^{(r)} - \mathbf{b}_r\mathbf{g}_r^{\mathrm{T}}\right\|_F^2 \quad \text{for } r = [0, 1, \ldots R\text{–}1], \tag{3.26}$$

where $\mathbf{b}_r$ is the r^{th} column of **B**, $\mathbf{g}_r^{\mathrm{T}}$ is the corresponding row of **G**, and

$$\mathbf{A}^{(r)} = \mathbf{A} - \sum_{j \neq r} \mathbf{b}_j\mathbf{g}_j^{\mathrm{T}}. \tag{3.27}$$

Each sub-problem corresponds to computing a distinct column of **B** and a corresponding row of **G** respectively, which corresponds to the interpretation of **BG** as being composed of the summation of R rank 1 matrices. These R local cost functions are minimized successively by alternating least squares updates, which are derived similarly to those for ALS. This approach is referred to as Hierarchical Alternating Least Squares (HALS) [245].

Various ANLS techniques were experimentally compared in [242] to both ALS and multiplicative updates for the $C_{\text{SED}}(\cdot)$, for the NMF of synthetic data. The results of the study show that ANLS takes significantly less time to converge while achieving the same or lower value of the cost function as both ALS and the multiplicative updates, which have comparable performance. In [246], HALS and multiplicative NMF for $C_{\text{SED}}(\cdot)$ where compared, and they

were shown to have approximately the same computational load in terms of floating point operations, but HALS was demonstrated to have faster convergence speed, especially for dense matrices. To improve the performance of both algorithms, it is proposed in [246] that for each iterative step **B** should receive more than one update, i.e. **B** should have more inner iterations than **G**, which was demonstrated on text and facial image data to result in faster convergence for HALS, the multiplicative updates, and PSD algorithms. This modification was also shown to significantly reduce the disparity in performance between the ANLS methods and the multiplicative updates for $C_{\mathrm{SED}}(\cdot)$ and HALS.

Another scheme for NMF, which is related to Simultaneous Multiplicative Algebraic Reconstruction Techniques (SMART), that is applicable to a wide variety of cost functions was proposed in [234]. For this approach, **B** and **G** are estimated using multiplicative Exponentiated Gradient (EG) descent updates, given for $C_{\mathrm{KL}}(\cdot)$ as,

$$\mathbf{G} \leftarrow \mathbf{G} \circ \exp\left(\mu_{\mathbf{G}} \circ \left(\mathbf{B}^{\mathrm{T}} \mathrm{In}\left(\frac{\mathbf{A}}{\mathbf{BG}}\right)\right)\right), \tag{3.28}$$

$$\mathbf{B} \leftarrow \mathbf{B} \circ \exp\left(\mu_{\mathbf{B}} \circ \left(\mathrm{In}\left(\frac{\mathbf{A}}{\mathbf{BG}}\right)\mathbf{G}^{\mathrm{T}}\right)\right). \tag{3.29}$$

Like the multiplicative NMF update framework, the multiplicative EG descent NMF approach was applied to a host of different, typically generalizing, cost functions in [234]. Compared to the multiplicative updates, the EG updates were demonstrated to produce sources with better fidelity for the benchmark data provided in [233].

Newton updates have also been applied to the NMF problem for a wide variety of cost functions [245, 247]. Incorporating a projection step, the general form of the Newton update for NMF may be expressed as,

$$\mathbf{G} \leftarrow \max[0, \mathbf{G} - [[\nabla_{\mathbf{G}}^2 C(\mathbf{A} \,||\, \mathbf{BG})]^{-1} \nabla_{\mathbf{G}} C(\mathbf{A} \,||\, \mathbf{BG})]], \tag{3.30}$$

$$\mathbf{B} \leftarrow \max[0, \mathbf{B} - [[\nabla_{\mathbf{B}}^2 C(\mathbf{A} \,||\, \mathbf{BG})]^{-1} \nabla_{\mathbf{B}} C(\mathbf{A} \,||\, \mathbf{BG})]]. \tag{3.31}$$

The Hessian matrices in these updates i.e. the second derivatives, may need to be regularized to mitigate the effects of ill-conditioning, which leads to what is known as Quasi-Newton updates [248]. The Quasi-Newton NMF updates generally produce fast convergence for a relatively high computational load. As stated in [232] however, it may be necessary to steer the solutions away from local minima's of the cost function to produce a useful solution.

A multilayer framework for computing the NMF was presented in [249] that aims to mitigate the problems of ill conditioning and poor local minima that beset NMF in certain applications. For this approach **A** is factorized through L successive NMF decompositions, which can be computed using any of the above-discussed methods. For the first NMF, **A** is factorized as above to produce $\mathbf{A} \approx \mathbf{B}^{(1)}\mathbf{G}^{(1)}$, where the factors $\mathbf{B}^{(1)}$ and $\mathbf{G}^{(1)}$ have the same dimensions as **B** and **G** above. Then for the next NMF, the gain matrix from the first decomposition, $\mathbf{G}^{(1)}$, is decomposed to yield $\mathbf{G}^{(1)} = \mathbf{B}^{(2)}\mathbf{G}^{(2)}$, where $\mathbf{B}^{(2)}$ is a $R \times R$ nonnegative matrix, and so on until $\mathbf{G}^{(L)} = \mathbf{B}^{(L)}\mathbf{G}^{(L)}$; the factors are initialized with nonnegative random values each time. The resulting factorization can then be expressed as,

$$\mathbf{A} = (\mathbf{B}^{(1)}\mathbf{B}^{(2)}\ldots\mathbf{B}^{(L)})\mathbf{G}^{(L)} + \mathbf{E}. \quad (3.32)$$

By multiplying the consecutive **B** terms a conventional bi-linear NMF approximation of **A** is attained. The multiplayer procedure for computing the NMF has been demonstrated to significantly improve the fidelity of the source separations achieved by various NMF optimization schemes under various cost functions and auxiliary constraints [234, 239, 245, 250], for the benchmark data provided in [233].

NMF under the Itakura-Saito based cost function, i.e. $C_{IS}(\cdot)$, was analyzed in a statistical framework in [251]. From this analysis, a new optimization procedure based on SAGE (space–alternating generalized expectation-maximization), guaranteeing convergence to a stationary point, was proposed. This SAGE-NMF approach was demonstrated to converge slower than the multiplicative updates for $\beta \rightarrow 1$, with both algorithms having comparable computational load. A benefit of SAGE-NMF however is its underlying statistical framework, which allows for priors on **B** and **G**, with which maximum a posteriori estimation can be performed. Relatedly, a Bayesian treatment of NMF was given in [252], in which the likelihood function assumed the elements of **E** are i.i.d. zero mean normally distributed, and exponential priors are assumed for **B** and **G**. An efficient Gibbs sampler was then derived to estimate the posterior densities of **B** and **G**. This approach was evaluated on chemical shift image data and was demonstrated to have a faster convergence rate, and to reach a lower value of the objective function, than the corresponding multiplicative NMF update algorithm.

A number of techniques for initializing the factors **B** and **G** before the NMF optimization procedure commences such that more unique or better decompositions are obtained have been proposed; see [253] for a review. Typically these approaches involve populating **B** with values extracted from **A** before the NMF is performed, and are particularly useful for large data sets [232]. One approach [254] involves performing a spherical k-means clustering of the columns of **A** to compute R centroids, which then initialize the columns of **B** resulting in faster initial convergence. Noting the computational cost of spherical k-means, a number of efficient initialization schemes were proposed in [255]; for example, each column of **B** is populated with an average of a random sampling of the columns of **A**. In [256] three initialization techniques based on PCA, fuzzy clustering, and Gabor wavelet functions are proposed and evaluated, while in [257] an ICA based initialization scheme is proposed and is demonstrated to allow for better performance than comparable techniques. In addition, a SVD based initialization approach is described in [258], and it is proposed in [259] to use a genetic algorithm based approach to enhance initialization.

Most of the above NMF optimization techniques can be customized to deal with auxiliary constraints on **B** and/or **G**, with the most common modification being scaling constraints to mitigate trivial satisfaction of the constraints; a general discussion about this topic can be found in [171, 189] for the multiplicative updates, in [191] for EG updates, and in [250] for ALS. In particular, NMF with an auxiliary sparsity constraint on **G** has been frequently addressed in the literature. In [215], it is proposed to minimize the NNSC cost function, defined in (3.10), using a multiplicative update rule for **G** and a PGD update for **B**,

which involves a rescaling of the columns of **B** to unit norm to prevent trivial satisfaction of the sparsity constraint. By using a PGD update for **B**, the cost function is guaranteed to be non-increasing for each update. For CNMF with a sparsity constraint in [227], it is also proposed to employ a multiplicative update for **G** and a PSD update for **B**(t), again preserving the non-decreasing convergence property. In contrast, the sparse NMF (SNMF) algorithm proposed in [216] uses multiplicative updates for both **B** and **G**. While the convergence of this algorithm was observed in practice, a formal proof of a non-decreasing cost function is not provided. Similarly in [260], multiplicative updates rules are derived for Sparse CNMF, again without a formal proof and with convergence observed in practice. Sparse ANLS is proposed in [261] with a convergence property to a stationary point guaranteed. In [218] a number of algorithms for various different sparsity regularizations of the $C_{\text{SED}}(\cdot)$ cost function are evaluated. Sparsity constraints and PSD are discussed in [262].

3.3 Monaural Sound Source Separation

The objective of Monaural Sound Source Separation (MSSS) is to estimate an arbitrary number of sound source signals from a single mixture, $y(n)$, of these sources. In this chapter, we restrict ourselves, without loss of generality, to two sources, which are denoted as $x_1(n)$ and $x_2(n)$. Taking $y(n)$ to be the linear superposition of the sources gives,

$$y(n) = a_1 x_1(n) + a_2 x_2(n) + w(n), \tag{3.33}$$

where it is assumed that $a_1 = a_2 = 1$, and where $w(n)$ denotes independently and identically distributed (i.i.d) noise, distributed according to some Probability Distribution Function (PDF), an example of which is specified in section 3.3.2. Before proceeding to describe model-based MSSS, we first give an overview of the source-driven approach of CASA.

3.3.1 Computational Auditory Scene Analysis

The underdetermined nature of MSSS coupled with the lack of spatial information has meant that computational MSSS is considered a relatively difficult problem in the field of digital signal processing. The human auditory system on the other hand is very effective at hearing out individual sources of sound from a monaural mixture of sound sources, i.e. the cocktail party problem [263]. Auditory Scene Analysis (ASA) [264] is believed to be the process by which the human auditory system achieves this feat. ASA is based on Gestalt grouping principles, for which harmonicity cues such as frequency/amplitude co-modulation and non-harmonic cues such as common onset/offset serve to partition a time-frequency representation of a sound mixture into perceptually meaningful elements, which are linked over time to produce an auditory stream corresponding to a source signal of interest [264]. Computational ASA (CASA) [163-165] attempts to emulate ASA by computational means. CASA algorithms seek to exploit the grouping mechanisms that underlie ASA to perform source separation in a spectral representation, which can be psycho-acoustically motivated so as to approximate the early processing performed by the human auditory system [265]. Some specific CASA approaches include the use of common onset/offset cues [160] and

amplitude/frequency co-modulation cues [161] to group localized segments of the time-frequency energy of a speech signal, with temporal cues such as pitch used to connect spectral elements to achieve a separation [161, 162]. Currently, the main drawbacks of CASA are difficulty in dealing with unvoiced speech, and the dependence on multi-pitch detection algorithms, for which discerning between overlapping pitch contours is problematic [266].

Another drawback of CASA is the ad hoc nature of the resulting algorithms, which are comprised of heuristically defined grouping rules based on the natural cues of speech signals. Model-based algorithms in contrast, which are described in detail in the next section, attempt to learn the characteristic spectral features of each of the sources in a mixture in a systematic machine learning/probabilistic framework [152]. This approach therefore allows for a priori training and statistical tractability, albeit for speaker dependence. In order to create MSSS algorithms with better overall characteristics, there have been attempts [267-273] to combine aspects of CASA, such as speaker independence, with some of those of model-based approaches. For instance, in [267, 268] a speaker independent MSSS algorithm is presented that first trains, in an offline manner, a probabilistic model that is tasked with generalizing the sound source separation process as performed using various CASA grouping rules. Learning is performed by optimizing the parameters of this model with respect to sound source separation performance, which is measured by comparing the separations achieved by the model on numerous artificially created mixtures of different speakers with an optimal separation of the sources. The resultant model is then applied to the spectrogram of a mixture where, driven by the constraints imposed by training, it clusters time-frequency points to effect speaker independent source separation. The resulting separations from this approach are rated as acceptable, but this algorithm has a large hardware resource requirement. Modeling the human speech production mechanism, that is, modeling the vocal tract as a time-varying filter that is excited by the signal produced by the vocal folds or hiss, has also facilitated a hybridized source-/model-driven approach to MSSS, which is described in [271, 272, 274]. A priori modeling of speech features has also been endowed with perceptual semantics, for instance, in [270], it is hypothesized that humans memorize semantic information in the form of learned spectral patterns that is then leveraged for separation; this topic is also discussed in [275].

3.3.2 Spectral Representations for Model-based MSSS

In this section, we discuss the various spectral representations, and the associated mixing models, in which model-based MSSS algorithms operate. Spectral representations are ubiquitous in model-based MSSS because they represent speech/music signals sparsely and elucidate their regularity, i.e. those distinctive features attributable to harmonicity and onsets/offsets. These properties make spectral representations amenable to inference, or low-rank matrix factorization, because they allow a source to be modeled by a small set of characteristic spectral features, resulting in compact and distinct source models, which are effective at discriminating their sources features in the mixture. Also in this section we

discuss some of the issues that influence the choice of a particular spectral representation, and the consequences for the resulting inference problem or matrix factorization problem.

To begin, we define the STFT $\Xi(f, k)$ (complex) of a signal $\xi(n)$ as

$$\Xi(f, k) = \sum_{s=0}^{N-1} \xi(km+s) q(s) \exp(-j2\pi fs/N). \tag{3.34}$$

for discrete frequencies $f = 0, 1, \ldots, N-1$, for frames $k = 0, 1, 2, \ldots$, where N is the time-domain frame length in samples, and $q(\cdot)$ is a length-N analysis window function that advances with the frames in steps of size m, and contains a symmetric tapering function that serves to mitigate spectral leakage inherent to finite window length analysis. In what follows $m = N/2$ unless otherwise stated. In this section, we ignore noise, which will be dealt with in the following section.

From the linearity of the STFT the mixture in (3.33) can be expressed in the time-frequency domain as,

$$Y(f, k) = X_1(f, k) + X_2(f, k), \tag{3.35}$$

which can be expressed equivalently in polar complex number format as,

$$|Y(f, k)| \angle Y(f, k) = |X_1(f, k)| \angle X_1(f, k) + |X_2(f, k)| \angle X_2(f, k), \tag{3.36}$$

where $|\cdot|$ denotes the absolute value of a complex number and $\angle$ denotes the phase angle of a complex number. Despite the fact that this representation follows from (3.33), and conserves source additivity, it is seldom used for MSSS, with phase invariant spectral representations generally being preferred.

Before describing such representations, we first make a few general comments. Phase invariant spectral representations are approximate, and their use generally implies that only the spectral content of the sources is estimated. This practice is justified by perceptual principles, which state that the human auditory system is relatively insensitive to phase [276-278]. As such, when synthesizing a time-domain rendition of an estimate of the spectrogram of a source, it is common practice to substitute the mixture phases for the source phases, a process which entails pairing this estimate with the mixture phases, followed by polar to Cartesian complex number format conversion, and then transforming the resultant complex frequency response using the Inverse STFT (ISTFT), which consists of the IDFT, overlap and add, and in certain cases, a synthesis window. In this thesis, we employ a phase invariant spectral representation and adopt this approach to source synthesis; we therefore ignore source phase estimation. However, it is worth mentioning that large deviations from the source phases can produce significant perceivable distortion, as noted in [279, 280] for instance. This has motivated some recent efforts to take account of, or estimate, the phase information of the sources, in addition to their spectral information; some of this work is documented [281-285].

To mathematically elucidate the approximate nature of phase invariant spectral representations, we take the square of (3.35) which gives,

$$|Y(f, k)|^2 = |X_1(f, k)|^2 + 2|X_1(f, k)||X_2(f, k)|\cos(\theta(f, k)) + |X_2(f, k)|^2, \tag{3.37}$$

where $\theta(f, t)$ is the phase difference between the two sources at frequency bin f and frame t. It is apparent that the presence of the cross-term obstructs a mathematically sound phase invariant spectral representation. If it is assumed however that the signals $x_1(n)$ and $x_2(n)$ are uncorrelated, i.e. $E[x_1(n)x_2(n)] = 0$, the cross-term in (3.37) tends to zero as the frame length N tends to infinity; thus, rendering a phase invariant spectral representation that preserves the additivity of the sources. However, an infinitely long window implies the loss of all temporal information. To create a practical phase invariant spectral representation that preserves source additivity, it is common in speech enhancement to assume that the cross-term in (3.37) is negligible over the short term, which gives rise to,

$$|Y(f,k)|^2 \approx |X_1(f,k)|^2 + |X_2(f,k)|^2, \tag{3.38}$$

which is known as the power spectrogram. If it assumed that the source signals are stationary over a short interval of time, i.e. N, which generally holds for speech signals over intervals of 20-30 ms, each spectral frame of the power spectrogram of a source can be considered the PSD for that source over that interval. This representation therefore suggests a Weiner based approach to MSSS [166, 286], similar to Weiner-based speech enhancement [287], by which, in a frame-wise manner, the power spectrogram of the sources are estimated to construct a pair of Weiner filters, each of which is then applied to the mixture in order to render estimates of the sources. Such a framework is appealing for model-based MSSS because it is generally not possible to construct an adequate estimate of a source signals PSD directly from its source model alone [275, 288]; this topic will be discussed in greater detail in the next section. However, such estimates are typically adequate to construct a Weiner filter for each source, which can then be applied to $Y(k, f)$ to estimate the sources. While this framework is regularly invoked for MSSS based on its performance alone, particularly for matrix factorization techniques, a theoretical justification for adaptive Weiner filtering in the context of probabilistic model-based MSSS is provided in [166]; this will be outlined in the next section.

Another useful phase-invariant, approximate spectral representation is the magnitude spectrogram, given as,

$$|Y(f,k)| \approx |X_1(f,k)| + |X_2(f,k)|. \tag{3.39}$$

This representation simply ignores the cross-term, though is nonetheless widely used for speech enhancement [289]. Moreover, under the aforementioned Windowed Disjoint Orthogonality (WDO) assumption [153, 155], which can be expressed as,

$$\mathbf{0} = X_1(f,k)X_2(f,k), \tag{3.40}$$

the following is implied,

$$|X_1(f,k) + X_2(f,k)| = |X_1(f,k)| + |X_2(f,k)|, \tag{3.41}$$

and as such, WDO offers a justification for (3.39) indeed, it similarly justifies the power spectrogram representation in (3.38). Additionally, WDO implies that phase estimation is superfluous if given a spectral representation of each source and the mixture phases, offering an empirical justification for the practice of mixture phase substitution. The extent to which

the WDO assumption holds was investigated in [155] with respect to various mixture orders and to the parameters related to the STFT i.e. N, m and p. In general, it was found that speech signals approximately satisfy WDO, with the empirical veracity of this assumption depending primarily on the mixture order (proportionally) and reverberation. In particular, it was found that WDO is maximized (93.6 % WDO) for pair wise mixtures (M = 2) of anechoic speech signals sampled at 16 KHz for window lengths of 64 ms (N = 1024 samples), overlap of 32 ms (M = 512 samples) and for hanning-type analysis windows.

Another phase invariant spectral representation is the log spectrogram, given as,

$$\log\left(|Y(f,k)|^2\right) = \log\left(|X_1(f,k)|^2 + |X_2(f,k)|^2\right). \tag{3.42}$$

This representation is often approximated for MSSS as,

$$\log\left(|Y(f,k)|^2\right) \approx \max\left(\log\left(|X_1(f,k)|^2\right), \log\left(|X_2(f,k)|^2\right)\right). \tag{3.43}$$

This is referred to as the log-max assumption [152] or alternatively in [290] as mixmax, and it was empirically demonstrated in [152, 291] and a theoretical justification is given in [290]; it can also be justified by appealing to WDO. Modeling the mixture using the log-max assumption implies that each of its log spectral frames is the point-wise maximum of the log spectral frames generated by the sources. The log-max assumption therefore (and the WDO assumption) implies that for each source in a mixture there is a binary time-frequency mask that when multiplied by $Y(f, k)$ recovers the STFT of that source. It has been argued therefore that computing the binary time-frequency masks of the sources such that they may be applied to the mixture in order to render the sources is a reasonable computational goal for MSSS algorithms [152, 155, 291, 292]. Given the limitations of source models, the time-frequency masks are often constructed by using the realizations of the source model estimates as interim estimates that are then used to construct a binary time frequency mask for each source. In [152, 291], for instance, the masks are constructed according to the point wise maximum of the source model realizations.

A perceptually objectionable side effect of binary time-frequency masking however is tone like noise in the resulting time-domain signals, which is referred to as musical noise [293]. Musical noise arises due to isolated peaks and ridges of spectral energy in the resulting source spectrograms due to hard assignment of the time-frequency points, and the approximate nature of the WDO/log-max assumptions. This phenomenon has motivated soft mask approaches to MSSS [294-296], (the Weiner based approach discussed above may also be construed as a soft mask technique) which are discussed in section 3.3.3.2.

Model-based MSSS algorithms that operate in perceptually motivated spectral representations have also been proposed. These representations accentuate characteristics of the sources that are known to be perceptually important, and are thus used to attain perceptually optimized source separations. One such representation is the Mel-scale frequency representation, used for MSSS in [288, 297], in which the frequency axis is scaled in accordance with the perceptually relevant Mel scale. Another is described in [298], where a perceptually motivated transformation is proposed which maps the columns of the STFT of a

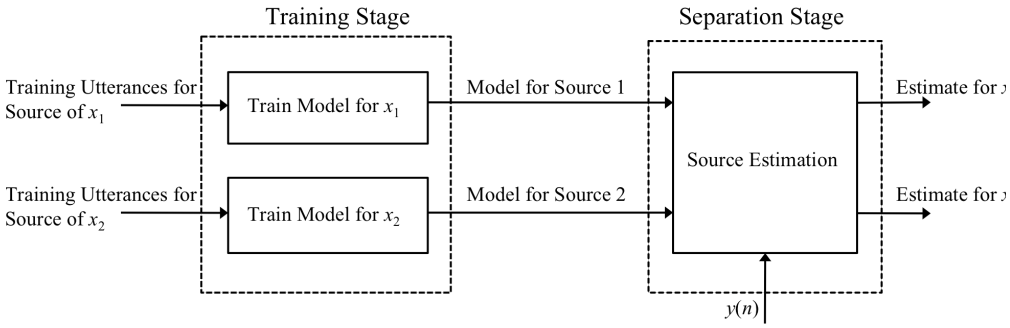

Figure 3.1: Block diagram of model-based MSSS.

speech source into a domain in which important perceptually aspects of speech sources are accentuated. In the same spirit, model-based MSSS was performed in an altered version of the Bark-scale scaled wavelet domain packet decomposition in [299]. In [300], a multi-window STFT approach is presented, which allows for analysis of spectral features at different time scales.

3.3.3 Probabilistic Model-based MSSS

In this section, we describe probabilistic model-based MSSS. This description is divided over two sub sections, namely, 3.3.3.1 and 3.3.3.2. In 3.3.3.1, we formulate model-based MSSS in a general probabilistic framework; in parallel, we also describe a well-known model-based approach that is conformable with this approach. This subsection is adapted from [166, 275, 301, 302], in particular [167], in which the various techniques that constitute probabilistic model-based MSSS are neatly generalized as a probabilistic inference problem or through the Bayesian formalism; we also discuss in 3.3.3.1 some Bayesian estimators that have been deployed to estimate the sources. Subsection 3.3.3.1, also forms the basis for 3.3.3.2, in which we review probabilistic model-based MSSS algorithms. In section 3.3.3.2, we discuss the algorithms with respect to their mixing model, source model, and the estimator employed. Figure 3.1 contains a schematic diagram of model-based MSSS, (adapted from [303]) in which it is characterized as being comprised of two stages, namely, source model training and separation.

3.3.3.1 Model-based MSSS as an inference problem

In this section we formulate model-based MSSS in a general probabilistic framework. This formulation is accompanied by a specific example based on Gaussian Mixture Modeling, which is a popular source model structure in this class. To this end, for generality, we let $\mathbf{y}(k)$ denote a $N/2 + 1$ nonnegative vector corresponding to the $N/2 + 1$ unique values (hermitian symmetry) of the k^{th} frame of some unspecified spectral representation of the mixture $y(n)$, and likewise for the source vectors, $\mathbf{x}_1(k)$, $\mathbf{x}_2(k)$, and the noise term, $\mathbf{w}(k)$; possible spectral representations are discussed in the preceding section.

In a probabilistic setting, the mixture is conventionally modeled by specifying a likelihood function, which expresses the likelihood of the mixture given the sources, and in the case of an additive mixing model can be expressed as,

$$p(\mathbf{y}(t)|\mathbf{x}_1(t), \mathbf{x}_2(t)) = p_w(\mathbf{y}(t) - \mathbf{x}_1(t) - \mathbf{x}_2(t)), \tag{3.44}$$

or in the alternate case of the log-max/mixmax model the likelihood function is given as,

$$p(\mathbf{y}(k)|\mathbf{x}_1(k), \mathbf{x}_2(k)) = p_w(\mathbf{y}(k) - \max[\mathbf{x}_1(k), \mathbf{x}_2(k)]). \tag{3.45}$$

The likelihood function, $p(\mathbf{y}(k)|\ \mathbf{x}_1(k),\ \mathbf{x}_2(k))$, accounts for the uncertainty regarding the noise term $\mathbf{w}(t)$, which is characterized by the PDF $p_w(\cdot)$; the likelihood function can also be used to characterize error attributable to the mixture model. An example $p_w(\cdot)$, is the Gaussian distribution, i.e. $p_w(\boldsymbol{w}(k)) = \boldsymbol{N}(\mathbf{w}(k); \mu_w, \sigma_w)$, where the mean and variance terms, μ_w and σ_w respectively, are shared across frequency. In the absence of noise, and/or error, the likelihood function can be expressed using the Dirac delta function.

The difficulty in solving the MSSS problem blindly or in an unsupervised fashion can be demonstrated by considering the Maximum Likelihood (ML) estimate of the sources, which is expressed as,

$$(\hat{\mathbf{x}}_1^{\mathrm{ML}}(k), \hat{\mathbf{x}}_2^{\mathrm{ML}}(k)) = \underset{\mathbf{x}_1(t), \mathbf{x}_2(t)}{\operatorname{argmax}}\ p(\mathbf{y}(k)|\mathbf{x}_1(k), \mathbf{x}_2(k)), \tag{3.46}$$

where $\hat{\mathbf{x}}_1^{\mathrm{ML}}(k)$ and $\hat{\mathbf{x}}_2^{\mathrm{ML}}(k)$ denote the ML estimates of $\mathbf{x}_1(t)$ and $\mathbf{x}_2(t)$ respectively. Ignoring noise, it is evident, for example, that owing to its underdetermined nature, $\hat{\mathbf{x}}_1^{\mathrm{ML}}(k) = \mathbf{z}$ and $\hat{\mathbf{x}}_2^{\mathrm{ML}}(k) = \mathbf{y}(k) - \mathbf{z}$ for any $\mathbf{z}$ are solutions to this problem. Note that ML problems are conventionally solved by finding the values of the parameters, source estimates in this case, that maximize the log likelihood function, which can be solved analytically if its derivative with respect to the parameters is available in closed form, but is more generally computed using an optimization procedure, such as the Expectation Maximization (EM) algorithm [304].

From a Bayesian perspective it is natural to consider incorporating additional information concerning the sources in order to constrain the set of solutions, which leads to model-based MSSS; acknowledging however, that this represents a much stronger assumption than conventional BSS assumptions and a commensurate loss of generality. Continuing the general probabilistic formalism, such information can be introduced by specifying a prior distribution over the sources, and is denoted by $p(\mathbf{x}_1(k), \mathbf{x}_2(k))$, which factorizes to $p(\boldsymbol{x}_1(t))p(\boldsymbol{x}_2(t))$ assuming the independence of the sources. As a specific example of a source model structure, we consider a Gaussian Mixture Model (GMM), by which, as described in [166], each power spectral frame of a source is considered a realization from a $N/2 + 1$ dimensional zero-centered Gaussian that is selected from a weighted set of such Gaussians. The prior distributions are expressed, in this case, as,

$$p(|X_1(f,k)|^2) = \sum_{i=1}^{J} \varpi_1^i \boldsymbol{N}(|X_1(f,k)|^2 \big| \sum\nolimits_1^i), \quad p(|X_2(f,k)|^2) = \sum_{j=1}^{J} \varpi_2^j \boldsymbol{N}(|X_2(f,k)|^2 \big| \sum\nolimits_2^j), \tag{3.47}$$

where the pair of sets $\{\varpi_1^i\}_{i=1}^J, \{\sum_1^i\}_{i=1}^J$, and $\{\varpi_2^j\}_{j=1}^J, \{\sum_2^j\}_{j=1}^J$ contain, respectively, J matching weights and diagonal covariance matrices (as they are considered in the frequency domain, they each contain a PSD) for the GMMs, indexed by i, j, respectively, with $\sum_{i=1}^J \varpi_1^i = \sum_{j=1}^J \varpi_2^j = 1$.

The priors or source models express source specific knowledge that is learned during an offline training stage, during which their parameters are optimized to fit the statistics of their respective source's training data. This entails the availability of prior knowledge or training data, which typically takes the form of isolated examples of each sound source, which for speech mixtures comprises of isolated utterances of each speaker. Ideally, each prior should be an exact generative model for its source; however, the broad range of variability inherent to speech or musical sound sources has meant that this is an unrealistic goal, some of the implications of which where discussed in section 3.3.2. Nonetheless, it is desirable that each source model captures a maximal amount of the statistical variability exhibited by their source without sacrificing source specificity, i.e. closely approximate the generative model, such that the source priors satisfactorily represent their source during the subsequent inference procedure. In general, this requires that each sources' training data be sufficiently representative, and that the chosen source model structure is apt for the task of modeling sound sources. Considering now the training, per [166], of the GMMs specified in (3.47), the EM algorithm was employed to optimize the parameters of the source GMMs under the ML criteria to fit examples of short-time PSDs of their respective sources. This procedure results in a pair of GMMs each containing a set of J Gaussians, each of which express a characteristic spectral shape (PSD) of their source each weighted by their probability in the training data.

The source priors and the likelihood function jointly specify a generative model for the mixture, and are combined using Bayes theorem to infer the joint conditional posterior distribution over the sources, given as,

$$p(\mathbf{x}_1(k), \mathbf{x}_2(k)|\mathbf{y}(k)) \propto p(\mathbf{y}(k)|\mathbf{x}_1(k), \mathbf{x}_2(k))p(\mathbf{x}_1(k))p(\mathbf{x}_2(k)), \tag{3.48}$$

where the posterior expresses the probability of the sources given the mixture; the normalizing constant is omitted for notational convenience. Intuitively speaking, the role of the priors in (3.48) is to encourage consistency with the source models while the role of the likelihood function is to encourage consistency with the mixture, such that a region of high probability in the posterior indicates source estimates that are consistent with both the priors and the likelihood. To infer the posterior, model-based MSSS algorithms conventionally combine the priors in a factorial architecture [305]. Using the example of GMM source priors, the factorial architecture assumes that each GMM evolves independently resulting in two independent state sequences. Inference then involves computing the posterior probability for each pair of states from each GMM, which in the case of an additive mixture model with Gaussian noise, is computed using the following analytical expression [166],

$$p(i, j \,|\, |Y(f,k)|^2) \propto \varpi_1^i \, \varpi_2^j \, \boldsymbol{N}(|Y(f,k)|^2, \textstyle\sum_1^i + \sum_2^j + \sigma_w^2 I). \tag{3.49}$$

Different assumptions regarding the generative model of the mixture will result in different expressions for the posterior, which is discussed further in the next section.

Having inferred the conditional posterior distribution, a variety of point estimators may be deployed to obtain estimates of the sources. Two common estimators are the Maximum A Posteriori (MAP) estimator, which is defined by the following criteria,

$$(\hat{\mathbf{x}}_1^{\text{MAP}}(k), \hat{\mathbf{x}}_2^{\text{MAP}}(k)) = \underset{\mathbf{x}_1(k),\mathbf{x}_2(k)}{\text{argmax}}\ p(\mathbf{y}(k)|\mathbf{x}_1(k), \mathbf{x}_2(k))p(\mathbf{x}_1(k))p(\mathbf{x}_2(k)), \tag{3.50}$$

and the Posterior Mean (PM) estimator, which is defined by,

$$\begin{aligned}\hat{\mathbf{x}}_1^{\text{PM}}(k) &= E\{\mathbf{x}_1(k)|\mathbf{y}(k)\} = \int \mathbf{x}_1(k)\, p(\mathbf{x}_1(k)|\mathbf{y}(k))\ \ d\mathbf{x}_1(k),\\ \hat{\mathbf{x}}_2^{\text{PM}}(k) &= E\{\mathbf{x}_2(k)|\mathbf{y}(k)\} = \int \mathbf{x}_2(k)\, p(\mathbf{x}_2(k)|\mathbf{y}(k))\ d\mathbf{x}_1(k),\end{aligned} \tag{3.51}$$

where the marginal conditional posterior distributions of $\boldsymbol{x}_1(t)$, and $\boldsymbol{x}_2(t)$ are given as,

$$\begin{aligned}p(\mathbf{x}_1(k)|\mathbf{y}(k)) &= \int p(\mathbf{x}_1(k),\mathbf{x}_2(k)|\mathbf{y}(k))\ d\mathbf{x}_2(k),\\ p(\mathbf{x}_2(k)|\mathbf{y}(k)) &= \int p(\mathbf{x}_1(k),\mathbf{x}_2(k)|\mathbf{y}(k))\ d\mathbf{x}_1(k),\end{aligned} \tag{3.52}$$

and where each integral is computed over the range of the corresponding 'nuisance' source. The MAP source estimates are the most probable per the joint conditional posterior, and are found similarly to those of the ML problem in (3.46); indeed, for uninformative source priors such as uniform distributions, the MAP estimates are equivalent to the ML estimates. The PM estimator of a source, also known as the (Bayesian) Minimum Mean-Squared Error (MMSE) estimate, corresponds to the expected value of that source conditioned on the mixture, and requires computing an integral or more generally integrals. An important consideration for this estimator therefore is analytical tractability, which implies that the integrals in (3.51), (3.52) and the normalizing constant of (3.48) are available in closed form, which is a highly desirable property especially for high dimensional problems; although for analytically intractable problems, approximate techniques such as Monte Carlo Markov Chain (MCMC) [306] techniques are applicable. The MAP and PM estimators have different properties and generally give different estimates; as such, the choice of either estimator depends on their appropriateness to the specific problem. To highlight a particular issue, we consider a multi-modal conditional posterior over the sources; the MAP estimator will correspond to the mode or maximum probability of the distribution, which may, in certain cases, correspond to an isolated spike that is not representative of the majority of the posterior. The PM estimator on the other hand may correspond to a point with low probability, i.e. the expected value of a bi-modal distribution may reside in a valley between both peaks giving estimates with low probability. For a Gaussian distribution the MAP and PM estimators coincide.

Returning once more to the GMM-based example, the MAP estimates of the sources can be expressed analytically as [166],

$$\left|\hat{X}_1^{\text{MAP}}(f,k)\right|^2 = \frac{\Sigma_1^{\hat{i}}}{\Sigma_1^{\hat{i}} + \Sigma_2^{\hat{j}} + \sigma_w^2 I}|Y(f,k)|^2, \quad \left|\hat{X}_2^{\text{MAP}}(f,k)\right|^2 = \frac{\Sigma_2^{\hat{j}}}{\Sigma_1^{\hat{i}} + \Sigma_2^{\hat{j}} + \sigma_w^2 I}|Y(f,k)|^2, \tag{3.53}$$

where $\hat{i}$ and $\hat{j}$ indicate the pair of states from the source GMMs with the maximum likelihood per $p(i,j\,||Y(f,k)|^2)$. As is apparent, if noise is excluded; in this case, using the MAP estimates of the sources may be construed as performing *adaptive* Weiner filtering, theoretically justifying this approach in this context. The PM estimate of the sources can also be expressed analytically [166] owing to the analytical tractability of this example,

$$\left|\hat{X}_1^{\mathrm{PM}}(f,k)\right|^2 = \sum_{i=1}^{J}\sum_{j=1}^{J} p(i,j \mid |Y(f,k)|^2)\left(\frac{\sum_1^i}{\sum_1^i + \sum_2^j + \sigma_w^2 I}|Y(f,k)|^2\right),$$
$$\left|\hat{X}_2^{\mathrm{PM}}(f,k)\right|^2 = \sum_{j=1}^{J}\sum_{i=1}^{J} p(i,j \mid |Y(f,k)|^2)\left(\frac{\sum_2^j}{\sum_1^i + \sum_2^j + \sigma_w^2 I}|Y(f,k)|^2\right), \quad (3.54)$$

with the Gaussian likelihood function and the Gaussian state being conjugate pairs. The PM estimator for a source is the sum of J sums of J weighted *adaptive* Wiener filters, with the weighting for a pair of states i, j corresponding to the posterior $p(i, j \mid |Y(f, k)|^2)$, and with J adaptive Wiener filters for each of the J Gaussian states of the corresponding GMM. It is apparent therefore that this expression reflects the computation of the marginal conditional posterior distribution for each source.

3.3.3.2 Review of Probabilistic Model-based MSSS algorithms

In this section, we review various probabilistic model-based MSSS algorithms. The GMM approach featured in the preceding section was proposed in [166, 286], and corresponds to a power spectrogram representation in which the sources sum with Gaussian noise. An inherent limitation of GMMs is their scale insensitivity, which in the context of MSSS renders their Gaussian states insensitive to the occurrence of their spectral shapes at different scales throughout a mixture. This is addressed in [166], where an additional nonnegative scale parameter is introduced for each trained Gaussian. The resulting source models, referred to as Gaussian Scaled Mixture Models (GSMMs), can be regarded as GMMs with covariance matrices $\{a_1^i \cdot \sum_1^i\}_{i=1}^J$, and $\{a_1^i \cdot \sum_1^i\}_{i=1}^J$, and yield expressions for the PM and MAP estimators similar to those specified in (3.53)(3.54); a multiplicative update optimization approach is proposed to find the optimum gains, exploiting the non-negativity of the problem [307]. In an accompanying experimental evaluation/comparison of GSMM/GMM source models and the PM/MAP estimators in [166], for the task of separating monaural music mixtures, it was demonstrated that initially as the number of states in either the GMM or GSMM models increase, increasing the expressive power of the models, there is an increase in the quality of the separations for both estimators. This trend diminishes somewhat though after 16 Gaussians, which is attributed to over-fitting and/or initialization issues related to the EM algorithm. It was also found that GSMM source models generally perform better than GMM models, though many exceptions exist. In terms of estimator criteria, for the GSMM case, the PM criterion gives slightly better results than the MAP, and for the GMM, the MAP gives modest performance, with the PM performing adequately. A downside of the PM estimator is its computational load, which is considerable greater than that of the MAP estimator.

In [294-296, 308] the use of GMMs source models under the log-max assumption is examined. As above, the algorithms proposed in [294-296, 308] employ a GMM model for each source and use a factorial GMM architecture for inference, but unlike above, the mixture is modeled using the log-max assumption, whereby each log spectral frame of the mixture is the point-wise maximum of the realizations produced by the proposed pair of states. In [294, 308] it is shown how the log-max assumption facilitates an analytical expression for the PM estimator. Also in [294] and earlier in [295], a soft mask approach is described, where instead

of a hard assignment of the log-spectral energy of the mixture, a soft assignment is performed. The soft masks correspond to the probability that a source generated a certain time-frequency point, such that each of their elements attains a value between 0 and 1. Experimental comparisons of these masks with a comparable log-max based Vector Quantization MSSS algorithm [291] that employs binary masking, show that the soft-mask estimates produce higher signal-to-interference ratios and higher subjective ratings per Mean Opinion Scoring (MOS). In the same context, a similar soft mask approach is proposed in [296], where it is shown that the PM estimator leads to a closed form filter, akin to the adaptive Weiner filter specified in section 3.3.3.1, which the authors term the soft mask filter; it is also shown that the binary mask is a simplified form of this soft mask filter. The soft mask filter is then compared to both the adaptive Weiner filter approach [166] and binary masking, and is shown to yield superior quality as measured using objective measures of speech quality. In [309] the use of univariate GMM source models in the Discrete Cosine Transform (DCT) domain is described. This source model structure allows for reduced computational complexity in comparison with the multi-dimensional GMM approaches; it also makes for analytical tractability, with the PM estimator being used to estimate the sources. Under the mix-max assumption, the problem of estimating sources that have different scales is tackled in [310].

GMM model-based MSSS was extended in [303, 311] to consider scenarios in which it may be operationally feasible to adapt trained source models to the mixture at hand; referred to as source-adapted models. This capability can be advantageous if the training data upon which the source model is trained is inadequate, i.e. not sufficiently representative of the source, or when the source models are class rather than source specific, i.e. a speech/music source model. This approach was exemplified for the task of separation of the singing voice from music [311]. A detector was described that can distinguish between intervals of music only, voice only, and both music and voice, such that the music/voice source model is adapted during music/voice only intervals, with the adapted models being used to separate the sources during the next detected period of both music and voice. In [303] a number of the issues related to this approach were discussed, including the allowed flexibility for the source-adapted models, and various techniques for updating the various parameters of the source-adapted GMM models within this context. It terms of performance, it was demonstrated that source-adaptation allows for approximately twice the separation performance, as measured using an objective speech quality measure, compared with no source adaptation; though the performance is negatively affected by detector inaccuracy. It can be concluded therefore, that the performance gain offered by this approach depends on both the availability, and accurate detection of, singular occurrences of the sources in the mixture. A further generalization of GMM model-based MSSS encompassing source-adapted models is described in [312].

Another common source modeling structure is the Hidden Markov Model (HMM). Essentially, for model-based MSSS, HMMs are used to extend the GMM source model structure by additionally modeling the transition probabilities between the GMM states. This model structure therefore admits temporal information concerning the sources to be

considered, so that this information, along with the spectral information encoded within the Gaussians, influence the resulting inference problem, and possibly constrain the set of solutions further. This application of HMMs was first proposed by Roweis in [152], where each source HMM consists of ($N/2 \times 1$) dimensional Gaussian states or emission probabilities, each of which, as above, encode a spectral feature of the source, and a state transition matrix that stores the state-to-state transition probabilities, which model the temporal dependencies between the Gaussians. These source models are trained by initializing the states of each HMM with Gaussian densities learned from fitting a GMM to the training data, before then learning the state transition matrix. The mixture is then decomposed in a factorial HMM architecture under the log-max assumption with nonnegative Gaussian noise; the MAP estimator of the sources is then used for the source estimates.

Also addressed in [152] is the computational complexity of performing inference under the factorial HMM/GMM architecture. To adequately model the mixture under the log-max assumption, it is proposed to model each source with a HMM with a large number of Gaussian states, i.e. 8149 states for each HMM. However, inference in a factorial HMM architecture requires searching over each pair of states from each HMM, a computational task with a complexity that grows exponentially with the number of states. To tackle this issue it is proposed to exploit the log max assumption to mitigate searching over all pairs of states of each HMM. The specific aspect that is exploited is that the energy of the mixture provides a bound on the energy in each time-frequency point of each state, and thus, states which imply energy above this bound are precluded from the search space. Relatedly, in [313] it is shown how by modeling the sources using a set of small HMMs, each of which models a subband of their source, with coupling between adjacent HMMs, the number of HMM states for MSSS source–modeling can be reduced without a decrease in performance relative to full-band source models.

HMMs were also incorporated into the GMM technique that was featured in section 3.3.3.1. At the training stage, the parameters of the GMM where learned using the EM algorithm, after which the state transition matrix corresponding to the HMM are learned. Experimental results show that the HMM/GMM achieves similar performance to a GMM based source model approach, and requires greater computational load, suggesting that the use of a HMM to model the temporal GMM state dependencies is superfluous. A similar point concerning the utility of HMM in conjunction with GMM is made in [291]. Similar to the GSMM-based approach described above, the issue of scale insensitivity as it pertains to HMM-based source models is addressed in [314], where a scaled factorial HMM architecture is proposed to compensate for the gain differences that may arise between sources. It is shown that this approach outperforms existing scaled/unscaled VQ-based MSSS algorithms. The use of the MAP estimator for HMM source modeling is discussed in [315].

A number of model-based MSSS algorithms where proposed as part of the monaural speech separation and recognition challenge task that is described in [316], along with the proposed algorithms. The objective of this task was to recognize the speech of a target speaker

in a monaural mixture also containing an interfering speaker. The speakers, both target and interfering, were constrained to be from a closed set, with each utterance from each speaker complying with the same restricted grammar set. Training data was provided for each speaker. The algorithm that produced the lowest Word Error Rate (WER) employed a Model-based MSSS technique based on graphical modeling [317-320]. This technique jointly models the spectro-temporal characteristics of the source and the temporal constraints of the sources pertaining to the allowed grammar of the task. The spectro-temporal characteristics were modeled using a HMM with Gaussian emission probabilities (as described above), while the grammar dynamics are modeled by grammar state transitions consisting of left to right phone models. The HMM and grammar transitions were then combined into a single source model, one for each speaker, by creating a graphical model in which each grammar state is associating with each HMM state through a state transition matrix, which is learned during training, such that the states of the HMM are conditioned on the grammar state. Then at the test stage, having identified the two speakers in a test mixture, the factorial graphical model architecture is employed for inference, for which the authors describe two efficient, customized, techniques, both of which find the most probable state paths for each source. It was demonstrated that grammar conditioning improves the Word Error Rate (WER) of the overall approach significantly in comparison with the HMM source modeling alone. Interestingly, this approach was demonstrated to achieve a slightly lower WER than human listeners.

In [321] this task was addressed by training a HMM model on Mel-Frequency Cepstral Coefficients (MFCC) for each speaker for each word of the allowed grammar. The HMMs source are then combined in a factorial HMM architecture to infer the distribution of MFCCs for the mixture, which was performed in the power spectrogram domain by transforming the MFCCs. Estimates of the power spectrograms of the target and interference sources are then drawn from the MFCCs corresponding to the most likely state transition paths for each speaker, and are used to populate two adaptive Weiner filters which are applied to the mixture to synthesize either source. In [288, 322], it is proposed to adapt the source models to the speakers in the mixture, an approach somewhat akin to the source-adapted GMM approach discussed earlier [303], but without the availability of singular instances of the sources. Given the mixture, source-adaptation is performed by representing the space of speaker variation with a parametric source model. To temporally constrain the state path of each source-adapted model, within-phone temporal information is incorporated into each source model. While this approach achieves a higher WER than the two previously described algorithms, it incorporates less task-specific information, and hence, it better generalizes to other tasks. The computational speech of this approach was accelerated in [323].

Vector Quantization (VQ) is another widely studied approach to source modeling. VQ was first proposed for model-based MSSS in [291], in which the sources are modeled by codebooks that contain spectral features of the sources. These features or codewords are learned by vector quantizing the frames of the log-spectrogram of the sources training data

REF	DOMAIN	SOURCE MODEL	MIXTURE MODEL	SOURCE ESTIMATE
[166, 286],	Power spectrogram	Guassian Mixture Model	Factorial GMM	MAP and MMSE,
[166]	Power spectrogram	Guassian Mixture Model	Factorial GSMM	MAP and MMSE
[294, 308]	Log spectrogram	Guassian Mixture Model	Factorial GMM, Log Max	MAP and MMSE
[152]	Log spectrogram	Hidden Markov Model-GMM emisions	Factorial HMM, Log max	MAP
[321]	Mel-Frequency Cepstral coeffiencts	Hidden Markov Model	Factorial HMM	Wiener Filter
[317-320]	Log spectrogram	Graphical model	Factorial Graphical modeling, log max	MAP
[291]	Log spectrogram	Codebook	Factorial Vector quantization, log Max	MAP

Table 3.1 : Summary table listing some aspects of tha main probabilistic-based approaches to model-based MSSS.

using a clustering algorithm, such as *k*-means, or if this approach is formulated in a probabilistic setting, as it is [291], using the EM algorithm. Similar to GMM/HMM source modeling, each codeword of the codebook can be considered a discrete state of the source, though now with deterministic realizations. To separate the mixture, the factorial architecture was extended to VQ under the log-max assumption with nonnegative Gaussian noise, whereby the pair of codewords from each codebook that best fit the current mixture frame are found. To mitigate under or over estimation of the mixture due to the finite number of codewords in the codebooks, without overly increasing the number of codewords, a binary mask is constructed to reconstruct each source based on the point-wise maximum of the selected codewords; this approach to source reconstruction is referred to as MAXVQ. Similarly to [152], to reduce the size of the search space, the log-max assumption is exploited to preclude certain codeword combinations from representing the mixture. The author reports that although the VQ approach does not incorporate temporal constraints, it has comparable performance to MSSS algorithms that do, such as [152]. Computational efficiency of VQ model-based MSSS is the topic of the work presented in [324], where a fast hierarchical VQ procedure for model-based MSSS is proposed, and also in [325], where two search heuristics are proposed to reduce the computational burden of VQ model-based MSSS. CASA and MAXVQ are combined in [326] to address the monaural speech separation challenge.

In [288], a VQ model-based MSSS system is proposed that trains a large codebook for a single target source, and then aims to estimate this source's contribution to a mixture also containing non-stationary noise by approximating the mixture using the vectors in the codebook; the idea being that each mixture frame will be approximated, or vector quantized, with respect to the codebook, such that the spectral feature corresponding to the chosen codeword will approximate the source leaving the interference. The authors show that a prohibitively high number of codeword's are required to learn an adequate codebook for this task; adequate in the sense that an acceptable quality reconstruction of the source is achieved. This approach was then extended to incorporate temporal information between the codeword's, using a HMM, which resulted in slightly improved performance. Similar results

were obtained from applying this approach in the Mel-scale frequency domain, though the authors report that some reconstructions sounded more pleasant. The authors also investigate the use of a phase vocoder representation to mitigate problems associated with the use of the original mixture phase to synthesize sources. A similar VQ technique, with a codebook containing a few orders of magnitude more codewords was investigated in [327]. Given the large size of the codebook, naïve inference, i.e. computing the conditional posterior distribution in its entirety is computationally unfeasible; instead, the authors employ a particle filtering approach to make inference computationally tractable.

In [328], a VQ based MSSS that operates in the log spectrogram is presented that sections the training data of each speaker into a low frequency band and a high frequency band, and then computes a separate codebook for each section for each speaker. At the separation stage, the mixture frames are divided into the same sections. The optimal output codewords from the low and high frequency band codebooks of each source are then concatenated, and the resulting vectors are combined to create a soft mask for each source. This approach is demonstrated to offer enhanced separation performance compared to a comparable VQ algorithm that does not perform such sectioning.

A VQ based MSSS is presented in [329] that operates on a sinusoidal representation of speech. For this approach, the source codebooks contain sinusoidal parameters learned from sinusoidal representations of the training data. Estimates of the sources are constructed from the sinusoidal parameters stored in the optimum set of codewords from each codebook. The evaluation of this approach shows that it outperforms other VQ based MSSS algorithms. Further work based on this approach was presented in [330], in which a metric is proposed for sinusoidal based VQ method that takes into account the signal characteristics, and a split VQ implementation of this technique is presented in [331]. VQ model-based MSSS with soft masking is investigated in [332], where a non-linear masking technique is proposed, and demonstrated to achieve better interference rejection than other techniques.

In [333], the performance of VQ-based MSSS approach is compared in three different representations; the log spectrogram, modulated lapped transform (MLT) coefficients, and a pitch and envelop representation. Using pre-trained codebooks of quantized spectral feature vectors as above, it is shown the log spectrum offers the best performance for speaker-dependent scenarios, whereas, for the speaker-independent scenario, the best results are obtained from the pitch-envelop representation. Also, a reduced complexity VQ based MSSS algorithm is evaluated on these representations in [334]. The effect of window size on the performance of VQ-based MSSS approach is discussed in [335], in which the author concludes that the optimal window length for VQ based MSSS is slightly longer than that commonly employed in speech coding applications.

Speech enhancement for non-stationary interference sources has also been addressed in the model-based MSSS framework. Similar to the VQ approach, it is proposed in [336, 337] to learn a codebook containing AR processes for both the source and the interference. The AR processes, parameterized by Linear Prediction Coefficients (LPC), and an excitation variance,

model the various spectral envelopes of the sources, both speech and interference, and are commonly used for this purpose in speech related applications. In [337] a iterative ML estimation technique is employed to find the optimum pair of states from each codebook for the current mixture frame. This procedure equates to finding the pair of states from each codebook closest to the mixture frame according to the Itakura-Saito distance measure; the excitation variances for the various sources must also be estimated. The power spectrum of the target source is then constructed from the chosen state and excitation variance, and is used to construct a Weiner filter for the target source. The computational complexity associated with an exhaustive search over all possible pairs of states is also addressed in [336, 337]; in particular, to alleviate the computational cost of performing separation with a large noise codebook that encompasses a wide variety of noise source, a small noise codebook is trained for each of a number of different noise sources, and a classification scheme is used to determine the type of noise in the mixture. In [338, 339] the LPC coefficients of the AR processes are considered as random variables and the PM estimator is derived, with numerous approximations for analytical tractability. The transition probabilities between the AR processes in the source codebooks are also taken into account in [337].

Several strategies for semi-supervised MSSS were evaluated for mixtures containing a speech and a music source in [340]. Specifically, three different techniques for capturing the features of the sources from training data in codebooks were considered; GSMM models [166], autoregressive models [336, 337] and Amplitude Factorization models [307] (Non-negative sparse coding). By experimental comparisons, it was found that separation performance can be optimized by using a different model for the speech and music sources, with speech best modeled by an AR codebook and music best modeled by an AF based codebook.

In [341], unsupervised separation of two Auto Regressive (AR) processes from a single mixture is examined. It is shown that the parameters of the AR models can be uniquely identified by factorizing the power spectral density of the mixture, for which three different algorithms are proposed. This approach was demonstrated for synthetic AR processes, and was demonstrated to separate a mixture of two selected speech signals, proving that an AR model of speech signals can be employed to achieve unsupervised MSSS. Unsupervised MSSS using both short and long term AR models is discussed in [342].

3.3.4 Matrix Factorization Model-based MSSS

In this section, we describe the matrix factorization class of model-based MSSS techniques. As in the previous section, we divide this description over two sections. In section 3.3.4.1 we outline a prototypical matrix factorization model-based MSSS approach and highlight some general properties. Then in section 3.3.4.2 we review specific techniques, which differ primarily by the representation used and the matrix factorization technique employed; we will also describe some relevant unsupervised MSSS techniques in this section. Many of these techniques can also be formulated in a ML or MAP framework. Note that we ignore noise is the following sections.

3.3.4.1 Model-based MSSS via matrix factorization

In general, this class of techniques takes advantage of the inherent low-rank nature of audio spectrograms [343] to learn a basis for each source. We let the bases $\mathbf{B}_1$ and $\mathbf{B}_2$ denote $(N/2 + 1) \times R$ matrices containing the trained basis vectors (columns) for the sources $\mathbf{x}_1(k)$ and $\mathbf{x}_2(k)$ respectively. Similar to the source codebooks of VQ model-based MSSS methods, $\mathbf{B}_1$ and $\mathbf{B}_2$ contain characteristic spectral features of their sources, which were learned from their source's respective training data using an appropriate matrix factorization technique. In the case of NMF, which as mentioned in section 3.2 is related to VQ, both $\mathbf{B}_1$ and $\mathbf{B}_2$ typically store a low rank NMF of a spectral representation of the sources training data; a discussion concerning the similarities between NMF and VQ in relation to model-based MSSS is available in [266].

Given the source bases, it is assumed that each spectral frame of the mixture, i.e. $\mathbf{y}(t)$, can be approximated by a linear combination of the union of the bases. Using block matrix notation, this factorization is expressed as,

$$\mathbf{y}(k) = [\mathbf{B}_1\ \mathbf{B}_2]\begin{bmatrix}\mathbf{g}_1(k)\\ \mathbf{g}_2(k)\end{bmatrix} + \mathbf{e}(k), \tag{3.55}$$

where $\mathbf{g}_1(k)$ and $\mathbf{g}_2(k)$ respectively denote randomly initialized length R vectors associated with $\mathbf{B}_1$ and $\mathbf{B}_2$, and $\mathbf{e}(k)$ denotes residual error. This factorization implies that the sources, $\mathbf{x}_1(t)$ and $\mathbf{x}_2(k)$, are modeled as a linear combination of their basis functions, expressed as,

$$\mathbf{x}_1(k) \approx \mathbf{B}_1\mathbf{g}_1(k), \qquad \mathbf{x}_2(t) \approx \mathbf{B}_2\mathbf{g}_2(k), \tag{3.56}$$

where we assume that an appropriate projection of any nonnegative values has been performed. We term the matrix products $\mathbf{B}_1\mathbf{g}_1(k)$ and $\mathbf{B}_2\mathbf{g}_2(k)$ as the source model estimates to distinguish from them from source estimates that may arise from further processing. In the case of NMF, the factorization in (3.55) is achieved by combining the NMF source bases to form the composite basis, i.e. $[\mathbf{B}_1\ \mathbf{B}_2]$, and then performing what we term a restricted NMF procedure whereby only the composite gain component, i.e. $[\mathbf{g}_1(k)\ \mathbf{g}_2(k)]^T$, is updated while the composite basis is held fixed throughout the procedure. By holding the composite basis fixed, the features in the mixture are matched to similar features in the composite basis. This typically results in each source basis being used to represent the bulk of the sources contribution to the mixture.

Similar to probabilistic source models, the spectral features that can be represented by a linear combination of the columns of the source bases does not encompass the entire variability of their source, and as such, the entire contribution of every source in a mixture may not be matched onto its basis or any other. Consequently, a portion of the spectral energy of the sources may remain unaccounted for after decomposition of the mixture, resulting in a residual. As described in section 3.3.2, the corresponding distortion in the model source estimates has meant that these estimates are often used to construct a hard (binary mask) or soft mask (Wiener Filter) to decompose the mixture. The performance of matrix factorization techniques is also affected by cross matching, where a portion of the spectral energy of a

source is matched onto the bases of others; it is especially prevalent between source bases that are insufficiently distinct and contain spectral features similar to those of others.

3.3.4.2 Review of Model-based MSSS via Matrix Factorization

We now review matrix factorization model-based MSSS techniques. Model-based MSSS via NMF with the $C_{KL}(\cdot)$ cost function, optimized using multiplicative updates, and operating in the magnitude spectrogram domain, was investigated in [344, 345]. An unsupervised and supervised approach, termed directed and undirected NMF model respectively, were presented. The directed NMF model employs the generic procedure described above, and constructs binary masks from the source model estimates to separate the sources. The undirected approach applies a low rank NMF to the mixture magnitude spectrogram and then groups the resulting basis vectors either manually or automatically, to eventually construct a binary time frequency mask for each source. Both methods were applied to a number of different MSSS datasets and the resulting separations were evaluated using the BSS-Eval toolbox [346]. The results indicate that both approaches achieve separations of the sources, with directed NMF having superior performance to undirected NMF. The results from an oracle source estimator [347] also indicate potential for improvement under this model.

Sparse NMF [216] was used to perform model-based MSSS in [297]. Two approaches were presented for training the speaker bases. The first conventionally performs a sparse NMF on the mel-scale magnitude spectrogram of the training data of the speakers, and the second first partitions the training data into segments corresponding to phonemes, and then performances a separate sparse NMF procedure on the set of examples of each phoneme; the latter approach was developed in part to address the monaural sound source separation challenge in [316]. It is reported that the computational load of the latter approach is less than the former. The resulting source model estimates of the two approaches were compared using objective performance measures, from which it is shown that the conventional training approach performs slightly better than the phoneme segmented method. It was found that best separations were achieved for mixtures of sources of different gender. It was also found that the performance improved with an increase in the number of basis functions for each approach; and for a very large number of basis vectors, increased sparsity, by way of an increase in the sparsity regularization parameter, improves the separations of both approaches.

The characteristics of CNMF decompositions, cost function defined in (3.13), of the magnitude spectrogram of speech signals was elucidated in [224], in which it is demonstrated that the CNMF basis functions correspond roughly to speech phones. A CNMF model-based MSSS was also described in [224]; this approach outputs the CNMF source model estimates. An extensive parameter study was performed, from which it is apparent that the time extent of the basis functions i.e. T, has a negligible influence over separation performance, which given the large increase in computational load for $T > 1$ suggests that $T = 1$ i.e. NMF, is more suited to MSSS than CNMF; the number of bases functions and the window length were found to be the most influential parameters. It was also reported that mixtures containing a male and female speaker, or a mixture containing a speaker and a distinctive interference source, are

more amenable to separation by this approach, which was ascribed to the spectral dissimilarity of these sources, and suggests a dependency on the dissimilarity of the sources for model-based MSSS via NMF. A number of post processing techniques are also suggested to improve performance after the initial separation stage. This CNMF approach is compared to CNMF with the $C_{SED}(\cdot)$ cost function for audio pattern separation in [226], in which it is reported that the later approach achieves better performance for less computational load.

Sparse CNMF was shown in [260] to discover an over-complete phone-like basis from the magnitude spectrogram of speech signals, with the captured phones typically containing richer detail than those captured by the same number of basis functions of CNMF. Sparse CNMF was then also exploited for model-based MSSS in the same paper [260], and was demonstrated to achieve superior separation performance than CNMF.

Motivated by the temporal structure evident in the magnitude spectral representation of music, NMF with a auxiliary temporal continuity constraint, defined in (3.11) and a sparsity constraint, each imposed on **G**, was proposed for unsupervised MSSS of music sources in [212]. Multiplicative updates where derived for the $C_{KL}(\cdot)$ cost function, augmented with penalty terms for each constraint. These updates where subsequently used to factorize the magnitude spectrogram of a mixture of music sources. A clustering algorithm was described that groups the resulting basis vectors to form an estimate of each source spectrogram. In the subsequent evaluation, this approach produced better separation performance than Independent Subspace Analysis (discussed below) and NMF. In particular, the temporal continuity constraint was found to improve the detection of pitched musical sounds, though the sparsity constraint was found to have negligible impact on separation performance. It is also reported in this work that the magnitude representation facilitates better performance than the power spectral representation. The same author proposes a perceptually weighted NMF scheme in [348] for audio spectrogram factorization for which a weighting is assigned to each frequency in proportion to the loudness perception of the human auditory system. This is demonstrated to produce more perceptually agreeable unsupervised MSSS separations. NMF is presented in a Bayesian framework in [349] for audio spectrogram factorization, in which prior distributions are imposed on the factors such that some characteristic feature of audio is imposed on the resulting factorization. This approach is demonstrated to outperform existing NMF for an unsupervised MSSS task. A more general NMF model is presented in a Bayesian framework in [350].

The attributes of 2D-CNMF where elucidated by comparing its factorization of the log spectrogram of a music mixture comprised of two instruments to that of CNMF [240]. For 2 basis functions, 2D-CNMF was shown to separate the two music sources, whereas CNMF for 6 basis functions was shown to capture the spectral signatures of the individual notes played by each instrument. This capability of 2D-CNMF was ascribed to its ability to shift its basis functions vertically, or in this case over frequency, allowing pitch changes to be described by vertical modulations of its basis functions, which is encoded in the accompanying convolutive gain matrix; in contrast, the CNMF basis functions can only capture temporal structure. A

perceptually enhanced version of 2-D CNMF is described in [351] for the separation of audio sources. This version improves separation performance over 2D-CNMF as measured using objective measures of speech, and for less computational load.

The matrix factorization approach to model-based MSSS may also be developed in a probabilistic framework using Probabilistic Latent Component Analysis (PLCA) [352, 353]. In this framework, $\mathbf{y}(k)$ is scaled to unit magnitude such that it can be considered a conditional (on frame index k) discrete probability distribution (multinomial) over frequency, i.e. each normalized time-frequency point represents the probability of drawing a particular frequency from a set containing all frequencies given a frame-index k. As above, each source contributes R components to the mixture factorization, which in this case correspond to a set of R source-specific frequency marginals that are learned a priori on the magnitude spectrogram of training data using the EM algorithm. The normalized mixture frame is then modeled as a convex combination of the $2R$ frequency marginals, or latent components. To decompose the mixture, the EM updates pertaining to the frequency marginals are held fixed, while the time marginal distributions are iterated. The R frequency marginals of each source, weighted by the resulting time-marginals, are then summed to estimate the distribution of each source, which can be considered an estimate of the magnitude spectrogram of the source. As is apparent, this approach is analogous to the matrix factorization approach above. Furthermore, as pointed out in [354], the EM updates for PLCA are equivalent, save for a normalization constraint, to the multiplicative updates of NMF for a cost function similar to $C_{\mathrm{KL}}(\cdot)$ [354]. The advantage of the PLCA is its probabilistic formulation, which allows it to be more easily generalized, interpreted and extended [354]. With this in mind, various extensions have been proposed for PLCA that have in turn been used for model-based MSSS, including; sparse PLCA in [355, 356], where an entropic prior is specified over the marginal's, and a shift invariant form of PLCA [355] which is similar to CNMF.

It is proposed in [357] to employ the spectrogram of the training data itself as the basis. This, it is argued, means that the resulting source bases are less likely to be used to explain competing sources, i.e. less cross-matching, relative to trained bases, which can contain more general and less discriminative spectral features. This approach uses a sparse latent variable decomposition technique [355] to factorize the mixture, such that a sparse weighting of the basis vectors is obtained for the mixture. To mitigate the large computational load associated with a composite basis comprised of the training data of each of the sources, an energy threshold is employed to discard redundant basis vectors during the factorization of the mixture. The proposed algorithm is demonstrated to have slightly better performance over regular trained bases for artificial mixtures, and significantly better performance for realistic sources.

In [343] the ability of PLCA to represent the spectral structure of audio signals, and the ability of HMMs to model the temporal structure of sound sources where combined for model-based MSSS in the magnitude spectral domain. For this approach, a nonnegative HMM (N-HMM) model is proposed for the sources, by which each source model contains a set of

states each associated with a small PLCA basis, and where the temporal dependencies between the bases, i.e. basis-to-basis transitions, are modeled by a state transition matrix. After training an instance of such a structure for each source using the EM algorithm, the mixture is decomposed in a factorial N-HMM architecture by which the current state for each source is computed, and given these states the mixture is decomposed as a weighted combination of a composite basis comprised of the bases associated with the these states. The use of temporal dynamics in these source models is demonstrated to improve separation performance over no temporal dynamics. A similar idea using a single spectral vector for each state basis and using the NMF multiplicative updates is proposed in [358]

The model-based MSSS paradigm has also been proposed to address speech enhancement for non-stationary interference sources. In [359], the specific problem of wind noise is tackled using Sparse NMF (multiplicative updates) in the magnitude spectral domain; the author reports that this representation often leads to better performance than the power spectral domain. For this approach a basis for the wind noise is trained a priori on representative training data, and the target speech basis is simply initialized with nonnegative random numbers in an effort to make it speaker independent. The mixture of speech and wind noise is then decomposed onto the composite basis of the wind basis and the speaker basis by way of a customized sparse NMF procedure, for which the wind basis is held fixed while the remaining factors, including the speech basis, are optimized. This approach was demonstrated to have performance comparable to existing speech enhancement algorithms as measured by objective measures. The influence of sparsity, controlled by the regularization parameter, was examined and was shown to effect a negligible improvement in terms of objective performance measures, but the author reports that increased sparsity greatly improves the subjective quality of the output speech. Wind noise is also addressed in the model-based MSSS framework in [360], in which sparse CNMF is used as the matrix factorization technique.

For NMF model-based MSSS, the information encoded in the gain matrices that are learned alongside the source bases during their training, and which is typically discarded, is utilized for model-based MSSS speech enhancement in [361]. The rows of these gain matrices contain information regarding the co-occurrence of the source basis vectors over the frames of the training data. This temporal information was extracted by taking the row-wise means and covariance matrices of each training gain matrix; learned with the source bases using the $C_{\mathrm{KL}}(\cdot)$ cost function an the multiplicative NMF algorithm. The multiplicative NMF algorithm used to decompose the mixture is also based the $C_{\mathrm{KL}}(\cdot)$ cost function but is regularized with a penalty term that serves to encourage a weighting of the basis vectors that is consistent with the temporal statistics, i.e. the co-occurrence of two basis vectors in a mixture that do not co-occur in the training data is discouraged. For the evaluation of this technique, two types of speaker bases were employed; a speaker independent basis that was trained on utterances sourced from a variety of different speakers, these speakers where not used for the test data; and a speaker specific basis, one for each speaker, each trained separately on isolated

examples of that particular speakers speech. A wide variety of noise sources were employed, with a separate noise specific basis trained for each. Perceptual relevant objective measures reveal that compared to NMF the proposed regularized NMF approach produces significantly higher quality output speech for both types of bases for noise sources dissimilar from speech; both algorithms where comparable when speech babble was used as a noise source. It was also found that despite the more general spectral shapes stored in its columns, the independent speaker basis is efficacious, for both NMF and regularized NMF, and achieves performance close to that of the dependent speaker basis for many noise sources. This technique was extended to employ additional inter-frame temporal information in [362], with noticeable increases in performance.

Speech enhancement for non-stationary noise is also addressed in model-based MSSS framework in [363]. In this work, each source basis is trained by factorizing a training spectrogram using the K-SVD algorithm [364], which is a technique devised for learning sparse over complete bases. The resulting mixture basis is then used to explain the mixture using Lasso regression, which results in a sparse linear combination of the composite basis column vectors. Final separation is performed using the Wiener-like filter defined by the source magnitude spectrograms rather than the source power magnitude spectrograms. This approach was evaluated for speaker independent and speaker dependent bases for a range of noise sources. It was found to have superior performance to comparable speech enhancement algorithms, with speaker dependent speech performing better than speaker independent basis.

Conventional speech enhancement algorithms assume that the noise source is stationary or quasi-stationary such that a model for the noise can be learned or updated during detected pauses in the speech signal. In [365] it is proposed to apply this approach to non-stationary noise signals by learning or updating an over complete basis for the noise source during detected non-speech periods using nonnegative latent variable decomposition. This model is then used to jointly estimate the speech and noise. This approach was shown to significantly outperform conventional spectral subtraction particularly if the noise is highly non-stationary.

Sparse Robust Convolutive NMF (SRCNMF) is presented in [211] for speech enhancement and for learning speech spectral features in the presence of noise. The SRCNMF basis is a composite of a speaker basis, initialized with nonnegative random numbers, and a noise basis, which is an identity matrix. SRCNMF is an unsupervised approach and does not employ speaker or noise specific training data, instead, the spectrogram of a noisy speech mixture is approximated by a linear combination of its basis vectors, which is computed using a customized CNMF multiplicative algorithm, for which a sparsity constraint is imposed on the speech component of the gain matrix, and a smoothness constraint is imposed on the noise gain matrix; all the factors are updated during this procedure except the noise basis. By imposing a sparsity constraint on the speech section of the coefficient matrix the speech basis has a proclivity to learn dense spectral features in the mixture, those exhibiting regularity across frequency and time, i.e. speech; in conjunction with this, by imposing a smoothness

constraint on the noise portion of the coefficient matrix the noise basis is encouraged to represent slowly time varying spectral energy, such as noise. This configuration of constraints and basis vectors engenders SCRNMF to segregate the noise and speech spectral features of the mixture onto their respective bases. With the sources segregated, the estimate of the enhanced speech signal is resynthesized from the result of subtracting the estimate of the noise from the mixture. In an experimental comparison study, SRCNMF learns more speech-like features from the noise than SCNMF, and produces more enhanced speech than SCNMF but slightly less than a spectral subtraction technique. It was also shown that the performance of SRCNMF is heavily influenced by the choice of sparsity and smoothness regularization terms.

Speech enhancement is also addressed in the model-based framework in [366]. For this approach, it is proposed to train an NMF basis for the noise source only. Then, to separate the sources given this basis, a novel NMF cost function is proposed that incorporates a disjointness constraint that is imposed by a penalty term that measures the disjointness between the speech and noise source models, i.e. between $\mathbf{B}_1\mathbf{g}_1(t)$ and $\mathbf{B}_2\mathbf{g}_2(t)$, where the speech basis has been randomly initialized. Multiplicative updates are derived for this cost function, and during the optimization procedure the noise basis is held fixed. The proposed approach is shown to yield significantly higher performance than sparse NMF for the same numbers of basis vectors.

Independent Subspace Analysis (ISA) was proposed for MSSS in [367]. ISA applies ICA to the subband signals of the magnitude or power STFT of a mixture, effectively treating each subband signal as a separate mixture, thereby overcoming the limitation of having only a single mixture. Moreover, unlike traditional ICA, the two factors generated by ISA are considered as a basis and a gain matrix, rather than as a demixing system and estimates of the sources. ISA was proposed for the unsupervised MSSS problem, i.e. no prior knowledge of the sources. A low rank ISA decomposition of the mixtures' spectrogram was computed, from which source estimates are formed by grouping the resulting basis vectors based on the Kullback-Liebler differential entropy. ISA has also been applied to other related problems such as musical note transcription [368], and drum transcription [369, 370], with [371] also containing a comparative description of the various features captured by PCA and ISA when applied to the spectrogram of audio signals. The conditions under which ISA-based monaural mixture source separation can be performed were examined in [372], where it was shown that the sources are required to be reasonable spectrally disjoint in order to allow separation from the mixture.

It is proposed in [373-375] to use ICA in the time-domain for model-based MSSS, which represents an exception to the spectral based MSSS algorithms that are prevalent in the literature. In this approach, ICA is used to learn a basis for each source from time domain training data, i.e. a matrix consisting of columns of successive overlapped frames of speech samples. A maximum likelihood approach is then used to separate the mixture given the

REF	DOMAIN	FACTORISATION	ALGORITHM	SOURCE MODEL	MIXTURE MODEL
[344, 345]	Magnitude STFT	NMF (Kullback-Leibler)	Multiplicative updates	Speaker specific bases	Use source estimates to construct Binary Time-Frequency Mask/user directed clustering
[297]	Mel scaled time-frequency domain	Sparse NMF (Kullback-Leibler Sparseness constraint)	Multiplicative updates	Speaker specific/phoneme specific bases	Source basis estimates
[224]	Magnitude STFT	CNMF (Kullback-Leibler)	Multiplicative updates	Speaker Specific bases	Source basis estimates
[260]	Magnitude STFT	Sparse CNMF (Kullback-Leibler Sparseness constraint)	Multiplicative updates	Speaker Specific bases	Source basis estimates
[352, 353]	Normalised Magnitude STFT	PLCA (Kullback-Leibler)	EM	Speaker Specific bases	Source basis estimates
[343]	Normalised Magnitude STFT	N-HMM (Kullback Leibler)	EM	Speaker Specific bases	Source basis estimates
[359]	Magnitude STFT	Sparse NMF (Kullback-Leibler Sparseness constraint)	Multiplicative Updates	Speaker independent, wind noise specific basis	Speaker basis is permitted to update, use final estimate
[361]	Magnitude STFT	NMF, (Kullback-Leibler with temporal constraints on gain matrix)	Multiplicative updates	Speaker Specific bases.	Source basis estimates
[363]	Magnitude STFT	K-SVD Lasso-Regression	K-SVD	Speaker Specific and independent Speaker bases	Wiener-like filter
[377]	Mel scaled time-frequency domain	Sparse Coding $L1$ norm	Linear Programming	Speaker and noise specific bases	Source basis estimates
[159]	Time domain	ICA and maximum likelihood	ICA	Speaker Specific bases	Maximum likelihood estimation using source basis

Table 3.2: Summary table listing aspects of some different matrix factoisation based model-based MSSS approaches for speech mixtures.

composite basis. For certain separation problems, this approach is reported to achieve near-perfect separation given the availability of adequate training data.

The differential filtering imposed by the Head Related Transfer Function (HRTF) on sound sources emanating from different positions in space, and Sparse coding [376], are employed to perform MSSS in [159]. This approach first chooses a suitable sparse over-complete basis, and then filters each of its vectors by a filter corresponding to the effect a known HRTF imparts on a source arriving from a certain azimuth. This is then repeated for a number of different azimuths to create a differentially filtered copy of the original sparse over-complete basis for each HRFT azimuth; the bases are then concatenated to form a composite basis. The mixture is factorized as a sparse linear combination of the composite basis, where coefficient sparseness is measured using the L_{I} norm and where the factorization is efficiently computed using linear programming. Assuming each source arrives from a different azimuth, the filtering by the HRTF means that the energy of the sources clusters in the basis coefficients corresponding to their azimuth's basis. Scaling these basis vectors by their coefficients then yields an estimate of the source. This technique was further explored in [378], in which the composite basis is comprised of filtered replicas of an NMF basis, and

where the significance of this approach in explaining seemingly redundant neuronal processing in the auditory cortex is proposed.

Sparse coding is also used for model-based MSSS algorithm in [377]. For this approach, linear programming is employed to learn a sparse over-complete basis for each source, in the mel-spectrogram domain, and to find a sparse linear combination of the vectors of the resulting composite basis to approximate the mixture. This algorithm achieves approximately 8 dBs increase in source-to-interference ratio for mixtures of different genders, and 5 dBs source-to target for mixtures of the same gender.

3.4 Source modeling of Reverberant Sources via NMF

In this section, we discuss speech source modeling via NMF in the context of model-based MSSS. Our interest in this topic relates to MSSS of reverberated sources e.g. the acoustic echo problem, for which we aim to estimate reverberated source signals using models trained on non-reverberated training data. As part of this discussion, we compute a number of NMF speech spectrogram reconstructions to demonstrate some techniques that will be employed throughout the remainder of the thesis. Prior to this, it is necessary to first describe some test signals and a particular NMF algorithm.

3.4.1 Test Signals and NMF Algorithm

The speech data for the experiments was sourced from the TIMIT speech corpus [379], which is a database of phonetically rich sentences uttered by a variety of different speakers. The RIRs for these experiments, used to simulate reverberation, where sourced from the MARDY database [380], which consists of RIRs recorded at different loudspeaker microphone geometries in a varechoic chamber. Four utterances of approximately 2 seconds in length were chosen arbitrarily from two female and two male speakers. Each utterance was convolved with a unique MARDY RIR (T_{60} = 0.4 s) to produce a dataset comprised of four non-reverberated and four reverberated speech utterances which where then down-sampled to 8 kHz; note that each RIR was normalized to unit energy for illustrative purposes. The magnitude spectrogram representation is employed in these experiments, where N = 64 ms, M = 32 ms, and $q(n)$ is a hanning window. The magnitude spectrogram is routinely employed in model-based MSSS, and is reported in [359] and [212] to facilitate better MSSS performance with NMF than the power spectrogram. The Log Spectral Distance (LSD) speech distortion measure is used an objective measure of the quality for the NMF reconstructions, with each LSD value, in dBs, corresponding to the average LSD value computed over the frame-wise LSDs values of the reconstructed spectrogram; the LSD measure is defined in (4.8).

An aim of this section is to motivate NMF as a source modeling technique for reverberated sources when given non-reverberated training data. NMF was considered for this task over probabilistic source modeling techniques for the following reasons. Firstly, probabilistic model-based MSSS techniques that employ a MAP estimator select a single spectral feature or state from each source model to model the sources in the current mixture frame. This prevents spectral features other than the selected feature from contributing to the

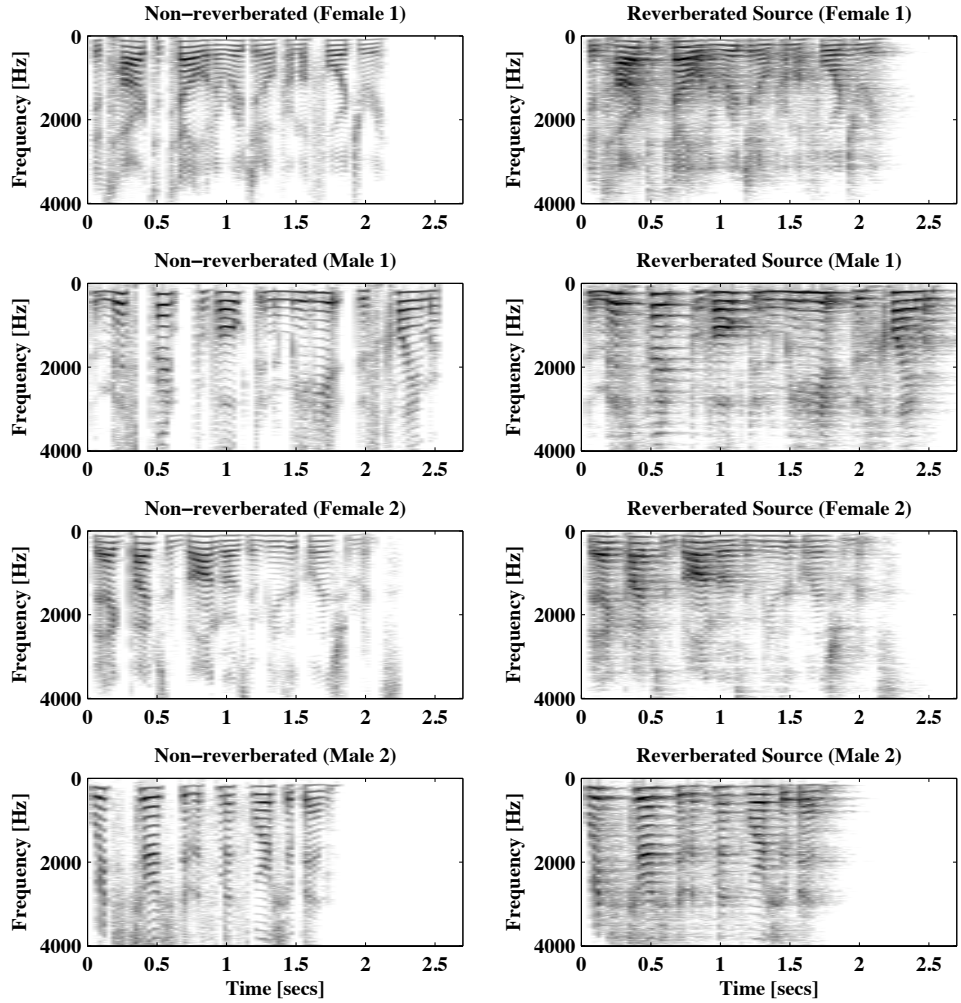

Figure 3.2: Reverberated (left column) and non-reverberated (right column) spectrograms of different male and female speech signals.

mixture, preventing combinations of such features from modeling the mixture. It follows, that this approach entails a large number of states or features to adequately model non-stationary sources such as speech, resulting in a large hardware resource requirement. In contrast, matrix factorization techniques, such as NMF, allow features comprised of combinations of the basis vectors to approximate the mixture, reducing the need for a large number of features. Secondly, and perhaps most importantly, the basic versions of the discrete state probabilistic models are inherently insensitive to scaling of their modeled features rendering such models unsuitable for reverberant speech mixtures where spectral features are typically temporally scaled; more sophisticated models in which scaling is addressed, such as the GSMM model [166] or scaled F-HMM [314], add significant algorithmic complexity. In contrast, matrix factorization techniques, such as NMF, are inherently able to deal with spectral features that temporally vary in scale.

The multiplicative NMF algorithm has been almost universally employed in audio applications, and as such, is well validated for this application domain. In particular, the multiplicative NMF algorithm with the $C_{\mathrm{KL}}(\cdot)$ cost function has been widely employed, especially for model-based MSSS [212, 224, 345, 361, 381], and is reported to provide good

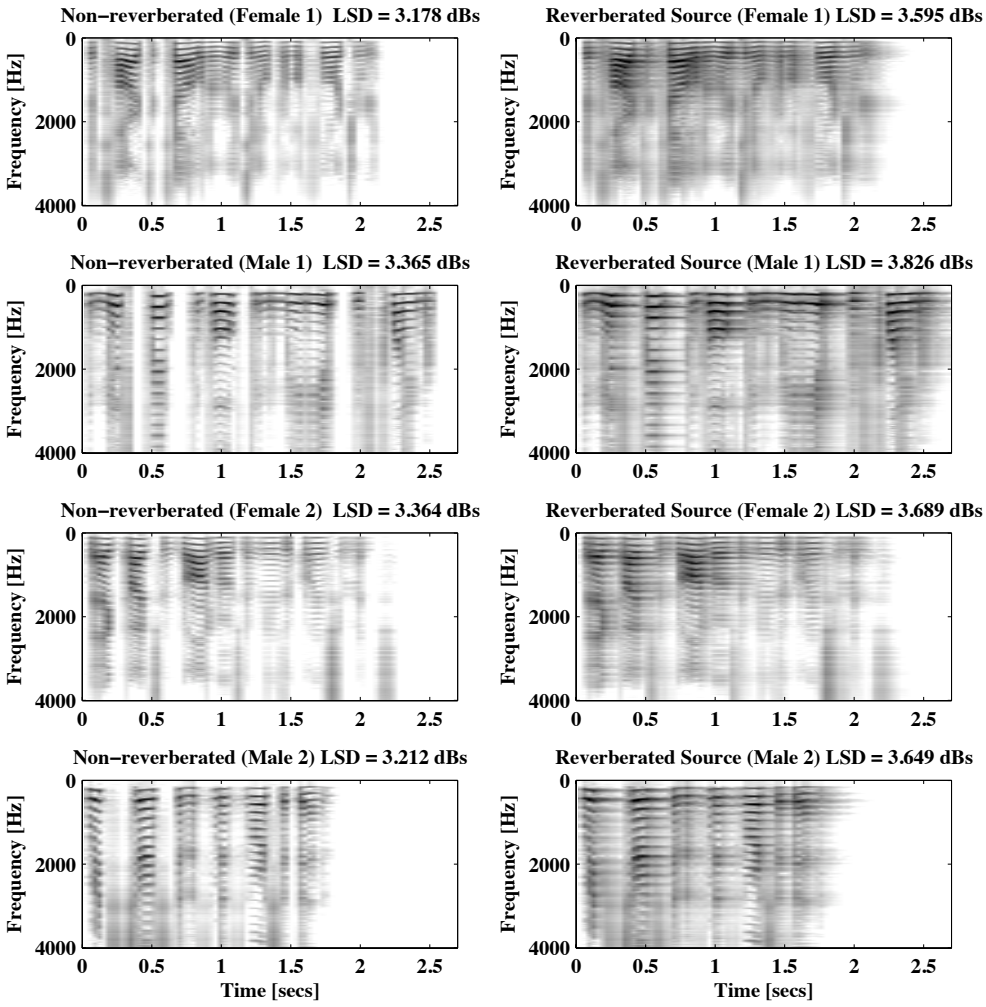

Figure 3.3: NMF reconstructions, R = 12 and 500 iterations, of the spectrograms displayed in Figure 3.2

performance for this task [361, 381]. As discussed in section 3.2.1, this popularity may be due to the perceptually relevant properties of the $C_{\mathrm{KL}}(\cdot)$ cost function, that is, it penalizes under-estimation more than over-estimation estimation, which suits perceptually relevant spectral factorization of audio signals [382]. For these reasons, and given the relatively low complexity of the multiplicative updates, as noted in [230], we employ the multiplicative NMF algorithm with the $C_{\mathrm{KL}}(\cdot)$ cost function throughout this thesis. As for auxiliary constraints, sparsity constraints on the gain matrix **G** are beneficial for model-based MSSS problems in which a large number of basis vectors are employed [297]. In this thesis, because of the application domain, we favour a minimal hardware resource requirement, and accordingly, favour a small number of basis vectors. We therefore do not consider sparsity constraints in this work. We also do not consider any other auxiliary constraints, or any convolutive extensions.

3.4.2 Source modeling of Reverberant Sources via NMF

Reverberation alters the magnitude spectrogram of clean speech signals considerable, with concomitant implications for source modeling. This is illustrated in Figure 3.2, where the magnitude spectrogram (in dBs) of the non-reverberated speech signals and those of

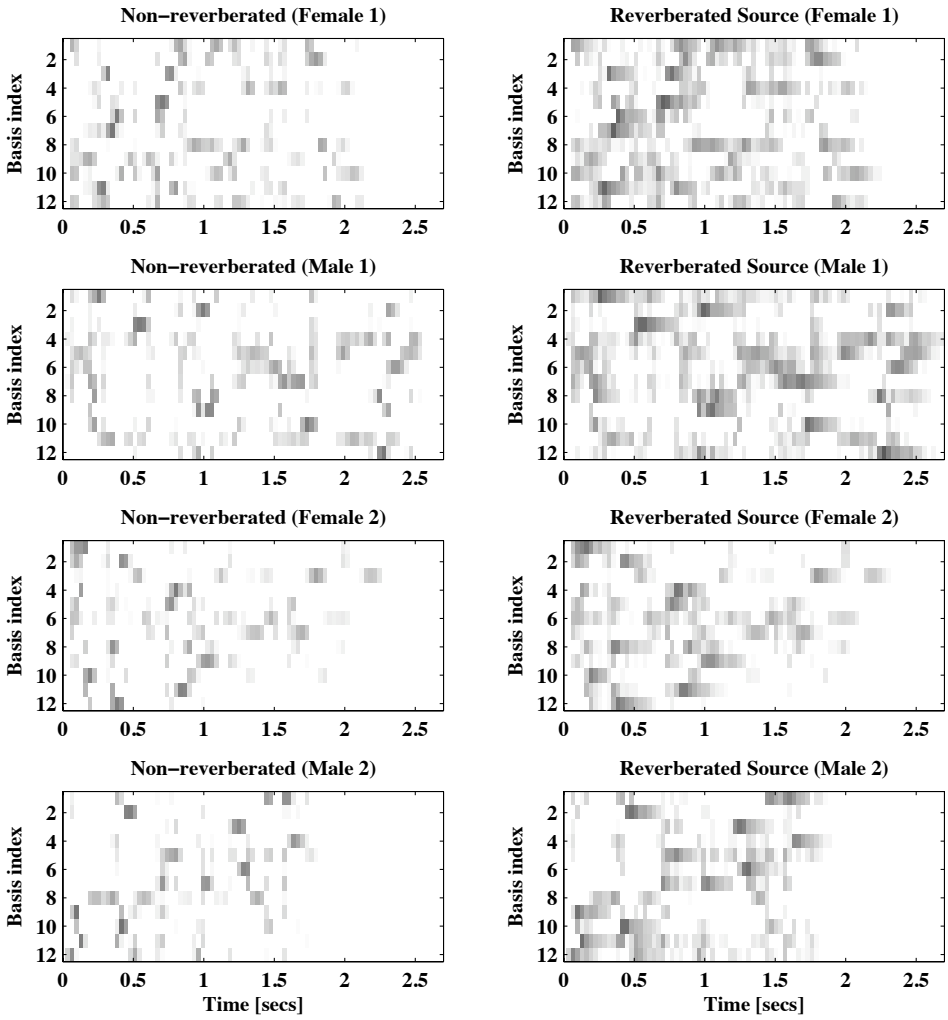

Figure 3.4: left column : Gain matrices from NMF decomposition of the non-reverberated spectrograms in Figure 3.2. Right column; Gain matrices from restricted NMF decomposition of the non-reverberated spectrograms in Figure 3.2

reverberated speech signals are displayed. Broadly speaking, it can be seen that the primary effect of reverberation is to smear the spectral content of the speech signals over time, with the profile of this smearing generally following the profile of the corresponding RIR. Another effect is coloration, which alters the spectral envelopes of the individual spectra; although, the spectral patterns of the clean speech sources are still distinguishable in the reverberated spectrograms, suggesting that non-reverberated spectral features are efficacious for model-based MSSS of reverberant sources. In terms of source modeling therefore, Figure 3.2 shows that for a source model to effectively model a reverberant speech source in the magnitude spectral domain so that good quality MSSS can be attained, it should encompass these smearing and coloration effects.

To examine the modeling capacity of NMF bases trained on non-reverberated data with respect to reverberation, we display, in Figure 3.3, the reconstructed spectrograms from the NMF (R = 12) of each of the spectrograms in Figure 3.2, where in the case of the reverberated spectrograms the reconstructions are derived from a restricted NMF procedure that uses the NMF basis yielded from the corresponding non-reverberated magnitude spectrogram; we also display the gain matrix of each spectrogram reconstruction in Figure

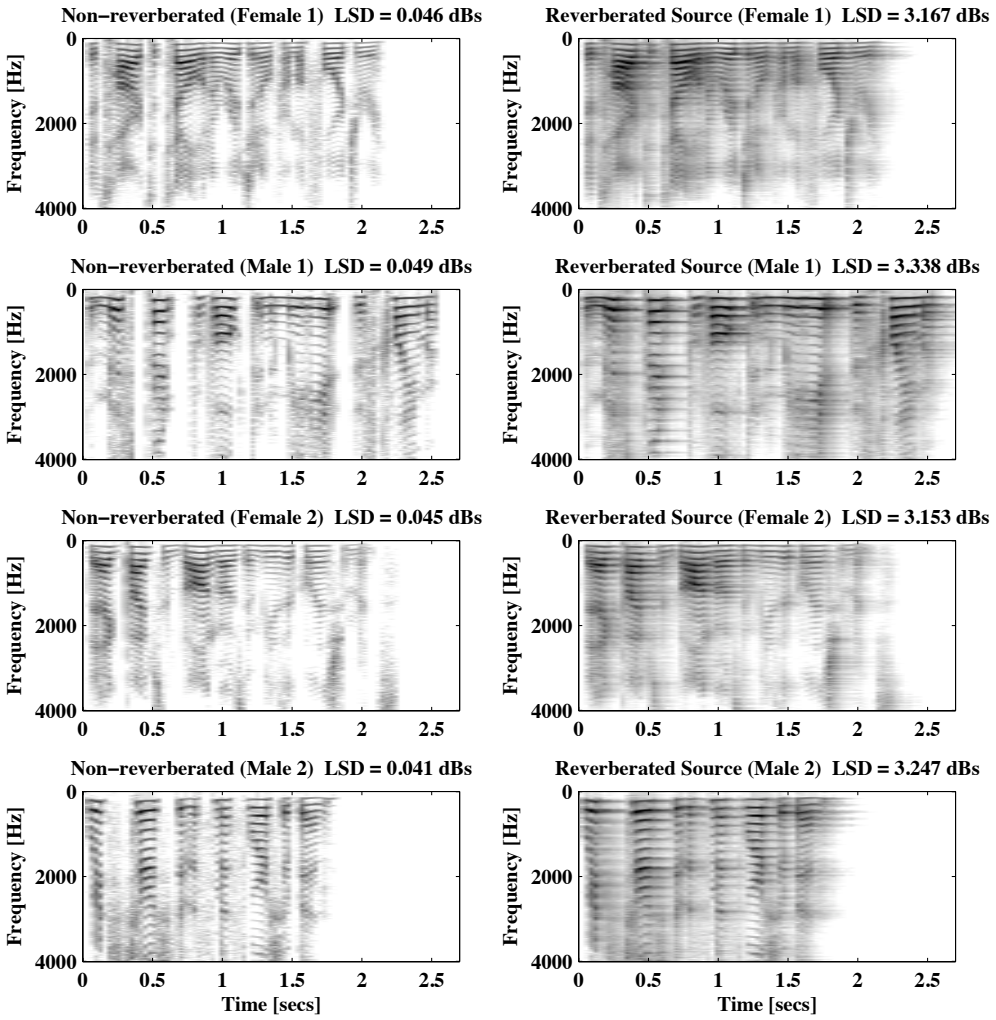

Figure 3.5: Restricted NMF reconstructions, 500 iterations and using the non-reverberated spectrograms displayed in Figure 3.2 as bases, of the spectrograms displayed in Figure 3.2

3.4. It is evident from Figure 3.3 that an accurate reconstruction of the spectrograms is generated in both the reverberant and non-reverberant scenarios. It is also evident that the effects of reverberation, such as smearing, are clearly distinguishable in the reconstruction of the reverberated spectrograms, entailing that the NMF bases encompass such features; though, comparing the LSD values, the reverberated reconstructions have slightly more distortion than the non-reverberated reconstructions. This versatility is explained by the gain matrices displayed in Figure 3.4; for non-reverberated spectrograms the gain matrices are sparse with activations usually occurring in single spikes over time, whereas for reverberated spectrograms they are considerable less sparse and contain more smeared activations, which indicates that the restricted NMF procedures choose to use repeated and scaled copies of the basis vectors to match the smeared spectral energy. A similar interpretation of reverberation with respect to NMF is proposed in [202], in which a robust speech dereverberation algorithm using a customized CNMF algorithm is proposed.

Audible renditions of the reconstructed spectrograms in Figure 3.3 (using the phases of the mixtures) reveal some distortion in the time-domain signals, which corresponds to the residuals for both sets of spectrograms; the reverberant and non-reverberated signals are

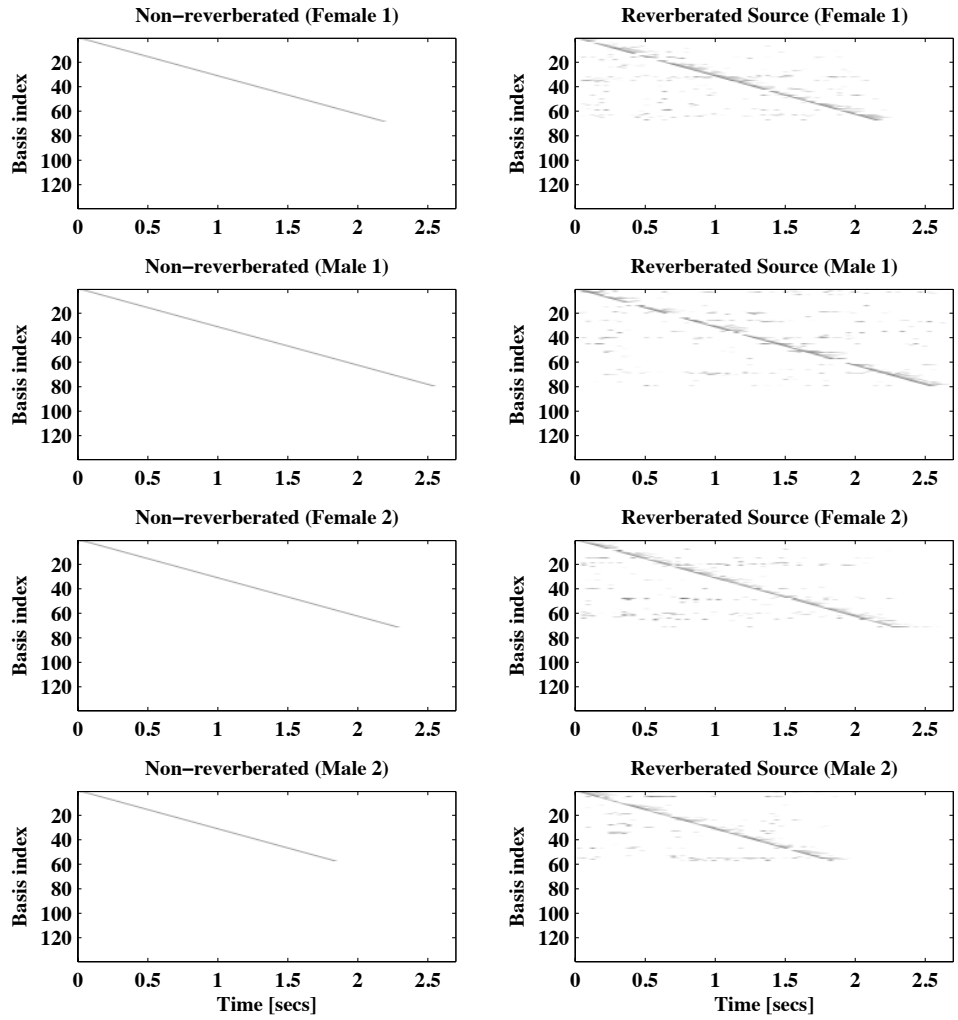

Figure 3.6: Gain matrices from the restricted NMF reconstructions displayed in Figure 3.5

clearly distinguishable however. These residuals are to be expected given the aforementioned limitations of source models, but should not be overstated, since what is important for model-based MSSS is that the sources in a mixture are sufficiently represented by their source models such that an adequate soft or hard mask can be created for the task of suppressing the interfering sources.

Another approach to NMF source modeling that we seek to explore is the concept of directly employing the training data as the source basis. This was proposed in [357] as a means of reducing the likelihood of cross-matching, and was shown to result in significantly better separation in comparison with trained bases for real mixtures. To investigate this concept for reverberated sound sources, Figure 3.5 contains reconstructions of each of the spectrograms in Figure 3.2 resulting from a restricted NMF procedure in which each non-reverberated spectrogram is used as the basis for itself and for its counterpart reverberated spectrogram; the corresponding gain matrices are displayed in Figure 3.6. As expected, in the case of the non-reverberated spectrograms, it is apparent that the restricted NMF procedure generated reconstructions that match the original spectrograms almost exactly, which is evident from the diagonal gain matrices, and the corresponding LSD values. In the case of the reverberated spectrograms, as reflected by the LSD values, the accuracy of the reconstruction

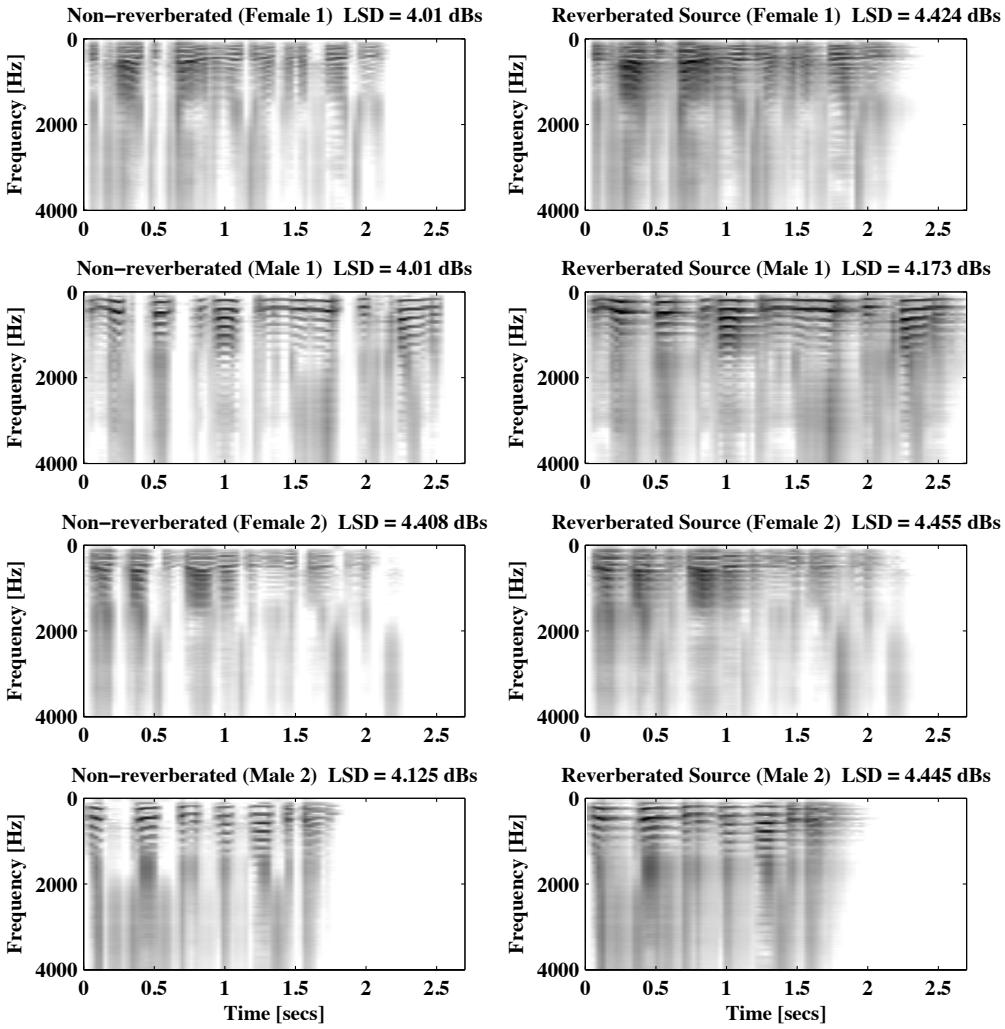

Figure 3.7: Restricted NMF reconstructions, $R = 12$ speaker independent basis and 500 iterations, of the spectrograms displayed in Figure 3.2

is limited as only a nonnegative combination of fixed non-reverberated magnitude spectra can be used to approximate each reverberant magnitude spectra; to fully model reverberation requires cross-band filters in the STFT domain, or data constraints in the frequency domain, see Chapter 2. Nonetheless, it is apparent that a reasonably accurate reconstruction of the reverberated spectrograms is achieved; moreover, such an approximation is apt for use in conjunction with a masking technique. Upon examination of the gain matrices in Figure 3.6, it is apparent that the frames of the non-reverberated spectrograms that contribute most to the reverberated spectrogram reconstruction are concentrated around the diagonal and skewed to the right, indicating that the restricted NMF choose mainly current and previous frames of the non-reverberated spectrogram to approximate the reverberated spectrogram, which is congruent with intuition regarding echo. This observation suggests a computationally efficient way of computing such a restricted NMF, where only a small number of previous frames of the non-reverberated spectrogram, are used to approximate each reverberant spectral frame. This approach will be central to the contributions in the following two chapters.

A final aspect of source modeling via NMF that we wish to explore is modeling a source for which training data is unavailable. An example of such a situation is speaker

independent speech enhancement applications. In the context of matrix factorization model-based MSSS for speech enhancement, this was tackled in [361, 363] by using a speaker independent basis that was trained on a variety of different speakers not specific to the speakers; another approach is to randomly initialize the speaker basis and then estimate it singularly when factorizing the mixture [359]. A speaker independent basis was demonstrated to be a reasonable substitute for source dependent bases, especially if the interference source is spectrally dissimilar, or if the interference source basis is highly source specific [361]. To demonstrate the utility of a speaker independent basis for our purposes, we trained an NMF basis (R = 12) on a training sequence comprised of various utterances from ten different speakers, none of the speakers are in the dataset. The reconstructions of the spectrograms from a restricted NMF using the speaker independent basis were computed and then plotted in Figure 3.7. As can be seen from Figure 3.7 the speaker independent basis is able to represent a considerable amount of the spectral energy of all the speech sources for both the reverberated and non-reverberated spectrograms, despite its lack of source specificity. As expected however, these reconstructions are visible less accurate than those in Figure 3.3 and Figure 3.5, and have higher LSD values. When used in conjunction with a masking technique, less effectual source modeling implies less accurate apportionment of the mixture, which in turn implies more distortion and/or source interference in the source estimates.

Although not investigated in this section, cross-matching is another important consideration for source modeling via NMF. Using a speaker independent basis for example, is likely to result in an increase in cross-matching due to the lack of specificity of its basis vectors. The effects of cross-matching is a topic of the next Chapter.

In the next chapter, we will address the acoustic echo problem as a MSSS problem some of the source-modeling techniques found in this section; intuitively speaking, this approach may be described as treating the acoustic echo problem in a spectral feature space. Although in comparison with existing echo mitigation approaches, some of the these source modeling techniques may appear crude and heavily constrained at first sight, even when used in conjunction with a masking technique; however, it will be demonstrated in the next chapter that these techniques can be configured to allow for an echo mitigation technique that innately addresses both DT and room change, while also being able to produce near-end speech of adequate quality. In chapter 5, a novel DT based on these techniques is also presented, while in chapter 5 all-pass phase distortion suppression is addressed by these techniques.

4 NEAR-END SPEAKER EXTRACTION USING NONNEGATIVE MATRIX FACTORIZATION

This chapter describes the application of monaural sound source separation (MSSS) techniques to the problem of single channel acoustic echo reduction. In the MSSS framework, the objective is to extract the near-end speaker's signal from a mixture containing that signal, as well as echo and noise. Separation is achieved in the magnitude Short Time Fourier Transform domain using Nonnegative Matrix Factorization to decompose spectral features of the microphone signal onto two bases of such features. An echo basis is constructed from the spectrum of the incoming far-end signal, while a speaker basis is trained on the spectra of multiple speakers a priori. An estimate of the near-end speaker's magnitude-spectrum is formed from the features of the microphone spectrum that are modeled by the speaker basis during decomposition. A time domain signal is then synthesized using the inverse Fourier transform of the estimated magnitude spectrum, together with the phase of the original mixture. The technique is quantitatively compared to existing Acoustic Echo Cancellation Double-Talk Detector (AEC-DTD) methods, in terms of echo reduction, distortion and resource requirements. The separation approach is shown to provide a consistent level of echo reduction, which is attained immediately upon initiation, and that is unaffected either by room change or double-talk.

4.1 Introduction and Background

Acoustic Echo (AE) mitigation by way of adaptive system identification naturally fits the standard linear model of AE, and as such is a theoretically appealing approach to solving this problem. However, as discussed in chapter 2, there are a number of frequently occurring operational conditions that place conflicting requirements on this approach. One such condition is Double-Talk (DT), that is, contemporaneous echo and near-end speaker activity

in the near-end enclosure: $v(n) \neq 0$ and $d(n) \neq 0$, and another is enclosure or room change, such as a door opening, a change in the near-end microphone-loudspeaker displacement, or movement within the near-end enclosure. Both of these conditions can cause the parameters of an adaptive system to diverge rapidly from optimality thereby increasing the echo disturbance for the far-end user during the ensuing convergence period. Thus, while adaptive system identification techniques are effective for the AE problem, the amount of echo reduction they provide varies.

Motivated by these problems, in this chapter we employ a novel framework for AE mitigation. In this framework, the single-channel AE problem is reframed as a Monaural Sound Source Separation (MSSS) problem, with one mixture, the near-end microphone signal $y(n)$, comprised of two sources, the near-end speaker signal $v(n)$, and the echo signal $d(n)$. The goal within this framework is to extract an estimate of the signal of interest, $v(n)$, from $y(n)$, thus enabling this estimate alone to be transmitted to the far-end user, an approach we call Near-end Speaker Extraction (NSE). Such an approach is conceptually distinct from adaptive system identification techniques in that it seeks to mitigate AE by extracting $v(n)$ from the near-end output $y(n)$, rather than by estimating and subtracting an estimate of $d(n)$ from $y(n)$. An inherent virtue of this approach is that separation is performed irrespective of the status of the near-end speaker with the estimate of $v(n)$ varying between zero during near-end speaker inactivity and $v(n)$ during near-end speaker activity, thus the DT problem is innately addressed in this framework; this approach also enables room change robustness as will be described shortly.

To perform NSE, we propose separating the near-end microphone signal into its near-end speaker and echo components using Nonnegative Matrix Factorization (NMF) [6] and techniques from the field of model-based MSSS, which were described in Chapter 3. We call this novel technique NMF-NSE. The approach is formulated in the magnitude-Short Time Fourier Transform (STFT) domain, in which the near-end microphone signal, $y(n)$, or mixture in the MSSS context, is decomposed onto two bases of spectral features. The first is a speaker-independent, general basis, trained off-line on the magnitude-spectra of many sample speakers, and is recalled from memory upon commencement. The second basis is created during operation, and is continually replenished by the incoming far-end speech signal, $x(n)$. As such, the second basis is specific to the far-end speaker, and the spectral features it contains are congenial to representing the echo component, $d(n)$, of the near-end microphone signal, $y(n)$. Given the two bases, NMF is then employed to approximate the near-end microphone mixture as a combination of vectors from each basis, by minimizing a cost function that measures the divergence of the approximation from the actual. Once the optimal combination of vectors from each basis has been identified, echo reduction is realized by inverse transforming only those vectors drawn from the first basis (using the phase from the complete mixture). In contrast with adaptive system identification, NMF-NSE does not invoke a model of the room response; as such, it will be shown to provide a relatively constant

level of echo reduction, including during room change and initiation, without the need for DTD.

The remainder of the chapter is organized as follows: in the next section, we describe the NMF-NSE method along with its hardware requirements. In section 4.3, the proposed method is evaluated experimentally over two stages: the first stage, section 4.3.2, investigates the influence of various NMF-NSE parameters on the overall level of echo reduction achieved and distortion during DT, and the second stage, 4.3.3, compares NMF-NSE performance and characteristics to those of established AEC-DTD algorithms, in terms of echo reduction, introduced distortion and computational load. Note that in the application of NMF to model-based MSSS, the columns of **A** typically contain successive frames of speech magnitude-spectra. In the next section, NMF-NSE will perform separation in a frame-wise manner; the matrix **A** will contain just a single column.

4.2 Nonnegative Matrix Factorization Near-end Speaker Extraction

4.2.1 Formulation of NMF-NSE

To begin, ignoring noise for the present, we adopt the model of AE that is commonly employed in the Acoustic Echo Suppression (AES) literature [130, 131, 135, 383],

$$|Y(f,\,k)| = |D(f,\,k)| + |V(f,\,k)|, \tag{4.1}$$

where $|Y(f, k)|$ is the STFT of $y(n)$, f denotes discrete frequency, k is the frame index, and $|\cdot|$ is the magnitude of a complex value. The transform of $y(n)$ is taken using a length N hanning window, advancing in steps of size m. The terms $|V(f, k)|$ and $|D(f, k)|$, likewise represent the $v(n)$ and $d(n)$ components of the mixture in the magnitude STFT domain. Such a model satisfies our need for nonnegativity and linearity of the terms, and is routinely employed for model-based MSSS. Though this model of AE does not follow from (1.4), its utility has been well demonstrated in [130, 131, 135, 383]. However, it does follow from (1.4) under the assumption that speech signals possess pair-wise disjoint supports in the STFT domain [155], as discussed in section 3.3.2.

Using the model (4.1), speaker extraction by NMF is carried out on each frame separately, which we divide into two phases: *Separation* and *Residual Elimination*.

Separation For each frame k, the column vector $\mathbf{y}(k)$ is defined to contain the $N/2+1$ unique values of $|Y(f, k)|$, which is symmetric about $f = N/2$. Thus for each frame $\mathbf{y}(k) = \mathbf{v}(k) + \mathbf{d}(k)$, and we seek approximations $\hat{\mathbf{v}}(k)$ and $\hat{\mathbf{d}}(k)$ through separation of $\mathbf{y}(k)$. The parameterization k has been retained (and is used henceforth) to denote the variables' frame dependence.

Now following the model-based MSSS framework described in section 3.3.4, we wish to create two source bases a priori, such that the mixture $\mathbf{y}(k)$ can be expressed in the form of (3.55), i.e. as an additive combination of the vectors in the union of the two source basis, plus a residual term. To train a basis for the echo signal, we note that the AE problem is singular among MSSS problems in that the far-end speaker signal $X(f, k)$, from which the source

$D(f, k)$ is composed, is observable. To exploit the availability of $X(f, k)$ we deviate from the model-based MSSS framework and create an echo basis once per frame. Motivated by the analysis in section 3.4.2, rather than train an echo basis per frame, which incurs a significant computational expoesne, for each frame the echo basis, $\mathbf{B}_d(k)$, is constructed directly from the $N/2+1$ unique values of the current and previous $R_d - 1$ frames of the far-end speaker signal $|X(f, k)|$. This approach results in substantially reduced computational load in comparison with a trained basis; moreover, in practice we found $\mathbf{B}_d(k)$ to have comparable performance to a trained basis.

To ensure that the time resolution of the basis $\mathbf{B}_d(k)$ is sufficiently fine to model $\mathbf{d}(k)$ as a combination of its columns, the far-end speaker signal $x(n)$ is processed in length-N frames with a step size $m_x \leq m$, with a corresponding frame index k_x.

To train a speaker independent basis, that is not specific to any, as yet unknown, near-end speaker, a rank R_v NMF is performed on the magnitude-spectrogram of a training sequence, comprised of phonetically rich utterances spoken by a variety of speakers. The process yields a gain matrix, which is discarded, and a basis matrix $\mathbf{B}_v$, the speaker basis, that is to be recalled from memory upon commencement of speaker extraction and at the start of every frame thereafter.

Defining a composite basis, given by the block matrix $\mathbf{B}(k) = [\mathbf{B}_v\ \mathbf{B}_d(k)]$, and defining a corresponding gain vector $\mathbf{g}(k)$, of length R_v+R_v, initialized with small, nonnegative, random entries for each frame k, separation of $\mathbf{y}(k)$ is achieved by a restricted NMF procedure, whereby the gain vector $\mathbf{g}(k)$ is subject to ϕ iterations of its update rule, during which the composite basis $\mathbf{B}(k)$ is held fixed. The update rule for $\mathbf{g}(k)$ is given by,

$$\mathbf{g}(k) \leftarrow \mathbf{g}(k) \circ \frac{\mathbf{B}(k)^{\mathrm{T}}\left[\dfrac{\mathbf{y}(k)}{\mathbf{B}(k)\mathbf{g}(k)}\right]}{\mathbf{B}(k)^{\mathrm{T}}\mathbf{1} + \delta}, \tag{4.2}$$

where δ is a small positive regularization term to prevent division by zero, and ϕ is chosen such that the cost function has equilibrated for a fixed $\mathbf{B}(k)$ ($50 \approx \phi$).

The restricted NMF procedure results in a factorization of $\mathbf{y}(k)$ of the form in equation (3.55), such that $\mathbf{y}(k)$ can now be expressed as,

$$\mathbf{y}(k) = \mathbf{B}(k)\mathbf{g}(k) + \mathbf{e}(k) = \left[\mathbf{B}_v\ \mathbf{B}_d(k)\right]\begin{bmatrix}\mathbf{g}_v(k)\\ \mathbf{g}_d(k)\end{bmatrix} + \mathbf{e}(k), \tag{4.3}$$

where $\mathbf{e}(k)$ is a residual term.

In conventional model-based, two source specific, trained bases would be employed to separate the sources; this is not possible in the AE setting, since the near-end speakers are not known in advance. However, the strong congeniality of $|X(f, k)|$ and $|D(f, k)|$ is sufficient to produce the desired separation, which is given by the multiplication of terms in (4.3),

$$\mathbf{y}(k) = \mathbf{B}_v\mathbf{g}_v(k) + \mathbf{B}_d(k)\mathbf{g}_d(k) + \mathbf{e}(k). \tag{4.4}$$

Though $\mathbf{B}_v$ is a generic basis, the speaker specific nature of $\mathbf{B}_d(k)$ means that its vectors provide an approximation of $\mathbf{d}(k)$ and offer almost no explanation of $\mathbf{v}(k)$. In choosing

features (vectors) from $\mathbf{B}_v$ and $\mathbf{B}_d(k)$ to represent those of $\mathbf{y}(k)$, the restricted NMF procedure results in a relatively small amount of erroneous vector selection, or *cross-matching error*, where vectors from $\mathbf{B}_d(k)$ are chosen to represent $\mathbf{v}(k)$ and where vectors from $\mathbf{B}_v$ are chosen to represent $\mathbf{d}(k)$. However, as is typical at the corresponding stage of conventional model-based MSSS techniques, significant energy resides in $\mathbf{e}(k)$ after the restricted NMF procedure, which accounts for the features of the speaker and echo spectra that were not assigned to either basis during separation, or indeed over-estimated features of $\mathbf{y}(k)$. Using the approximation $\hat{\mathbf{v}}(k) = \mathbf{B}_v\mathbf{g}_v(k)$ at this point would produce excessively distorted near-end output speech.

Residual Elimination To eliminate the energy in the residual $\mathbf{e}(k)$, for each frame, an additional unrestricted NMF with ψ iterations, is carried out, during which both the basis $[\mathbf{B}_v\ \mathbf{B}_d(k)]$ and the recently identified $\mathbf{g}(k)$ are modified. The update rule for $\mathbf{g}(k)$ is given in (4.2) while the update for $\mathbf{B}(k)$ is,

$$\mathbf{B}(k) \leftarrow \mathbf{B}(k) \circ \frac{\left[\frac{\mathbf{y}(k)}{\mathbf{B}(k)\mathbf{g}(k)}\right]\mathbf{g}(k)^{\mathrm{T}}}{\mathbf{1}\mathbf{g}(k)^{\mathrm{T}} + \delta}, \tag{4.5}$$

At this point the vectors in $\mathbf{B}_v$ are modified, but the modifications will be discarded at the completion of the frame, with $\mathbf{B}_v$, $\mathbf{g}(k)$ and $\mathbf{B}_d(k)$ all reinitialized in the succeeding frame such that the separation and approximation quality of one frame has no bearing on that of the next; this also serves to mitigate the build up of numerical error. Element-wise convergence of $\mathbf{e}(k)$ to 0 is rapid during the unrestricted NMF, and in practice, $\psi = 3$ is suffice. By adopting this approach the approximation error $\mathbf{e}(k)$ is effectively eliminated and $\mathbf{y}(k)$ can thus be expressed as,

$$\mathbf{y}(k) = \hat{\mathbf{v}}(k) + \hat{\mathbf{d}}(k), \tag{4.6}$$

Our intuition behind the unrestricted NMF procedure may be described as follows. By virtue of their initialization, $\mathbf{B}_d(k)$ and $\mathbf{B}_v$, though in particular $\mathbf{B}_d(k)$, contain basis vectors that are somewhat congenial to expressing their respective sources in $\mathbf{y}(k)$, such that a good separation may be stated as corresponding to a critical point of the cost function for which $\mathbf{y}(k)$ is completely expressed without extensive modifications to the initialized $\mathbf{B}(k)$. During the restricted NMF with ϕ updates the salient components of $\mathbf{y}(k)$ are expressed by the initial basis vectors, which, by steering the solution towards a neighborhood of critical points that correspond to less extensive adjustments to $\mathbf{B}(k)$, establishes a solution away from critical points that correspond to large modifications to $\mathbf{B}(k)$ and therefore good separations. The ψ unrestricted NMF iterations ($2 \approx \psi$), which modify both $\mathbf{B}(k)$ and $\mathbf{g}(k)$ to model the remaining fine details or residual of $\mathbf{y}(k)$, are then employed to converge (rapidly) to a critical point in this neighborhood, leaving zero residual and separated sources.

As described in Chapter 3, MSSS techniques employ various masking approaches to residual elimination; NMF is not among them, however, we found the unrestricted NMF procedure to have comparable performance to adaptive Weiner filtering constructed using the

source model estimates $\mathbf{B}_v\mathbf{g}_v(k)$ and $\mathbf{B}_d(k)\mathbf{g}_d(k)$. For this reason, and because of its novelty, we feature it in this thesis.

Speaker Extraction Speaker extraction (and thus acoustic echo mitigation) is accomplished by synthesizing the time-domain frame, by the IFFT, using the magnitude vector $\hat{\mathbf{v}}(k)$ (after residual elimination) together with the phase of $Y(f, k)$. Overlapping and adding successive time-domain frames forms the output speech signal, $\hat{v}(n)$, with much of the echo removed. An algorithmic summary of NMF-NSE is provided in Table 4.1.

4.2.1.1 Distortion of the Output

Distortion of the synthesized speech occurs when features in $\mathbf{v}(k)$ are represented by vectors from the echo basis and are thus erroneously omitted from the output speech. No distortion of the output occurs when $\mathbf{d}(k) = \mathbf{0}$, provided the parameter R_d is chosen such that the time spanned by $\mathbf{B}_d(k)$ is less than or equal to the duration of the room response, i.e. $R_d \leq (\tau - m_x)/(N - m_x)$. Then, $\mathbf{B}_d(k) = \mathbf{0}$ when $\mathbf{d}(k) = \mathbf{0}$, and the restricted NMF procedure may choose only vectors from $\mathbf{B}_v$ to represent $\mathbf{v}(k)$ during the separation phases. Thus, when the unrestricted NMF cycle begins, $\mathbf{B}(k) = [\mathbf{B}_v\ \mathbf{0}]$ and $\mathbf{g}(k) = [\mathbf{g}_v(k)\ \mathbf{0}]^T$, the zero values in $\mathbf{g}(k)$ and $\mathbf{B}(k)$ cannot change, since the update functions, given in (4.2) and (4.5), are element-wise multiplicative. With the elimination of $\mathbf{e}(k)$ during unrestricted NMF, (4.4) reduces to $\mathbf{y}(k) = \mathbf{B}_v\mathbf{g}_v(k)$, and so $\mathbf{v}(k)$ is fully represented by $\hat{\mathbf{v}}(k)$, precluding distortion.

When $\mathbf{d}(k) \neq \mathbf{0}$, during the restricted NMF procedure, some features of $\mathbf{y}(k)$ that are attributable to $\mathbf{v}(k)$, will be incorrectly modeled by vectors chosen from the echo basis, $\mathbf{B}_d(k)$, an error we call speaker matching. At other times, echo matching will occur, for which vectors from $\mathbf{B}_v$ will be incorrectly selected to represent features contributed to the mixture by $\mathbf{d}(k)$. The former is manifest in the output speech as distortion, and the latter as echo; collectively referred to as cross-matching. Moreover, during the unrestricted NMF some additional cross-matching may also occur. Thus an amount of distortion is introduced during DT (Note that the scenario $\mathbf{d}(k) \neq \mathbf{0}$, collectively includes both instances of doubletalk and only echo, when the near-end speaker is inactive). The integrity of the separation performance therefore depends only on the suitability of the bases and is uninfluenced by changes to $\mathbf{h}$.

During DT, a constant amount of signal distortion is associated with returning to the time domain, using the phase of $Y(f, k)$, in place of the unknown phase of $V(f, k)$, in the IFFT. However, this distortion is known to be of minimal perceptual significance [276].

The influence of the various parameters of NMF-NSE on the level of echo reduction and distortion, or equivalently, the level of speaker-matching and echo-matching, attained by NMF-NSE is examined in a series of experimental study described in section 4.3.2. Distortion and echo are also quantitatively compared to that of AEC-DTD systems in section 4.3.3.

4.2.1.2 Noise

With the inclusion of noise again, $\mathbf{y}(k)$ contains a third signal for which the separation procedure has no basis; consequently, when $\mathbf{d}(k) \neq \mathbf{0}$, any noise present is divided, in some

ALGORITHM STEP	ARITHMETIC OPS	MEMORY
Process far-end Signal x		
Step-size m_x, frame index k_x		
$X(f, k_x)$,	$*R_d\left(2N\log_2(N) - \frac{3N}{2} - 4 + N\right)$	N
$\lvert X(f, k_x)\rvert$	$*R_d\left(\frac{N}{2} + 1\right)$	Stored in $\mathbf{B}_d$
Create echo basis $\mathbf{B}_d(k)$		
For each frame k and step-size m, such that $mk = m_x k_x$,		
$\mathbf{B}_d(k) = [\,[\lvert X(0, k_x)\rvert, \lvert X(1, k_x)\rvert \ldots, \lvert X(N/2, k_x)\rvert]^{\mathrm{T}},$ $[\lvert X(0, k_x-1)\rvert, \lvert X(1, k_x-1)\rvert \ldots, \lvert X(N/2, k_x-1)\rvert]^{\mathrm{T}} \ldots,$ $[\lvert X(0, k_x-R_d-1\rvert, \lvert X(1, k_x-R_d-1)\rvert \ldots, \lvert X(N/2, k_x-R_d-1)\rvert]^{\mathrm{T}}\,]$		$R_d\left(\frac{N}{2} + 1\right)$
$\mathbf{B}_v$ (Near-end basis)	Computed offline	$R_v\left(\frac{N}{2} + 1\right)$
$\mathbf{B}(k) = [\mathbf{B}_v\ \mathbf{B}_d(k)]$		
$Y(f, k)$	$2N\log_2(N) - \frac{3N}{2} - 4 + N$	N
$\lvert Y(f, k)\rvert \angle Y(f, k)$	$6\left(\frac{N}{2} + 1\right)$	$\left(\frac{N}{2} + 1\right)$
$\mathbf{y}(k) = [\lvert Y(0, k)\rvert, \lvert Y(1, k)\rvert, \ldots, \lvert Y(N/2, k)\rvert]^{\mathrm{T}}$		$\left(\frac{N}{2} + 1\right)$
Initialize $\mathbf{g}_y$ with random nonnegative numbers	Computed offline	
Separation:		
For $l = 1 : 1 : \phi$ (restricted NMF updates) $+\psi$ (unrestricted NMF updates) **do**		
$\mathbf{g}(k) \leftarrow \mathbf{g}(k) \circ \frac{\mathbf{B}(k)^{\mathrm{T}}\left[\frac{\mathbf{y}(k)}{\mathbf{B}(k)\mathbf{g}(k)}\right]}{\mathbf{B}(k)^{\mathrm{T}}\mathbf{1} + \delta}$	$\left(R_d + R_v\right)\left(\frac{5N}{2} + 4\right) + \frac{N}{2} + 1\right)(\phi + \psi)$	$5\left(R_d + R_v\right) + N + 2,$
If $l > \phi$ **do** (Residual elimination)		
$\mathbf{B}(k) \leftarrow \mathbf{B}(k) \circ \frac{\left[\frac{\mathbf{y}(k)}{\mathbf{B}(k)\mathbf{g}(k)}\right]\mathbf{g}(k)^{\mathrm{T}}}{\mathbf{1}\mathbf{g}(k)^{\mathrm{T}} + \delta}$	$\left(\left(R_d + R_v\right)(3N + 4) + \frac{N}{2} + 1\right)\psi$	$\left(\frac{5N}{2} + 4\right)\left(R_d + R_v\right) + N + 2,$
End if		
End for		
$\hat{\mathbf{v}}(k) = \mathbf{B}_v\mathbf{g}_v(k)$	$R_v(N + 1)$	$\frac{N}{2} + 1$
Speaker extraction		
$\hat{\mathbf{v}}(k) \angle Y(f, k)$	$2N\log_2(N) + \frac{N}{2} + N(N/m)$	$N/m + 1$

Table 4.1 : Algorithmic Summary of NMF-NSE with Indicative Computational Load and Memory Requirement over one frame of processing

*if $R_d > m/m_x$, R_d in this expression can be replaced with m/m_x as frames of $|X(f, k_x)|$ calculated for the $k^{\text{th}}-1$ frame can be reused.

way among the approximations $\hat{\mathbf{v}}(k)$ and $\hat{\mathbf{d}}(k)$. When $\mathbf{d}(k) = \mathbf{0}$, and $\mathbf{B}_d(k) = \mathbf{0}$, all the noise is manifest in $\hat{\mathbf{v}}(k)$.

4.2.2 Hardware Resource requirement of NMF-NSE

Accompanying the algorithm summary in Table 4.1 are expressions for the number of arithmetic operations and the memory requirements arising from each algorithm step of NMF-NSE during the processing of one frame; without reference to a specific hardware, the expressions provide an indicative rather than absolute measure of computational load and memory requirement. An arithmetic-step is considered to be any real multiplication, addition, subtraction, or division. The square root operation, arctangent, sine and cosine are also counted as one operation. Therefore, 4 operations are required to obtain the absolute value of a complex number, 2 operations to calculate its phase, 6 operations to convert a complex number from polar to Cartesian form, and 4 operations to convert from Cartesian to polar form. The number of arithmetic operations for the FFT/IFFT is taken from [37] with N additional operations incurred by applying the windowing function for the FFT and N

additional operations required to normalize the IFFT. In enumerating the memory requirements, we assume that each complex value requires two memory locations, and that for a Fourier-transform vector of length N, only $N/2 + 1$ elements are stored. Note that the computational cost of training $\mathbf{B}_v$ offline is not counted in Table 4.1 since it is not associated with any frame.

From Table 4.1, the initial ϕ updates of $\mathbf{g}(k)$ can be seen to be the predominant algorithmic step of NMF-NSE in terms of arithmetic operations, which are proportional to the values of R_d, R_v and ϕ for a given value of N. Also from Table 4.1, the memory requirement of NMF-NSE can be seen to depend predominately on the parameters N, R_d and R_v.

4.3 Performance of NMF-NSE

In this section we evaluate the performance of NMF-NSE over two stages. For the first stage (section 4.3.2), several properties of NMF-NSE are explored by examining the performance effect of some of its parameters and various test signal parameters. The results of this stage are then used to inform the choice of NMF-NSE parameters for the second stage (section 4.3.3), in which several aspects of NMF-NSE performance are evaluated by way of comparison with conventional AEC-DTD approaches. Prior to discussing the testing methods and results of each stage, it will be necessary to establish some suitable performance metrics and to describe some test signals.

4.3.1.1 Performance Measures

To appraise the overall performance of an AE mitigation approach it is appropriate to quantify, separately, the performance during periods of DT when there is concurrent echo and near-end speaker activity, and in the absence of DT. To measure the level of echo reduction in the absence of DT, in which $y(n) = d(n) + w(n)$, we employ the commonly used echo return loss enhancement (ERLE) measure defined here as

$$ERLE(n) = 10\log_{10}\left(\frac{E[y^2(n)]}{E[e^2(n)]}\right). \tag{4.7}$$

In calculating $ERLE(n)$ the expectation functions in (4.7) are estimated by the arithmetic mean over 1024 samples centered on n. The signal $e(n)$ in (4.7) is the output signal from an AEC, and will contain noise and residual echo due to filter misadjustment. To calculate ERLE for NMF-NSE, $e(n)$ in (4.7) is replaced by the NMF-NSE output signal $\hat{v}(n)$. The signal $\hat{v}(n)$ will contain the component of the noise matched onto the near-end speaker basis, and residual echo due to echo-matching error. Note that the inclusion of noise (with $\sigma^2 > 0$) precludes either expectation in (4.7) becoming zero.

During periods of DT, the perceptual quality of the output near-end speech is paramount, and it is therefore appropriate to characterize algorithm performance during DT by a measure of speech distortion. The Log Spectral Distance (LSD) is one such objective measure of speech distortion used to assess speech enhancement algorithms [384]. During DT the LSD in dB between each input and output frame of speech is calculated using the expression

$$LSD(k) = \sqrt{\frac{\sum_{f=0}^{N/2}\left(10\log_{10}\left|V(f,\,k)\right| - 10\log_{10}\left|\hat{V}(f,\,k)\right|\right)^2}{N/2+1}}\,. \tag{4.8}$$

For NMF-NSE, $LSD(k)$ will capture distortion, due to matching of $\mathbf{v}(k)$ onto $\mathbf{B}_d(k)$ i.e. speaker matching, residual echo due to matching of $\mathbf{d}(k)$ onto $\mathbf{B}_v$ i.e. echo matching, and noise due to matching of noise onto $\mathbf{B}_v$. For AEC, the LSD value will measure both the residual echo due to filter misadjustment as well as noise. It is recognized that, as a measure between magnitudes of spectra, LSD is insensitive to phase distortion, which is inherent to NMF-NSE through the substitution of the phases of $Y(f,\,k)$ to synthesize $\hat{v}(n)$. However, as described in section 3.3.2, the omission is justifiable here given that the typical phase distortion introduced by this substitution is known to be perceptually insignificant [276]. Note that the LSD values were calculated using a frame size of 64 ms with a frame overlap of 50%.

For the parameter study in section 4.3.2, it was necessary to gauge the level of echo, distortion, and noise in the output $\hat{v}(n)$ signals, so that NMF-NSE performance during DT may be analyzed in terms of echo matching, speaker matching and noise matching respectively. For this purpose, the BSS_toolbox evaluation framework [346, 385] was employed. This framework is commonly used to assess the performance of sound source separation algorithms, whose objectives are similar to those of NMF-NSE during DT. Given the original sources of a mixture, the BSS_toolbox returns a set of time-domain performance measures for each of the source estimates produced by a separation algorithm. These measures are correlation-based, calculated from the waveforms of the source estimates. To compute these measures for a source estimate, $\hat{s}(n)$, this signal is first decomposed such that it can be expressed as:

$$\hat{s}(n) = s_{\text{target}}(n) + e_{\text{interf}}(n) + e_{\text{noise}}(n) + e_{\text{artif}}(n)\,, \tag{4.9}$$

where, $s_{\text{target}}(n)$ is the component of the original source signal in $\hat{s}(n)$, $e_{\text{interf}}(n)$ is the error of $\hat{s}(n)$ corresponding to interference from other sources, $e_{\text{noise}}(n)$ is the error corresponding to noise in $\hat{s}(n)$, and $e_{\text{artif}}(n)$ represents the remaining error, which consists of distortion and algorithm artifact error, including model error [346]. Taking these error signals, the following performance measures are calculated:

$$\begin{aligned}
&\text{SIR}=10\log_{10}\frac{\left\|s_{\text{target}}(n)\right\|^2}{\left\|e_{\text{interf}}\right\|^2}, \qquad &&\text{SAR}=10\log_{10}\frac{\left\|s_{\text{target}}(n)+e_{\text{interf}}(n)+e_{\text{noise}}(n)\right\|^2}{\left\|e_{\text{artif}}(n)\right\|^2},\\
&\text{SNR}=10\log_{10}\frac{\left\|s_{\text{target}}(n)+e_{\text{interf}}(n)\right\|^2}{\left\|e_{\text{noise}}(n)\right\|^2}, \qquad &&\text{SDR}=10\log_{10}\frac{\left\|s_{\text{target}}(n)\right\|^2}{\left\|e_{\text{interf}}(n)+e_{\text{noise}}(n)+e_{\text{artif}}(n)\right\|^2}.
\end{aligned} \tag{4.10}$$

Signal to Interference Ratio (SIR) measures the level of interference in $\hat{s}(n)$ from the other mixture sources, Signal to Noise Ratio (SNR) measures the SNR of $\hat{s}(n)$, Signal to Artifacts ratio (SAR) measures the level of algorithm artifacts and distortion introduced by the separation algorithm in $\hat{s}(n)$, and Signal to Distortion Ratio (SDR) measures the total error in

$\hat{s}(n)$ from all contributions. To compute these measures for NMF-NSE, $\hat{v}(n)$ during DT is substituted for $\hat{s}(n)$ and, with the contemporaneous segments of the signals $v(n)$, $d(n)$, and $w(n)$; is decomposed into the aforementioned error terms, from which the ratios in equation (4.10) are calculated. In this context, SIR will indicate the level of echo interference in $\hat{v}(n)$, SNR will indicate the level of noise in $\hat{v}(n)$, SAR will measure the remaining error in $\hat{v}(n)$ due to distortion and algorithm artifacts, including modeling error, and SDR will provide an overall measure of error in $\hat{v}(n)$.

The resulting values for NMF-NSE will be influenced by the error related to the use of the mixture phases to construct $\hat{v}(n)$, and the model approximation error related to the magnitude STFT model of AE assumed in (4.1), and as such will not exclusively reflect error attributable to cross-matching. Nonetheless, given that the phase and model errors are independent of many of the parameters of NMF-NSE, and assuming that a change in echo-matching induces a proportional change in SIR, and similarly, for speaker-matching and SAR, and for noise and SNR, then these measures may be used to indicate the relative level of cross-matching, that is, SIR can be used to indicate the relative level of echo matching, SAR can be used to indicate the relative level of speaker matching, SNR can indicate the relative level of noise matched onto $\mathbf{B}_v$, and SDR will indicate the overall relative level of cross matching, i.e. the combined level of speaker matching and echo matching, and noise matched onto $\mathbf{B}_v$. Note that from its definition (4.10) SAR depends on $e_{\text{noise}}(n)$ and $e_{\text{artif}}(n)$, such that an increase in the level of either of these signals will affect a rise in averaged SAR; in a similar manner, SNR is dependent on $e_{\text{interf}}(n)$.

The BSS_toolbox measures and ERLE are expressed in dBs, with higher values indicating better performance. LSD is also expressed in dBs, but a lower value of LSD indicates better performance.

4.3.1.2 Creation of Test Signals

All speech data was taken from the TIMIT speech database [379], which is comprised of phonetically rich sentences spoken by an wide assortment of speakers. An excitation signal $x(n)$ was created by concatenating speech utterances (downsampled to 8 kHz) from a single speaker to form a 16 second sequence. An echo signal $d(n)$ was then generated by convolving $x(n)$ with an enclosure transfer function, or room impulse response (RIR), $\mathbf{h}$. RIRs were selected from the MARDY RIR database [379], which contains responses recorded in a real room that are used for the evaluation of blind dereverberation algorithms. MARDY RIRs ($RT_{60} \approx 0.4$ s) were modified to reduce the initial lag between source and receiver by 0.8 ms as the microphone and loudspeaker are typically closer in the AE problem than in the dereverberation problem. The resulting RIR was truncated at 256 ms or 2048 samples. Six different pairs of excitation and echo signals were created from 3 female and 3 male speakers. Four near-end speaker signals $v(n)$ were also created from two male and two female speakers. Each $v(n)$ signal was comprised of a speech utterance at 6 seconds, denoted as utterance I, and another at 11 seconds, denoted as utterance II. Both utterances were convolved with an RIR,

truncated at 256 ms, from the MARDY database to simulate reverberation of the near-end speaker signal as may occur during hands free operation. The signal $v(n)$ was reverberated using the same MARDY RIR as that used to produce $d(n)$, but without modifying the time of first arrival, allowing that the near-end speaker may be farther from the microphone than the loudspeaker. The average utterance duration in $v(n)$ was 2 seconds. The near-end microphone output signal $y(n)$ is formed by the sum of an echo signal $d(n)$, a near-end speaker signal $v(n)$ and a noise component $w(n) \sim N(0, \sigma^2)$. The energy in $v(n)$ relative to $d(n)$, or the near-end to far-end ratio (NFR), is defined as

$$NFR(n) = 10\log_{10}\left(\frac{E[v^2(n)]}{E[d^2(n)]}\right). \tag{4.11}$$

The expectation functions in (4.11) are estimated in practice by the arithmetic mean over 1024 samples centered on n. The a priori signal-noise ratio, SNR_{in} (so-called to distinguish it from the SNR performance measure defined above) is given by the ratio $E[d^2(n)]$ to σ^2. The six $d(n)$ signals and four $v(n)$ signals and different $w(n)$ signals were combined to form 24 distinct near-end microphone signals, $y(n)$; SNR and NFR are to be specified. A second set of near-end microphone signals, $y_c(n)$, incorporating a room change at 9 s, was constructed by the same method. The room change was introduced to the echo signals $d(n)$ by linearly fading between two sets of RIR coefficients over 256 ms (2048 samples) in such a way so that no spurious discontinuities were introduced to the signals. A similar room change was introduced to the near-end speaker's RIR at the same instant. Physically these room changes correspond to a sudden 20 cm displacement of the near-end microphone.

4.3.2 Parameter Study

The objective of this stage is to relate various NMF-NSE parameters and test signal parameters to the performance of NMF-NSE, in terms of distortion, echo, and noise residing in output $\hat{v}(n)$ signals. Consequently, this study will provide an overview of the NMF-NSE parameter space from which an informed choice of parameter value may be made. Since the parameter space of NMF-NSE is of high dimension, for tractability, the objective of this stage is accomplished through 7 separate experimental studies, each of which evaluates a different subset of NMF-NSE parameters or test parameters, with each subset inducing a pertinent aspect of NMF-NSE performance. Before describing and analyzing the results from each study we will outline the generic experimental method followed in each study, and specify a set of default parameters for NMF-NSE.

4.3.2.1 Generic Parameter Study Method

For each parameter study, two experiments were performed for each combination of prescribed parameter(s) values; the first experiment examines performance in the absence of DT, and the second experiment examines performance during DT. The effect of room changes is not considered at this stage. To examine performance in the absence of DT, NMF-NSE was applied to each $y(n)$ signal with $v(n) = 0$, ($v(n) = 0$ implies 6 distinct $d(n)$ signals, however we employ all the 24 $y(n)$ signals here which will have different noise signals) resulting in 24

$\hat{v}(n)$ sequences, from which 24 ERLE sequences were computed. A single averaged value for ERLE was then calculated by ensemble averaging over the resultant ERLE sequences, and by then time averaging the resulting ensemble averaged ERLE sequence. To examine performance during DT, NMF-NSE was again applied to each $y(n)$ signal, this time including $v(n)$ for a NFR to be specified below. From each NMF-NSE output signal, i.e. each $\hat{v}(n)$ signal, two average values of LSD were computed, the first over the frames of utterance I, and the second over the frames of utterance II. The values for utterance I and II were then averaged over all such values from each of the 24 output signals, after which the average values for utterance I and II were averaged to produce a single averaged value for LSD. Similarly, a single averaged value for SNR, SIR, SDR and SAR was computed by first calculating two values, one over the samples of utterance I and the second over the samples of utterance II, and then averaging over each output signal, and then averaging the average values for utterance I and II. The two experiments are repeated for each combination of parameter(s) in each parameter study, resulting in an averaged value for ERLE, LSD, SIR, SAR, SNR and SDR for each combination of parameter(s) in each study.

Any unspecified parameters in the following studies have the subsequent values by default, $\phi = 50$, $\psi = 3$, $R_v = 4$, $R_d = 8$, $N = 512$, $m = 256$, and $m_1 = m/2$ (128), and the default test signal parameters were $SNR_{in} = 30$ dB, and NFR = 0 dB, simulating the realistic condition of equivalent echo and near-end speaker energy during DT. The default near-end basis $\mathbf{B}_v$ was speaker independent, and was constructed by applying a rank-R_v NMF procedure with 2000 iterations to the spectrogram (64 ms frames, 50% overlap) of a 30 s speech signal comprised sentences spoken by ten different arbitrarily chosen speakers from the TIMIT database [379]; both male and female speech were employed. These speakers were not employed again throughout the evaluation for any purpose, and as such $\mathbf{B}_v$ is independent of all near-end speakers. Similarly, each speaker dependent $\mathbf{B}_v$, required for the final study, was trained from a 15 s speech signal comprised exclusively of sentences spoken by the specific speaker. These utterances were not employed in the corresponding test signals.

4.3.2.2 Study of R_v and R_d and N

This study elucidates the inherent trade-off between echo reduction and distortion (during DT) by examining the influence that different values for R_d and R_v have on NMF-NSE performance. This study also examines the effect of window size on NMF-NSE performance. The specific range of values for these parameters tested were: R_d = [1, 2…16], R_v = [1, 2…16], and N = [64, 128, 256, 512, 1024, 2048] samples. The results for this study are displayed in Figure 4.1 in which each surface plot contains the results for a particular performance measure and window size for each combination of R_v and R_d. A separate close up view of the results for $N = 512$ is displayed in Figure 4.2.

In general, the averaged ERLE results in Figure 4.1, and in Figure 4.2, indicate that in the absence of near-end speech (no DT), less echo and noise reside in the output $\hat{v}(n)$ signals for higher values of R_d (more basis vectors in $\mathbf{B}_d(k)$) and for lower values of R_v (less basis vectors in $\mathbf{B}_v$), with $R_d = 16$ and $R_v = 1$ producing the lowest averaged ERLE for each N, and

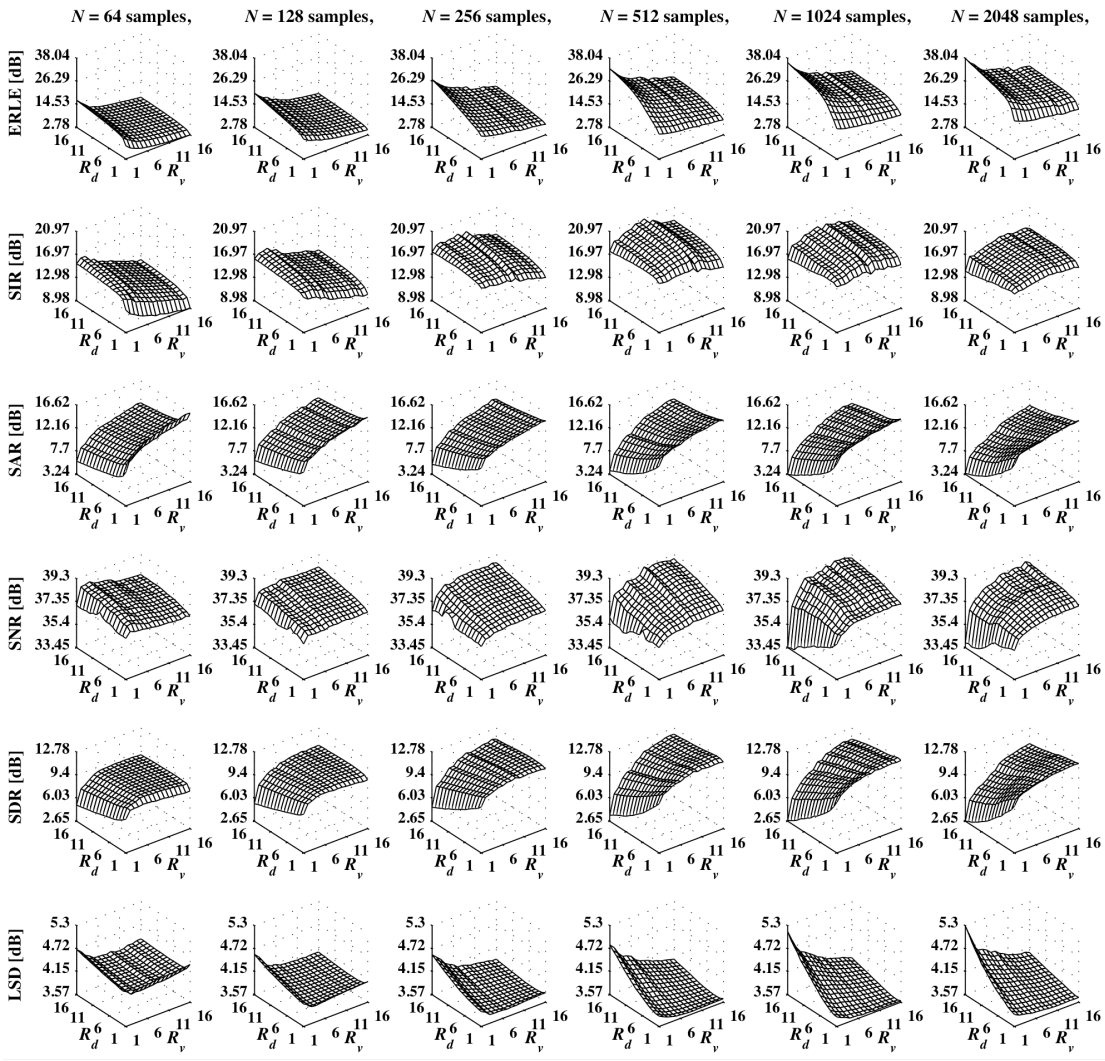

Figure 4.1: Experimental Results illustrating the influence of R_d, R_v and N on NMF-NSE performance in the absence of DT and during DT. Each column of plots displays a different value for N while each row displays a different performance measure; the z-axis label of the leftmost plot indicates the particular measure. For each row all plots are plotted across the same scale, with the lowermost and uppermost z-axis labels indicating the minimum and maximum values (rounded to nearest 1/100) attained across all R_d, R_v and N.

$R_d = 1$ and $R_v = 16$ producing the highest averaged ERLE for N. Ignoring noise for the present, the averaged ERLE values imply that by increasing the number of basis vectors in $\mathbf{B}_d(k)$, for a fixed R_v and N, less echo matching occurs. This outcome was expected since an increase in the number of basis vectors in $\mathbf{B}_d(k)$ better enables this basis to express the variability of its source $\mathbf{d}(k)$. The averaged ERLE results also imply that by decreasing the number of basis vectors in $\mathbf{B}_v$ (until the minimum of $R_v = 1$) for a fixed R_d and N echo matching is also reduced. This reduction is attributable to the reduced speech variability that a lower rank $\mathbf{B}_v$ can express; therefore, it is less able to erroneously express $\mathbf{d}(k)$. It is apparent that the rise in ERLE for decreases in R_v or increases in R_d is not consistent across all the values of N, R_v or R_d. For example, for $N = 1024$ for R_d between 1 and 7 for all R_v the averaged ERLE results exhibit a relatively sharp rise in averaged ERLE, followed by relatively small increases for $R_d > 7$, particularly for $R_v > 2$.

The averaged ERLE results also depend on N, for each increase in N from 64 to 2048 samples there is a rise in averaged ERLE for each pair of R_d and R_v values, demonstrating that

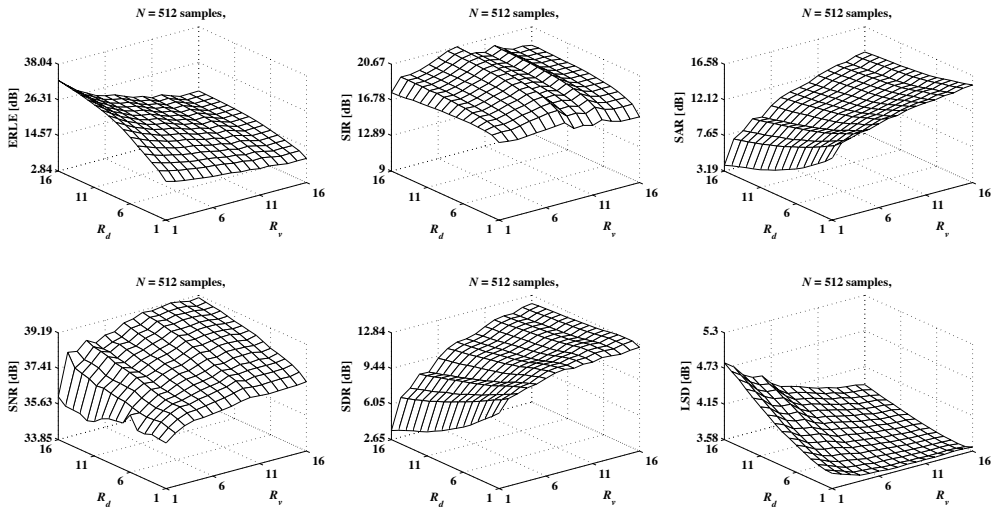

Figure 4.2: Experimental Results illustrating the influence of R_d, R_v for N = 512 samples on NMF-NSE performance in the absence of DT and during DT.

the output $\hat{v}(n)$ signals contain less echo for longer windows in the absence of DT. However, if R_d is adjusted such that $\mathbf{B}_d(k)$ spans approximately the same time interval of $x(n)$ for each N, these performance disparities are reduced, for example: for R_v = 2, R_d = 8, and N = [128, 256, 512, 1024, 2048], averaged ERLE = [15.5089, 19.7951, 25.8791, 30.1866, 32.0126] dBs respectively; while for R_d adjusted, i.e. R_d = [16, 8, 4, 2, 1] the corresponding averaged ERLE values become [18.0708, 19.7951, 20.2526, 20.8338, 21.2105] dBs. Nonetheless, it is apparent that longer frame lengths are still preferable for maximal echo reduction in the absence of DT. It is probable that less echo-matching occurs for longer frames because the basis vectors of the composite basis $\mathbf{B}(k)$ are required to fit more frequency bins, and so the more general basis vectors in the $\mathbf{B}_v$ component of $\mathbf{B}(k)$ are less likely to be matched a portion of $\mathbf{y}(k)$. Furthermore, shorter windows generate spectrums with lower resolution of their constituent spectral components, and therefore, $\mathbf{B}_d(k)$, $\mathbf{B}_v$ and $\mathbf{y}(k)$ are populated with less distinctive and more blurred spectral features, enabling the less specific basis vectors of $\mathbf{B}_v$ to more easily fit portions of $\mathbf{y}(k)$.

Assessing now the averaged SIR surface plots in Figure 4.1, which pertain to the echo reduction performance of NMF-NSE during DT; for each increase in R_d there is a slight increase in averaged SIR for all R_v, indicating lower levels of echo remaining in the output $\hat{v}(n)$ signals. A similar trend is seen in the averaged ERLE results, which would suggest, as expected, that in both the absence of, and during DT echo matching declines with an increase in R_d for a fixed R_v. As described in the context of averaged ERLE, the reductions in echo matching, are due to the enhanced ability of a higher rank $\mathbf{B}_d(k)$ to express $\mathbf{d}(k)$. The influence of R_v on the averaged SIR results is more complex, for example for N = 256 averaged SIR peaks for R_v = 9 and declines thereafter. Similar features are seen for higher values of N, though for different values of R_v in general. This feature of the averaged SIR results is somewhat surprising given that it was expected that a reduction in R_v, which is expected to consistently decrease echo matching by curtailing the speech variability $\mathbf{B}_v$ can express, would lead to a reduction in averaged SIR, as it did for averaged ERLE. For N = 64, and 128

however, averaged SIR decreases on average for an increase in R_v as expected. Although not shown here, for smaller NFR values, i.e. greater proportion of echo in $y(n)$ during DT, the averaged SIR values vary much more closely with those of averaged ERLE, as expected, suggesting that for NFR = 0 dB, SIR is somewhat ineffective at measuring the echo interference in $\hat{v}(n)$.

The averaged SAR results in Figure 4.1 are characterized by rising values for increasing R_v and rising values for decreasing R_d for all N. This is to be expected; by increasing the number of basis vectors in $\mathbf{B}_v$, extending the range of speech it can express, for a fixed $\mathbf{B}_d(k)$; or by decreasing the number of basis vectors in $\mathbf{B}_d(k)$, restricting the range of speech it can express, for a fixed $\mathbf{B}_v$; a larger portion of $\mathbf{v}(k)$ is matched onto $\mathbf{B}_v$ and therefore, speaker-matching is reduced, with a commensurate reduction of distortion in the output $\hat{v}(n)$ signals. However, as indicated by the averaged SIR values, a decrease in R_d promotes echo matching during DT; consequently, the averaged SIR and SAR results jointly demonstrate that during DT the choice of R_d is a trade-off between echo matching and speaker matching. A similar trade-off is seen for R_v for N = 64, 128; but for N = 248, 512, 1024, and 2048, the choice of R_v is less of a compromise, since both averaged SAR and SIR increase with R_v up to a point, after which, averaged SIR starts to decrease while SDR continues to increase. The averaged SAR results vary negligible across N.

The averaged SNR results in Figure 4.1 suggest that during DT more noise is omitted from the output $\hat{v}(n)$ signals for higher values of R_v, R_d and N. The omitted noise corresponds to the portion of noise that is matched onto $\mathbf{B}_d(k)$ and is therefore absent from $\mathbf{B}_v$. Given the lack of structure of $w(n)$ in the magnitude-STFT domain, it is natural to assume that the noise contribution in $\mathbf{y}(k)$ is divided equally between the basis vectors of $\mathbf{B}(k)$ such that the noise is assigned to $\mathbf{B}_v$ and $\mathbf{B}_d(k)$ in proportion to the ratio of R_v and R_d. This assumption is somewhat evinced from the averaged SNR results, which exhibit an increase in value for increasing R_d across all N; however, the results also exhibit a slight increase for increasing R_v for higher values of N, a result that does not corroborate this assumption. The averaged SNR values peak in value for N = 1024 for most values of R_v and R_d.

The averaged SDR values of this study, which can be taught of as a combination of the averaged SAR, SIR and SNR values, correlate substantially more with the averaged SAR values than with either averaged SIR or averaged SNR. This implies that distortion is the primary error in the output $\hat{v}(n)$ signals during DT, and thus, while averaged SAR and SDR are also reflecting model error and error due to phase substitution, relatively speaking, speaker matching as opposed to echo matching contributes most to the error during DT. The averaged LSD values, which like averaged SDR are an overall performance measure, vary negatively with the averaged SDR values for all N, implying that the averaged LSD values also reflect primarily distortion. The predominance of speaker matching error in NMF-NSE during DT is ascribable to the generality of the basis vectors in $\mathbf{B}_v$, which deter its ability to fit $\mathbf{v}(k)$, with the specificity of $\mathbf{B}_d(k)$ being insufficient to prevent it from being used to represent a portion

of $\mathbf{v}(k)$. On the other hand, relatively low echo matching error occurs during DT, owing to the likeness of the basis vectors in $\mathbf{B}_d(k)$ to $\mathbf{d}(k)$, and the lack of specificity of $\mathbf{B}_v$ for $\mathbf{d}(k)$.

Contrasting now the performance of NMF-NSE in the absence of DT (echo only) and during DT, according to the DT performance measures values, speaker matching can be reduced, or equivalently distortion during DT is reduced, by either lowering R_d (decreasing the number of basis vectors in $\mathbf{B}_d(k)$) or by increasing R_v (increasing the number of basis vectors in $\mathbf{B}_v$); however, either of these changes would also increase echo matching during periods of echo only, leading to increased residual echo for such periods. It follows therefore that both the choice of R_d and the choice of R_v is a trade-off between echo reduction during echo only periods, and distortion during DT. Furthermore, although the effect of echo matching during DT is less significant than that of speaker matching, a decrease in R_d, and to a lesser extent R_v, also increases residual echo during DT. Consequently, the choice of R_d and of R_v can be stated more generally as a trade-off between echo matching and speaker matching, or as a trade-off between increased echo reduction and increased distortion of the near-end speech.

Contrasting now the results in terms of N, it can be observed that in the absence of DT longer window lengths produce more echo reduction, while during DT optimal performance is attained for window lengths of size 512 or 1024 samples. The variability in the DT performance values across N may be related to the influence that N has on the validity of the assumption of pair-wise disjoint supports of speech signals in the STFT domain, which was given as a justification for the model (4.1). It was demonstrated in [155] that speech signals generally satisfy this assumption in practice, but that the level to which they do varies depending on N, as well as other factors. For example, for speech signals sampled at 16 kHz the optimal value for N was shown empirically to be 1024 samples/64 ms, which we assume corresponds to a window size of 512 samples/64 ms for speech sampled at 8 kHz. Relating this finding to the performance of NMF-NSE during DT, it is apparent from the results in Figure 4.1 that averaged SIR exhibits a peak in value at $N = 512$ for all R_v and R_d, indicating that optimum echo reduction occurs during DT for this value of N; this then influences averaged SDR, which also exhibits a small peak in value, though this peak is spread out over $N = 512$ and $N = 1024$; and the averaged LSD values, which have a slight trough in value at $N = 512$ and $N = 1024$ for the same R_v and R_d values. The results indicate therefore that an optimum value of N exists during DT, which we contend is linked to the level of pair wise disjointness between $\mathbf{d}(k)$ and $\mathbf{v}(k)$. This relationship may arise because increased disjointness of the sources facilitates more accurate matching of their energy onto their respective basis vectors in $\mathbf{B}(k)$ during the ϕ updates i.e. the restricted NMF, with less overlap between bases. Furthermore, increased disjointness between $\mathbf{d}(k)$ and $\mathbf{v}(k)$ implies that the energy in each time frequency bin of $\mathbf{y}(k)$ is more likely to belong exclusively to either $\mathbf{d}(k)$ and $\mathbf{v}(k)$, which if the spectral energy in this bin is assigned to the correct bases during the ϕ updates, mitigates the possibility of cross matching during the assignment of $\mathbf{e}(k)$ during the subsequent ψ updates of both $\mathbf{B}(k)$ and $\mathbf{g}(k)$.

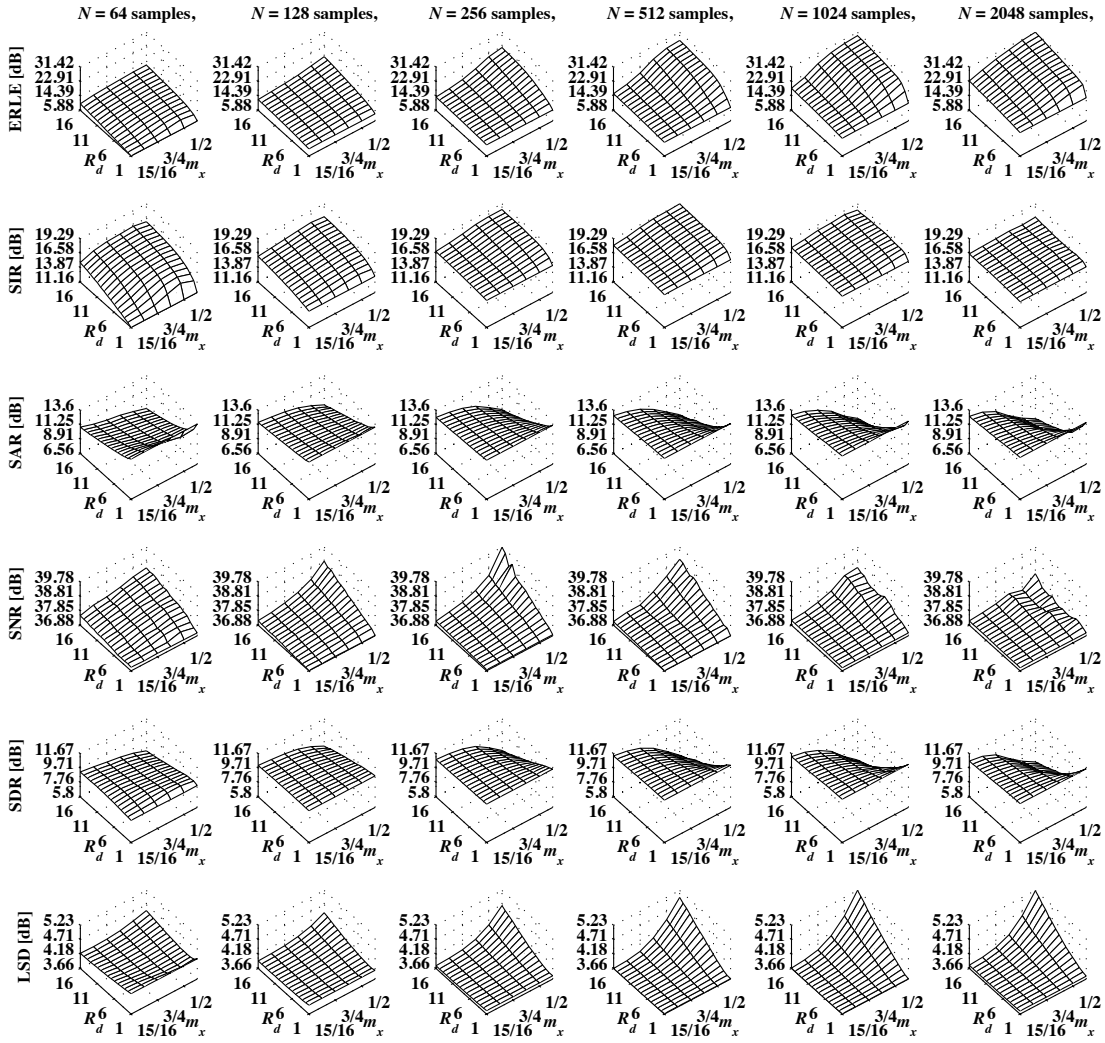

Figure 4.3: Experimental Results illustrating the influence of R_d, m_1 and N on NMF-NSE performance in the absence of DT and during DT. The axis labels across m_1 are displayed as fractions, with a scale factor of m implied. As in figure 1, each column of plots displays a different value for N while each row displays a different performance measure. For each performance measure all plots are across the same scale, with the lowermost and uppermost axis labels for a particular row indicating the minimum and maximum values a performance measure attained for all N.

4.3.2.3 *Study of R_d, m_1 and N*

This study seeks to examine the effect on performance of the time resolution of the basis vectors in $\mathbf{B}_d(k)$ by investigating the influence that the parameters R_d, m_x, and N have on NMF-NSE performance. The values of these parameters employed in this study were $R_d = [1, 2\ldots16]$, $R_v = [1, 2\ldots16]$, and $m_x = [31m/32, 15m/16, 7m/8, 3m/4, m/2, m]$, where for $m_x = m$ the STFT analysis windows for $x(n)$ and $y(n)$ advance at the same rate (same stepsize), with decreasing m_x implying greater time resolution in the echo basis i.e. $x(n)$ is processed with a smaller stepsize. The results for this study are displayed in Figure 4.1, in which each column of plots displays the results for a particular window size while each row displays the results for a particular performance measure, as in the previous study. Note that for space considerations the factor m is omitted from the labels along the m_x axis of the plots in Figure 4.1.

From the results in Figure 4.3, it is apparent that the choice of m_x is a trade-off between echo reduction and distortion. For example, from the averaged ERLE results for $N = 512$, for

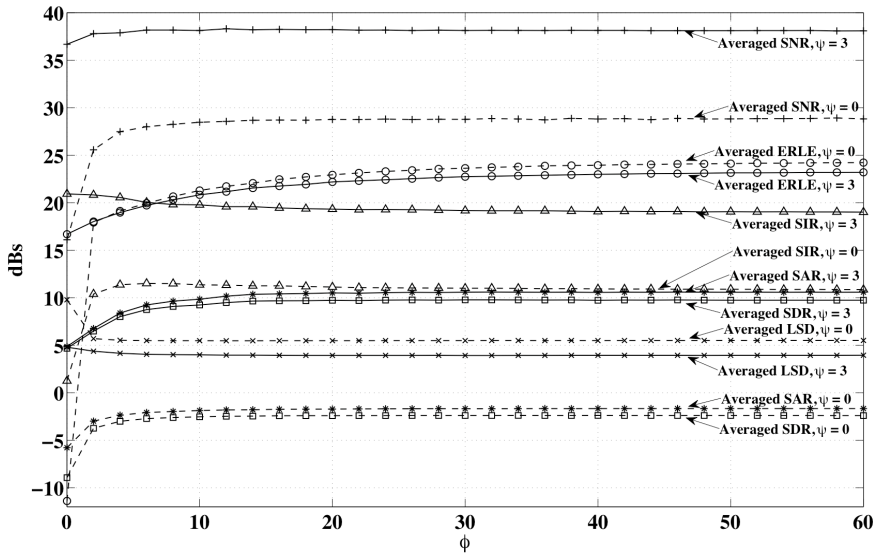

Figure 4.4: The effect of ϕ and ψ on the performance of NMF-NSE. The dashed line type signifies values for $\psi = 0$ i.e. no post separation updates, and the solid line type signifies values for $\psi = 3$. As indicated on the graph, the plus (+), circle (o), triangle (△), star (*), square (□), and x-mark (x), plot symbols correspond to averaged SNR, averaged ERLE, averaged SIR, averaged SAR, averaged SDR, averaged LSD respectively.

decreasing m_x, averaged ERLE decreases for each value of R_d, (except for $R_d = 1$) which implies, (ignoring $R_d = 1$) that by increasing the time resolution of $\mathbf{B}_d(k)$, (decreasing m_x) echo matching is increased. This result is most likely attributable to the narrowing of the time span of the basis vectors in $\mathbf{B}_d(k)$ that comes with reducing m_x for a fixed R_d, which decreases the variability in this basis, decreasing its ability to account accurately for the variability of $\mathbf{d}(k)$. However, by virtue of this decreased range of variability, such an $\mathbf{B}_d(k)$ has greater specificity, and thus, is less prone to speaker matching, a feature borne out by the averaged SDR results, which increase for all R_d, except for $R_d = 1$ for decreasing m_x.

Also from the averaged ERLE results for $N = 512$, averaged ERLE rises for increasing R_d for all m_x, implying that by increasing the number of previous spectral frames of $x(n)$ in $\mathbf{B}_d(k)$, less echo matching occurs. This reduction is because of the increase in R_d, which extends the length of time over which the basis vectors of $\mathbf{B}_d(k)$ span, which consequently increases the variability in $\mathbf{B}_d(k)$, better enabling it to express $\mathbf{d}(k)$. This reduction in echo matching however, and the resulting decrease of residual echo in $\hat{v}(n)$, comes at the expense of greater speaker matching during DT, which is evident from the falling values for averaged SDR for increasing R_d. This increased distortion is attributable to the concomitant ability of a higher rank $\mathbf{B}_d(k)$ to speaker match. The results show therefore that the lack of variability in $\mathbf{B}_d(k)$ arising from a high time resolution can be compensated by increasing R_d. This strategy however also results in a high hardware resource requirement.

4.3.2.4 Study of ϕ and ψ

This study assesses NMF-NSE performance for different numbers of $\mathbf{g}(k)$ updates during the restricted NMF procedure, ϕ, and demonstrates the beneficial performance effect of the post separation updates, i.e. $\psi > 0$, during the unrestricted NMF procedure. The values of ϕ and ψ employed were, $\phi = [1, 2 \ldots 150]$, and $\psi = [0, 3]$. The results of this study are plotted in Figure

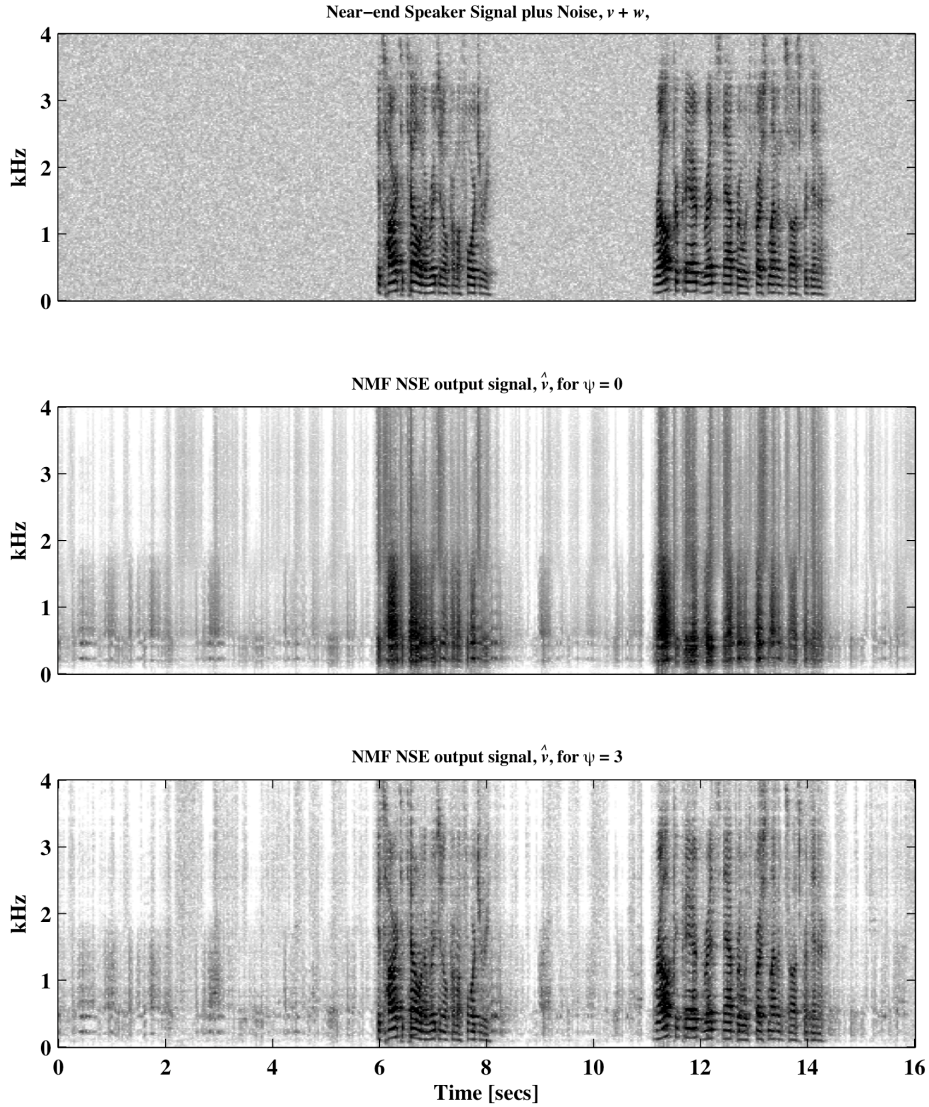

Figure 4.5: Spectrograms (in dBs) illustrating the performance benefit imparted on NMF-NSE from post separation NMF updates. The performance improvement is apparent from the finer spectral detail captured in $\hat{v}$ for $\psi = 3$ in comparison to $\hat{v}$ for $\psi = 0$. Note that for the signals displayed in this figure, the SIR, SAR, SNR, SDR and LSD values for utterance I for $\psi = 3$ were respectively 15.01 dBs, 7.82 dBs, 36.37dBs, 6.94 dBs, and 3.85 dBs, while for $\psi = 0$ they were, 8.3962 dBs, -1.5000 dBs, 28.7801 dbs, -2.4680 dBs, and 5.31 dBs. For utterance II, the results for $\psi = 3$ were respectively 16.10 dBs, 10.01 dBs, 34.51 dBs, 8.95 dBs, and 3.59 dBs, while for $\psi = 0$ they were respectively 8.46 dBs, -1.59 dBs, 23.84 dbs, -2.55 dbs and 4.88 dBs. Note that for the same y signal with $v = 0$, the average ERLE (across time only) for $\hat{v}$ with $\psi = 3$ was 18.74 dBs, while for $\hat{v}$ with $\psi = 0$ it was 17.04 dBs.

4.4. In Figure 4.5, spectrograms of an example $v(n)$, $\hat{v}(n)$ for $\psi = 0$, and $\hat{v}(n)$ for $\psi = 3$ are displayed to illustrate the benefit of the post separation NMF updates.

The profiles of the performance measures in Figure 4.4 exhibit a similar trend for $\psi = 0$ and for $\psi = 3$, typified by relatively large increases or decreases initially, followed by stabilization at the maximum or minimum value reached, or in the case of averaged SIR for $\psi = 3$, an asymptotic decrease from the maximum value attained; this result is discussed in more detail below. In general, it is evident from the results that the ψ updates ($\psi = 3$), for all ϕ, induce a sharp increase in each of the averaged BSS_toolbox performance measures and a fall in LSD, verifying that the post separation updates reduce the distortion, echo, and noise in the near-end speech during DT by accounting for the residual $\mathbf{e}(k)$. However, in the absence of DT, the post separation updates induce a slight drop of approximately 3 dB in averaged

ERLE, indicating that they also induce some echo matching in the absence of DT; thereby, increasing the echo in the output $\hat{v}(n)$ signals. On balance however, we contend that this increase in echo is outweighed by the increased performance during DT, justifying the use of post separation updates. Although not shown here, increasing the number of post separation updates, ψ, beyond three has a negligible effect on NMF-NSE performance.

The highest averaged SIR value in Figure 4.4 for $\psi = 3$ occurs at $\phi = 0$, i.e. no $\mathbf{g}(k)$ updates during the restricted NMF are required for optimum echo removal during DT. This result suggests that the basis vectors of $\mathbf{B}_d(k)$ are such that $\mathbf{d}(k)$ is assigned to this basis during the post separation updates irrespective of the initial activation pattern in $\mathbf{g}(k)$ (provided however $\mathbf{g}(k) > \mathbf{0}$). However, the relatively low values for the other DT measures for $\phi = 0$, $\psi = 3$, or relatively high value in the case of LSD, imply that the ψ updates for $\phi = 0$ also assign much of $\mathbf{v}(k)$ to $\mathbf{B}_d(k)$, owing to the arbitrary activation pattern in $\mathbf{g}(k)$, which manifests as distortion in the output $\hat{v}(n)$ signals. As is reflected by these results, it is not until several updates of $\mathbf{g}(k)$ are completed, when the spectral energy of $\mathbf{y}(k)$ is distributed across $\mathbf{B}(k)$ such that the late updates can eliminate $\mathbf{e}(k)$ without introducing excessive extra speaker-matching, that optimum performance during DT is attained.

4.3.2.5 SNR study

Due to the linearity of NMF, the fraction of $\mathbf{v}(k)$ matched onto $\mathbf{B}_d(k)$ (speaker-matching), and the fraction of $\mathbf{d}(k)$ matched onto $\mathbf{B}_v$ (echo matching), are both invariant to scaling of either $\mathbf{d}(k)$ or $\mathbf{v}(k)$. In this study, and the next, we examine the consequences of this property in relation to noise and near-end speech levels respectively. In this study a single parameter, the test signal parameter SNR_{in} is varied to test the robustness of NMF-NSE to noise. For high levels of SNR_{in}, for which $w(n)$ is negligible, this study will enable the error in the output $\hat{v}(n)$ signals introduced by the NMF-NSE algorithm alone to be examined. The values of SNR_{in} tested were [-40, -39, -38… 39, 40] dBs. The results of this study are displayed in Figure 4.6. For reference, Figure 4.6 also displays the ERLE performance of a conventional FDAF-based AEC; this AEC is also used in the experimental comparison, in which its parameters are specified.

The results in Figure 4.6 show that for decreasing SNR_{in}, for which the gain of $w(n)$ is increasing and those of both $d(n)$ and $v(n)$ are fixed; averaged ERLE, averaged SDR and averaged SNR all decrease, and averaged LSD increases; a trend attributable to the increasing proportion of the output $\hat{v}(n)$ signals that consists of noise. Moreover, for $\text{SNR}_{\text{in}} < 0$ the averaged SDR and averaged SNR values begin to converge, implying that for this range of SNR_{in} the error in the $\hat{v}(n)$ signals is overwhelmingly attributable to noise. Assuming that the proportion of $\mathbf{v}(k)$, $\mathbf{d}(k)$, and noise matched onto $\mathbf{B}_v$ is constant irrespective of SNR_{in}, the increasing prevalence of noise in the output $\hat{v}(n)$ signals for decreasing SNR_{in} is attributable to the increase in the gain of the portion of noise matched onto $\mathbf{B}_v$; therefore, while the level of near-end speech and echo in the output $\hat{v}(n)$ signals is relatively constant for decreasing SNR_{in}, the noise component of the output $\hat{v}(n)$ signals grows.

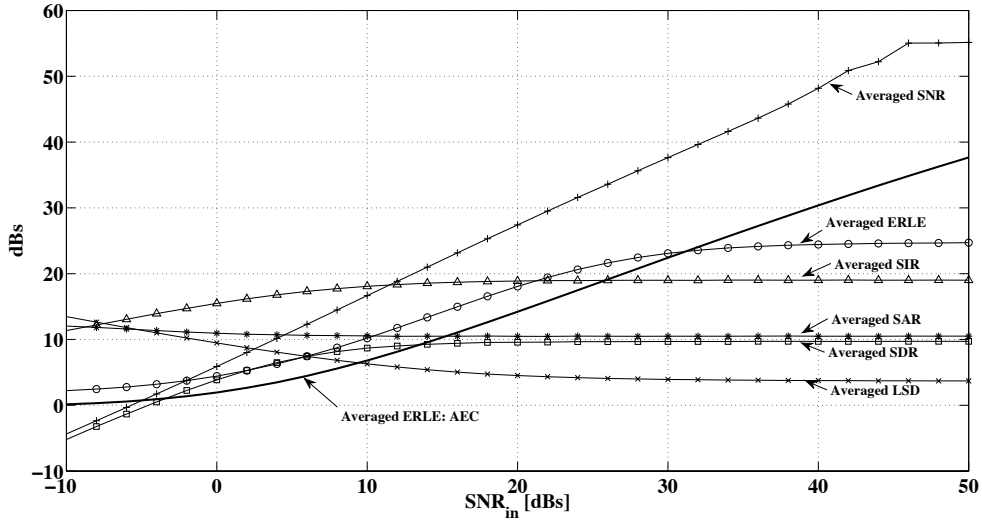

Figure 4.6: The effect of SNR on the performance of NMF-NSE. As indicated on the graph, the plus (+), circle (o), triangle (△), star (*), square (□), and x-mark (x), plot symbols correspond to averaged SNR, averaged ERLE, averaged SIR, averaged SAR, averaged SDR, and averaged LSD respectively. The solid black line represents the averaged ERLE results for AEC.

For more realistic SNR_{in}, such as $SNR_{in} > 10$ dB, the values of the performance measures in Figure 4.6 increase asymptotically for increasing SNR_{in}, or in the case of averaged LSD decrease asymptotically, with an increase in SNR_{in} above 35 dB eliciting a negligible effect on these measures. The increases are attributable to the diminishing proportion of $\hat{v}(n)$ that consists of noise due to the now decreasing gain of the noise component matched onto $\mathbf{B}_v$. The constant values above 35 dB, for which $w(n)$ becomes increasingly insignificant, expose the error inherent to NMF-NSE in the output $\hat{v}(n)$ signals; in this case for the default configuration of its parameters. It is apparent from these values that this error is attributable to distortion, which is evident from the convergence of the average SAR and SDR values. It can be asserted therefore, that a minimum level of distortion, and to a lesser degree echo, is present in the output $\hat{v}(n)$ signals irrespective of the level of noise; this is discussed further in the next section.

For typical values of SNR_{in}, such as $SNR_{in} > 20$ dB, the values are relatively stable implying that noise has a rather insignificant effect on the performance of NMF-NSE over this range, which demonstrates that NMF-NSE is robust to noise. In contrast, it can be seen that the performance of the reference AEC algorithm decreases monotonically with decreasing SNR_{in}; though it is evident that AEC can attain better echo reduction, higher averaged ERLE, for $SNR_{in} > 30$ dB. Note that for low SNR the averaged SAR results in Figure 4.6 are reflecting their dependence on $e_{noise}(n)$, the level of which is increasing for decreasing SNR_{in}. Note also that the prevalence of noise at lower SNR may be obfuscating $\mathbf{d}(k)$ and $\mathbf{v}(k)$ during the separation procedure, thereby rendering the measures unreliable for these levels of SNR_{in}.

4.3.2.6 NFR Study

In this study, NMF-NSE performance is examined for varying levels of the near-end speaker, which is achieved by varying the test signal parameter NFR for fixed levels of echo and noise. The following values of NFR were applied: [-40, -39, -38… 39, 40] dBs. Figure 4.7 displays

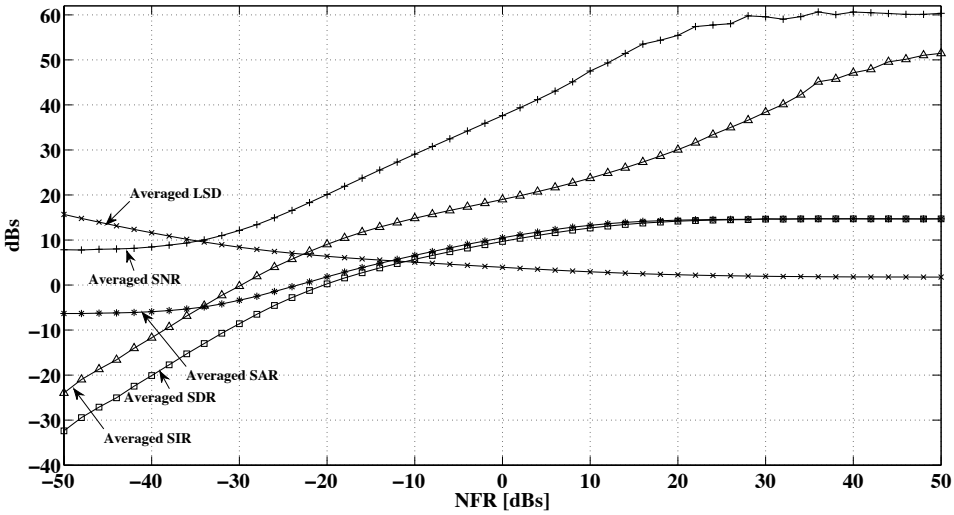

Figure 4.7: The effect of NFR on the performance of NMF-NSE. As indicated on the graph, the plus (+), circle (o), triangle (△), star (*), square (□), and x-mark (x), plot symbols correspond to averaged SNR, averaged ERLE, averaged SIR, averaged SAR, averaged SDR, averaged LSD respectively.

the results of this study; the averaged ERLE values are omitted, as they are constant for all NFR; as such, the discussion of the results is in the context of DT.

For decreasing NFR, for which the level of $v(n)$ in $y(n)$ is decreasing, for a fixed level of $d(n)$ and $w(n)$, the profiles of averaged SDR, SNR, LSD and SIR in Figure 4.7 indicate that NMF-NSE performance during DT degrades. There are two factors responsible for this; firstly, assuming again that the proportion of $\mathbf{v}(k)$, $\mathbf{d}(k)$ and noise matched onto $\mathbf{B}_v$ is constant irrespective of NFR, then the level of echo manifest in the $\hat{v}(n)$ signals due to echo–matching is approximately constant for all NFR, implying that residual echo will constitute an increasing proportion of the output $\hat{v}(n)$ signals for decreasing NFR, resulting in lower averaged SIR. This rise in the proportion of echo in $\hat{v}(n)$ also leads to increased averaged LSD and reduced averaged SAR, averaged SNR and averaged SDR, which covarys with averaged SIR for NFR > 30, signifying the prevalence of residual echo in the output $\hat{v}(n)$ signals for this range of NFR. Secondly, and as discussed in the previous section, the level of noise in $\hat{v}(n)$ is approximately constant irrespective of NFR, and therefore noise will constitute an increasing proportion of $\hat{v}(n)$ for decreasing NFR, pushing averaged SNR lower, and contributing to the lower values of the other performance measures, or higher values in the case of averaged LSD.

For increasing NFR, for which the level of $v(n)$ in $y(n)$ is now increasing for a fixed level of $d(n)$ and $w(n)$, both averaged SIR and SNR rise steadily, reflecting the diminishing proportion of the output $\hat{v}(n)$ signals that consist of echo and noise. Initially, this relative decrease in echo and noise is also reflected through increased averaged SAR, SDR and LSD; however, the increases taper off as the effect of distortion error becomes more prevalent, a prevalence apparent from the convergence of the averaged SDR and averaged SAR values for NFR > 0. The constancy of averaged SDR, SAR, and LSD above 20 dB NFR indicates that the absolute level of distortion in the output $\hat{v}(n)$ signals increases linearly with NFR, thereby showing, empirically, that an invariable percentage of $\mathbf{v}(k)$ is matched onto $\mathbf{B}_d(k)$ irrespective

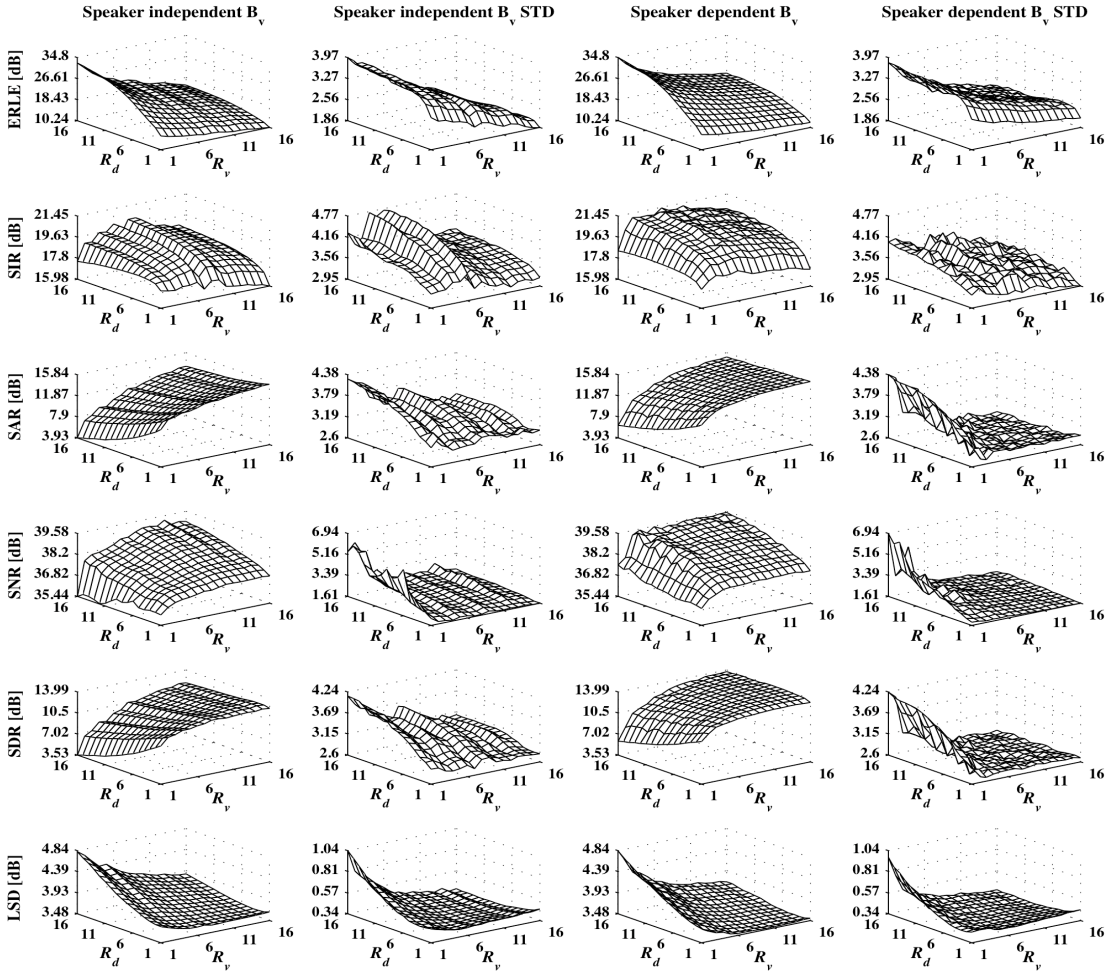

Figure 4.8 : NMF-NSE performance with a speaker independent $\mathbf{B}_v$ and with a speaker dependent $\mathbf{B}_v$ in the absence of and during DT. The 1st and 3rd column of plots display the results for a speaker independent $\mathbf{B}_v$ and a speaker dependent $\mathbf{B}_v$ respectively, while the 2nd and 4th columns display the standard deviation of the results for a speaker independent $\mathbf{B}_v$ and a speaker dependent $\mathbf{B}_v$ respectively. In each row a different performance measure is displayed, the particular measure is indicated along the z-axis of the leftmost graph. The results plots for a particular row are displayed across the same scale, and the plots of standard deviation for a particular row are displayed across the same scale. The lowermost and uppermost z-axis labels of each of indicate the minimum and maximum values of the measures attained.

of SNR and NFR. The results of this study therefore follow the results of the SNR study, in which NFR was 0 dB; both of which indicate that $\hat{v}(n)$ during DT will contain a minimum level of distortion due to speaker matching irrespective of the relative level of the near-end speaker or noise.

An obvious implication of this property of NMF-NSE is that during DT, or more precisely when $\mathbf{B}_d(k)$ contains nonzero values, even if the level of the near-end speaker far exceeds that of the echo and noise signal, $\hat{v}(n)$ will contain a significant amount of distortion. For example from Figure 4.7, if NFR > 20 dBs, $\hat{v}(n)$ will contain on average 18 dBs SAR of distortion, a level of distortion which may be more objectionable to the end-user than the original $y(n)$ signal. However, in realistic scenarios, such as hands free telephony, the near-end speaker and the echo signal can be expected to have equivalent levels, i.e. NFR = 0 dB or within a region of 5 dBs around this value, which, judging from the results in Figure 4.7, is the range of NFR where the trade-off between echo reduction and distortion is most favorable.

Furthermore, as demonstrated in the accompanying studies of this section, the performance of NMF-NSE can be manipulated by varying R_v, R_d, N, m_1, ϕ and ψ or by employing a source dependent $\mathbf{B}_v$ (next study).

4.3.2.7 Comparison of NMF-NSE performance with a independent speaker $\mathbf{B}_v$ to that of NMF-NSE with a speaker dependent $\mathbf{B}_v$

This final study of this section demonstrates the benefits of a speaker dependent $\mathbf{B}_v$ by comparing the performance of NMF-NSE with such a $\mathbf{B}_v$ to that of NMF-NSE with a speaker independent $\mathbf{B}_v$. This comparison was performed for the following values of R_d and R_v, R_d = [1, 2…16], R_v = [1, 2…16]. The results of this study are displayed in Figure 4.8, in which the results for NMF-NSE with a speaker independent $\mathbf{B}_v$ are displayed in the first column of plots while the results for NMF-NSE with a speaker dependent $\mathbf{B}_v$ are displayed in the third column of plots. To compare the variability in the NMF-NSE results for the two bases, each result in Figure 4.8, is accompanied by the standard deviation (STD) of the set of values used to compute that result. The STD values are displayed in the second (speaker independent $\mathbf{B}_v$) and fourth column (speaker dependent $\mathbf{B}_v$) of in Figure 4.8,.

Figure 4.8 shows that NMF-NSE with a speaker dependent $\mathbf{B}_v$ attained higher values for each performance measure for each value of R_d and R_v relative to NMF-NSE with a speaker independent $\mathbf{B}_v$, indicating that with a speaker dependent $\mathbf{B}_v$ the output $\hat{v}(n)$ signals contain less echo, distortion and noise during DT and less echo in the absence of DT. As a specific example, it can be seen that for the default parameters of R_v and R_d, NMF-NSE with a speaker dependent $\mathbf{B}_v$ offers an increase of 2.02 dBs in averaged ERLE, 1.34 dBs in averaged SIR, 1.71 dBs in averaged SAR, 1.2 dBs in averaged SNR, 1.71 dBs for averaged SDR and a decrease in averaged LSD of 0.15 dBs. Examining the variability in the performance measures computed from the output $\hat{v}(n)$ signals, the standard deviations in Figure 4.8 indicate that NMF-NSE with a speaker dependent $\mathbf{B}_v$ also achieved more consistent results than NMF-NSE with a speaker independent $\mathbf{B}_v$, which generally yields higher standard deviations for all performance measures across R_d and R_v.

The higher and more consistent values obtained for a speaker dependent $\mathbf{B}_v$ relative to a speaker independent $\mathbf{B}_v$ stems from the greater specifity of the speaker dependent $\mathbf{B}_v$, which decreases the likelihood of echo matching, and from the greater likeness of $\mathbf{v}(k)$ to the basis vectors in a speaker specific $\mathbf{B}_v$, which reduces the likelihood of speaker-matching. The use of the speaker dependent $\mathbf{B}_v$ with NMF-NSE however, entails information about the near-end speaker, a requirement not entailed with an independent $\mathbf{B}_v$, nor indeed, for conventional AEC-DTD.

4.3.3 Comparison study between NMF-NSE and conventional AEC-DTD

For this stage, the performance of NMF-NSE is evaluated by way of comparison with conventional AEC-DTD approaches. Similar to the previous stage, in this stage the performance of the algorithms was compared in the absence of DT and during DT separately; though in this study the effect of room change is examined. Before describing the testing

methods and discussing results, it will be necessary to choose representative values for the parameters of NMF-NSE, and to identify and describe comparative AEC-DTD algorithms.

4.3.3.1 Parameters of NMF-NSE

In this section, representative parameter values for NMF-NSE are chosen. This process was guided by both the results in section 4.3.2, and the hardware resource requirement of NMF-NSE described in section 4.2.2. The first parameter selected was N, for which we contend 512 samples is the longest window length with an acceptable buffering delay; moreover, NMF-NSE exhibits a peak in DT performance measures for this window length, which was linked to the disjointness of the underlying sources. The near-end microphone signal $y(n)$ therefore, was processed in 64 ms windows using a hanning window and a 50% overlap i.e. $N = 512$, and $m = 256$. The choice of m_1 was established as a trade-off between echo reduction and distortion, as such in this evaluation a compromise was reached and m_1 was set to 128 samples; therefore, the far-end speaker signal x was processed in 64 ms frames with a 75% overlap. The choice of R_d and R_v was also established as a trade-off between echo reduction and distortion of the near-end speech. Furthermore, recall from section 4.2.1.1, that R_d should be chosen such that $\mathbf{B}_d(k) = \mathbf{0}$ implies $\mathbf{d}(k) = \mathbf{0}$; the room responses (2048 samples), together with N and m_1 dictate that $R_d \leq 16$; with $R_d = 16$ resulting in a high hardware resource requirement. Given that in practice DT occurs approximately 20 % of the time [16], less than the occurrence of echo, it is appropriate to favor echo reduction at the expense of distortion, which can be realized by setting $R_v < R_d$; hence, for this comparative study, R_d was set to 8 and R_v was chosen to be 4. This value for R_d enables good echo reduction for $N = 512$, $m_1 = 128$ (see Figure 4.1 and Figure 4.3). To veraciously compare NMF-NSE with AEC DTD, which is speaker independent, a speaker independent near-end speaker basis $\mathbf{B}_v$ was employed, the construction of which is described in the previous stage. Note that the chosen values for R_d, R_v and ϕ entail a competitive hardware resource requirement, which is compared with traditional AEC-DTD below.

4.3.3.2 Comparative Conventional AEC-DTD algorithms

The performance of NMF-NSE was compared to that of two adaptive algorithms commonly used for AEC: the Generalized Multi-Delay Filter algorithm [7] (GMDFα) and the NLMS algorithm [3]. Recall from Chapter 2 that GMDFα is a type of FDAF algorithm that can accept block sizes less than $2L$ and an input block overlap greater than 50% [7]. As a member of the FDAF class, GMDFα also offers fast convergence for low computational load, and as such represents a highly capable algorithm against which to compare the performance of NMF-NSE. By contrast, time-domain based NLMS offers slower convergence than GMDFα with comparatively higher computational load, which can grow larger for long-duration room response. The utility of NLMS is that it is a widely used reference algorithm in the AEC literature and so is included here to facilitate broader comparison of our results with those of other studies.

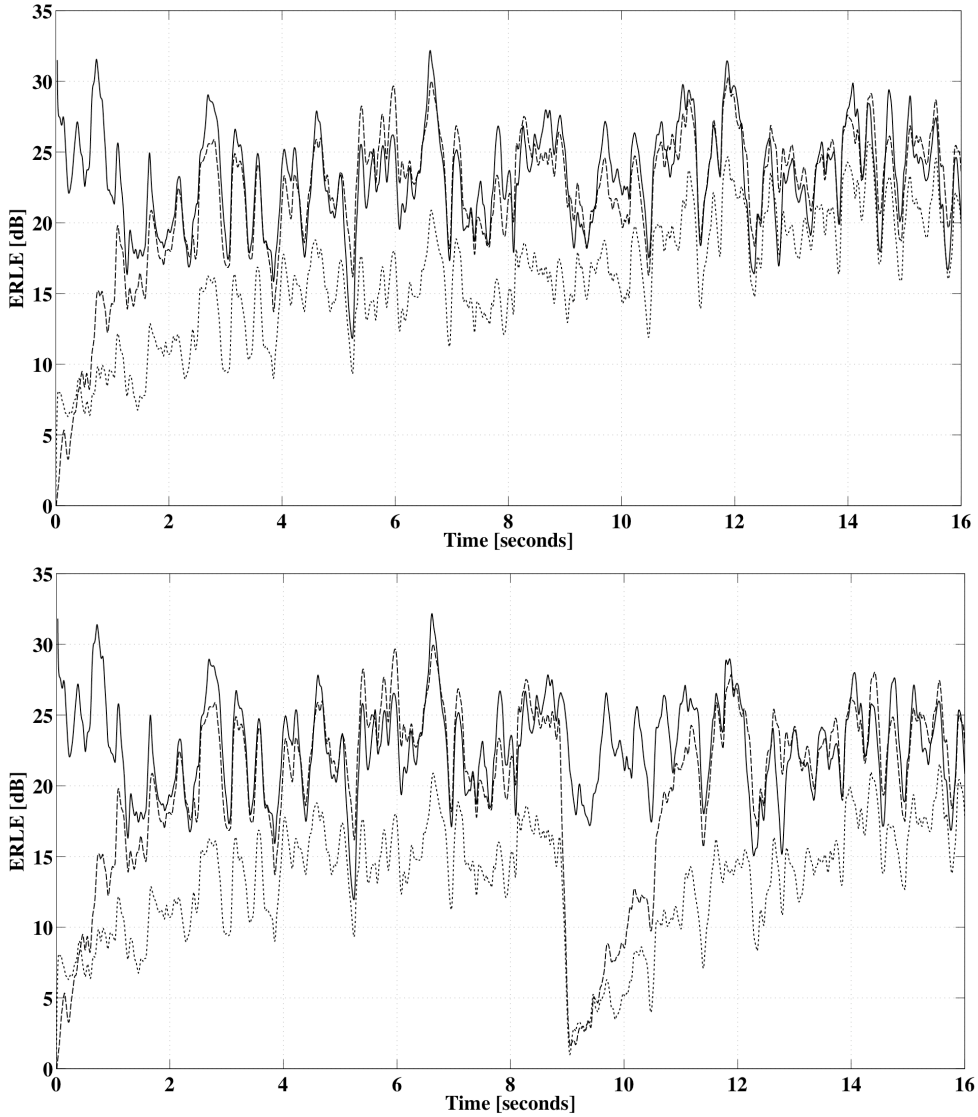

Figure 4.9 : Top: Comparison of ERLE values for NMF-NSE (solid line), GMDFα (dashed line) and NLMS (dotted line) with a stable near-end room throughout. Bottom : Comparison of ERLE values for NMF-NSE (solid lines), GMDFα (dashed lines) and NLMS (dotted lines) with a near-end room change at nine seconds.

The constrained, self-orthogonalizing, GMDFα implementation [7], as described in Chapter 2 section 2.2.5, was employed in this study, with the following parameters: overlapping factor $\alpha = 2$, input frame size $N_{\text{GMDF}} = 64$ ms/512 samples (therefore, FFT size or input block size = $2N_{\text{GMDF}}$ or 128ms/1024 samples, $2N = 2^b$, $b = 10$); $L_h = 2048$ (the impulse response is divided into 4 segments i.e. $K_{\text{GMDF}} = L_h/N_{\text{GMDF}}$), $\mu_{\text{GMDF}} = 0.3$; and the forgetting factor for the spectral normlaisation factor, denoted here by λ_{GMDF} was set to 0.9. Using these parameter values GMDFα was configured to produce 64 ms output frames, with 50% overlap, providing an output signal directly comparable to that of NMF-NSE. The parameters of NLMS are the step size μ_{NLMS}, which was set to 0.5, and the length of $\hat{\mathbf{h}}$, which was set to L, giving 2048 coefficients.

Both NLMS and GMDFα were paired with a DTD algorithm to prevent their filter models diverging during DT. NLMS was paired with the well known computationally efficient variant of the Normalized Cross Correlation (NCC) algorithm defined in (2.64). The

statistics for NCC were calculated using recursive estimates with a forgetting factor of 0.99, and each indication of DT from NCC was held for 200 samples. GMDFα was paired with the Multi-Delay block Frequency-domain DTD (MDF-DTD) [13], whose decision variable is defined in (2.69). Following the foreground/background MDF filter implementation presented in [13], in this study a block estimate of **h** from an independent background GMDFα is used to calculate the MDF-DTD decision variable. The forgetting factor of the background GMDFα was set to 0.8 (0.1 smaller than λ_{GMDF}) such that the background GMDFα will adapt to the statistics of the far-end speaker faster than the foreground GMDFα, and thus at the onset of DT, MDF-DTD can alert the foreground GMDFα before divergence occurs. The remaining parameters for the background GMDFα were the same as the foreground GMDFα specified above.

4.3.3.3 Performance in the absence of Doubletalk

The first set of experiments compared the echo mitigating performance of NMF-NSE, as measured by ERLE, to that of GMDFα and NLMS in the absence of DT during three distinct operational conditions: upon initiation, in a stable long-established room, and in the period following a sudden room change. In the absence of DT, $v(n) = 0$ and the number of distinct near-end microphone signals $y(n)$ and $y_c(n)$ is reduced in both cases from 24 to just 6. NMF-NSE, NLMS, and GMDFα were each applied to the 6 remaining distinct near-end microphone signals, with and without room change, with DTD disabled for the latter two algorithms. The ERLE results from this experiment for $y(n)$ (without room change) and for $y_c(n)$ (with room change) are displayed in Figure 4.9

The transient behavior exhibited by GMDFα and NLMS in Figure 4.9 is characteristic of conventional AEC approaches, which rely on an estimate of **h** to perform echo cancellation. From the ERLE values of GMDFα and NLMS in Figure 4.9 it is evident that both require an initial convergence period before reaching a steady ERLE level, with GMDFα exhibiting the faster convergence. Moreover, after the room change, evident at 9 seconds in Figure 4.9 (bottom), there is a sharp decline in the ERLE values of GMDFα and NLMS, with GMDFα again exhibiting faster re-convergence. The tangible implication of such an ERLE profile is that the far-end user will experience echo while the adaptive filters converge at initiation and again after a room change. In contrast, the level of the NMF-NSE ERLE values in Figure 4.9 are consistent throughout the experiment, matching the level reached by GMDFα at its steady state and exceeding NLMS throughout. The ERLE profiles of NMF-NSE indicate that, in contrast to conventional AEC-DTD, the far-end user will experience a consistent level of echo reduction, available upon initiation and undiminished by room change.

4.3.3.4 Performance during periods of DT

This section compares the performance of NMF-NSE, as measured by LSD, to that of GMDFα-MDF-DTD, and NLMS-NCC during DT for 3 different conditions, with a separate test for each condition. To emulate a room response long-established at the onset of DT, the first test applied NMF-NSE, GMDFα-MDF-DTD, and NLMS-NCC separately to each of the 24 y signals, which have a stable RIR throughout, ensuring that the adaptive filters of the AEC

ALGORITHM	UTTERANCE I	UTTERANCE II
GMDFα-MDF-DTD	3.03	3.29
NLMS-NCC	3.32	3.76
NMF-NSE	3.79	4.11

Table 4.2 : Averaged Log Spectral Distance results, average taken over 24 text signals.

ALGORITHM	UTTERANCE I	UTTERANCE II
GMDFα-MDF-DTD	3.03	5.15
NLMS-NCC	3.32	5.48
NMF-NSE	3.80	4.05

Table 4.3: Averaged Log Spectral Distance results with Room Change at 9 seconds, average taken over 24 text signals

ALGORITHM	UTTERANCE I	UTTERANCE II
GMDFα-MDF-DTD	3.04	5.58
NLMS-NCC	3.30	5.28
NMF-NSE	3.61	4.09

Table 4.4: Averaged Log Spectral Distance results with Room Change at 13 seconds, average taken over 24 text signals

approaches are as well adapted at the onset of DT as their DTDs will allow. For this test, two average values of LSD were computed for each algorithm, the first over the frames of utterance I, and the second over the frames of utterance II. These values were then averaged over all 24 output signals and are tabulated in Table 4.2.

From Table 4.2, NMF-NSE has the highest average value of LSD for utterances I and II, indicating that, under stable room conditions, both GMDFα-MDF-DTD and NLMS-NCC introduced less distortion to the near-end output speech during DT than NMF-NSE, with GMDFα-MDF-DTD introducing the least. NMF-NSE produced inferior quality speech during DT due to cross-matching error; discussed extensively in section 4.3.2.

To emulate a room change in the recent past that remains influential at the onset of DT, the second test applied NMF-NSE, GMDFα-MDF-DTD, and NLMS-NCC to each of the 24 y_c signals, in which a room change occurs at 9 seconds, after utterance I has ended and before utterance II has begun at 11 seconds. The averaged LSD values for utterances I and II for this test are tabulated in Table 4.3. Figure 4.10 displays representative examples of output speech waveforms from this test for each of the three algorithms.

The averaged LSD values of the conventional algorithms for utterance II in Table 4.3 have significantly increased relative to those of utterance II in Table 4.3. These larger values signify increased distortion attributable to greater levels of echo in the output near-end speech of both algorithms due to a higher level of misadjustment at the onset of utterance II. In contrast, NMF-NSE produces an average value of LSD for utterance II in Table 4.3 that is consistent with that in Table 4.2, the magnitude of the discrepancy being commensurate with noise. The results for this test show that the performance of NMF-NSE during DT is insensitive to recent room changes whereas conventional AEC-DTD approaches, which rely on an estimate of **h**, are negatively affected. Additionally, spurious DT, erroneously identified by the MDF-DTD and NCC DTDs in response to the room change, may have impeded the

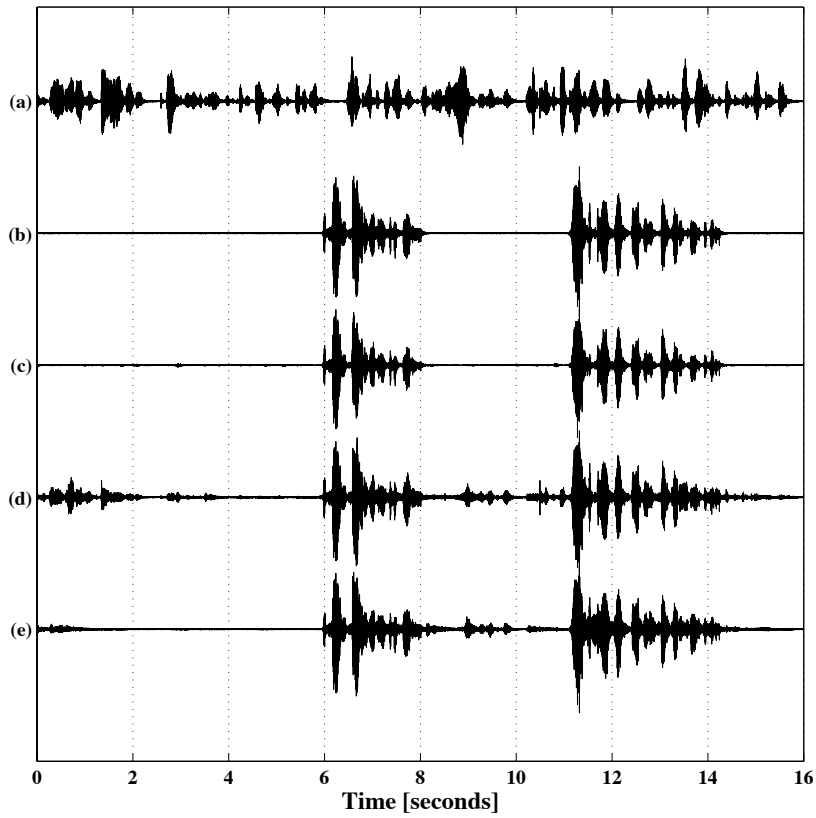

Figure 4.10: Example input and output signals from the second test, (a) echo signal, (b) near-end speaker signal plus noise, (c) Output from NMF-NSE, (d) NLMS-NCC output signal, (f) GMDFα-MDF-DTD output signal.

adaptation of GMDFα and NLMS algorithms after the room change, thereby contributing to the misadjustment that affected utterance II; this issue is examined in detail in Chapter 5.

For the third and final test, each algorithm was applied to the 24 y_c signals with the room change shifted to 12 seconds such that the change occurs during utterance II i.e. during DT; a room change is synchronously introduced into the near-end speaker RIR to reflect the change in that room response. The averaged LSD values for utterances I and II from this test are displayed in Table 4.4.

Examining the values in Table 4.4, it can be seen that the average LSD value of NMF-NSE during utterance II to be less than those of the AEC-DTD algorithms, and is still consistent with those in Table 4.2 and Table 4.3. The test results indicate that the distortion introduced by NMF-NSE and echo reduction is unaffected by room change during DT; this is in contrast to conventional AEC-DTD. During DT the invariant level of NMF-NSE distortion is less than that experienced by the AEC-DTD systems following a room change.

To establish that the various quantitative results reported to this point translate to perceptible, audible effects, an informal listening test was conducted. (Audio files are available online (http://www.eeng.nuim.ie/~ncahill/)) The participants individually reported that elevated echo levels are audible in the AEC algorithms' outputs following initiation and room change, and that those periods are not discernable from the NMF-NSE output. The periods of DT however, were judged to be discernable in the NMF-NSE output, coinciding with audible distortion. During stable room conditions only NLMS-NCC was judged to be

Algorithm	ARITHMETIC OPERATIONS	MEMORY LOCATIONS
NLMS-NCC	14,346	6,154
GMDFα-MDF-DTD	1,740	23,068
NMF-NSE	4,148	22,239

Table 4.5: Number of Arithmetic Operations per sample and number of Memory Locations required

clearly not preferable. The participants opined that while the distorting by NMF-NSE was perceptible and a distraction, the inconstancy of the AEC algorithms' performances was similarly distracting.

4.3.3.5 Resource requirement comparison

As a final important point of comparison between the performances of NMF-NSE and the conventional AEC-DTD methods, this section enumerates and compares the memory requirement, in terms of Memory Locations (MLs), and the computational load, in terms of Arithmetic Operations (AOs) per output sample, of each algorithm described above.

For GMDFα, $2\alpha((K_{\text{GMDF}})(N_{\text{GMDF}}+1) + (2K_{\text{GMDF}}+6)(N_{\text{GMDF}}+1) + 6N_{\text{GMDF}}$ MLs are required [46], and $\alpha[(8K_{\text{GMDF}}+12)b + 8K_{\text{GMDF}} - 13]$ AOs are required [7] to produce one output sample. For the combined GMDFα-MDF-DTD implementation, certain AOs are common to the foreground and background GMDFα filters, but this computational saving is offset by extra AOs required to calculate the detection statistic. Consequently the number of AOs required per-sample for GMDFα-MDF-DTD is approximated as twice that required for GMDFα. The number of extra MLs required for MDF-DTD is $2L + 3N_{\text{GMDF}}$. NLMS requires $2L + 3$ MLs and $4L + 7$ AOs per output sample [13]. Coupled with NLMS, NCC with recursive updates requires $L + 3$ extra MLs and $3L + 3$ extra AOs per sample [11]. For NMF-NSE, the number of AOs per-sample is obtained by calculating the total number of AOs in a frame, and dividing by $N/2$ i.e. frame size = N, overlap = 50%. The total number of MLs required for NMF-NSE is the total number of MLs arising from each algorithmic step as listed in Table I.

Table 4.5 lists the number of memory locations and arithmetic operations required by each algorithm to produce one output sample, based on the parameters specified in section III.C. Examining Table 4.5, NMF-NSE has resource requirements more comparable to those of GMDFα-MDF-DTD than those of NLMS-NCC: both NMF-NSE and GMDFα-MDF-DTD have a relatively low computational cost and a relatively high memory requirement, the latter requiring approximately ½ the number of AOs, and approximately the same number of MLs as the former. This balance of computational and memory requirements is characteristic of FDAF algorithms in general. In contrast, time-domain based NLMS-NCC has a relatively high computational load and a relatively low memory requirement.

Another practical consideration for the AE problem is the processing delay, with less delay being more desirable. For the parameters specified above, both GMDFα-MDF-DTD and NMF-NSE incur a 64 ms delay associated with block/frame (overlapping) that may be prohibitively large in certain applications, whereas NLMS-NCC operates on a sample-by-sample basis, and therefore has no buffering delay.

4.4 Chapter Summary

Echo arising from loudspeaker/microphone acoustic coupling at an opposing hands-free telephone user is a common complaint of telephone users. The conventional approach to mitigate such echo is based on adaptive system identification, which in general, is sensitive to both room change and DoubleTalk. To address these issues, we applied monaural sound source separation (MSSS) techniques to the problem of single channel acoustic echo reduction. In the MSSS framework, the objective is to extract the near-end speaker's signal from a mixture containing that signal, as well as echo and noise. To achieve separation, NMF was used in the magnitude STFT domain to decompose spectral features of the microphone signal onto two bases of such features. An echo basis was constructed from the spectrum of the incoming far-end signal, while a speaker basis was trained on the spectra of multiple speakers a priori. An estimate of the near-end speaker's magnitude-spectrum was formed from the features of the microphone spectrum that are modeled by the speaker basis during the decomposition. A time domain signal is then synthesized using the IFFT of the estimated magnitude spectrum, together with the phase of the original mixture. This approach was named NMF Near-end Speaker Extraction (NMF-NSE).

Numerous experiments were conducted to evaluate the proposed algorithm, by which the performance effect of various parameters of NMF-NSE was elucidated. The main relationship that arose from these experiments is that a trade-off exists between echo reduction and distortion of the near-end users speech during DT. This trade-off is controlled principally by the number of basis vectors in the echo and speakers bases, with more basis vectors in the echo basis resulting in more echo reduction and more distortion of the near-end user speech during DT, and vice versa, the near-end speakers basis. In a subsequent comparative study, it was shown that relative to conventional AEC-DTD methods, NMF-NSE exhibits robust performance upon initialization and after room changes, even during DT. For the far-end user, the comparative results demonstrate that upon initialization or after room changes no abrupt periods of disturbing echo are perceived. However, it was also evident that during DT the near-end speech signal from NMF-NSE is distorted, although subjectively the speech quality and intelligibility from NMF-NSE were deemed acceptable. Lastly, it was shown that the hardware resources required by NMF-NSE are comparable to that of conventional AEC and DTD approaches.

5 DOUBLETALK DETECTION USING NONNEGATIVE MATRIX FACTORISATION

This Chapter describes a novel Doubletalk Detection algorithm for block-based AEC algorithms. For each input block of the AEC, this DTD compares the normalised inner product of the smoothed short time magnitude spectra of the near-end microphone signal and the smoothed short time magnitude spectra of the NMF-NSE estimate of the echo signal to a preset threshold to detect DTD. This algorithm, NMF-DTD, has comparable Receiver Operating Characterisitc curves to a conventional DTD, and in constrast to this conventional DTD, allows for uninterrupted adaptation of its paired adaptive filter upon initation and following room change.

5.1 Introduction and Background

Doubletalk (DT) is a well-known complication for acoustic echo cancellers. During DT the near-end speaker $v(n)$ and the echo signal $d(n)$ are simultaneously active, i.e. $v(n) \neq 0$ and $d(n) \neq 0$, which can cause the coefficents of the adaptive filter of an acoustic echo canceller to rapidly diverge from optimality, resulting in increased echo being sent to the far-end user. A common strategy to cope with DT is to suspend adaptation during periods identified as DT by a Doubletalk Detector (DTD). As described in Chapter 2, to detect DT, conventional DTDs typically draw on the available signals and the signals created by the paired adaptive filter to compute a decision variable, which is compared to a preset or time-varying threshold to decide on the presence of DT. For tractable computational load, and to avoid ill-conditioning or numerical issues, Conventional DTD's often also employ the estimate of **h** from the AEC (foreground) adaptive filter or from an independent background adaptive filter to approximate **h** when computing the test variable. Under certain conditions, this substitution can give rise to transparent performance; however, during and after both initiation and enclosure changes this

estimate approximates the actual **h** poorly, giving rise to inaccurate values for the decision variable, which in turn, can cause false detection of DT. Such DT false positives impede the adaptation of the AEC adaptive filter after such events, thereby slowing convergence, prolonging the echo disturbance for the far-end user.

In this chapter we propose a new DTD approach for single channel block-based frequency domain AEC. For this approach, which we name NMF-DTD, the estimate of the echo signal generated by NMF-NSE algorithm, background instantiated, and the available signals are utilized to compute a DT decision variable to control a foreground block-based adaptive filter. The NMF-DTD decision variable is defined as the normalised inner product of the smoothed short time magnitude spectra of the near-end microphone signal and the smoothed short time magnitude spectra of the NMF-NSE estimate of the echo signal. Like conventional block-based DTD, a value for this test variable is computed for each block or frame of the foreground adaptive filter, and the presence of DT is decided by comparing this value to a preset threshold. Unlike conventional DTD however, NMF-DTD does not require **h** or an estimate of **h** to calculate its decision variable, and therefore, operates independently of fluctuations in its coefficents, such as after an enclosure change or upon initiation. Instead, NMF-DTD utilises the NMF-NSE estimate of the echo to calculate this variable, and consequently NMF-DTD inherits the proven room/enclosure change robustness of NMF-NSE. This robustness will be shown to mitigate DT false positives upon initiation and after enclosure changes, enabling largely unimpeded adaptation of the paired foreground adaptive filter during and after such conditions, and thus, allowing for approximately optimum echo cancellation.

This chapter is organised as follows. NMF-DTD is formulated in section 5.2, which follows from the formulation of NMF-NSE in section 4.2.1 of the previous chapter. In section 5.3 an experimental evalutation of NMF-DTD by way of comparison with a conventional DTD is outlined, and section 5.4 contains the chapter summary.

5.2 Nonnegative Matrix Factorization Doubletalk Detection (NMF-DTD)

5.2.1 Formulation of NMF-DTD

Recall from section 4.2.1 of Chapter 4 the NMF-NSE magnitude spectral estimates $\hat{\mathbf{v}}(k)$ and $\hat{\mathbf{d}}(k)$, which correspond to $\mathbf{v}(k)$ and $\mathbf{d}(k)$ respectively, and which combine to give the near-end microphone spectral frame $\mathbf{y}(k)$, that is, $\mathbf{y}(k) = \mathbf{v}(k) + \mathbf{d}(k) = \hat{\mathbf{v}}(k) + \hat{\mathbf{d}}(k)$. Assuming that the current near-end microphone spectral frame, $\mathbf{y}(k)$, contains non-zero echo energy, which can be expressed as $||\mathbf{d}(k)|| > 0$, the task of NMF-DTD is to detect if the near-end speaker is also active in the current frame, i.e. $||\mathbf{v}(k)|| > 0$, using the signals generated by NMF-NSE, i.e. $\hat{\mathbf{v}}(k)$ and $\hat{\mathbf{d}}(k)$, and the available signals, $\mathbf{y}(k)$ and the far-end users speech signal.

The algorithm structure we employ for NMF-DTD is similar to that used by conventional DTDs that employ a parallel foreground/background adaptive filter implementation. For each frame, NMF-DTD computes a decision variable using the output

from a background instantiated NMF-NSE. The computed value is compared to a threshold, the result of which controls the adaptation rate of a foreground block-based adaptive filter. For reasons outlined in Chapter 2, we wish to define an appropriately normalized correlation-based decision variable for NMF-DTD; by appropriately normalized, we mean that for $\|\mathbf{v}(k)\| = 0$, $\xi(k) = 1$ and for $\|\mathbf{v}(k)\| > 0$, $\xi(k) < 1$, where $\xi(k)$ denotes the NMF-DTD decision variable. We proceed by provisionally defining the NMF-DTD decision variable, describing some of the properties of this variable, and then motivating some modifications that enhance its robustness.

We define $\xi(k)$ as the normalized inner product between $\mathbf{y}(k)$ and $\hat{\mathbf{d}}(k)$, which is given by,

$$\xi(k) = \frac{\mathbf{y}(k)^{\mathrm{T}}\hat{\mathbf{d}}(k)}{\|\mathbf{y}(k)\|_2\|\hat{\mathbf{d}}(k)\|_2}. \tag{5.1}$$

This expression can be interpreted geometrically by viewing $\xi(k)$ as a measure of the angle between the vectors $\mathbf{y}(k)$ and $\hat{\mathbf{d}}(k)$; specifically, the cosine of the angle between $\mathbf{y}(k)$ and $\hat{\mathbf{d}}(k)$. This interpretation was adopted in [86], in which a time domain based DTD was presented with a decision variable similar to that of (5.1); this variable is defined by (2.67) in Chapter 2.

For $\hat{\mathbf{d}}(k) = \mathbf{d}(k)$, it can be easily deduced, using the Cauchy-Schwarz inequality, that (5.1) meets the criteria of an appropriately normalized DT test value, i.e. for $\|\mathbf{v}(k)\| = 0$, $\mathbf{y}(k)^{\mathrm{T}}\hat{\mathbf{d}}(k) = \|\mathbf{y}(k)\|_2\|\hat{\mathbf{d}}(k)\|_2$, and for $\|\mathbf{v}(k)\| > 0$, $\mathbf{y}(k)^{\mathrm{T}}\hat{\mathbf{d}}(k) < \|\mathbf{y}(k)\|_2\|\hat{\mathbf{d}}(k)\|_2$, which correspond to $\xi(k) = 1$ and $\xi(k) < 1$ respectively. However, as discussed at length in Chapter 4, in practice cross matching introduces error into the echo estimate $\hat{\mathbf{d}}(k)$, which in turn, introduces error into $\xi(k)$. Additionally, noise will induce error in $\xi(k)$. If noise is present in $\mathbf{y}(k)$ during DT it will be matched in some way between the bases $\mathbf{B}_d(k)$ and $\mathbf{B}_v$, and as such, $\hat{\mathbf{v}}(k)$ will contain a portion of this noise, and consequently, $\|\hat{\mathbf{d}}(k)\|$ will vary with respect to $\|\mathbf{y}(k)\|$, with concomitant variations in $\xi(k)$.

As $\xi(k)$ relies on a value of correlation between $\mathbf{y}(k)$ and $\hat{\mathbf{d}}(k)$ rather than on the levels of these signals, it is already somewhat robust to the effects of both cross matching and noise. However, to reduce spurious deviations in $\xi(k)$ due to these sources of error, we compute $\xi(k)$ using temporally smoothed versions of the terms in (5.1). To this end, we re-define $\xi(k)$ as,

$$\xi(k) = \frac{E^2_{\hat{\mathbf{d}}\mathbf{y}}(k)}{E_{\hat{\mathbf{d}}}(k)E_{\mathbf{y}}(k)}. \tag{5.2}$$

where $E_{\mathbf{y}}(k), E_{\hat{\mathbf{d}}}(k)$ and denote $E_{\hat{\mathbf{d}}\mathbf{y}}(k)$ the following recursively smoothed variables,

$$E_{\mathbf{y}}(k) = \lambda E_{\mathbf{y}}(k - 1) + (1 - \lambda)\mathbf{y}(k)^{\mathrm{T}}\mathbf{y}(k), \tag{5.3}$$

$$E_{\hat{\mathbf{d}}}(k) = \lambda E_{\hat{\mathbf{d}}}(k - 1) + (1 - \lambda)\,\hat{\mathbf{d}}(k)^{\mathrm{T}}\hat{\mathbf{d}}(k), \tag{5.4}$$

$$E_{\hat{\mathbf{d}}\mathbf{y}}(k) = \lambda E_{\hat{\mathbf{d}}\mathbf{y}}(k - 1) + (1 - \lambda)\hat{\mathbf{d}}(k)^{\mathrm{T}}\mathbf{y}(k), \tag{5.5}$$

where λ ($0 < \lambda < 1$) denotes the forgetting factor for the exponential smoothing, and is a trade-off between sufficient smoothing, i.e. a high value for λ, and fast tracking, i.e. a low value for

λ. Note that squaring the numerator in (5.2) is so the square root operation required for the norms given in the denominator of (5.1) are avoided.

To control the rate of false positive and true negative indications that may still arise due to noise and cross matching, the DT decision is made for the k^{th} frame by comparing $\xi(k)$ to a threshold T, below which DT is deemed present,

$$I(k) = \begin{cases} 0, & \xi(k) < T \\ 1, & otherwise \end{cases} \tag{5.6}$$

where $I(k)$ is a binary DT indicator function for $\mathbf{y}(k)$. If $I(k) = 0$ the adaptation in the contemporaneous block of the foreground adaptive filter is suspended until at least the k^{th}+1 frame, while if $I(k) = 1$, adaptation is permitted.

At the beginning of the k^{th}+1 frame, $\mathbf{B}_v$ is reinitialized, $\mathbf{g}(k+1)$ is reinitialized with random non-negative values, and a new $\mathbf{B}_d(k)$ is compiled, such that the DT decision for one frame has no bearing on that of the next.

5.2.2 Hardware Resource requirements of NMF-DTD

A complete algorithmic summary of NMF-DTD is provided in Table 5.1; the algorithmic steps of NMF–NSE necessary for NMF-DTD are restated. Accompanying this summary are expressions for the number of arithmetic operations and the memory requirements arising from each algorithm step of NMF-DTD during the processing of one frame. The computational and memory requirements are enumerated in accordance with the method described in section 4.2.2 of Chapter 4.

From in Table 5.1, the hardware resource requirement of NMF-DTD largely mirrors that of NMF-NSE, with NMF-DTD not requiring the resources to synthesize $\hat{v}(n)$ but requiring additional resources for the decision variable $\xi(k)$. This analysis is somewhat incomplete however, as the hardware resources of the foreground adaptive filter should be taken into account when enumerating the hardware resource requirement of an AE mitigation algorithm that incorporates NMF-DTD. To address this, we enumerate the combined hardware resource requirement of NMF-DTD and a conventional adaptive filter, and compare it to that of a conventional AEC-DTD pairing, in section 4.3.3.5.

5.3 Performance of NMF-DTD

This section empirically evaluates NMF-DTD by comparing its detection performance to that of a well-known conventional block based frequency domain DTD. A variety of fixed enslosures and enclosures that vary are employed. Each DTD is paired with an identical foreground adaptive filter, so that the influence of the DTDs on the performance of these foreground adaptive filters is analyzed and compared. The remainder of this section is organized as follows: section 5.3.1 specifies the parameters of NMF-DTD, describes both the foreground adaptive filter and the comparative DTD, and specifies their parameters; section 5.3.2 describes the test signals; section 5.3.3 describes some performance metrics; section

ALGORITHM STEP	ARITHMETIC OPS	MEMORY
Process far-end Signal x		
Step-size m_x, frame index k_x		
$X(f, k_x)$,	$*R_d\left(2N\log_2(N) - \frac{3N}{2} - 4 + N\right)$	N
$\lvert X(f, k_x)\rvert$	$*R_d\left(\frac{N}{2} + 1\right)$	Stored in $\mathbf{B}_d$
Create echo basis $\mathbf{B}_d(k)$		
For each frame k and step-size m, such that $mk = m_x k_x$,		
$\mathbf{B}_d(k) = [\,[\lvert X(0, k_x)\rvert, \lvert X(1, k_x)\rvert \ldots, \lvert X(N/2, k_x)\rvert]^{\mathrm{T}},$ $[\lvert X(0, k_x-1)\rvert, \lvert X(1, k_x-1)\rvert \ldots, \lvert X(N/2, k_x-1)\rvert]^{\mathrm{T}} \ldots,$ $[\lvert X(0, k_x-R_d-1\rvert, \lvert X(1, k_x-R_d-1)\rvert \ldots, \lvert X(N/2, k_x-R_d-1)\rvert]^{\mathrm{T}}\,]$		$R_d\left(\frac{N}{2} + 1\right)$
$\mathbf{B}_v$ (Near-end basis)	Computed offline	$R_v\left(\frac{N}{2} + 1\right)$
$\mathbf{B}(k) = [\mathbf{B}_v\ \mathbf{B}_d(k)]$		
$Y(f, k)$	$2N\log_2(N) - \frac{3N}{2} - 4 + N$	N
$\lvert Y(f, k)\rvert \angle Y(f, k)$	$6\left(\frac{N}{2} + 1\right)$	$\left(\frac{N}{2} + 1\right)$
$\mathbf{y}(k) = [\lvert Y(0, k)\rvert, \lvert Y(1, k)\rvert, \ldots, \lvert Y(N/2, k)\rvert]^{\mathrm{T}}$		$\left(\frac{N}{2} + 1\right)$
Initialize $\mathbf{g}_y$ with random nonnegative numbers	Computed offline	
Separation:		
For $l = 1 : 1 : \phi$ (restricted NMF updates) $+\psi$ (unrestricted NMF updates) **do**		
$\mathbf{g}(k) \leftarrow \mathbf{g}(k) \circ \frac{\mathbf{B}(k)\left[\frac{\mathbf{y}(k)}{\mathbf{B}(k)\mathbf{g}(k)}\right]}{\mathbf{B}(k)^{\mathrm{T}}\mathbf{1}}$	$\left(\left(R_d + R_v\right)\left(\frac{5N}{2} + 4\right) + \frac{N}{2} + 1\right)(\phi + \psi)$	$5\left(R_d + R_v\right) + N$ $+2,$
If $l > \phi$ **do** (Residual elimination)		
$\mathbf{B}(k) \leftarrow \mathbf{B}(k) \circ \frac{\left[\frac{\mathbf{y}(k)}{\mathbf{B}(k)\mathbf{g}(k)}\right]\mathbf{g}(k)^{\mathrm{T}}}{\mathbf{1}\mathbf{g}(k)^{\mathrm{T}}}$	$\left(\left(R_d + R_v\right)(3N + 4) + \frac{N}{2} + 1\right)\psi$	$\left(\frac{5N}{2} + 4\right)\left(R_d + R_v\right)$ $+N + 2,$
End if		
End for		
Compute $\xi(k)$		
$E_{\mathbf{y}}(k) = \lambda E_{\mathbf{y}}(k - 1) + (1 - \lambda)\,\mathbf{y}(k)^{\mathrm{T}}\mathbf{y}(k)$	$2\left(\frac{N}{2} + 2\right)$	1
$E_{\hat{\mathbf{d}}}(k) = \lambda E_{\hat{\mathbf{d}}}(k - 1) + (1 - \lambda)\,\hat{\mathbf{d}}(k)^{\mathrm{T}}\hat{\mathbf{d}}(k)$	$2\left(\frac{N}{2} + 2\right)$	1
$E_{\hat{\mathbf{d}}\mathbf{y}}(k) = \lambda E_{\hat{\mathbf{d}}\mathbf{y}}(k - 1) + (1 - \lambda)\,\hat{\mathbf{d}}(k)^{\mathrm{T}}\mathbf{y}(k)$	$2\left(\frac{N}{2} + 2\right)$	1
$\xi(k) = \frac{E^2_{\hat{\mathbf{d}}\mathbf{y}}(k)}{E_{\hat{\mathbf{d}}}(k)E_{\mathbf{y}}(k)}$	3	1
DT decision for frame		
$I(k) = \begin{cases} 0, & \xi(k) < T, \\ 1, & otherwise, \end{cases}$		1
If $I(k) = 0$, stall adaptation, **else** continue adaptation,		

Table 5.1 : Algorithmic Summary of NMF-DTD with Indicative Computational Load and Memory Requirement over one frame of processing

*if $R_d > m/m_x$, R_d in this expression can be replaced with m/m_x as frames of $|X(f, k_x)|$ calculated for the k^{th}–1 frame can be reused.

5.3.4 describes the experiments, analyzes and discusses the results; and 5.3.5 compares the hardware resource requirement of the two DTDs.

5.3.1 NMF-DTD and comparative algorithm parameters

The parameters of NMF-DTD common to NMF-NSE were set according to the default parameter values established in chapter 4, i.e. $\phi = 50$, $\psi = 3$, $R_v = 4$, $R_d = 8$, $N = 512$, $m = 256$, and $m_1 = m/2$, and a speaker independent $\mathbf{B}_v$, as described in Chapter 4, was used throughout. The remaining NMF-DTD parameter, λ, was set to 0.9.

Due to its overlapping frame nature, NMF-DTD is an unsuitable DTD for the time domain class of adaptive filters or, in general, for the FDAF class, for which the input block length is typically twice the filter length. However, as described in Chapter 2 section 2.2.5, the Generalized Multi-Delay adaptive Filter algorithm [7] (GMDFα, where α is an overlap factor) is a member of the FDAF class that can accept block sizes less than $2L$ and an input block overlap greater than 50%, facilitating straightforward integration with NMF-DTD. Futhermore, as a member of this class, it offers fast convergence for low computational load. For these reasons, GMDFα was chosen as the foreground adaptive filter for NMF-DTD, and likewise, for the comparative DTD. As in Chapter 4, the constrained, self-orthogonalizing, GMDFα implementation was employed. This implementation was configured to produce output frames directly comparable to those of NMF-DTD such that each value of $\xi(k)$, and the subsequent DT decision, pertain to a contemporaneous output frame of GMDFα. GMDFα was, as such, configured to produce 64 ms output frames, with 50% overlap, i.e, $\alpha = 2$, and frame size N_{GMDF} = 64 ms/512 samples (therefore, FFT size or input block size = $2N_{GMDF}$ or 128ms/1024 samples, $2N = 2^b$, $b = 10$). The remaining parameters of GMDFα were: $L = 2048$ (the impulse response is divided into 4 segments i.e. $K_{GMDF} = L/N$), $\mu_{GMDF} = 0.2$; and the forgetting factor λ_{GMDF} was set to 0.90.

As in Chapter 4, we choose the Multi-Delay adaptive Filter DTD (MDF-DTD) [56] as the comparative DTD for this evaluation. The MDF-DTD is a block frequency domain derivative of the well-known Normalized Cross-Correlation DTD, with proven performance [56]. To accommodate the GMDFα algorithm, MDF-DTD was implemented to generate DT decisions for overlapping frames. An independent background GMDFα was used to provide an independent block estimate of **h** when calculating the MDF-DTD detection value. As in Chapter 4, the forgetting factor of the background GMDFα was set to 0.8 (0.1 smaller than λ_{GMDF}) such that the background GMDFα will adapt to the statistics of the far-end speaker faster than the foreground GMDFα; consequently, the background GMDFα will converge faster, and thus at the onset of DT, MDF-DTD can alert the foreground GMDFα before divergence occurs. The remaining parameters of the background GMDFα were the same as for the foreground GMDFα specified above.

5.3.2 Creation of Test Signals

Following the method outlined in section 4.3.1.2 of Chapter 4, six sets of $y(n)$ signals were created for this evaluation. To compare performance in stable enclosures and in enclosures that vary, three of these sets contain no enclosure change in their $d(n)$ signals, and the remaining three sets contain a room change at 9 s in their $d(n)$ signals. Each set of signals employs a different RIR, or set of RIRs in the case of those sets that contain an enclosure change. The RIRs for the fixed enclosures are given by the MARDY IDs, ir_1_L_4, ir_1_C_4, and ir_1_R_4; and in the case of a varying enclosures, by the sets of RIRs (ir_1_L_2, ir_1_L_6), (ir_1_C_1, ir_1_C_8) and (ir_1_R_2, ir_1_R_6). Physically the first and third enclosure room changes correspond to a sudden 20 cm displacement of the near-end

microphone, while the second change corresponds to a 40 cm displacement. Note that the RIRs employed in all sets of $y(n)$ signals have varying energy, and, despite truncation at 256 ms, varying initial delays.

5.3.3 Performance metrics

To measure the performance of the DTDs we employ the following well-known pair of complementary DT performance measures [79]; the probability of false detection of DT, P_f, and the probability of missed detection of DT, P_m. In this work, P_f is measured as the percentage of overlapping frames of a $y(n)$ signal that contain only echo that are misclassified as DT by the DTD, also known as the false positive rate. P_m is measured as the portion of overlapping frames of a $y(n)$ signal containing DT that are misclassified as DT free by the DTD, also knwn as the true negative rate.

Per the description in [79], which is adapted here for overlapping frames, to compute P_f for a particular $y(n)$ signal applied to a particular DTD, two binary sequences are required; the DT indicator sequence from the DTD for $y(n)$ with $v(n) = 0$, and the voice activity sequence for the $d(n)$ signal in $y(n)$ (this sequence omits inactive frames of $d(n)$ from the calculations). The DT indicator sequence was obtained by applying $y(n)$ with $v(n) = 0$ to the DTD, generating the DT indicator function, $I(k)$, for that signal. The voice activity sequence was obtained by applying the corresponding $d(n)$ signal to a Voice Activity Detector (VAD). The VAD in this work identifies inactivity by calculating the energy in overlapping frames of length 64 ms (512 samples) and step size 32 ms (256 samples), whereby if the energy, expressed in decibels, is below a threshold of -30 dB then the corresponding frame is labeled inactive, i.e. an absense of echo; otherwise, it is deemed active, i.e. contained echo. The VAD output is a binary sequence in which zero denotes a frame of inactivity and a one denotes activity. The voice activity sequence for $d(n)$ is denoted as $\overline{D}(k)$. With both these sequences calculated, P_f is computed as,

$$P_f = \frac{1}{M}\sum_{k=0}^{M-1} I(k)\overline{D}(k), \tag{5.7}$$

where M is the total number of overlapping frames in $y(n)$. To obtain the corresponding P_m, the $y(n)$ signal is again applied to the DTD this time with $v(n) \neq 0$. Taking the resultant DT indicator function generated by the DTD, the voice activity sequence for $v(n)$, denoted by $\overline{V}(k)$, and $\overline{D}(k)$, P_m is calculated by,

$$P_m = 1 - \left(\frac{\sum_{k=0}^{M-1} I(k)\overline{D}(k)\overline{V}(k)}{\sum_{k=0}^{M-1} I(k)\overline{D}(k)} \right). \tag{5.8}$$

Low values for both P_m and P_f indicate a low false positive rate and low true negative rate respectively, and thus better detection performance.

To gauge the performance of the adaptive filter during these experiments, we employ the commonly used normalized misalignment measure, defined as,

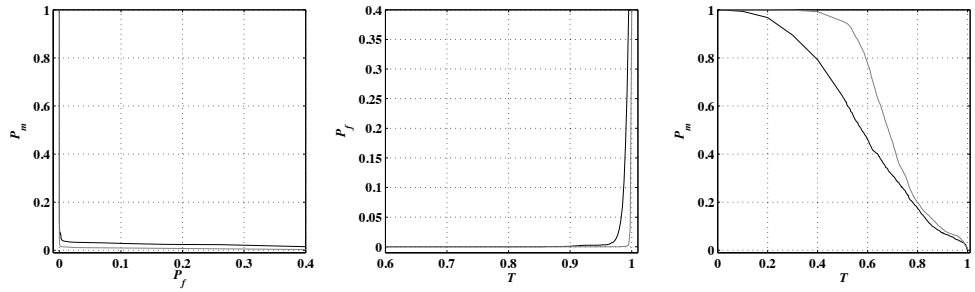

Figure 5.1: Left: A sample ROC curve for NMF-DTD (Black line) and MDF DTD (Grey line) for NFR = 0 dB and for a stable room enclosure. Middle: the corresponding plot of P_f as a function of threshold T for each algorithm. Right: the corresponding plot of P_m as a function of T are also displayed

$$\frac{\| \mathbf{h} - \hat{\mathbf{h}}(k) \|^2}{\| \mathbf{h} \|^2}. \tag{5.9}$$

The misalignment measure indicates the level of echo being sent to the far-end user, and is applicable both during DT and in the absence of DT. This measure is expressed in dBs below, with a smaller value indicating less misadjustment and this better AEC performance.

5.3.4 Experiments

A separate experimental test was performed for each set of $y(n)$ signals, as such three of these tests evaluate the performance of NMF-DTD and MDF-DTD for fixed enclosures, and the other three tests evaluate the DTDs for enclosures that change suddenly at 9 seconds. To benchmark the performance of NMF-DTD to that of MDF-DTD under operational conditions that are ideal for MDF-DTD and conventional AEC-DTD in general, i.e. converged background filter and fixed enclosure throughout, each foreground GMDFα and the background GMDFα for MDF DTD were initialized to $\mathbf{h}$ for the fixed enclosure tests; for the variable enclosure tests, each foreground GMDFα and the background GMDFα were initialized to $\mathbf{0}$, as is typical. The experimental framework employed in each test to evaluate and compare the performance of the two DTDs is based on the standard DTD evaluation technique first outlined in [79], which employs P_m and P_f. In what follows, we describe how this evaluation scheme was applied to each DTD during each test. We then discuss the results, including a close examination of the characteristics of the false positives of both algorithms, before demonstrating and comparing the influence the DTDs have on their respective foreground GMDFα performance using the misalignment measures.

Each test compiled the Receiver Operating Characteristic (ROC) curve for each DTD for a range of values of NFR, i.e. the output of each test is a set of ROC curves for each algorithm. The ROC curve of a DTD is a plot of P_m versus P_f for a range of values of its threshold variable T. This curve is useful for judging the classification performance of a DTD in terms of its false positive rate and true negative rate, with ROC curves closer to the origin while straddling the P_m and P_f axis signifying better DT detection. In this work, each point on a ROC curve is the average P_f and P_m taken over the set of such values (24 in total) obtained from applying a set of $y(n)$ signals, as described in section 5.3.3, to a DTD for a certain value of T; repeated over a range of values of T between 0 and 1 inclusive to form a complete curve. A separate ROC curve was generated for a number of values of NFR ranging from 15 to -15

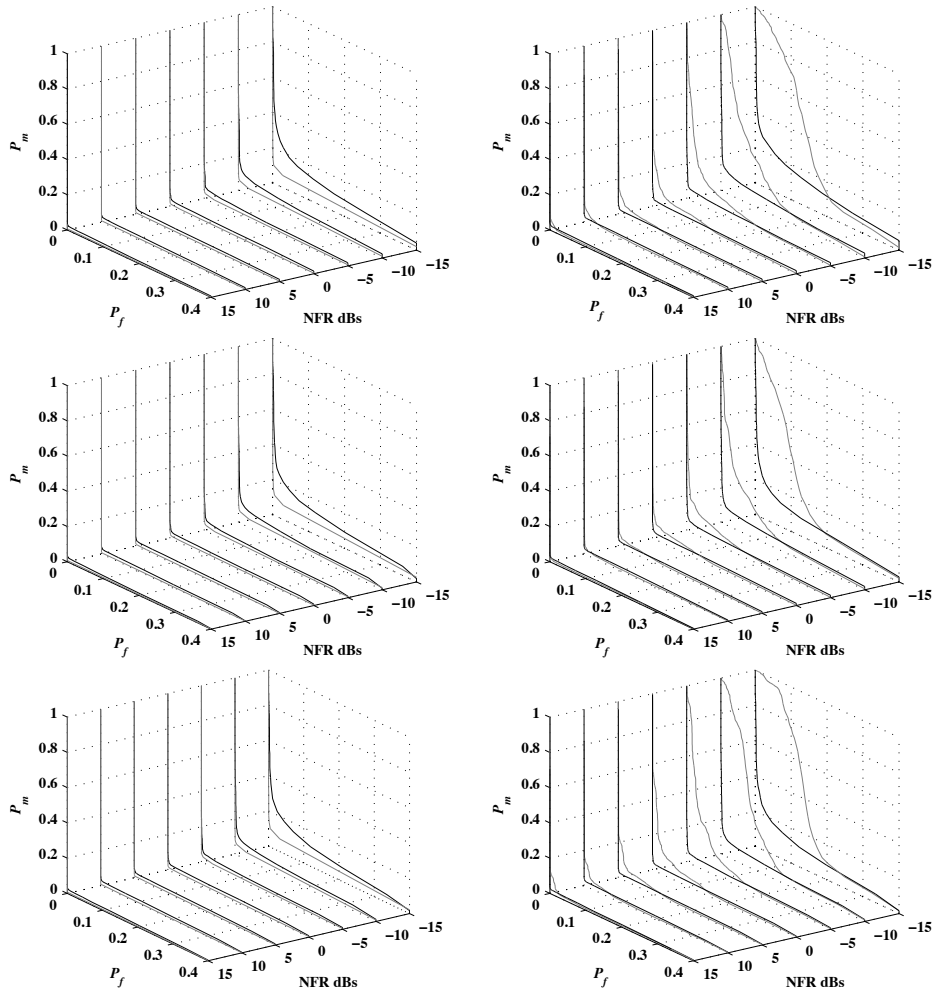

Figure 5.2: The ROC curves of NMF-DTD (Black lines) and MDF DTD (Grey lines) for a selection of NFRs for six different enclosures. The left column displays the results for the fixed enclosures tests, where the GMDFα filters, both background and foreground, were initialized with **h**, and the right column displays the results for the enclosures that change after 9 seconds, where the GMDFα filters initialized with **0**.

dB. The resulting set of ROC curves characterizes the true negative rate and the false positive rate of the DTDs over varying levels of the near-end speaker in a particular enclosure; a separate set of ROCs was produced for each algorithm for each test. The resulting ROC curves are displayed in Figure 5.2, in which the left column displays the results from the fixed enclosure tests, and the right column displays the results from the variable enclosure tests. One issue with these standard ROC curves, is the inability to unambiguously ascribe differences in performance in either P_m, P_f or both. Since we are interested in such differences, in Figure 5.3 we display P_m as a function of T for each ROC curve, while in Figure 5.4 we display P_f as a function of T for both algorithms in each test; a single P_f function is displayed for each algorithm since P_f is independent of NFR.

Prior to analyzing the results in full, we will explain a sample of the results in order to to aid with interpretation of the analysis. For this purpose, Figure 5.1 depicts an example ROC curve for each algorithm for 0 dB NFR for the same signal set (without a room change); the corresponding plot of P_m as a function of T, and P_f as a function of T are also displayed. ROC curves allow for detection algorithms to be visually compared in the context of their inherent trade-off between P_f and P_m, and in Figure 5.1, if a low P_m and a low P_f are equally desirable,

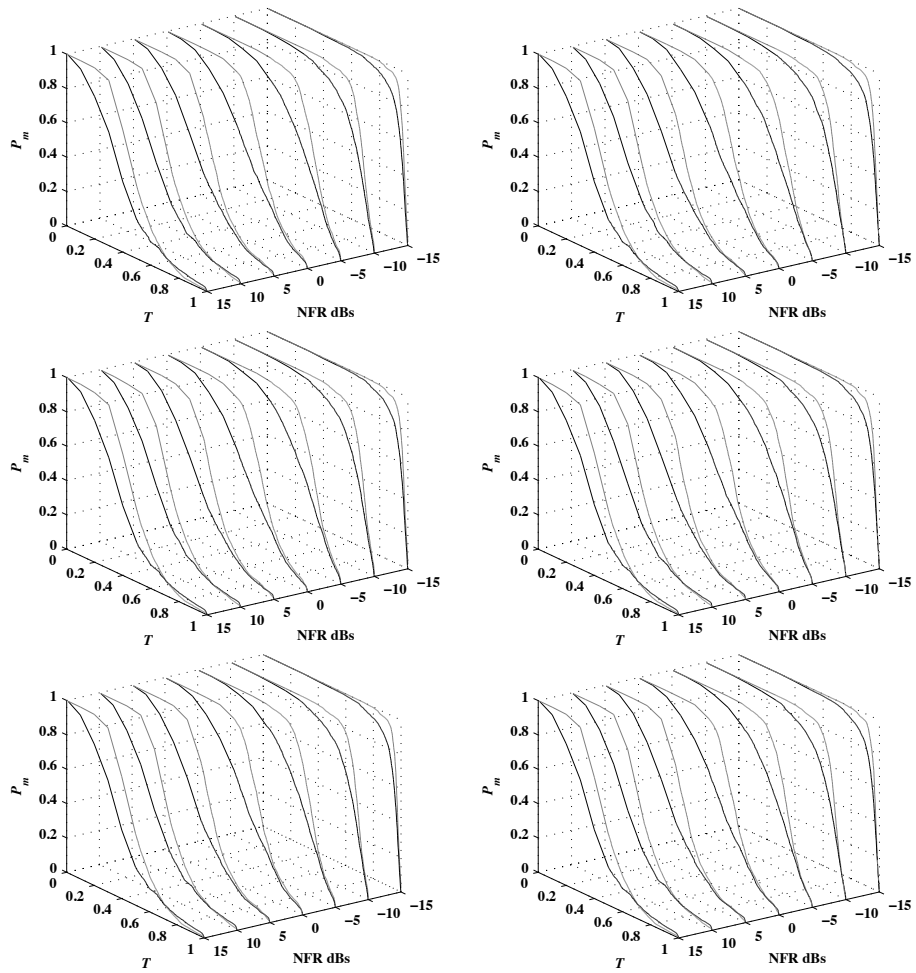

Figure 5.3: Plots of P_m as a function of T for the ROC curves in Figure 5.2.

it is apparent that the conventional DTD has superior performance because it attains the lowest P_m/P_f value; note though that the separate contribution of P_m and P_f to this result can not be asserted. The middle graph displays P_f as a function of the threshold T, and enables the specific effect P_f has on the performance of the algorithms to be assessed. As expected, T is directly proportional to P_f, with the conventional DTD exhibiting superior performance, as it is able to generate the least probability of false detection. The remaining plot in Figure 5.1 shows P_m as a function of T and elucidates the effect of P_m has on the performance of the detectors. Again T is directly proportional to P_m, as expected; with the proposed DTD exhibiting a lower P_m over much of the range of T. However, in this plot the lowest P_m for both algorithms is approximately equivilent and occurs for the same high values of T, which is the most preferable value of T given the results for P_f. Taking this into account it can be asserted that the conventional DTD has exhibits better performance because it generates less false positives for the same level of true positives than the proposed algorithm. Figure 5.1 shows results for each algorithm for a single NFR in a single room enclosure, in the figures that follow, for conciseness, Figure 5.2 contains an ROC for a set of NFRs for all enclosures; likewise, the separate results for P_f and P_m in Figure 5.3 and Figure 5.4 respectively.

The similarity of the ROC curves of NMF-DTD in Figure 5.2 in each test verifies that NMF-DTD discriminates between DT and echo consistently in different enclosures for both

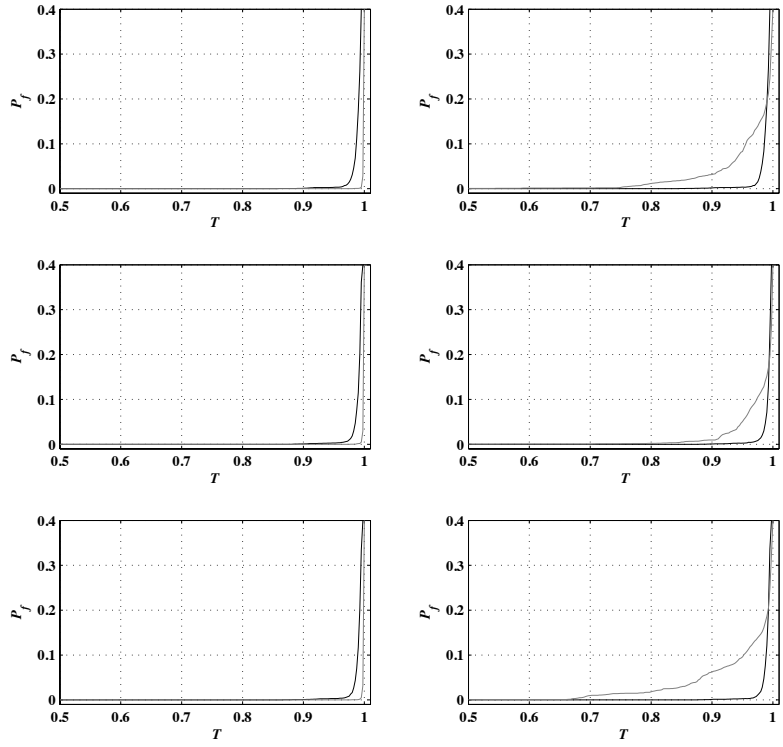

Figure 5.4: Plots of P_f as a function of T for the ROC curves of each test in Figure 5.2.

fixed and variable enclosures; in contrast, the ROC curves of MDF-DTD deteriorate significantly between fixed and variable enclosures, indicating that imposing initiation and enclosure change on this algorithm significantly degrades its ability to discriminate DT. For the fixed enclosure tests, the ROC curves show that MDF-DTD can attain a slightly lower P_m/P_f than NMF-DTD in the more desirable operational range of low P_f and low P_m near the origin. Examining Figure 5.3 and Figure 5.4, it is evident that these differences are due exclusively to differences in P_f, with MDF-DTD and NMF-DTD exhibiting approximately the same values of P_m for the relevant low values of P_m in Figure 5.3, and with MDF-DTD exhibiting lower P_f relative to NMF-DTD for the relevant values of T in Figure 5.4. The increase in P_m/P_f for MDF-DTD between the fixed and variable enclosure tests can also be discerned by jointly examining Figure 5.3 and Figure 5.4, from which it is apparent that the deterioration in performance is due exclusively to an increase in P_f. Examining the true positive rates or P_f of each algorithm from Figure 5.2 and Figure 5.3, the general trend across NFR for the ROC curves of both algorithms is wholly influenced by P_m, which increases for most T for decreasing NFR due to the increasing difficulty of detecting DT due to the diminishing energy of $v(n)$ relative to the echo in the $y(n)$ signals. Also from Figure 5.2 and Figure 5.3, it is evident that the P_m values of both DTDs converge at values of T close to 0 and 1 for most NFR, with NMF-DTD exhibiting lower P_m otherwise. The results in Figure 5.2 and Figure 5.3 establish therefore that the true positive rate of NMF-DTD at least matches that of MDF-DTD for most values of T for both fixed and variable enclosures, with both DTDs capable of providing a low P_m.

To more closely examine and compare the nature of the false positives of each DTD, which, as described, account for most of the inter-DTD variability in the ROC curves, Figure

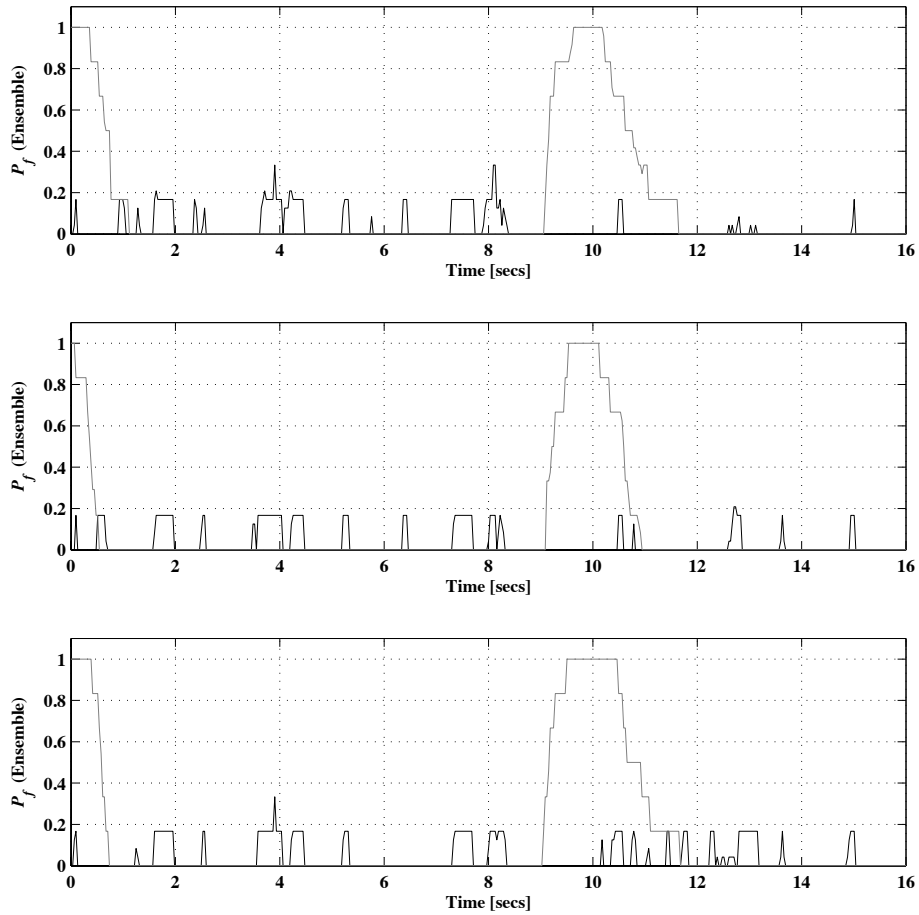

Figure 5.5 : Probability of false detection P_f , calculated by ensemble averaging, for GMDFα-NMF-DTD (black line) and GMDFα-MDF DTD (Grey line) for variable enclosures tests, each panel displays a different test.

5.5 displays frame wise P_f values for each algorithm for each of the three variable enclosure tests. These P_f functions were computed by ensemble averaging over the set of DT indicator functions obtained from applying a set of $y(n)$ signals ($v(n) = 0$) to each algorithm, with a separate function for each test, and with the DTD thresholds set such that $P_m \approx 0.02$ for a NFR of 0 dB for each algorithm. Note that equivalent plots for the fixed enclosure tests are not displayed. This is because MDF-DTD generated a negligible number of false positives in these tests, and NMF-NSE produced the same level of P_f in both sets of tests, obviating the need for further analysis of the results of the these experiments.

From the ensemble P_f functions in Figure 5.5, it is apparent that MDF-DTD erroneously stalls foreground adaptation for each $y(n)$ signal at the start and after nine seconds of each test, confirming that initiation and room change occasioned the false positives of MDF-DTD in these tests, a result typical of conventional DTD in general. These false positives are attributable to the background GMDFα, which like the foreground GMDFα requires time to converge upon initiation and after enclosure changes, during which the MDF-DTD test value is consequently inaccurate, leading to in turn an increased likelihood of false classification of DT. NMF-DTD also exhibits non-zero values for P_f in Figure 5.5, but these differ characteristically from those of MDF-DTD, in that they are relatively low in value and are distributed more uniformly in time, and in particular, are not significantly elevated either

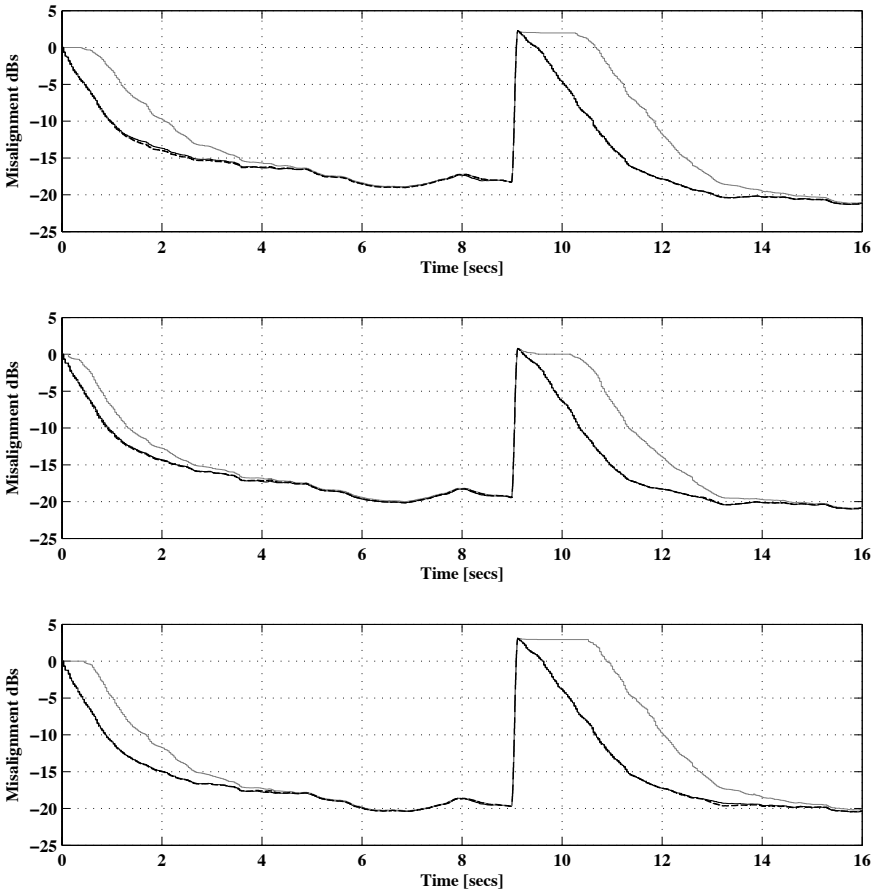

Figure 5.6: Ensemble averaged normalized misalignment functions for GMDFα-NMF-DTD (black line). GMDFα-MDF DTD (Grey line) and GMDFα with no DTD (black dashed line) for the three variable loudspeaker microphone tests with a room change at nine seconds. Note that each panel displays the misalignment functions for a different test, and that the misalignment functions for GMDFα-NMF-DTD and GMDFα (no DTD) overlap throughout.

upon initiation or after the enclosure change, demonstrating that NMF-DTD is insensitive to such events. As described in the formulation, this robustness is ascribable to the ability of NMF-DTD to calculate its test value accurately during initial convergence and after room changes, a capability that is fundamentally attributable to the ability of $\hat{\mathbf{d}}(k)$ to capture $\mathbf{d}(k)$ consistently including upon initiation and after room changes. However, the non-zero P_f values of NMF-DTD exhibit a rather similar pattern across time in each test, implying that NMF-DTD produced false positives in the same frames in each test. Since echo matching is largely independent of the room response and is dependent on the underlying signals and bases (as demonstrated in Chapter 4), then since the same speech signals were reverberated for each test, it can be inferred that spurious echo matching occasioned these false positives. More specifically, echo matching produced these false positives by giving rise to inaccurate values of $\xi(k)$ in the corresponding frames, with variations between each test commensurate with noise.

To elucidate and compare the influence that the respective false positives of the DTDs have on the convergence rate and steady state performance of GMDFα, Figure 5.6 displays averaged frame wise normalized misalignment values for GMDFα-NMF-DTD and GMDFα-MDF-DTD for each of the three variable enclosure tests. Similar to the frame wise P_f functions in Figure 5.5, the misalignment functions in Figure 5.6 were obtained by ensemble

averaging over the set of normalized misalignment functions obtained by applying a set of $y(n)$ signals ($v = 0$) to each algorithm separately for each variable enclosure test for $P_m \approx 0.02$ for NFR = 0 dB. For comparative purposes, the same values were obtained for GMDFα without DTD, which are also displayed in Figure 5.6.

The misalignment functions in Figure 5.6 show that upon initiation, GMDF*α*-NMF-DTD converges faster than GMDF*α*-MDF-DTD, exemplifying the adverse effect on GMDF*α* convergence of the false positives generated by MDF-DTD upon initiation. It is also apparent that upon initiation GMDF*α*-NMF-DTD converges at a rate similar to that of GMDF*α*, indicating that NMF-DTD allows GMDF*α* to converge at a rate approaching maximal. After the enclosure change at 9 seconds, the misalignment profiles in Figure 5.6 show that NMF-DTD-GMDF*α* re-adapts faster than GMDF*α*-MDF-DTD, which is impeded from adapting due to the erroneous false positives generated by MDF-DTD in response to the room change. Again, the convergence rate of GMDF*α*-NMF-DTD after the room change closely matches that of GMDF*α*, demonstrating that, as upon initiation, after an enclosure change NMF-DTD allows GMDF*α* to converge largely unimpeded. It is also apparent in Figure 5.6 that both GMDF*α*-MDF-DTD and GMDF*α*-MDF-DTD are able to reach the same steady state performance as GMDF*α*. For the far-end user, the profiles in Figure 5.6 show the tangible benefits of employing GMDF*α*-NMF-DTD, and indicate that he/she will receive a faster reduction in echo disturbance both upon initiation and after room changes relative to GMDF*α*-MDF-DTD. They also show that the level of echo cancellation in the absence of DT provided by GMDF*α*-NMF-DTD approaches that of the maximum that can be attained using GMDF*α*.

Returning to the spurious false positives generated by NMF-DTD, it can be inferred from the closely matching trajectories of the misalignment functions of GMDF*α* and GMDF*α*-MDF-DTD that these false positives have a negligible impact on echo cancellation. It is evident in Figure 5.6, that the benign nature of these false positives is because they tend to occur intermittently in time, and as such, do not give rise to long periods of stalled AEC adaptation; moreover, examining Figure 5.6 and Figure 5.5 it is evident that they occur when GMDF*α* has converged, when a short pause in adaptation is trivial. This characteristic of NMF-DTD implies that it can be configured to tolerate a somewhat higher P_f than would otherwise be dictated by analysis based on conventional DTD, with NMF-DTD therefore being capably of being configured for a lower P_m. In the context of the fixed enclosure tests, the results of which are displayed in Figure 5.2, this characteristic also implies that the slightly higher P_f attained by NMF-DTD for these tests has a less consequential effect on the performance of GMDF*α*, and as such the effective performance of the foreground adaptive filters of each DTD is similar.

Algorithm	ARITHMETIC OPERATIONS	MEMORY LOCATIONS
NMF-DTD	3,641	21,885
GMDFα	870	17,436
GMDFα-NMF-DTD	4,511	39,321
GMDFα-MDF DTD	1,740	22,244

Table 5.2 : Number of Arithmetic Operations per sample and number of Memory Locations required

5.3.5 Hardware Resource Requirement Comparison

For the purpose of appraising the hardware resource requirement of GMDF*α*-NMF-DTD, this section enumerates and compares the memory requirement, in terms of Memory Locations (MLs), and the computational load, in terms of Arithmetic Operations (AOs) per output sample, of each algorithm described above. This hardware resource requirement comparison is based on that which is described in Chapter 4, with the number of MLs and AOs required by GMDF*α*-NMF-DTD enumerated as the sum of those required separately for GMDF*α* and NMF-DTD, whose hardware resource requirement during the processing of one frame is described in section II.

Table 5.2 lists the number of memory locations and arithmetic operations required by each algorithm to produce one output sample, based on the parameters specified in section 4.3.6. Examining Table 5.2, GMDF*α*-NMF-DTD requires significantly more hardware resources than GMDF*α*-MDF-DTD, which entails 2271 less AOs and 17077 less MLs. A contributory factor for this finding is that much of the hardware resources required for the background and foreground GMDF*α* of GMDF*α*-MDF-DTD can be shared, allowing for resource savings. In contrast, the scope for such savings for GMDF*α*-NMF-DTD is considerable less, given that GMDF*α* and NMF-DTD each use different FFT sizes.

Another practical consideration for the AE problem, and therefore DTD, is the processing delay, with less delay being more desirable. For the parameters specified above, both GMDF*α*-MDF-DTD and GMDF*α*-NMF-DTD incur a 64 ms delay associated with block/frame (overlapping) that may be prohibitively large in certain applications. However, it was demonstrated in the previous chapter that NMF-NSE can be configured to process shorter frames (smaller N), and choosing R_v, R_d, and m_1 approprtately can optimize performance. By selecting these same parameters for NMF-DTD such that $\hat{\mathbf{d}}(k)$ to approximates $\mathbf{d}(k)$ optimally for the current frame size, with the corresponding parameters of GMDF*α* selected, the buffering delay of GMDF*α*-NMF-DTD can be reduced.

5.4 Chapter Summary

In general, conventional DTDs erroneously detect DT upon initiation and after room changes, effecting slower AEC adaptation, and thus, prolonging echo disturbance for the far-end user. To address this issue, a novel DTD approach, named Nonnegative Matrix Factorization DTD (NMF-DTD), was presented. NMF-DTD calculates it decision variable, smoothed normalised inner product, from the output of a background instantiation of NMF-NSE and the observable signals. This test variable controls a block-based foreground adaptive filter in a manner similar to the foreground/background filter structure prevalent in conventional AEC/DTD.

It was demonstrated using standard ROC analysis that NMF-DTD matches the true positive rate of a representative example of conventional DTD in both variable and fixed enclosures, verifying that NMF-DTD can protect an AEC from DT. In fixed enclosures, with the adaptive filters initialized to the correct room response, NMF-NSE falsely detects DT slightly more than conventional DTD. Although, in more realistic experiments, in which initial convergence is required and room changes occur, NMF-DTD exhibits consistent performance, whereas, the performance of conventional DTD deteriorates such that its false positive rate now exceeds that of NMF-DTD. In addition, it was shown, that the false positives of NMF-DTD differ characteristically from those of conventional DTD, in that, they are independent of the convergence of the adaptive filter, but are dependent on echo matching, and as such are distributed more evenly over time. By virtue of this trait, NMF-DTD was shown to allow its foreground adaptive filter to converge at a rate approximately matching that of the maximum rate, i.e. the convergence rate without DTD, enabling approximately optimum echo cancellation; in contrast, the conventional DTD significantly slowed the convergence of its AEC during initiation and after room change. From a comparison of the hardware resources of the algorithms it was shown that NMF-DTD requires approximately 250 % greater arithmetic operations and approximately 170 % more memory locations than conventional block based frequency domain AEC-DTD, implying that an AEC-DTD system incorporating NMF-NSE as a DTD requires a comparatively high hardware resource requirement; NMF-DTD though has a lower hardware resource requirement than most time-domain AEC-DTD (see Chapter 4).

6 On Mitigating All-Pass Phase Distortion in the context of Non-Minimum Phase Room Impulse Response Inversion for Low-Delay Dereverberation

This chapter explores the properties of all-pass phase distortion in the context of inverse filtering for dereverberation of speech signals using the minimum phase component of a non-minimum phase RIR. This chapter describes how all-pass phase distortion is suppressed in reverberated speech signals due to the magnitude response of the RIR attributable to maximum phase zeros of the RIR, but is exposed by the magnitude response of the minimum phase inverse filter attributable to these same zeros. Based on this description, it is explained how recent RIR inversion techiques, which typically employ smoothed RIRs, mitigate this distortion. This description also motivates two novel approaches to inversion that mitigate all-pass phase distortion while maintaining low delay, a desirable characteristic of minimum phase inverse filtering, especially for dereverberation applications. One approach modifies the inverse minimum phase filter prior to inversion such that the magnitude response of the minimum phase inverse filter attributable to maximum phase zeros is suppressed, thereby mitigating all-pass phase distortion in the processed speech, while the second approach is based on applying model-based MSSS and NMF, as used in previous chapters, to separate the speech and phase distortion in the magnitude STFT domain. The performance of these algorithms is demonstrated by way of comparative listening tests with an existing inversion algorithm.

6.1 Introduction and Background

As described in chapter 2, during hands-free telephone usage in an enclosure, the near-end users speech signal, $v(n)$, is typically reverberated and is thus less intelligible and of lower quality than compared to during conventional handheld telephone usage [1, 2]. To restore $v(n)$

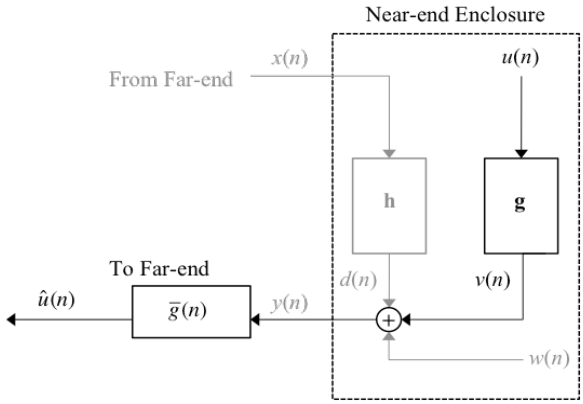

Figure 6.1: Block Diagram of inverse filtering for derevberation, acoustic echo and near-end noise have been grayed-out as they are assummed negligible in this chapter.

to $u(n)$ (or estimate of $\hat{u}(n)$), its non-reverberated original, a variety of dereverberation algorithms for both the single and multi-microphone scenarios have been proposed [2]. One well-known approach to dereverberation is inverse filtering, also known as deconvolution. For dereverberation, inverse filtering may be construed as solving two separate problems, namely, obtaining an estimate of the impulse response from the users lips to the recording microphone, $g(n)$, and then inverting this estimate; the inverse is convolved with $v(n)$ to perform dereverberation, as illustrated in Figure 6.1. In this chapter, we address the latter problem, that is, inversion of $g(n)$, specifically in the single channel/microphone case. While the single channel RIR inversion problem has received much less research attention than the related multi-microphone inversion problem [2, 386-388], which is perhaps due to the relative difficulty of the single channel RIR estimation problem [389-391], there are, nonetheless, many scenarios in which a single microphone recording of a speech signals is available and where dereverberation is desired.

Single RIR inversion has been researched extensively for room equalization applications [392], where it is typically assummed that a accurate measurement of the RIR is available. In this application, the inverse of the RIR is used to pre-filter an audio signal before it is radiated by a loudspeaker such that the effect of the room (reverberation, often also including the effect of the loudspeaker) at the point in the room where the RIR was measured is removed or suppressed; this is mathematically equivalent to post-filtering a reverberated signal. It has been shown that inverting RIRs is problematic because they typically possess non-minimum phase characteristics [393], and thus, a delay is required to realize the inverse filter [394]; moreover, such inverse filters typically require very long lengths and are sensitive to spatial displacement [392, 394, 395]. For dereverberation, which is typically aimed at real-time telecommunication applications, such a delay is prohibitive. An alternative inversion approach, also investigated primarily for audio equalization, is to equalize the magnitude response of $g(n)$ by inverting the minimum phase filter of $g(n)$ [393] and convolving this inverse with $v(n)$. This approach incurs no delay, but does not fully address phase distortion, which manifests as perceptually untenable audible artifacts in the processed speech [7, 393] in both the audio equalization and dereverberation applications.

In this chapter, we describe the cause and properties of this phase distortion. From this description, we present two novel approaches, both of which aim to optimally remove the magnitude distortion resulting from $g(n)$, while minimizing the delay and the effects of phase distortion. While these two approaches do not address the spatial sensitivity of inverse RIRs, which is a significant issue for audio equalization applications, their attributes render them suitable for low-delay inverse filtering or dereverberation applications when the speaker is quasi-stationary, or the RIR is adaptively estimated.

The remainder of this chapter is structured as follows: section 6.2 formulates and describes the single channel RIR inversion problem including a review of existing methods from the field of audio equalization, section 6.3 explains in detail the cause of all-pass phase distortion, highlights some of its properties, and motivates the novel schemes to be described in sections 6.4 and 6.5. Then, in section 6.6 the proposed schemes are evaluated on real RIRs by a comparative listening test with an existing RIR inversion technique, with the chapter summary contained in section 6.7. The purpose of this exploration is to understand its properties such that low delay dreveberation can be attained. Note that for tractability of scope, it is assumed henceforth that an accurate estimate of $g(n)$ is available. This estimate may have been obtained empirically, by direct measurement a priori [396, 397], or estimated, via blind system identification techniques [2]. We also make the simplifying assumptions of negligible noise and negligible acoustic echo i.e. $w(n) \approx 0$, $d(n) \approx 0$, and we assume that $g(n)$ is time-invariant.

6.2 Single Microphone Room Impulse Response Inversion

Given a stable, casual, and non-minimum phase RIR $g(n)$, which, recall from chapter 1, is truncated at sample L_g, the ideal objective of single channel RIR inversion is to obtain the inverse filter of $g(n)$, denoted $\overline{g}(n)$, such that,

$$\delta(n) = g(n) * \overline{g}(n), \quad n = 0, 1, \ldots \tag{6.1}$$

where $\delta(n)$ is a unit sample sequence, ($\delta(n) = 1$ for $n = 0$, and $\delta(n) = 0$ for all remaining n) and $*$ is the discrete linear convolution operator. This objective can be expressed equivalently in the z-domain as,

$$1 = G(z)\overline{G}(z), \tag{6.2}$$

where $G(z)$ and $\overline{G}(z)$ denote the transfer functions of $g(n)$ and $\overline{g}(n)$ respectively. Since $G(z)$ is non-minimum phase, that is, contains zero(s) that are located outside the unit circle, a straight inversion of $G(z)$ to obtain a casual and stable (right-sided) inverse filter to satisfy (6.1)(6.2) will result in poles located outside the unit circle. This implies an inverse filter with an unstable impulse sequence that is consequently unrealizable in practice. As such, the criterion of (6.1)(6.2) is not obtainable for non-minimum phase RIRs.

If the requirement is distortionless signal processing [398], that is, no waveshape change, then the requirements for inversion can be relaxed somewhat, such that the criteria for inversion of $g(n)$ in the time domain becomes,

$$\kappa\delta(n-t_g) = g(n)*\overline{g}(n), \quad n= 0,1,.. \tag{6.3}$$

where $\kappa > 0$ is a scaling factor, and $t_g \geq 0$ is a delay factor expressed in samples. The equivalent criteria in the z-domain becomes,

$$\kappa e^{-j\omega t_g} = G(z)\overline{G}(z), \tag{6.4}$$

which indicates that a constant magnitude response i.e. $|G(e^{j\omega})||\overline{G}(e^{j\omega})| = \kappa$, and a phase response proportional to frequency i.e. $\angle[G(e^{j\omega})\overline{G}(e^{j\omega})] = -\omega t_g$, are required for distortion-less processing; deviations from a constant magnitude response or from a linear phase are known as magnitude and phase distortion respectively [398], terms that will be used extensively henceforth. This criterion allows a two-sided (acausal) inverse filter to be considered, in which the poles of $\overline{G}(z)$ that are outside the unit circle manifest as convergent, anti-causal geometric sequences in its impulse response. Assuming that this two-sided or acasual impulse response may be truncated after some term in both the anti-casual and casual direction, this filter is realizable as a Finite Impulse Response (FIR) filter by delaying (right-shifting) the impulse response such that its anti-causal component is rendered casual [394]. In the audio equalization field, attempts to attain such a filter have been referred to as full or ideal RIR inversion [399].

An early method for obtaining an estimate of $\overline{g}(n)$, with a view to obtaining full RIR inversion satisfying (6.3)(6.4) is by Least Squares [400]. Obtaining such an estimate amounts to minimizing the following error,

$$\overline{g}_{\mathrm{LS}}(n) = \min_{\overline{g}_{\mathrm{LS}}(n)} \left\| g(n)*\overline{g}_{\mathrm{LS}}(n) - \kappa\delta(n- t_g) \right\|_2^2, \tag{6.5}$$

where $\overline{g}_{\mathrm{LS}}(n)$ is a length $L_{\overline{g}_{\mathrm{LS}}}$ least squares estimate of $\overline{g}(n)$, and $\delta(n)$ is a length $L_g + L_{\overline{g}_{\mathrm{LS}}} - 1$ unit sample sequence. Alternatively, an estimate of $\overline{g}(n)$ may be attained in the frequency domain. For this approach, the frequency responses of both $\delta(n - t_g)$ and $g(n)$ are computed using the DFT, the frequency response of the former sequence is divided (bin-wise) by the frequency response of the latter sequence, and the inverse filter is yielded by taking the Inverse DFT (IDFT) of the resulting sequence [401]. This frequency domain approach is less computational intensive than the time domain approach of Least Squares, due to the use of the Fast Fourier Transform (FFT) in performing the DFT and IDFT operations, but relative to Least Squares this approach was shown to produce more error [400], which was ascribed to time-aliasing from the use of a finite window length FFT.

For optimum full or ideal RIR inversion approach using Least Squares, the model filter length $L_{\overline{g}_{\mathrm{LS}}}$ and the model delay t_g should be chosen so that there is minimal truncation of the higher order terms in the inverse impulse response; similarly, in the case of DFT methods, assuming $L_{\overline{g}_{\mathrm{LS}}}$ is the FFT length, $L_{\overline{g}_{\mathrm{LS}}}$ and t_g should be chosen to minimise time-aliasing. It is known that room transfer functions typically contain some zeros, both minimum and maximum phase, located very close to the unit circle, referred to as high-Q zeros [399]. Upon inversion these zeros become 'ringing' poles that contribute slowly decaying, casual and anti-casual, geometric sequences to the inverse impulse response, which implies that both the

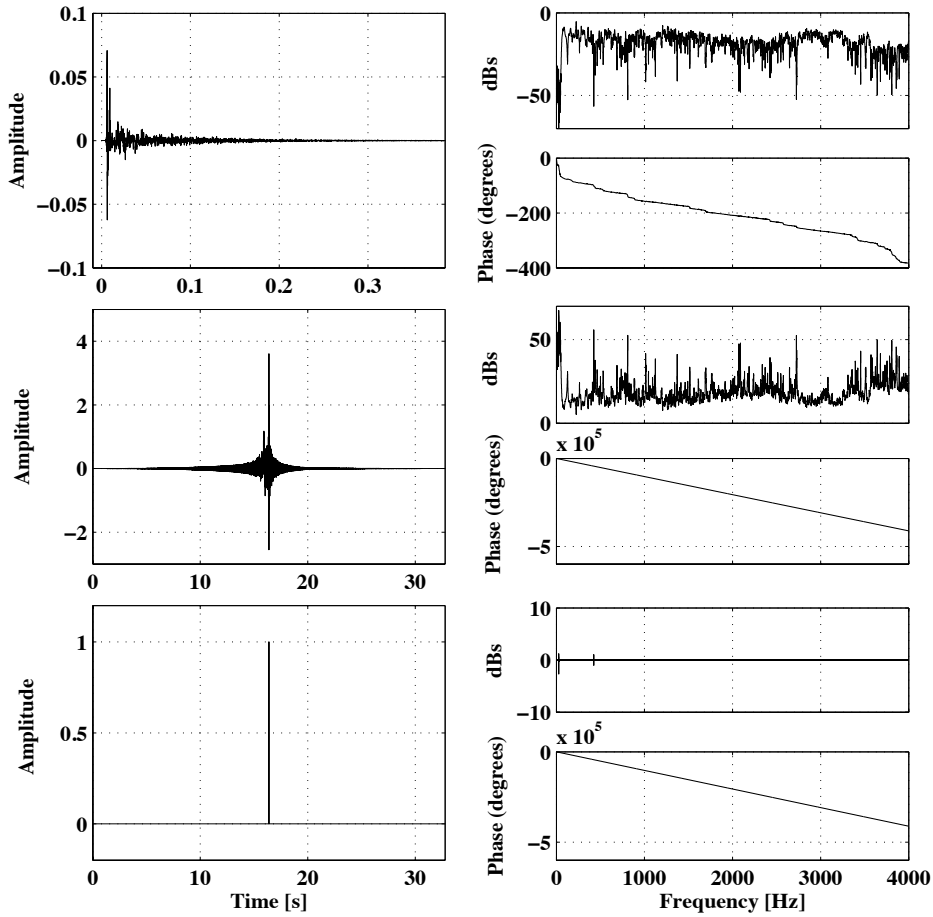

Figure 6.2: Left column from top: MARDY RIR recorded approximately 1 m from a loudspeaker (filename = ir_1_L_1.wav)), Corresponding inverse filter computed using DFT (FFT length 2^{17}, symmetric delay), resulting delayed unit sample function yielded from convolving the RIR with its inverse filter (also known as the Equalized response). Right column from top: MARDY RIR magnitude and phase response, inverse magnitude and phase response, equalized response magnitude and phase response.

length, $L_{\bar{g}_{LS}}$, and delay, t_g, are typically required to be very long [399]. To demonstrate this, Figure 6.2 displays the inverse of an RIR, computed using the FFT, along with the original RIR and the result from convolving this RIR with its computed inverse. As exemplified by this figure, for real-time applications, where low latency is required and where hardware resources are limited, the typical delay required for causality of inverse RIR filters and their overall length are excessively long, and thus, represent significant obstacles to the deployment of full or ideal inverse filtering in that context. Although these problems may be alleviated somewhat by truncating the inverse filter, this comes at the expensive of dereverberation performance, and perceptually detrimental audible artifacts [395, 401], which are discussed in further detail below.

Another significant problem for full or ideal RIR inversion is the spatial sensitivity of the inverse [392, 394, 395]. While we do not address this issue explicitly in this work, it has motivated many of the more recent approaches to single RIR inversion for audio equalization, for which it is desirable to equalize the effects of a room over a wide listening area. Typically, RIRs vary greatly within a room, and a relatively small difference between any two RIRs implies large differences between their inverses. In practice, this means that the inverse of an RIR recorded at one position in a room cannot, in general, be used to invert an RIR recorded

some distance away in the same room; indeed, this can actually increase distortion, particularly at high frequencies [392, 402]. In equalization applications, this lack of spatial robustness can manifest as pre-echo [401, 403, 404]. Pre-echo is attributable to the anti-casual component of the acausal RIR inverse, which, for listeners located away from the equalized point can be perceived as distinct echo before the main signal. As this echo arrives before the main signal rather than after, during which temporal masking has a greater effect, they are considerably more objectionable than the original unprocessed sound [401]. Analogously, in the post filtering dereverberation context this corresponds to filtering a reverberated signal with the inverse of an RIR measured in a different position, with equivalent resulting pre-echo. The use of ideal inverse filters therefore is limited to a region around the position at which the RIR was measured, or estimated; the extent of this region has been investigated in [401, 405], where it was shown to be typically very small. In the dereverberation context, this spatial sensitivity implies that the RIR should be adaptively estimated, or the speaker should be stationary if using a measured response.

Apart from ideal inversion, another approach to single channel RIR inversion is to partially invert $g(n)$ such that the resulting inverse filter equalizes the magnitude response of $g(n)$. A standard procedure to achieve this is to first decompose $g(n)$ into its minimum phase sequence, $g_{mp}(n)$, and its all-pass sequence, $g_{ap}(n)$, such that $g(n) = g_{mp}(n)*g_{ap}(n)$, and then invert $g_{mp}(n)$ it to obtain the inherently stable inverse filter, $\bar{g}_{mp}(n)$ [393]. As $g_{mp}(n)$ is minimum phase, $\bar{g}_{mp}(n)$ is minimum phase, and thus this inversion approach requires no extra delay and may, as such, be construed as an attempt to satisfy the criteria expressed in (6.1)(6.2).

Before describing an early method for obtaining this decomposition, we will first specify and describe some of the properties of the sequences $g_{mp}(n)$ and $g_{ap}(n)$. The magnitude responses of $g_{mp}(n)$ and $g_{ap}(n)$ are defined as [393, 406],

$$|G_{mp}(e^{j\omega})| = |G(e^{j\omega})|, \quad |G_{ap}(e^{j\omega})| = 1, \tag{6.6}$$

where $G_{mp}(z)$ and $G_{ap}(z)$ denote the transfer functions of $g_{mp}(n)$ and $g_{ap}(n)$ respectively. $G_{mp}(z)$ satisfies (6.6) by having all the minimum phase zeros of $G(z)$, and an additional zero, mirrored about the unit circle, for each maximum phase zero of $G(z)$, or stated differently, a zero located at the reciprocal radius of each maximum phase zero of $G(z)$. This satisfies (6.6) because mirroring a zero about the unit circle does not change the magnitude response of a filter; its phase response however, is altered. As its name implies, the phase response of $g_{mp}(n)$ is the minimum phase required for $|G_{mp}(e^{j\omega})|$, and is related to $|G_{mp}(e^{j\omega})|$ through the Hilbert transform. The remaining excess phase distortion is captured in the phase response of $g_{ap}(n)$, which contains each maximum phase zero of $G(z)$ and an additional pole for each such zero, located at the reciprocal radius, giving a unity magnitude response per (6.6). When reconstructing $G(z)$, the introduced poles in $G_{ap}(z)$ are cancelled by the introduced zeros of $G_{mp}(z)$ such that $G(z) = G_{mp}(z)G_{ap}(z)$.

The inverse of $G_{\text{mp}}(z)$ will contain poles that are inside the unit circle, giving a stable and causal inverse filter. A straight inversion of $G_{\text{ap}}(z)$ will contain both poles and zeros, in recipricol radius pairs, with the poles and zeros located, respectively, outside and inside the unit circle. By definition, inverting $g_{\text{mp}}(n)$ and convolving the inverse with $g(n)$, corresponds to compensating for the magnitude response of $g(n)$, thereby eliminating magnitude distortion; the remaining phase distortion however, represented by $g_{\text{ap}}(n)$, is unaddressed.

It is worth noting that, while we refer to the sequence $g_{\text{mp}}(n)$ as the minimum phase filter in this chapter in accordance with contemporary literature, as pointed out in [394], when it was originally introduced in [406] $g_{\text{mp}}(n)$ was referred to as the effective minimum phase filter, with the terms minimum phase filter and maximum phase filter referring to, respectively, those filters that result from decomposing a non-minimum phase filter into its minimum phase zeros, and its maximum phase zeros. In this context, the magnitude response of the effective minimum phase filter (referred to as minimum phase filter herein), contains the combined magnitude response of both the maximum and minimum phase filters, while its phase response is made up of the phase reponse of the minimum phase zeros and the phase response of the maximum phase zeros reflected about the unit circle.

The standard early method [406] for computing an estimate of $\overline{g}_{\text{mp}}(n)$, is to first decompose $g(n)$ into $g_{\text{mp}}(n)$ and $g_{\text{ap}}(n)$ via homomorphic processing and then invert $g_{\text{mp}}(n)$ in the frequency domain. In this chapter, we employ this approach exclusively; an estimate of $\overline{g}_{\text{mp}}(n)$ may also be obtained using Least Squares by setting $t_g = 0$ [400]. The homomorphic decomposition approach exploits the fact that the cepstrum of the minimum phase sequence can be obtained by zeroing the top half of the real cepstrum of $g(n)$. Such a non-parametric method is preferred over explicit factorization of $g(n)$ into its roots (zeros), which can be numerically problematic [407]. The homomorphic method for obtaining this factorization may be stated as follows [7, 406, 408, 409].

1. Compute the Discrete Fourier Transform (DFT) of $g(n)$,

$$G(l) = \sum_{n=0}^{L_{ap}-1} g(n) e^{-j(2\pi/L_{ap})ln}, \tag{6.7}$$

   where l denotes frequency bin index and L_{ap} represents the number of DFT points; as L_{ap} is generally greater than L_{g}, $g(n)$ is zero-padded up to L_{ap}. Note that for notational convenience, we refer to the estimates of $g_{\text{mp}}(n)$, $g_{\text{ap}}(n)$ and $\overline{g}_{\text{mp}}(n)$ as $g_{\text{mp}}(n)$, $g_{\text{ap}}(n)$ and $\overline{g}_{\text{mp}}(n)$ respectively.

2. Compute the real part of the complex cepstrum of the sequence $g(n)$, using the Inverse DFT (IDFT),

$$\tilde{g}(n) = \frac{1}{L_{ap}} \sum_{l=0}^{L_{ap}-1} \log |G(l)| e^{j(2\pi/L_{ap})ln}. \tag{6.8}$$

3. Calculate the real cepstrum $\tilde{g}_{\text{mp}}(n)$ of the minimum phase sequence/filter,

$$\tilde{g}_{\text{mp}}(n) = \begin{cases} \dfrac{\tilde{g}(n)}{\varphi}, & n = 0, \dfrac{L_{\text{ap}}}{2}, \\ \dfrac{2\tilde{g}(n)}{\varphi}, & 1 \le n \le \dfrac{L_{\text{ap}}}{2}, \\ 0, & \dfrac{L_{\text{ap}}}{2} < n \le L_{\text{ap}} - 1, \end{cases} \tag{6.9}$$

where φ is a positive integer [408]. For the standard decomposition $\varphi = 1$, which we assume henceforth unless otherwise stated.

4. Transform the resulting minimum phase cepstrum, $\tilde{g}_{\text{mp}}(n)$, back to the frequency domain using the DFT

$$\tilde{G}_{\text{mp}}(l) = \sum_{n=0}^{L_{\text{ap}}-1} \tilde{g}_{\text{mp}}(n) e^{-j(2\pi/L_{\text{ap}})ln}. \tag{6.10}$$

5. Compute the frequency response of the minimum phase filter,

$$G_{\text{mp}}(l) = \exp(\tilde{G}_{\text{mp}}(l)). \tag{6.11}$$

6. The all-pass frequency response, $G_{\text{ap}}(l)$, is computed by dividing out $G_{\text{mp}}(l)$ from $G(l)$,

$$G_{\text{ap}}(l) = \frac{G(l)}{G_{\text{mp}}(l)}. \tag{6.12}$$

7. The frequency response of the minimum phase inverse filter, $\overline{G}_{\text{mp}}(l)$, is computed as,

$$\overline{G}_{\text{mp}}(l) = \frac{1}{G_{\text{mp}}(l)}. \tag{6.13}$$

8. The equalized response, $G_{\text{eq}}(l)$, which is equivalent to $G_{\text{ap}}(l)$, is expressed as,

$$G_{\text{eq}}(l) = G(l)\overline{G}_{\text{mp}}(l). \tag{6.14}$$

Note that $G_{\text{eq}}(l)$ is useful for analyzing the overall performance of $\overline{G}_{\text{mp}}(l)$, and will be employed for this purpose later in this chapter. A flat equalized magnitude response i.e. $|G_{\text{eq}}(l)| = \kappa \ \forall l$, signifies full magnitude equalization and a linear phase response i.e. $\angle G_{\text{eq}}(l) = -\omega t_g \ \forall l$, signifying no phase distortion; deviations from a constant magnitude or linear phase indicate magnitude distortion or phase distortion at the corresponding frequencies in the processed signal.

9. Equipped with $G_{\text{mp}}(l)$, $G_{\text{ap}}(l)$ and $\overline{G}_{\text{mp}}(l)$, the respective impulse responses are obtained using the IFFT; for example

$$\overline{g}_{\text{mp}}(n) = \frac{1}{L_{\text{ap}}} \sum_{l=0}^{L_{\text{ap}}-1} \overline{G}_{\text{mp}}(l) e^{j(2\pi/L_{\text{ap}})ln}, \tag{6.15}$$

where $g_{\text{mp}}(n)$, $g_{\text{ap}}(n)$, and $\overline{g}_{\text{mp}}(n)$ denote the impulse responses of $G_{\text{mp}}(z)$, $G_{\text{ap}}(z)$ and $\overline{G}_{\text{mp}}(z)$, respectively.

For adequate inversion of $g_{\text{mp}}(n)$, L_{ap} should be set such that time aliasing is minimized in $\overline{g}_{\text{mp}}(n)$ and $g_{\text{ap}}(n)$, both of whose transfer functions contain poles. Similar to full RIR inversion therefore, this implies large DFT lengths due to the effect of high-Q zeros; the computed $g_{\text{mp}}(n)$ however may be truncated to L_{g} as $G_{\text{mp}}(z)$ contains no poles. Figure 6.3

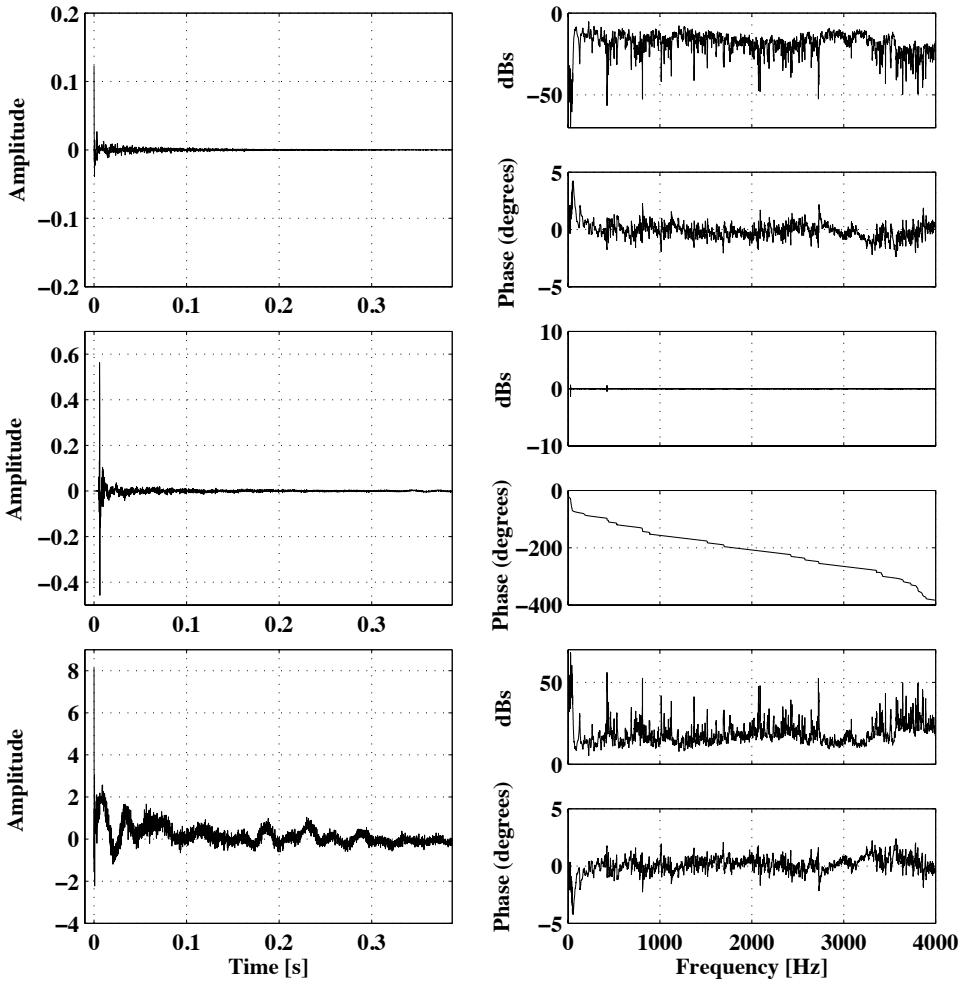

Figure 6.3: Left column from top: Minimum phase impulse response, All-pass impulse response (truncated), and Inverse minimum phase impulse response (truncated); from RIR in Figure 6.2. Right column from top: Minimum phase impulse response magnitude and phase response, All-pass magnitude and phase response, and Inverse minimum phase magnitude and phase response.

displays an example $g_{\text{mp}}(n)$, $g_{\text{ap}}(n)$ and $\overline{g}_{\text{mp}}(n)$ impulse response, which were computed using the homomorphic approach for the RIR displayed in Figure 6.2, the corresponding magnitude, and unwrapped phase responses, are also displayed.

The filter $\overline{g}_{\text{mp}}(n)$ can be applied to the reverberated speech signal $v(n)$ to remove the effect of the minimum phase component, $g_{\text{mp}}(n)$, of the RID, $g(n)$,

$$v_{\text{ap}}(n) = v(n) * \overline{g}_{\text{mp}}(n), \tag{6.16}$$

where $v_{\text{ap}}(n)$ denotes the partially dereverberated, minimum phase inverted speech signal, or simply the processed speech signal.

As is evident from Figure 6.3, $g_{\text{ap}}(n)$ generally contains a significant proportion of the reverberant energy of $g(n)$, and as such, $\overline{g}_{\text{mp}}(n)$ is generally unable to approximate the criteria for inversion dictated by (6.1) and (6.2). Moreover, it is known that the processed signal, $v_{\text{ap}}(n)$, typically contains distinct audible artifacts that have a perceptually detrimental effect on its quality [7, 393]; these artifacts have been attributed to the unaddressed phase distortion, of $g_{\text{ap}}(n)$ i.e. $\angle G_{ap}(k)$ [7, 393]. In [7], these artifacts were described as tonal and metallic

sounding, and have been likened to the sound of a bell chime in [7, 393, 408]. From comparing numerous different $v(n)$ and $v_{ap}(n)$ signals, generated from various MARDY RIRs and different TIMIT speech signals, we can concur with these descriptions, and similarly opine that they are considerably distracting, so much so that the unprocessed $v(n)$ signals are more preferable, particularly for highly reverberant speech signals, confirming that inverting $g_{mp}(n)$ alone is not advisable for RIR inversion. However, by focusing attention solely on the speech component of the various $v_{ap}(n)$ signals, which is possible given the distinctly different characteristics of the artifacts and speech component of $v_{ap}(n)$, it is apparent that this speech is dereverberated, an observation also noted in [393]. This feature of $v_{ap}(n)$, perhaps unsurprising given that the magnitude distortion of $g(n)$ has been removed, suggests that if the phase related artifacts could be removed form $v_{ap}(n)$, the resulting signal would, subjectively speaking, contain artifact-free dereverberated speech. In addition to the audible artifacts, the minimum phase inverse filter $\overline{g}_{mp}(n)$ is sensitive to spatial displacement, but unlike ideal RIR inverse filters, $\overline{g}_{mp}(n)$ does not have an anti-casual component, and therefore does not give rise to pre-echo effects.

To directly address the audible artifacts in $v_{ap}(n)$, the all-pass filter $g_{ap}(n)$ may also be inverted to satisfy the criteria in (6.3) and (6.4). The resulting inverse, $\overline{g}_{ap}(n)$, can be applied to $v_{ap}(n)$ to further dereverberate $v(n)$, or alternatively, the minimum phase and all-pass inverse filters can be combined i.e. $\overline{g}_{mp}(n) * \overline{g}_{ap}(n)$, before then being convolved with $v(n)$. The all-pass inverse, $\overline{g}_{ap}(n)$, can be obtained using either Least Squares or the FFT, or more simply in this case, by time reversing $g_{ap}(n)$, and shifting the reversed $g_{ap}(n)$ until it is rendered casual [7]. However, as is evident in Figure 6.3, the long length of $g_{ap}(n)$, attributable to 'ringing' poles in $G_{ap}(z)$ located at the reciprocal radius of the high-Q maximum phase zeros in $G(z)$, implies a long delay in order to render $\overline{g}_{ap}(n)$ casual. Indeed, by separately inverting both $g_{mp}(n)$ and $g_{ap}(n)$, and convolving their inverses, the resulting filter is equivalent to the full or ideal inverse described above, and hence, has the same issues regarding delay and length; additionally, this appraoch leads to greater error (due to time-aliasing) in $\overline{g}_{mp}(n) * \overline{g}_{ap}(n)$ relative to the Least Squares inverse [400].

As mentioned above, perceptually detrimental artifacts have also been reported in the context of full RIR inversion [401, 404]; specifically, they were referred to as cymbals in [401], matching the bell chime description of the artifacts in $v_{ap}(n)$. From comparing the speech signals produced after processing by various $\overline{g}_{mp}(n)$ and undermodeled $\overline{g}_{LS}(n)$ filters, we note that the audible artifacts have a similar quality; though, they are much more prominent in $v_{ap}(n)$. Recalling that $\overline{g}_{LS}(n)$ is equivalent to $\overline{g}_{mp}(n)$ for $t_g = 0$, and that $\overline{g}_{ap}(n)$ manifests in the anti-casual component of $\overline{g}_{LS}(n)$, we assume, for the present, that for both full RIR inversion and minimum phase RIR inversion, for which $\overline{g}_{ap}(n)$ is explicitly not modeled and partially modeled depending on the modeling delay, under-modeling of $\overline{g}_{ap}(n)$, and the

APPROACH	FILTER SHAPE	MAGNITUDE COMPENSATION	PHASE COMPENSATION	AUDIBLE ARTIFACTS	DELAY
Full RIR inversion	Two-sided (acasual)	Full compensation, provided filter length and delay are sufficiently long	Full compensation, provided filter length and delay are sufficiently long	No, provided filter length and delay are sufficiently long	Yes, necessary for causality
Minimum phase RIR inversion	Right-sided (causal)	Full compensation, provided filter length is sufficiently long	Excess phase distortion (all-pass phase distortion) remains unaddressed	Yes, due to all-pass phase distortion	No.

Table 6.1 : Summary table contrasting properties of full or ideal RIR inversion and minimum phase RIR inversion.

resulting residual phase distortion, cause the described audible artifacts in the processed speech. We will return to this topic in more detail later in section 6.3.

The preceding single channel RIR inversion schemes, i.e. full or ideal inversion and minimum phase filter inversion, constitute the early approaches to this problem; these schemes are summarised in Table 6.1. More recently, to address the various problems associated with these early schemes, numerous alternative inversion schemes have been devised, aimed mostly at room equalization applications [392, 402, 410-417]. In general, these schemes produce more tractable inverse filters, but do not admit an inversion of $g(n)$ by the criteria set forth in (6.1),(6.2), or (6.3),(6.4), and aim instead to optimize inversion performance by some other criteria.

One such approach involves applying either uniform or non-uniform smoothing [410, 411] directly to the frequency response (complex) of $g(n)$, and then inverting the smoothed response. This prior smoothing, known as complex smoothing, smoothes sharp/high-Q dips or nulls in the frequency response of $g(n)$ attributable to zeros near the unit circle (high-Q zeros), such that the peaks due to the corresponding poles are obviated in the magnitude response of the inverse filter, which is computed from the smoothed response using Least Squares. This implies that the original high-Q zeros are not fully inverted by the inverse filter, and as such, the inverse filter impulse response contains faster decaying casual and acausal sequences. Relative to ideal inversion using Least Squares, this approach has been shown to typically attain shorter inverse filters, with less delay, greater spatial robustness, and with an absence of audible artifacts [399, 410]; though for less overall dereverberation performance. In [412], the feasibility of inverting all-pole approximations of an RIR was explored. In the case of a minimum phase all-pole RIR approximation, the inability of the all-pole model to express zeros results in a smoothed frequency response without sharp dips due to high-Q zeros, similar to complex smoothing. Accordingly, this results in an (all-zero) inverse containing smoothed peaks, and a shorter impulse response, with no delay. It was also found that relative to all-zero models, all-pole approximations can model RIRs with a much lower order (factor of 40), and are more robust to spatial displacement, a fact exploited for common acoustical pole modeling for multi-point applications in [402]. The all-pole model also has a perceptual justification in that RIR transfer function zeros are perceptually less relevant than RIR poles [418, 419]. Warped frequency scales, particularly those based on perceptual principles such as

the bark scale or the fractional-octave scale have also been exploited for room equalization. Using these techniques, RIRs are approximated using low order warped FIR/IIR filters [414, 420], and Kautz filters [415, 416], with greater frequency resolution in perceptually pertinent frequencies. The resulting inverse filters, typically have a lower model order than complex smoothed responses, while having similar performance benefits [392]. Warped filtering is also beneficial from a spatial robustness perspective [413], as the frequency scales employed typically have greater resolution at low frequencies where the effect of poles corresponding to room resonances, which are more spatially uniform, reside. To mitigate these room resonances explicitly, which are problematic for sound reproduction in small rooms, modal equalization was proposed in [417]. For this approach, a selective filter implementation is employed, whereby the parameters (frequency and radius) of the responsible low frequency poles are identified, using peak finding and Q-factor or temporal decay rate estimation to estimate the freweqicies and radii, and are replaced with poles at the same frequency but closer to the origin (reduced radius) [421-424], thereby increasing the temporal decay rate of the poles. However, this approach is restricted to low frequencies, typically less than 200 Hz, the frequency range in which the poles corresponding to stable room modes reside [392]. Least squares RIR inversion with regularization has also been investigated for RIR inversion [425], where regularization is applied to influence the resulting inverse in a desirable manner, to prevent loudspeaker saturation for example, or to optimize the inversion performance by imposing perceptually motivated constraints [401, 404]. Instead of the 2-norm error criterion of Least Squares, the feasibility of a number of different norms, including the infinity norm, as the error criterion to compute the inverse filter along with perceptual principles was employed in [426]. For this approach, the perceptually adverse effects of the RIR, identified using a computational model of perception, are removed by the resultant inverse filter while the perceptually irrelevant parts of the RIR are not fully inverted such that the inverse can be shortened. Channel shortening concepts, imported from the telecommunications field, were applied to the problems of RIR inversion in [427]. The problem of audio equalization in car cabins has received special attention in [428]. The issues of audio equalization are summarized in the review paper [424].

Extensions have also been proposed to the standard homomorphic approach for computing the minimum phase inverse filter of an RIR. An iterative approach to obtaining $\overline{g}_{\mathrm{mp}}(n)$ was presented in [7], where for each iteration a fraction of $g_{\mathrm{mp}}(n)$ is computed and inverted. The fraction of the inverse filter generated at each iteration is then applied to $g(n)$ prior to the next iteration, with each iteration repeating steps 3-9. This process effects an iterative flattening of $|G_{\mathrm{eq}}(k)|$, and results in a set of short inverse filters, one for each iteration, which when convolved together form the complete inverse filter. A benefit of this approach is that the accumulating partial inverse filters can be monitored at each iteration, which enables the inverse process to be controlled according to some criteria; in [7] this was used to study perceivable phase distortion, from which an objective measure of phase distortion, discussed in section 6.3., was proposed. A second benefit of this approach is that time-aliasing error,

attributable to the use of a finite length FFT size, is minimized, since the cumalative time aliasing error from each partial inverse is less than that incurred from computing the entire inverse in one iteration.

This iterative homomorphic approach was analyzed in [408], where it was shown that the analysis of magnitude distortion can be simplified to the parameter φ, which was introduced in this chapter at step 3 above. Specifically, the influence of φ on the level of magnitude inversion by this approach is given by the following relationship [408],

$$\log \left|\overline{G}_{\text{mp}}(k)\right| = -\frac{1}{\varphi}\log \left|G_{\text{mp}}(k)\right|. \tag{6.17}$$

For $\varphi = 1$, the standard homomorphic decomposition of $g(n)$ is attained, and the resulting $\overline{g}_{\text{mp}}(n)$ removes the effect of $g_{\text{mp}}(n)$ from $g(n)$, thus equalizing the magnitude response of $g(n)$ i.e. $|G_{\text{eq}}(k)| = 1\ \forall k$, but leaving the excess phase corresponding to the filter $g_{\text{ap}}(n)$ unaddressed. For $\varphi > 1$, partial inversion of $g_{\text{mp}}(n)$ is achieved, with $\overline{g}_{\text{mp}}(n)$ compensating for a fraction, $1/\varphi$, of the log magnitude response of $g_{\text{mp}}(n)$, or equivalently, $1/\varphi$ of the log magnitude response of $g(n)$. Apart from the beneficial reduction in the length of $\overline{g}_{\text{mp}}(n)$ for $\varphi > 1$, which is due to an increase in the decay rate of the poles of $\overline{g}_{\text{mp}}(n)$, subjective listening tests in [7] report that as φ is increased the audibility of phase distortion is reduced in the processed speech signal. However, increasing φ also increases the fraction, i.e. $1\text{-}1/\varphi$, of magnitude distortion remaining in the processed speech.

As discussed in [408], the effect of $\varphi > 1$ on the standard homomorphic approach is to reduce the radii of the poles of $\overline{G}_{\text{mp}}(z)$. However, by acting on all the poles of $\overline{G}_{\text{mp}}(z)$ insufficient inversion performance may be attained [408]. In [408], a more selective inversion approach was proposed, specifically for RIRs that are dominated by a small number of high-Q zeros; implying a small number of dominant 'ringing' poles in the corresponding $\overline{g}_{\text{mp}}(n)$. Given such a $\overline{g}_{\text{mp}}(n)$, this approach aims to identify the dominant complex pole pairs of its transfer function $\overline{G}_{\text{mp}}(z)$, and replace them with a complex pole pair at the same frequency but with reduced radii. In this way, the energy in $\overline{g}_{\text{mp}}(n)$ decays quicker by selectively reducing the radii of those pole pairs that contribute most to the length of $\overline{g}_{\text{mp}}(n)$. This preserves the remaining poles such that magnitude distortion is not overly compromised. This approach was implemented using an iterative selective filter approach similar to that of modal equalization, with the location of spectral peaks of $|\overline{G}_{\text{mp}}(e^{j\omega})|$ and the their respective Q-factors being used to identify the parameters of the poles. This approach was evaluated through listening tests on data generated from car cabin responses, so chosen because that are dominated by a small number of zeros. It was reported that this approach effects a subjective improvement over the standard approach for $\varphi > 1$. It was also found that bell chime interference (phase distortion) is mitigated provided the radius of the poles is decreased sufficiently, a result congruent with the experiments with φ in [7]; likewise however, such a reduction in phase distortion comes at the expense of magnitude distortion.

These more recent inversion approaches can be broadly characterized as appropriately modifying the response $g(n)$ or $g_{mp}(n)$ prior to inversion, or alternatively, modifying the inverse, $\bar{g}(n)$ or $\bar{g}_{mp}(n)$, i.e. by smoothing, low order modeling, selective filtering; such that the resulting inverse has the following desirable properties: low delay, short length, spatial robustness, and an absense of audible artifacts in the processed speech. To achieve an inverse satisfying these properties, these approaches effectively obviate the full inversion of high-Q zeros in $G(z)$, such that the undesirable effects of the corresponding poles in the inverse are mitigated. While the reason this general approach is successful at producing inverse filters satisfying the three former properties is well known, we contend that the literature has yet to satisfactorily explain why the audible artifacts induced by all-pass phase distortion are mitigated by this approach; a point that has also been noted in [7, 399]. Indeed, save for references to phase distortion as bell chime artifacts [7, 393, 408] in the context of minimum phase equalization, and as audible artifacts in [399, 401, 404] in the context of full RIR equalization, and despite its perceptually deleterious effects, the issue of phase distortion/audible artifacts in the context of RIR inversion, both full and minimum phase, has received little attention in the literature; save for, as described above, a thorough investigation into the subjective effects of such distortion in the context of minimum phase inverse filtering [7], wherein the iterative homomorphic approach was devised. To remedy this, in the next section, we describe the effects of all-pass phase distortion on processed speech in the context of minimum phase RIR inversion, and explain how recent inversion techniques manage to mitigate such distortion. Furthermore, to achieve inverses with the above attributes, these more recent inversion approaches typically discard a significant amount of the detail of the response, which in certain real-time applications, such as dereverberation, in which the full RIR may be available and the speaker is quasi-stationary, or is adaptively estimated, may result in overly compromised inversion performance. In such situations, it is desirable to remove the maximum amount of magnitude distortion introduced by the RIR, using an inverse filter satisfying the properties of low delay, no audible artifacts and reasonable length; spatial sensitivity therefore is ignored. In the next section, we motivate two inversion approaches that realize these properties.

6.3 All-Pass Phase Distortion

The phase distortion introduced by a digital filter is commonly described by its group delay function, which is defined as the derivative of the unwrapped phase response of the filter. Using the DFT, the group delay function of $g(n)$ can be computed by the following formulas,

$$\tau_g(l) = -\mathrm{Im}\left[\frac{G'(l)}{G(l)}\right], \tag{6.18}$$

$$G'(l) = -i\sum_{n=0}^{L_{ap}-1} n g(n) e^{-j(2\pi/L_{ap})ln}. \tag{6.19}$$

The group delays functions of $g(n)$, $g_{mp}(n)$ and $g_{ap}(n)$ are denoted as $\tau(l)$, $\tau_{mp}(l)$ and $\tau_{ap}(l)$ respectively, and $\tau(l) = \tau_{mp}(l) + \tau_{ap}(l)$. Figure 6.4 displays the group delay function of the RIR in Figure 6.2, along with the group delay functions of the corresponding $g_{mp}(n)$ and $g_{ap}(n)$

sequences (displayed in Figure 6.3) computed using these formulas. Note that, from the definition of group delay, deviations from a constant group delay correspond to discontinuities in the unwrapped phase response, and therefore, such deviations indicate phase distortion at the corresponding frequencies.

To measure phase distortion in a perceptually meaningful way, a modified group delay function was proposed in [7], given as,

$$\tau_{\mathrm{M}}(l) = -\mathrm{Im}\left[\frac{G'_{\mathrm{eq}}(l)}{G_{\mathrm{eq}}(l)}\right]\cdot\frac{|G_{\mathrm{eq}}(l)|}{\max(|G_{\mathrm{eq}}(l)|)}. \tag{6.20}$$

In the context of minimum phase inverse filtering, this measure was proposed [7] under the hypothesize that perceivable phase distortion is due to disrupted magnitude and phase response relationships arising from such filtering, and implies that as the magnitude distortion of $g(n)$ is reduced by minimum phase inverse filtering, that is, by setting $\varphi \geq 1$ closer to 1, the all-pass phase distortion becomes more prominent, i.e. for $\varphi = 1$, $|G_{\mathrm{eq}}(l)| = \max(|G_{\mathrm{eq}}(l)|) = 1$. By using $\tau_{\mathrm{M}}(l)$ to monitor the iterative homomorphic inversion approach described earlier, it was shown that this measure correlated with the perception of all-pass phase distortion in $v_{\mathrm{ap}}(n)$. In what follows, we examine the properties of $g_{\mathrm{ap}}(n)$, and describe how spikes in the group delay function of $g_{\mathrm{ap}}(n)$ cause audible artifacts, as was first noted in [393]. Then, we explain the prominence of the phase distortion of $g_{\mathrm{ap}}(n)$ after minimum phase inverse filtering by contrasting $g(n)$ and $g_{\mathrm{ap}}(n)$, before assessing the above hypothesis, and describing how more recent inversion approaches mitigate the audible artifacts of phase distortion.

Given that, $\bar{g}_{\mathrm{mp}}(n) * v(n) = g_{\mathrm{ap}}(n) * u(n) = v_{\mathrm{ap}}(n)$, (assuming L_{ap} is sufficiently long), all-pass phase distortion can be described by examining the properties of the filter $g_{\mathrm{ap}}(n)$, which, being all-pass, is completely described by its group delay function, $\tau_{\mathrm{ap}}(l)$. As is evident from Figure 6.4, the phase distortion of $g_{\mathrm{ap}}(n)$ typically manifests as sharp peaks or spikes in $\tau_{\mathrm{ap}}(l)$, each of which are due to the combined group delay of a high-Q maximum phase zero, and a pole that lies at the recipricol radius. The remaining frequencies of $\tau_{\mathrm{ap}}(l)$, between the spikes, are typically flat with a constant group delay corresponding to the direct path delay, t_d (linear component of the unwrapped phase response). By the convolution of $u(n)$ with $g_{\mathrm{ap}}(n)$, the majority of $u(n)$, at the frequencies in the flat portions of $\tau_{\mathrm{ap}}(k)$, is subject to the same group delay i.e. t_d, and as such this portion of $u(n)$ experiences no phase distortion. In contrast, at the frequencies of group delay function spikes, $u(n)$ is sharply and variously delayed, which given the significant delays involved, results in the energy at these frequencies being de-synchronized from the rest of the processed signal; typically, outside the range of any temporal masking effects. These de-synchronized spectral components then give rise to the perceivable tones, or more descriptively, bell chime interference, in $v_{\mathrm{ap}}(n)$.

The distinctive bell chime interference of $v_{\mathrm{ap}}(n)$ is not perceived in $v(n)$, even though by definition $\tau(l) = \tau_{\mathrm{mp}}(l) + \tau_{\mathrm{ap}}(l)$. This seeming incongruence can be explained by contrasting some of the properties of $g(n)$ and $g_{\mathrm{ap}}(n)$. Firstly, the introduced poles of $g_{\mathrm{ap}}(n)$ are cancelled by the introduced zeros of $g_{\mathrm{mp}}(n)$ such that the phase distortion corresponding to the poles of $\tau_{\mathrm{ap}}(l)$ is cancelled in $\tau(l)$. The remaining phase distortion of $\tau_{\mathrm{ap}}(l)$, that is, excluding the effect

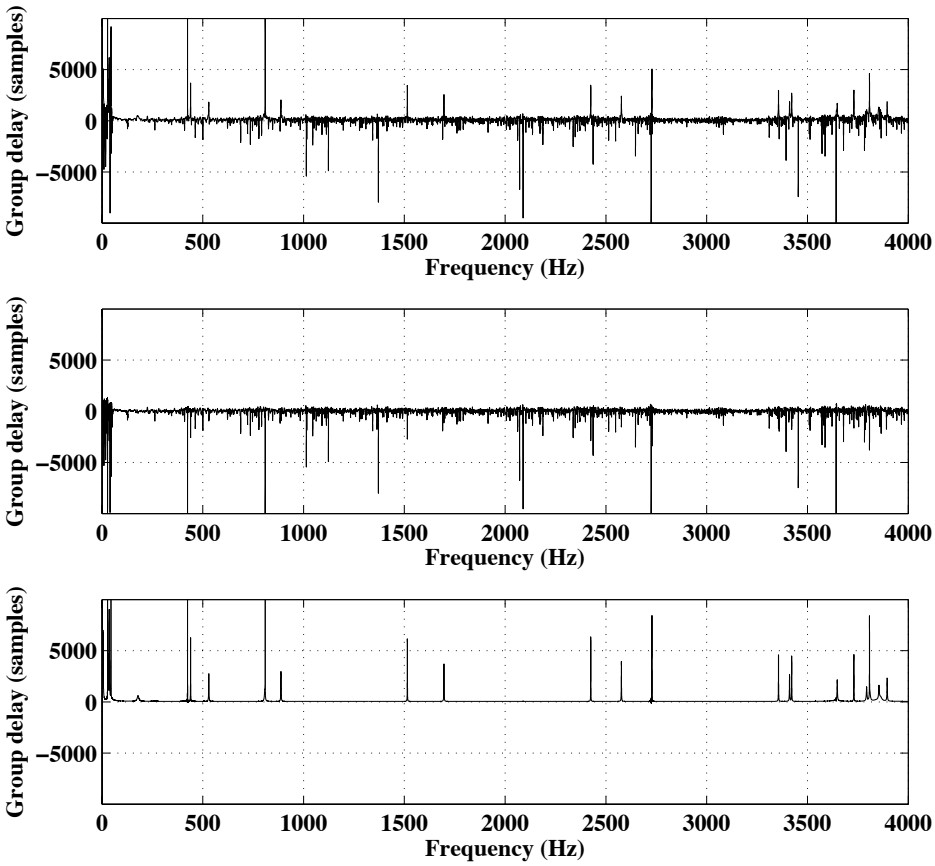

Figure 6.4: Top: Group delay function, $\tau(k)$, of the RIR in Figure 6.2. Center, minimum phase group delay function of RIR, $\tau_{mp}(k)$, and bottom, all-pass group delay function of RIR, $\tau_{ap}(k)$. The minimum phase and all-pass impulse responses are displayed in Figure 6.3.

of the poles, is attributable to maximum phase zeros. This component of the phase distortion of $\tau_{ap}(l)$ is perceivable absent from $v(n)$ because $g(n)$ contains the phase *and* magnitude response of the maximum phase zeros; $\tau_{ap}(k)$ contains only the phase response of these zeros. The magnitude response of $g(n)$ therefore, contains sharp dips at the frequencies of the spikes in $\tau_{ap}(k)$ (as is evident by comparing Figure 6.3 and Figure 6.4) that serve to suppress the phase distortion introduced at these frequencies. Therefore, in the case of $g(n)$, it may be stated that the phase distortion introduced by the maximum phase zeros is simultaneously suppressed by their magnitude response, which explains the perceived absence of the bell chime interference in reverberated speech signals such as $v(n)$.

In the context of minimum phase inverse filtering, since the minimum phase inverse filter, $\bar{g}_{mp}(n)$, contains a stable pole for each zero of $g(n)$ (located at the recipricol radius for maximum phase zeros), its magnitude response will exhibit sharp peaks at the frequencies of the high-Q zeros of $g(n)$, which includes the frequencies of the spikes in $\tau_{ap}(k)$, i.e. the frequencies of the high-Q maximum phase zeros of $g(n)$ (this again, is evident by comparing Figure 6.3 and Figure 6.4). When applied to $v(n)$ therefore, $\bar{g}_{mp}(n)$ will expose the hitherto suppressed phase distortion corresponding to the maximum phase zeros of $g(n)$. Therefore it is the magnitude response of $\bar{g}_{mp}(n)$ that enables this phase distortion to becaome audible in

$v_{ap}(n)$. Note that, the poles of $\bar{g}_{mp}(n)$ located at the recpirical radii of these zeros introduce additional phase distortion at these frequencies.

The success of the modified group delay, $\tau_M(l)$, as a perceptual measure of phase distortion may be explained similarly, whereby as the level of magnitude distortion removed by $\bar{g}_{mp}(n)$ is increased, by setting φ closer to 1 in (6.9), the level of phase distortion in $v(n)$ is increased, which reaches a maximum for $\varphi = 1$. The premise that underlies this measure, which was stated above, can now be modified to incorporate the new understanding put forth in this section, that is: the perception of audible artifacts in $v_{ap}(n)$ is due to the disrupted relationship between the *magnitude and phase response of the maximum phase zeros of the RIR*, a disruption which is inherent to minimum phase inverse filtering. In the case of acasual RIR inverses, with insufficient length or delay to encompase higher order terms, we ascribe similiar audible artifacts to the same cause, that is, a disruption between the magnitude and phase response of the maximum phase zeros reslting in phase distortion, but we are unclear as to how exactly this arises in that context.

Apart from full or ideal inverse filtering (with sufficient delay and length), the phase artifacts can be avoided by inverting the magnitude and phase of $g(n)$ attributable to the minimum phase zeros in $G(z)$ only, such that the relationship between the magnitude and phase response of the maximum phase zeros of $g(n)$ is not disrupted. In theory, such an inverse could be attained by zeroing the anti-causal component of a full (two-sided) derived inverse filter and left shifting to the origin, or by zeroing the anti-casual part of the complex cepstrum of $g(n)$ and transforming the resultant vector from the quefrency domain, the independent variable in this domain, to the time domain. These approaches however would result in modest inversion performance as only the minimum phase zeros are inverted; and as discussed above, a fraction of the magnitude response of maximum phase zeros can be inverted without giving rise to perceivable all-pass phase distortion [7].

This point may also be explained in the context of the briefly aforementioned in section 6.2 decomposition of a non-minimum phase into two filters; one containing its minimum phase zeros and the other containing its maximum phase zeros; recall that these filters were referred to originally as the minimum and maximum phase filters, respectively, with $g_{mp}(n)$ being referred to as the effective minimum phase filter. In this context, the inverse of the effective minimum phase filter ($\hat{g}_{mp}(n)$) can be described as equalizing the magnitude response of both the minimum and maximum phase filters and the phase response of minimum phase filter and the phase response of the minimum phase equivalent of the maximum phase filter, but not equalizing the remaining excess phase distortion of the maximum phase filter. It may also be described, perhaps more clearly, that this disruption of the magnitude and phase response of the maximum phase filter is avoided by either inverting only the minimum phase filter, or inverting both the minimum phase and maximum phase filters i.e. full RIR inversion.

From the preceding description, recent RIR inversion schemes are robust to all-pass phase distortion because they avoid fully compensating for the dips in the magnitude

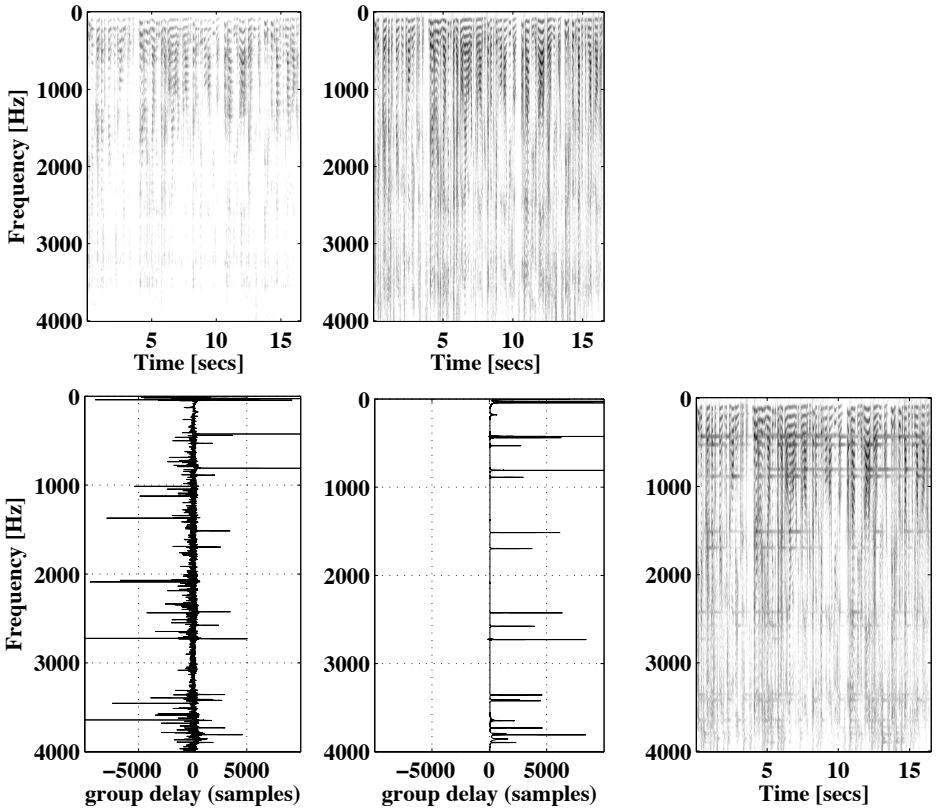

Figure 6.5 : Top, from left: Time-frequency responses of reverberated, $v(n)$, and clean speech signal $u(n)$; RIR from Figure 6.2 was employed to reverberate clean speech signal. Bottom from left: Group delay functions of RIR and its all-pass sequence, and the time-frequency response of minimum phase inverted speech, $v_{ap}(n)$, which exhibits all-pass phase distortion at the frequencies of spikes in all-pass delay function.

responses of $g(n)/g_{mp}(n)$ attributable to the high-Q zeros (both minimum and maximum phase). This implies that the magnitude response peaks are reduced in the modified inverses, which mitigates the exposure of the all-pass phase distortion in $v(n)$. As the preceding explanation also shows however, perceivable all-pass phase distortion arises at the frequencies of high-Q maximum phase zeros (assuming their phase is uncompensated) and not at the frequencies of the high-Q minimum phase zeros. It follows therefore, that all-pass phase distortion can be more precisely mitigated by not fully removing the magnitude distortion attributabile to the maximum phase high-Q zeros of an RIR. This approach would allow the magnitude response of the remaining zeros, including all minimum phase zeros, to be fully inverted allowing for less magnitude distortion, albeit with a relative increase in the inverse filter length. This motivates the single microphone partial RIR inversion scheme to be presented in section 6.4.

All-pass phase distortion, and the direct path delayed speech of $v_{ap}(n)$ are readily discernable in the spectrogram of $v_{ap}(n)$, which is denoted by $|V_{ap}(f, k)|$, where f and k denote discrete frequency bin and frame index respectively. This is due to the temporal resolution of the spectrogram, which enables the sharply delayed spectral components of $u(n)$ in $v_{ap}(n)$, corresponding to the spikes in $\tau_{ap}(n)$, to be discerned from the direct path speech, provided the processing window is short relative to the delays indicated by the spikes of $\tau_{ap}(n)$; this property was first shown in [393]. To illustrate this, Figure 6.4 displays the spectrogram (64

ms hanning window, overlap 32 ms) of a $v_{ap}(n)$ signal, which was constructed by convolving a clean speech signal $u(n)$, taken from the TIMIT speech corpus and downsampled to 8 KHz, with the $g_{ap}(n)$ of Figure 6.5; for illustrative purposes, the spectrograms of the corresponding $u(n)$ and $v(n)$ signals, and $\tau(n)$ and $\tau_{ap}(n)$ functions are also displayed, note that the frequency resolution of $\tau(n)$ and $\tau_{ap}(n)$ is 256 times higher than that of the spectrogram. Examining Figure 6.5, it is apparent that the all-pass phase distortion manifests as prominent tones in $|V_{ap}(f, k)|$ occurring at the frequencies containing the effect of the maximum phase zeros of $G(z)$; such tones are absent at all other the frequencies, including those that contain the effect of high-Q minimum phase zeros.

As is evident from Figure 6.5, the spectrogram representation of $v_{ap}(n)$ matches with the earlier subjective appraisal of $v_{ap}(n)$, where we opined that the all-pass phase distortion and the direct path delayed speech appear as two distinct components in $v_{ap}(n)$, and the direct path delayed speech was derevberated. This motivated our second approach to all-pass distortion mitigation, for which $|V_{ap}(f, k)|$ is considered to contain two components, namely, a distortion component, consisting of the variously delayed spectral features of $u(n)$, and a speech component, corresponding to the direct path delayed spectral features of $u(n)$. Within this interpretation of $|V_{ap}(f, k)|$, all-pass phase distortion is mitigated by applying a suitable speech enhancement algorithm to remove or suppress the distortion while preserving the speech component, thereby treating the distortion as an interference source. Conventional spectrogram based speech enhancement techniques that employ framewise stationary or slowly time-varying models of the interference source are not suited to this task due to the inherent non-stationarity of the all-pass phase distortion component in the magnitude STFT domain. NMF in a MSSS setting on the other hand has been successfully applied to speech enhancement for non-stationary interference sources; moreover, NMF has proven especially suited to processing music signals for applications such as separation [212] and automatic note transcription [197], and therefore, is amenable to processing non-stationary tonal features such as all-pass phase distortion. Therefore, given these attributes, and since the RIR is available a priori to generate training data, we applied NMF in a MSSS setting to the problem of all-pass phase distortion suppression; this approach is described in section 6.5.

6.4 A Partial Non-Minimum Phase Room Impulse Response Inversion Technique

In this section, we describe a novel partial inversion technique for single microphone non-minimum phase RIRs. This technique is based on minimum phase/all-pass decomposition of $g(n)$ using the homomorphic approach, and seeks to invert the minimum phase filter $g_{mp}(n)$ without producing perceptually detrimental all-pass phase distortion in the processed speech signal. As motivated in the previous section, to achieve such an inversion, this technique avoids fully inverting the magnitude response of the high-Q minimum phase zeros of $G_{mp}(z)$ located at the recpiricol radii of the high-Q maximum phase zeros of $G(z)$. By this way the

most perceptual detrimental all-pass phase distortion is suppressed in the resulting processed speech, and the remaining zeros of $g_{mp}(n)$ are fully inverted.

For this approach, we operate on the minimum phase inverse $\overline{g}_{mp}(n)$ rather than on $g_{mp}(n)$; similar to the selective pole replacement approach presented in [408], which was outlined in section 6.2. As such, we seek to identify and replace high Q-poles in $\overline{G}_{mp}(z)$ that correspond to the high-Q minimum phase zeros of $G_{mp}(z)$ that in turn correspond to the high-Q maximum phase zeros of $G(z)$. To identify the frequencies and radii of such poles, we make use of the all-pass group delay function $\tau_{ap}(k)$. As discussed in section 6.3, each maximum phase zero of $G(z)$ manifests as a sharp and discernable peak or spike in $\tau_{ap}(k)$. The location in frequency of each identified spike is deemed to correspond to the frequency of a desired pole in $\overline{G}_{mp}(z)$, with perceptually insignificant spikes below a prescribed minimum spike threshold, τ_{apmin}, in samples, ignored.

Another useful property of the all-pass group delay function, $\tau_{ap}(k)$, is that the radius of a pole in $\overline{G}_{mp}(z)$ that corresponds to a spike in $\tau_{ap}(k)$, can be determined using the following expression [429],

$$a_p = \frac{\tau_{ap}(k_p) - 1}{\tau_{ap}(k_p) + 1}, \tag{6.21}$$

where a_p and k_p denote the radius and frequency of the p^{th} indentified pole, with P corresponding to the number of poles, or equivalently the number of group delay spikes, that are identified. This expression enables the radius of each pole to be determined from $\tau_{ap}(k)$, without resorting to Q-factor estimation or temporal decay rate estimation in the time-frequency domain, both of which are prone to error at high frequencies [421, 430]. The utility of the group delay function $\tau(k)$ for RIR pole estimation in general is described in [401].

To proceed, we wish to replace each identified pole with a new pole at the same frequency but with a reduced radius, i.e. replace the p^{th} pole (a_p, k_p) with $(\hat{a}_p, k_p)$, where $\hat{a}_p < a_p$. We perform this by using a selective filter, as employed in [431] and [408], which we specify for the p^{th} identified pole (or complex pole pair given that $g_{mp}(n)$ is real), as,

$$G_s^{(p)}(z) = \alpha_p \frac{(1 - |a_p| e^{jk_p} z^{-1})(1 - |a_p| e^{jk_p} z^{-1})}{(1 - |\hat{a}_p| e^{jk_p} z^{-1})(1 - |\hat{a}_p| e^{jk_p} z^{-1})}, \tag{6.22}$$

where α_p is the gain of the p^{th} selective filter, $G_s^{(p)}(z)$, which is also known variously as a bi-quadratic filter, or a 2^{nd} order IIR notch filter. By applying $G_s^{(p)}(z)$ to $\overline{G}_{mp}(z)$, the p^{th} identified pole in $\overline{G}_{mp}(z)$ is cancelled by the numerator of $G_s^{(p)}(z)$, and is replaced by the p^{th} replacement pole in the denominator of $G_s^{(p)}(z)$; likewise for the complex conjugate pole.

The radius of the p^{th} replacement pole influences the depth of the notch, centered at k_p, in the magnitude response of the p^{th} selective filter $G_s^{(p)}(z)$. For $\hat{a}_p = a_p$, $G_s^{(p)}(z)$ has a flat magnitude response, and thus it imparts no effect on the magnitude response of $\overline{G}_{mp}(z)$, such that $|G_{eq}(k)| = 1 \ \forall k$, and all-pass phase distortion is fully exposed. For $\hat{a}_p < a_p$ the radius of the pole is reduced, implying a notch or dip centered on $k = k_p$ in $G_s^{(p)}(z)$, which when applied to

$\overline{G}_{\text{mp}}(z)$ implies that the magnitude response at $G(k_p)$ is reduced, which in turn implies an uncompensated dip (magnitude distortion) centered on $|G_{\text{eq}}(k_p)|$. By reducing the magnitude response at this frequency, the exposure of all-pass phase distortion in the processed speech at is mitigated; $\hat{a}_p < a_p$ also increases the decay rate of the geometric sequence in the inverse impulse response corresponding to the pole at (a_p, k_p).

To enable the trade-off between the magnitude distortion and all-pass phase distortion corresponding to the identified poles to be controlled using a single parameter, the replacement pole radii were set according to,

$$\hat{a}_p = a_p - \gamma, \quad \forall p \tag{6.23}$$

where γ is a user prescribed replacement pole radius offset. The offset γ controls the trade-off between magnitude distortion and all-pass phase distortion of the identified maximum phase zeros of $g(n)$, with γ being proptioal ito magnitude distortion and inversely proportional to all-pass phase distortion. The offset γ is also independent of a_p, which means that the relative properties of the P selective filters depends on their corresponding identified poles, with poles closer to the unit circle receiving a replacement filter with a sharper notch (lower gain at k_p and narrower bandwidth), while poles further away from the unit circle receiving a replacement filter with a wider bandwidth and higher gain.

The gain of each selective filter, α_p, is used to ensure that the gain at the Nyquist frequency of each selective filter is equal to 1; α_p is calculated as [432],

$$\alpha_p = \frac{1 + a_p a_p}{1 + \hat{a}_p \hat{a}_p}. \tag{6.24}$$

The P selective filters may then be cascaded and applied to $\overline{G}_{\text{mp}}(z)$ as follows,

$$\hat{\overline{G}}_{mp}(z) = \overline{G}_{mp}(\text{z}) G_s^{(1)}(z) \ldots G_s^{(P)}(z), \tag{6.25}$$

where $\hat{\overline{G}}_{\text{mp}}(z)$ is the resultant modified minimum phase inverse filter, or in the discrete case,

$$\hat{\overline{G}}_{mp}(k) = \overline{G}_{mp}(k) G_s^{(1)}(k) \ldots G_s^{(P)}(k), \tag{6.26}$$

This may also be described as a series of convolutions in the time domain,

$$\hat{\overline{g}}_{mp}(n) = \overline{g}_{mp}(n) * g_s^{(1)}(n) \cdots * g_s^{(P)}(n), \tag{6.27}$$

where $g_s^{(P)}(n)$ and $\hat{\overline{g}}_{mp}(n)$ are the impulse responses of $G_s^{(p)}(k)$ and $\hat{\overline{G}}_{mp}(k)$ respectively, obtained using the IFFT.

The implementation of this algorithm can be stated as follows:

1. Perform steps 1-8 of the homomorphic approach to decompose $g(n)$ into $g_{\text{mp}}(n)$ and $g_{ap}(n)$ and compute $\overline{g}_{\text{mp}}(n)$, see section 6.2.
2. Compute $\tau_{\text{ap}}(k)$ from $g_{\text{ap}}(n)$ using the formulas in (6.18)(6.19)
3. Identify the frequencies in $\tau_{\text{ap}}(k)$ that contain spikes using a standard peak finding algorithm. (Due to the spikedness of $\tau_{\text{ap}}(k)$ a basic peak-picking algorithm suffices, for

this work, we specified, τ_{apmin}, in samples, after which the frequencies of local maxima above τ_{apmin} are identified.) Each identified frequency is deemed to correspond to the frequency, k_p, of a desired high-Q pole in $\overline{G}_{\text{mp}}(k)$, with the number of poles identified denoted as P.

4. Determine the pole radius, a_p, for each of the P identified poles using (6.21),
5. Determine the replacement pole radii, $\hat{a}_p$, for each of the P identified poles using (6.23),
6. Determine the gain α_p for each selective filter using (6.24),
7. Having obtained a_p, $\hat{a}_p$, k_p and α_p for P high-Q poles in $\overline{G}_{\text{mp}}(z)$, we now adopt an iterative approach to selective filtering,
8. $\hat{\overline{G}}_{mp}(k) = \overline{G}_{mp}(k)$, Initialize the modified minimum phase inverse filter $\hat{\overline{G}}_{mp}(k)$.
9. For $p = 1: P$
    a. Using the parameters a_p, k_p, $\hat{a}_p$ and α_p, construct the selective filter, $G_s^{(p)}(k)$, for the p^{th} pole, as specified (6.22)
    b. Remove and replace p^{th} pole by applying $G_s^{(p)}(k)$ to $\hat{\overline{G}}_{mp}(k)$ as,

$$\hat{\overline{G}}_{mp}(k) = \hat{\overline{G}}_{mp}(k) G_s^{(p)}(k), \tag{6.28}$$

    c. $p = p + 1$;
10. Compute the modified time domain output inverse filter, $\hat{\overline{g}}_{mp}(n)$, by taking the IFFT of the resultant $\hat{\overline{G}}_{mp}(k)$.

The resulting modified filter $\hat{\overline{G}}_{mp}(k)$ contains notches at the frequencies of maximum phase zeros such that the phase distortion at the corresponding frequencies is not exposed after processing by $\hat{\overline{G}}_{mp}(k)$. This approach differs from existing approaches, which mitigate phase distortion by avoiding the inversion of high-Q zeros in general. The experimental evaluation of this technique is presented in section 6.6.

6.5 NMF based All-Pass Phase Distortion Suppression

In this section, we apply NMF in a MSSS setting to the all-pass phase distortion problem, an approach we call NMF All-Pass Distortion Suppression (NMF-APDS). Central to this approach is the assumption that $v_{\text{ap}}(n)$ consists of two components, namely, an interference component, the portion of $u(n)$ subject to phase distortion by $g_{\text{ap}}(n)$, and a target component, the portion of $x(n)$ subject to the direct path delay of $g_{\text{ap}}(n)$ only. To suppress the interference, the two components are segregated in the magnitude spectral domain using NMF, which exploits their distinct features in this domain, from which only the target speech component is resynthesized. Formally then, NMF-APDS assumes the following representation of the All-Pass Distortion problem,

$$|V_{ap}(f, k)| = |U_{tar}(f, k)| + |U_{interf}(f, k)|, \quad (6.29)$$

where $|U_{tar}(f, k)|$ and $|U_{interf}(f, k)|$ correspond to the decomposition of $|V_{ap}(f, k)|$ into a target and interference component respectively. Note that by this approach the components of $u(n)$ that are subject to phase distortion are considered interference and are therefore not recovered; as such, some distortion is inherent to this method.

To illustrate the aptness of the model in (6.29) for all-pass phase distortion suppression, the terms in (6.29) were estimated for the representative example $g_{ap}(n)$ in Figure 6.3, via the following procedure; note that for this example we employed the same speech signal $u(n)$ as for Figure 6.6 above. The $|U_{interf}(f, k)|$ component in Figure 6.6, was calculated by subtracting the spectrogram, $|U_t(f, k)|$, of the direct path delayed $u(n)$ signal i.e. $u(n - t_d)$, from $|V_{ap}(f, k)|$, by which the target component in $|V_{ap}(f, k)|$ (direct path delayed speech) is suppressed leaving the interference component (all-pass phase distortion artifacts); any negative components in the resulting spectrogram were assigned to zero, this may occur due to over subtraction at the frequencies containing the effects of phase distortion. This operation may be expressed as, $|U_{interf}(f, k)| = \max(0, |V_{ap}(f, k)| - |U_t(f, k)|)$. The $|U_{tar}(f, k)|$ component in Figure 6.6 was obtained by subtracting the resultant $|U_{interf}(f, k)|$ from $|V_{ap}(f k)|$ i.e. $\max(0, |V_{ap}(f, k)| - |U_{interf}(f, k)|)$, an operation which suppresses the phase distortion captured by $|U_{interf}(f, k)|$ from $|V_{ap}(f, k)|$ to leave the target component.

As is demonstrated by Figure 6.6, this procedure, as expected, neatly partitions $|V_{ap}(f, k)|$ into a target (direct path delayed speech) and interference (all-pass phase artifacts) component, serving to demonstrate the aptness of the model in (6.29); the Log Spectral Distance, defined in Chapter 4, between $|U_{interf}(f, k)|$ and $|U_t(f, k)|$ of Figure was 2.12 dB. To confirm that the resulting partition of the interference and target fits with the subjective experience of these as two distinct mixed components in $v_{ap}(n)$, a time domain signal was synthesized from both $|U_{interf}(f, k)|$ and $|U_{interf}(f, k)|$ using the phase angles of $|v_{ap}(f, k)|$. Each of the resulting signals were found to contain a neatly partitioned component, that of the described speech and bell chime distortion respectively, of $v_{ap}(n)$; this was repeated for numerous examples with similar results.

To formulate the NMF-APDS algorithm, the column vector $\mathbf{v}_{ap}(k)$ is defined to contain the $N/2+1$ unique values of $|V_{ap}(f, k)|$, which is symmetric about $k = N/2$, such that $\mathbf{v}_{ap}(k) = \mathbf{u}_{tar}(k) + \mathbf{u}_{interf}(k)$. The aim of NMF-APDS is to suppress $\mathbf{u}_{interf}(k)$ in a framewise manner, or equivalently, to extract $\mathbf{u}_{tar}(k)$ in a framewise manner.

To begin, it is necessary to train a non-negative basis for each component a priori, each of which will contain spectral features that compactly characterize their respective source. We define a basis for $\mathbf{u}_{tar}(n)$, $\mathbf{B}_{tar}$, of dimensions $(N/2) + 1 \times R_{tar}$, and a basis for $\mathbf{u}_{interf}(k)$, $\mathbf{B}_{interf}$, of dimensions $(N/2) + 1 \times R_{interf}$. A speaker independent $\mathbf{B}_{tar}$ was trained per the description given in Chapters 5 and 6. To train $\mathbf{B}_{interf}$ it is necessary to obtain a magnitude spectrogram consisting exclusively of representative all-pass phase distortion artifacts for the room response $g(n)$; $\mathbf{B}_{interf}$ is therefore RIR dependent. To attain such data, the procedure described

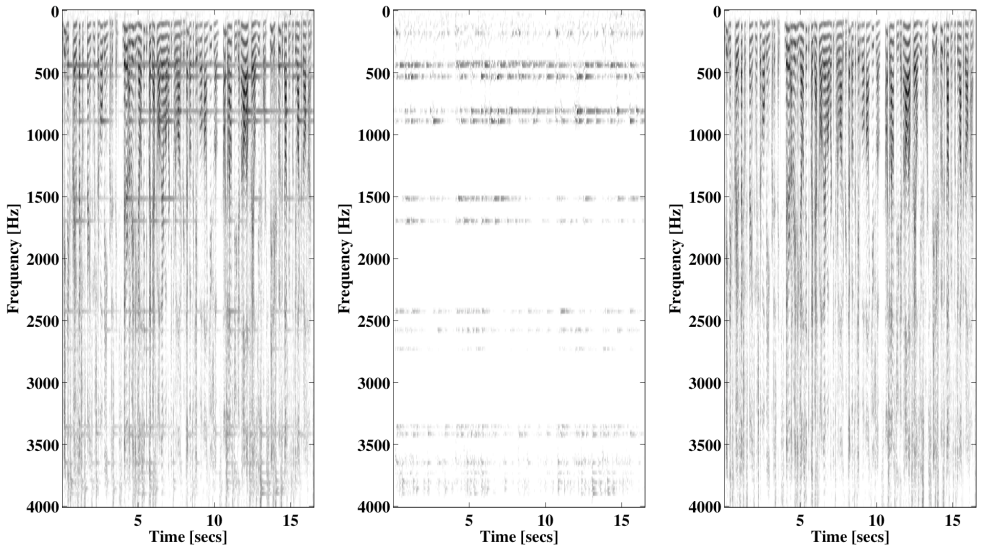

Figure 6.6 :Left: Example $|V_{ap}(f, k)|$, also displayed in Figure 6.5. Center, example $|U_{interf}(f, k)|$, and right, example $|U_{tar}(f, k)|$.

above to create the example $|U_{interf}(f, k)|$ in Figure 6.5 was employed. The data was generated from the $g_{ap}(n)$ derived from $g(n)$ using the homomorphic approach, and using the same clean speech signal (non-reverberated) used to train $\mathbf{B}_{target}$, with a rank R_{interf} applied to the resulting spectrogram to train $\mathbf{B}_{interf}$. The composite basis $\mathbf{B}$ contains the union of the two trained basis i.e. $\mathbf{B} = [\mathbf{B}_{tar}\ \mathbf{B}_{interf}]$, and remains static throughout the suppression procedure.

To obtain a separation of $\mathbf{u}_{tar}(k)$ and $\mathbf{u}_{interf}(k)$, $\mathbf{v}_{ap}(k)$ is decomposed onto the composite basis $\mathbf{B}$ by a restricted NMF procedure, (described in chapter 4) during which $\mathbf{B}$ is fixed and only the updates for the gain component, $\mathbf{g}(k)$, which is a $R_{tar} + R_{interf}$ vector initialized with non-negative random numbers for each k, is updated. The restricted NMF procedure yields the following factorization of $\mathbf{v}_{ap}(k)$,

$$\mathbf{v}_{ap}(k) = \mathbf{B}\mathbf{g}(k) + \mathbf{e}(k) = [\mathbf{B}_{tar}\ \mathbf{B}_{interf}]\begin{bmatrix}\mathbf{g}_{tar}(k)\\ \mathbf{g}_{interf}(k)\end{bmatrix} + \mathbf{e}(k), \tag{6.30}$$

where $\mathbf{e}(k)$ denotes the residual error, and where $\mathbf{g}_{tar}(k)$ and $\mathbf{g}_{interf}(k)$ are the parts of the gain vector $\mathbf{g}(k)$ corresponding to $\mathbf{B}_{tar}$ and $\mathbf{B}_{interf}$ respectively. By holding $\mathbf{B}$ fixed during the restricted NMF procedure the characteristic spectral features of each source in $\mathbf{v}_{ap}(k)$ are expressed by the basis vectors of their respective bases. An estimate of each source can be obtained by parsing the factors in (6.30) to give,

$$\mathbf{v}_{ap}(k) = \mathbf{B}_{tar}\mathbf{g}_{tar}(k) + \mathbf{B}_{interf}\mathbf{g}_{interf}(k) + \mathbf{e}(k). \tag{6.31}$$

However, the compact nature of the source bases means that they are generally unable to model the entire variability that their sources may exhibit in any one frame, and so, in each frame the residual vector $\mathbf{e}(k)$ contains a significant amount of energy. This means that using the estimate of the magnitude of the target speech at this stage results in heavily distorted speech.

To obtain a final estimate of the speech, therefore, we apportion $\mathbf{v}_{ap}(k)$, as follows,

$$\hat{\mathbf{u}}_{\text{tar}}(k) = \mathbf{v}_{\text{ap}}(k)\frac{\mathbf{B}_{\text{tar}}\mathbf{g}_{\text{tar}}(k)}{\mathbf{B}(k)\mathbf{g}(k)}, \tag{6.32}$$

where $\hat{\mathbf{u}}_{\text{tar}}(k)$ is the NMF-APDS estimate of $\mathbf{u}_{\text{tar}}(k)$. By adopting this approach, some distortion is inevitable in the estimate (in addition to the distortion inherent to this approach) given the coarseness of the original magnitude estimates. The approach specified by (6.32) is widely used by MSSS algorithms in varying forms to complete separation. Note that the unrestricted NMF technique employed in Chapters 4 and 5 was found to be ineffective in this context.

By overlapping and adding successive time-domain frames, obtained by taking the IFFT of $\hat{\mathbf{u}}_{\text{tar}}(k)$ with the phase angles of $V_{\text{ap}}(f,k)$, the output speech signal, $\hat{u}_{\text{tar}}(n)$, is synthesized, with the phase distortion artifacts excised. This approach is experimentally evaluated in section 6.6.

6.6 Experimental Evaluation

This section evaluates the performance of the two single channel RIR inversion schemes outlined in section 6.4 and 6.5 respectively. Initially, this section examines some aspects of the parameters of both algorithms, afterwhich each algorithm is evaluated separately by way of a comparative listening test, in which samples of partially dereverberated speech signals from each scheme are compared to those of a competing algorithm in the field, such that the performance of the proposed algorithms is placed in context.

The evaluation was performed in the Matlab simulation environment, using pre-recorded RIRs from the MARDY database, downsampled to 8 kHz, and pre-recorded speech data from the TIMIT speech data, also downsampled to 8 KHz; a variety of speakers with an equal proportion of male and female speakers was used. To avoid circular convolution effects (time aliasing) it was necessary to set the length of the FFT, L_{ap}, used for homomorphic processing, to 2^{18}.

6.6.1 Parameters of NMF-APDS

The frame length N and stepsize m of NMF-APDS were set to 64 ms and 32 ms respectively, which was deemed short enough such that the de-sycronised spectral components attributable to phase distortion are discernable from the target speech, and such that the delay incurred by NMF-APDS is not prohibitive. As in previous chapters, the number of restricted NMF iterations per frame was set to 60.

The speaker independent $\mathbf{B}_{\text{tar}}$ was trained as per previous chapters, and the RIR dependent $\mathbf{B}_{\text{inter}}$ was trained as described in section 6.5. Per previous chapters, because of cross-matching error the choice of R_{tar} and R_{inter} is a trade-off; in this case between phase distortion, i.e. mitigation of audible bell chime artifacts, and distortion of $\hat{u}_{\text{tar}}(n)$, or magnitude distortion. Through informal listening, for which we varied $\mathbf{B}_{\text{tar}}$ and $\mathbf{B}_{\text{interf}}$ and subjectively assessed the output speech, we arrived at a value of 20 for both R_{tar} and R_{inter}.

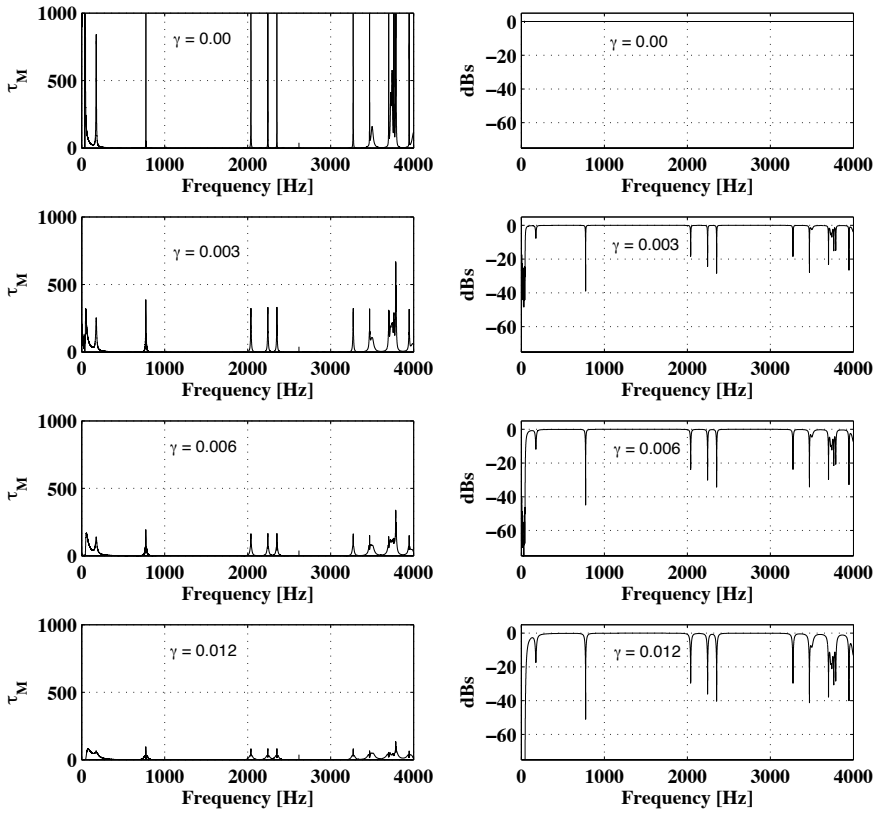

Figure 6.7: Left Column: Modified group delay functions, $\tau_M(k)$, defined in (6.20), for $\hat{\hat{g}}_{mp}(n)$ for σ= 0, (top), σ= 0.003, σ = 0.006, σ=0.0012 (bottom). Right Column: Magnitude equalized responses, i.e. $|G_{eq}(k)|$, corresponding to σ= 0, (top), σ= 0.003, σ = 0.006, σ=0.0012.

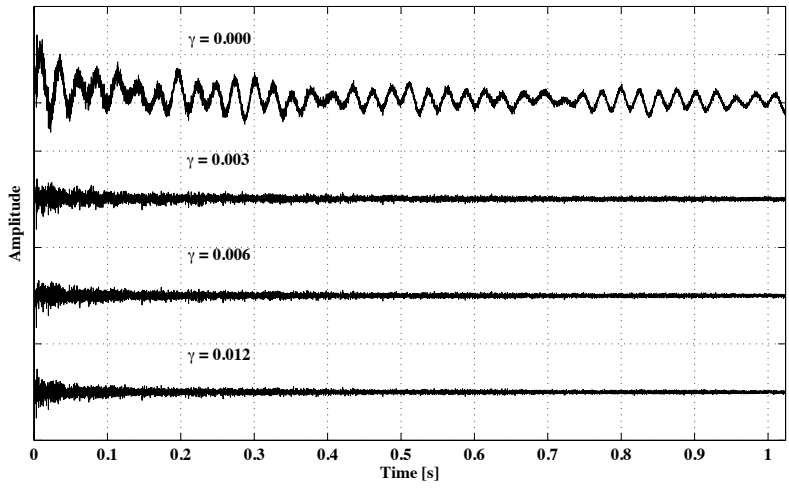

Figure 6.8 : Impulse responses of modified filter, $\hat{\hat{g}}_{mp}(n)$, for a range of values of γ.

6.6.2 Parameters of the Partial inversion scheme

The two parameters of the partial dereverberation algorithm described in section 6.4 are the minimum group delay threshold for the peak finding algorithm, τ_{apmin}, and the pole radius offset, γ. The integration time of the ear (between 30-200 ms for different frequencies [7]) was proposed as a threshold for the perception of phase distortion in [393] and [7]. We incorporate this knowledge by setting τ_{apmin} to 150 samples (approx. 20 ms at 8 kHz), such that

perceptually insignificant poles are not selectively filtered; this threshold also prevents spurious low level peaks in $\tau_{ap}(k)$ from being erroneously identified as poles by the peak finding algorithm.

As described in section 6.4 the proposed partial inversion approach implies a trade-off between the magnitude distortion of the maximum phase zeros and all-pass phase distortion. This trade-off was anticipated by the modified group delay function defined in (6.20), which can be used as an objective measure of phase distortion, and thus, to gauge the influence of the offset γ. To this end, Figure 6.7 displays a modified group delay function, $\tau_M(k)$, for each modified inverse filter, $\hat{\tilde{g}}_{mp}(n)$, produced for various values of γ, accompanied by the corresponding equalized magnitude response. As is evident from in Figure 6.7, as γ is increased the spikes of $\tau_M(k)$ are reduced, owing to lower notch gains at the corresponding frequencies in the corresponding $|G_{eq}(k)|$ (evident from the $|G_{eq}(k)|$ plots in Figure 6.7), which implies less exposure of phase distortion in the processed signals. However, the lower gains at the selective filter notches will increase the magnitude distortion of the processed speech for the same frequencies.

To subjectively appraise the influence of the replacement pole radii, we varied the offset γ and assessed the resulting speech. As expected, as γ is set closer to 0 (greater gain at notch frequencies of the selective filters) the artifacts associated with all-pass distortion become more prominent, with the artifacts first becoming noticeable for $\gamma = 0.01$ (approx); magnitude distortion however was noticeable decreasing. Conversely, for decreasing γ (gain at notch decreasing), informal subjective listening tests revealed that less phase artifacts and more magnitude distortion in the processed speech. For the listening test, we deemed it appropriate to sacrifice some magnitude distortion such that the phase artifacts are almost completely removed, we therefore set $\gamma = 0.012$.

The parameter γ also affects the required length of the modified inverse filter $\hat{\tilde{g}}_{mp}(n)$. Figure displays $\hat{\tilde{g}}_{mp}(n)$ for a range of values of γ; zoomed in to the origin for emphasis. It can be seen from Figure 6.8, that as γ increases, reducing the radius of the high-Q maximum phase poles, the impulse response of $\hat{\tilde{g}}_{mp}(n)$, accordingly, decays more quickly owing to the commensurate increase in the decay rate of the poles corresponding geometric sequences. This effect is most noticeable between $\gamma = 0$ and $\gamma = 0.003$, for which there is a dramatic decrease in the influence of low frequency poles on the impulse response. In practice, this means that for higher values of γ the modified inverse filter can be truncated sooner with less adverse effects to allow a shorter inverse filter length.

6.6.3 Comparative algorithm

The comparative algorithm employed in both comparative listening tests was the homomorphic approach outlined above for $\varphi = 2$ [409]. This algorithm was chosen for its simplicity and because like both proposed algorithms, it is based on homomorphic processing and seeks to partially compensate for the magnitude response of a time-invariant RIR without introducing all-pass phase distortion. As described in (6.17) the parameter φ controls the

	Left	Centre	Right
1 Metre	ir_1_L_4.wav	ir_1_C_4.wav	ir_1_R_4.wav
2 Metre	ir_2_L_4.wav	ir_2_C_4.wav	ir_2_R_4.wav
3 Metre	ir_3_L_4.wav	ir_3_C_4.wav	ir_3_R_4.wav

Table 6.2 Selected MARDY RIRs. Top row corresponds to loudspeaker location. Leftmost column corresponds to distance from loudspeaker.

proportion of the magnitude response of $g(n)$ that is inverted, and as for γ and the choice of R_{interf} and R_{tar}, is a trade-off between RIR magnitude and phase distortion. For this comparison $\varphi = 2$ was chosen for each RIR, which is close to the optimum value of this parameter found in [7], and in [408], making our work comparable with these works; we also informally verified the optimality of this choice. Below, we will refer to the partially dereverberated speech signal produced by this algorithm as the comparative processed speech.

6.6.4 Listening Test procedure

Each listening test comprised of nine separate comparisons, with each comparison corresponding to a different RIR from the MARDY database; a different TIMIT speaker was also used for each comparison, both male and female where used. A broad sampling of MARDY RIRs where employed from various room positions; the MARDY IDs for these RIRs are tabulated in Table 6.2, in which rows indicate the distance of the microphone array from the loudspeaker, while the columns indicate the position of the loudspeaker relative to the array (4 at the end of each file name refers to the microphone on the array). All audio listening test files are available online (http://www.eeng.nuim.ie/~ncahill/). Each $v(n)$ signal was created by convolving a MARDY RIR with a TIMIT speech utterance, before being passed to each algorithm, along with the RIR.

A panel of sixteen subjects were recruited, fifteen males and one female, all with normal hearing, and with an average age of 32. The test signals were presented through good quality consumer headphones at a comfortable level. We adopted a similar listening test procedure to that outlined in [7], which was also employed in [408]. For each comparison, each subject was first presented with the original clean speech signal (non-reverberated), and then presented with the processed or partially dereverberated speech signals from the comparative algorithm and the test algorithm; the order of these two signals was randomized to avoid bias. The subjects were instructed to evaluate the quality of the two output speech signals, using the original clean signal as a reference, and to quantify their opinion according to the following ratings,

8 – Very Good
7 –
6 – Good
5 –
4 – Fair
3 –
2 – Poor
1 –

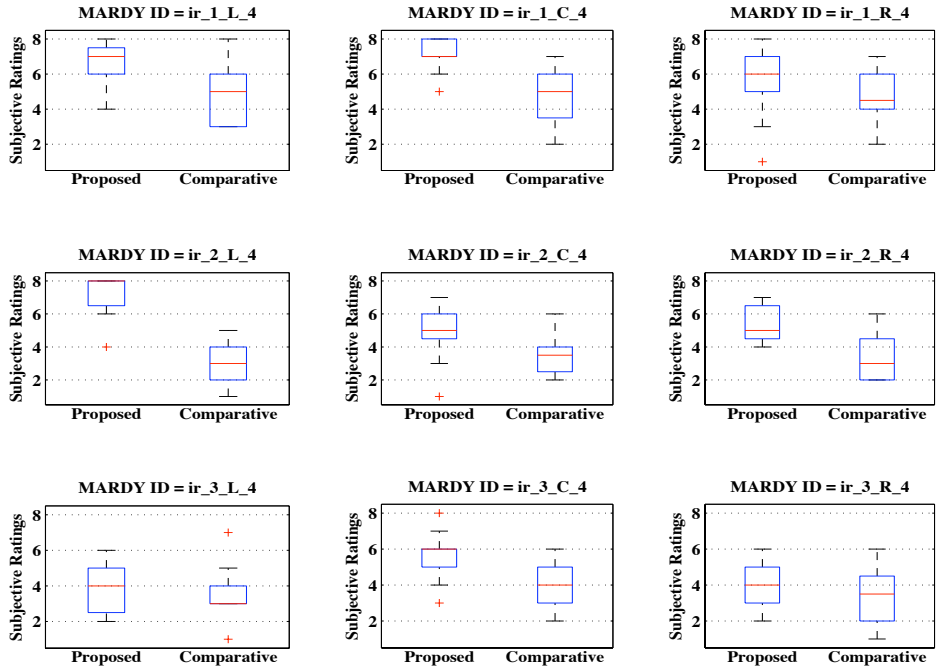

Figure 6.9 : Box plots of subjective ratings from listening test between the proposed partial inversion scheme and the comparative algorithm.

with 8 signifying a quality equivalent to the unprocessed clean speech. The ratings were then collated and displayed using box plots in Figure 6.9 for the partial minimum phase inversion and in Figure 6.12 for NMF-APDS. The results from all sixteen listeners are included in the aggregate ratings. A qualitative assessment of the presented signals was taken from each subject after each listening test was completed.

To compliment the listening test results and to objectively demonstrate all-pass phase distortion mitigation, three modified group delay functions where computed for each listening test, and are displayed in Figure 6.10. The first $\tau_M(k)$ corresponds to full inversion of the minimum phase inverse filter i.e. $\gamma = 0$, $\varphi = 1$, the second $\tau_M(k)$ corresponds to the inverse filter from the proposed minimum phase partial inversion approach, and the third $\tau_M(k)$ corresponds to the inverse filter of the comparative approach. The NMF-APDS algorithm is not suitable for this type of analysis.

6.6.5 Discussion of the Partial Minimum Phase Inversion Scheme Results

Examining the box plots of the subjective ratings in Figure 6.9, it is apparent that, on average, the listening panel rated the speech signals produced by the proposed algorithm as being better quality than that of the comparative algorithm for each RIR. From a qualitative perspective, the panel reported that the proposed algorithm's speech sounded consistently less distant, or less reverberated, than the comparative speech. The panel also reported that bell chimes or similar artifacts (phase distortion) were almost non-existent or where few and faint in the proposed algorithms output speech, while such artifacts were noticeable in the comparative speech output. The reported absence of all-pass phase distortion in the proposed algorithms processed speech is congruent with the modified group delay functions in Figure 6.10, where it is apparent from the complete absence of peaks in the proposed algorithms

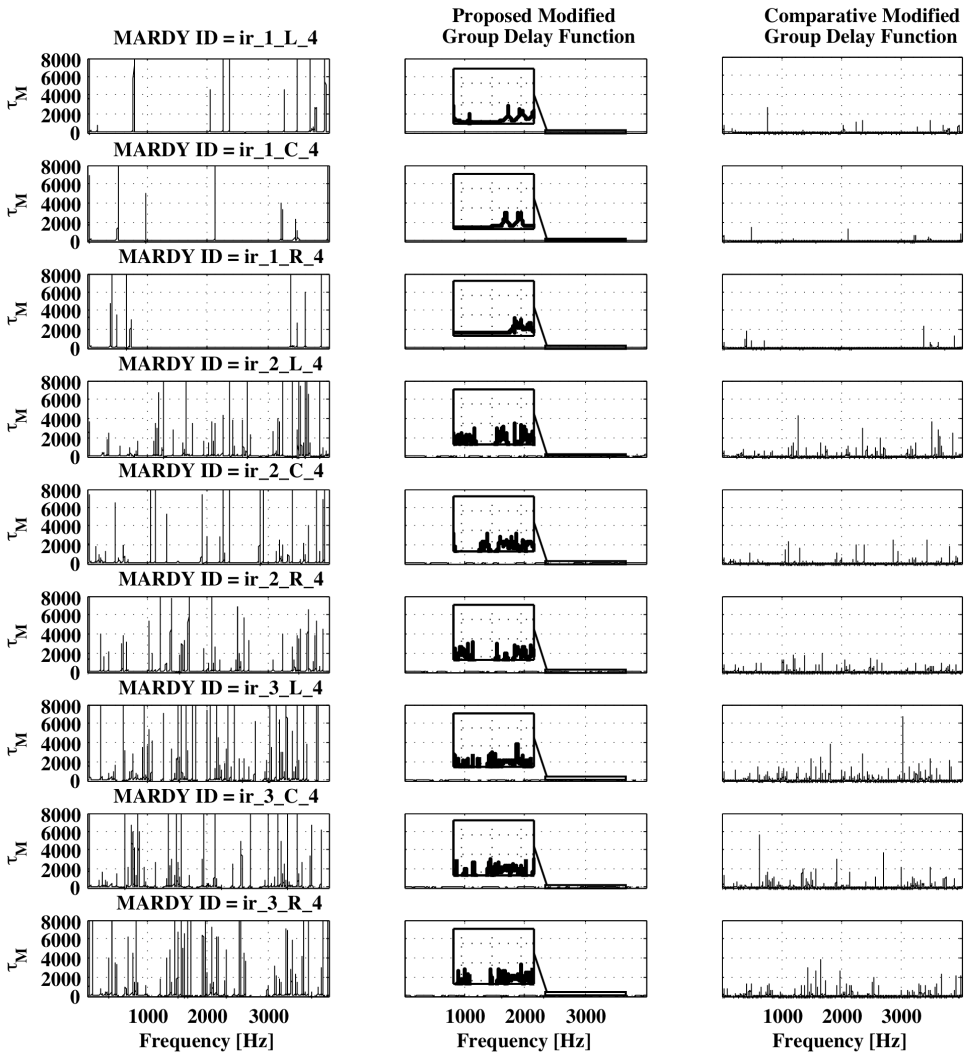

Figure 6.10: Left Column: Modified Group delay functions ($\tau_M(k)$) for the all-pass filters of the MARDY RIRs. Center, Modified Group delay functions for the modified inverse of the proposed approach. Right column, Modified Group delay functions of the comparative algorithms inverse filter.

modified group delay functions that it successfully mitigates the introduction of all-pass phase distortion into its processed speech. The opinions of the panel suggest that the quality of the processed speech from the proposed inversion scheme is subjectively better than that of the comparative scheme, and that this improvement is achieved without introducing phase distortion into the proposed speech, with both algorithms registering their best performance for 1 meter MARDY RIRs (i.e. all MARDY RIRs recorded one meter from the loudspeaker), for which the proposed scheme attained an average rating between 6 (good) and 8 (very good), and the comparative algorithm was rated on average between 4 (fair) and 6 (good).

For RIRs recorded at microphones further away from the loudspeaker, it is evident from Figure 6.9 that the listening panel perceived a worsening in the quality of the processed speech from both inversion algorithms. On this result, the panelists commented that relative to the original clean speech signal, the processed speech generally sounded more echoic and distant for RIRs further away from the loudspeaker, with a distinct increase in the number of perceivable tones or chimes in the comparative output speech. A number of subjects also noticed that for the 2 and 3 meter RIRs, the clarity or intelligibility of the proposed

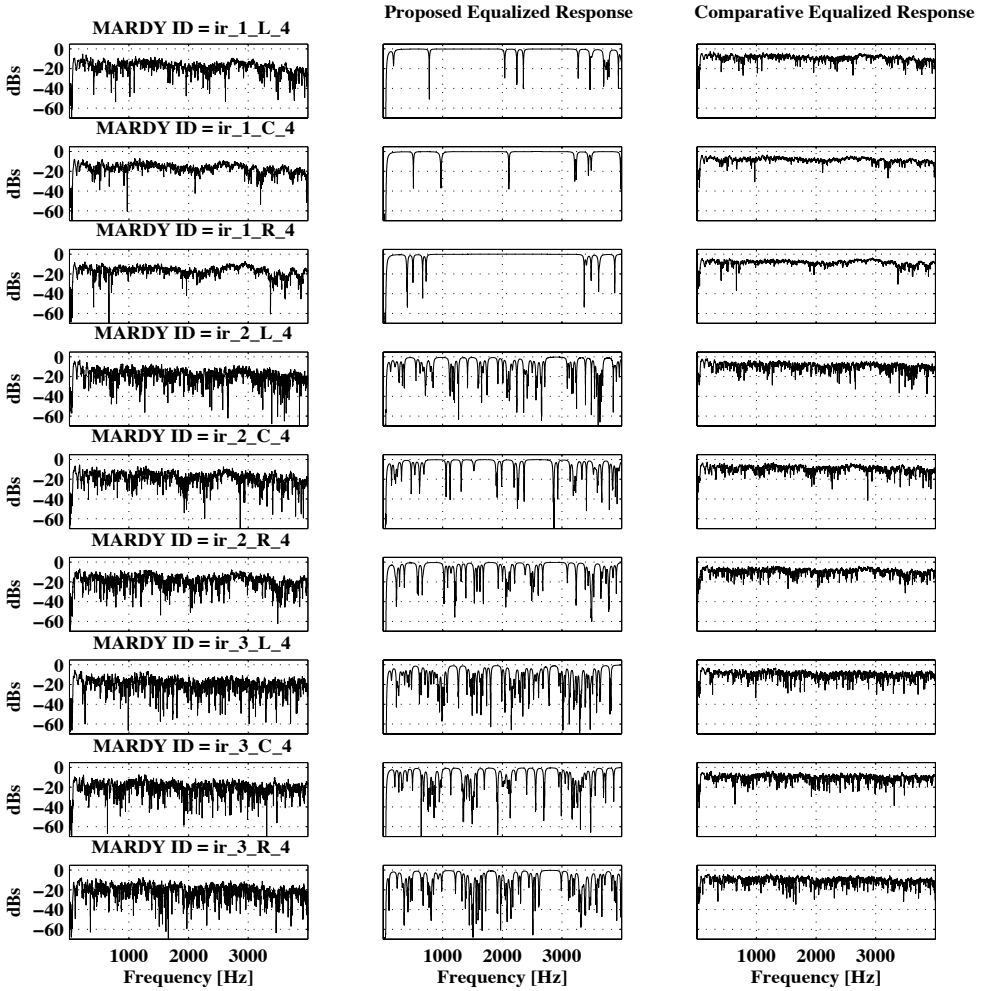

Figure 6.11 : Magnitude response of MARDY RIRs, i.e. $|G(k)|$, left column. Equalized Responses of proposed approach ie. $|G_{eq}(k)|$, centre column, and Equalized responses of comparative algorithm, right column $|G_{eq}(k)|$.

algorithm's output speech became somewhat impaired relative to both the original clean speech signal and the comparative speech signal; the proposed algorithms speech was still rated higher however than the comparative algorithm for these RIRs. Given that L_{ap} was fixed for each RIR, a contributory factor to the general deterioration in quality of both algorithms is that RIRs recorded further away from the loudspeaker typically contain a higher number of high-Q zeros (both minimum and maximum phase), and as such, require inverse filters with longer length to achieve equivalent inversion performance. The reported reduction of speech intelligibility is also ascribable to magnitude distortion, which becomes perceptually noticeable for such RIRs because of the greater concentration of maximum phase high Q-zeros in the transfer functions of these RIRs; the concentration of maximum phase high Q-zeros for such RIRs is apparent from Figure 6.10. This is evident from Figure 6.11, where the magnitude responses of both the proposed and comparison algorithms are displayed. Examining Figure 6.11, it can be seen that the equalized magnitude responses of the proposed approach for RIRs recorded progressively further away from the loudspeaker bear the imprint of a greater number of selective notch filters, each required to mitigate all-pass phase distortion. For the 2 and 3 meter RIRs in particular, the equalized magnitude responses exhibit

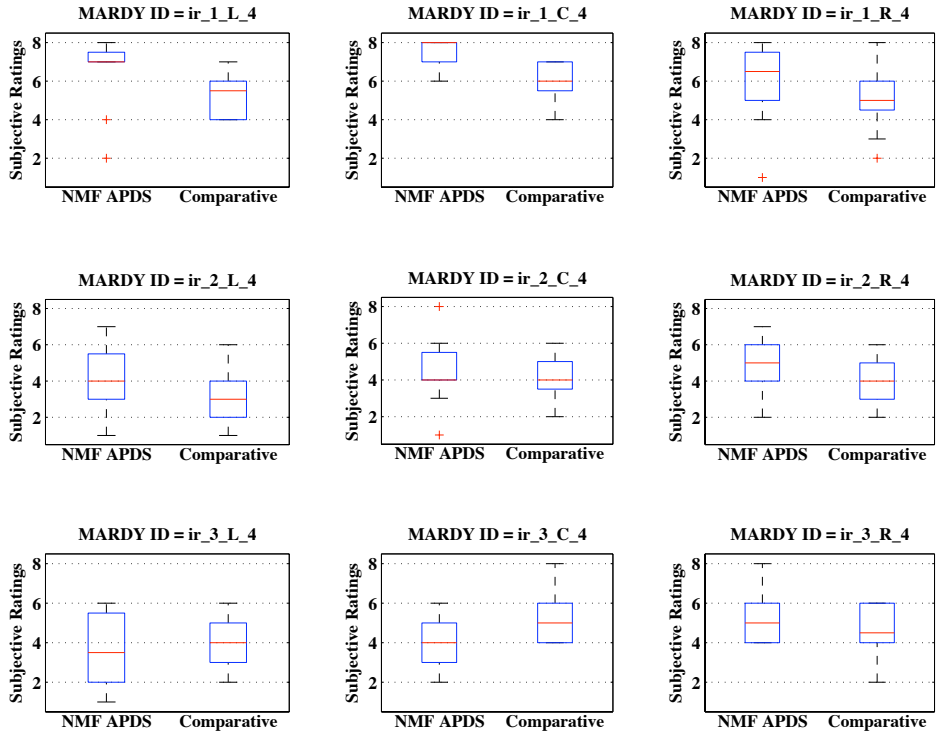

Figure 6.12: Box plots of subjective ratings from listening test between NMF APDS and the comparative algorithm.

a significant number of notches, indicating that the magnitude spectrum of the processed speech is imparted with an uneven spectrum profile for such RIRs, serving to reduce the intelligibility of the processed speech as described. Moreover, it is evident from Figure 6.11, that in some cases the selective filters introduce some additional distortion into the processed speech by having notches of lower gain than the original RIR. This is most likely attributable to the use of a single offset γ to tune the radii of all the replacement poles. In contrast to the proposed approach, the comparative algorithm distributes the magnitude distortion more evenly across frequency such that intelligibility is not overly affected, but on other hand, this approach does not precisely mitigate the all-pass phase distortion.

6.6.6 Discussion of NMF-APDS Results

Examining the box plots of the subjective ratings in Figure 6.12, the NMF-APDS output speech received higher ratings, on average, than the comparative output speech for all 1 meter MARDY RIRs (RIR recorded 1 meter from the loudspeaker). For these RIRs, NMF-APDS received an average subjective rating between 6 and 8 implying that the panel contended that the output speech was good to very good, with the comparative algorithm again obtaining an average rating between 6 and 4, implying good to fair output. The listeners reported that the NMF-APDS output speech sounded spatially closer and much less reverberated than the reference; similar to the test algorithm in the previous listening test. Unlike the previous listening test however, the listeners noticed some artifacts, attributable to residual phase distortion, in NMF-APDS output speech. While most listeners found this residual phase distortion not to be distracting, two listeners contended that it was somewhat distracting. On the distortion inherent to this algorithm, the listeners did not perceive distortion of the target

speech component relative to the original clean speech signal, suggesting that this distortion may be masked or insignificant for the 1 meter RIRs. In general, for 1 meter RIRs the opinions of the panel may be summarized as: NMF-APDS enables RIR magnitude equalization by greatly suppressing, though not removing entirely, all-pass phase distortion.

Like the previous listening test, it is evident from Figure 6.12 that for RIRs recorded at microphones further away from the loudspeaker, the panel perceived a progressive deterioration in the quality of the processed speech from both NMF-APDS and the reference inversion algorithms. And as in the previous test, a contributory factor to this result is the relative difficulty of inverting RIRs measured further away from a loudspeaker. Contrasting the results of the two algorithms however, it is apparent NMF-APDS ratings deteriorate further for 2 and further again for 3 meter RIRs than those of the comparative algorithm, which attains comparable to better average ratings than NMF-APDS for the 3 meter RIRs. Broadly speaking, the listening panel reported that while the audible distortions increased for both algorithms the introduced artifacts in the NMF-APDS processed speech became considerable more distracting. The level of residual phase distortion remaining in the NMF-APDS output speech is related to the ability of $\mathbf{B}_{\text{interf}}$ to account for the phase distortion component, $|U_{\text{target}}(k,t)|$, of $|V_{\text{ap}}(k,t)|$ during the restricted NMF procedure. A larger number of maximum phase zeros in a RIR means that the corresponding $|U_{\text{target}}(k,t)|$ component can exhibit a wider variety of spectral patterns, meaning $\mathbf{B}_{\text{interf}}$ is required to express more spectral patterns to achieve equivalent performance. Since the number of basis vectors in R_{interf} was fixed at 20 for each RIR, the task of $\mathbf{B}_{\text{interf}}$ therefore becomes more difficult for the 2 and 3 meter RIRs, which contain significantly more maximum phase zeros than the 1 meter RIRs, and thus there is an inevitable decrease in matching of phase distortion onto $\mathbf{B}_{\text{interf}}$, leading to more audible distortions in the resynthesized speech, as reported. An increase in R_{interf} would reduce the number of artifacts by enabling $\mathbf{B}_{\text{interf}}$, to express a more varied range of artifacts, but this would also allow $\mathbf{B}_{\text{interf}}$ to cross-match more of the target component leading to greater magnitude distortion.

Although the listeners rated the NMF-APDS output speech fair for the 2 and 3 RIRs, they didn't notice any significant distortion of the target speech signal, i.e. the direct path delayed speech signal, relative to the original (non-reverberated) speech signal. For these RIRs however, it was difficult to discern between magnitude distortion introduced by the algorithm, or magnitude distortion remaining from the RIR.

6.7 Chapter Summary

Assuming the existence of the Room Impulse Response (RIR) between a users lips and a microphone, this chapter addressed the issue of inverting this estimate so that the users speech may be dereverberated or deconvolved. As is well-known, inverting an RIR, which are generally non-minimum phase, results in an inverse filter requiring a long modeling delay and long length, both of which are unsuitable for real-time applications such as dereverberation. Alternatively, the RIR may be decomposed into a minimum phase/all-pass decomposition and

the inverted, which results in zero delay but gives rise to perceptually detrimental artifacts attributable to the phase distortion of the all-pass filter.

In this chapter we described all-pass phase distortion; its properties and how it arises in the context of minimum phase filtering. It was described that this distortion is a consequence of a disrupted relationship between the magnitude response of the maximum phase zeros and the phase response, which in the context of minimum phase inverse filtering, arises from the amplification of the all-pass phase distortion in the reverberated speech signal by the minimum phase inverse filter. Based on some of the properties of all-pass phase distortion we proposed two alternative inversions schemes both of which expressly aimed for optimal inversion in terms of the magnitude and phase distortion, and delay. These properties were targeted with a view to the real-time application of single channel inverse filtering in the context of dereverberation.

The first presented algorithm avoids fully inverting the magnitude response of high-Q maximum phase zeros of the RIR, such that the suppressed all-pass phase distortion in $y(n)$ is not exposed in the processed speech. This approach employs the group delay function of the all-pass component of the RIR to estimate the frequencies and radii of the desired high-Q poles in the minimum phase inverse response. For each identified pole, a selective filter is used to reduce the pole radius, thereby reducing the magnitude response at the corresponding frequency, and increase the decay rate of the impulse response. The proposed algorithm was evaluated by a comparative listening against an existing partial inversion technique. Although the listening panel was composed of mainly non-expert listeners, the difference in quality between the speech produced by both algorithms was such that the panel consistently preferred the output speech from the proposed algorithm. The panel also reported the complete absence of phase distortion related audible artifacts, which was backed up by objective results. However, it was reported that some intelligibility of the proposed approach was lost by the proposed approach, which is ascribable to the magnitude distortion at the frequencies of maximum phase zeros that is required to mitigate phase distortion.

For the second inversion scheme, the all-pass phase distortion problem was cast in a speech enhancement framework, in which the phase distortion is considered an interference source that is to be removed and the remaining speech is considered to be a clean speech target to be preserved. This approach was motivated by the distinct features of all-pass distortion in the magnitude spectral domain, which differ markedly from the target speech. This property was exploited by employing NMF in a MSSS setting to segregate the interference and retain the target in the magnitude STFT domain, an algorithm we named NMF All Pass Distortion Suppression. A procedure for obtaining representative spectral data of the all-pass phase distortion a priori was presented, from which a NMF basis for the interference component is pre-trained, and together with a pre-trained speaker independent NMF basis, are used during a restricted NMF procedure to segregate the spectral frames of the post minimum phase inverted speech signal into interference and target components. A final rendition of the target signal is obtained using a re-filtering procedure. We employed the same

procedure to evaluate this algorithm as for the first proposal. The same panel of listeners preferred the output speech from NMF-APDS over the output speech of the comparative algorithm, for RIRs with a relatively small number of maximum phase zeros. For RIRs with a relatively large number of maximum phase zeros, such as those recorded further away from the loudspeaker, the panel noticed a considerable increase in distracting phase distortion, which is reflected in the decline in subjective ratings.

7 DISCUSSION AND FURTHER WORK

This thesis has addressed a number of speech signal processing problems that are pertinent to hands-free telephony, namely, the acoustic echo problem, the associated doubletalk detection problem, and the near-end speaker reverberation problem. This chapter gives a synopsis of these contributions and offers future research directions.

7.1 A synopsis of the contributions of this thesis

In chapter 4, the single channel acoustic echo problem was treated as a reverberant MSSS problem, in which acoustic echo is mitigated by extracting the near-end users speech signal from a mixture also containing echo and noise. This approach resulted in an algorithm (NMF-NSE) that employs NMF and some of the techniques of model-based MSSS, customized for the inherently online and reverberant nature of the AE problem, to extract the near-end users speech signal by tracking its spectral features in the magnitude STFT domain. It was shown that NMF-NSE has comparable echo mitigating capabilities to those of effective AEC-DTD methods with similar computational loads. The characteristics of the echo mitigation achieved by NMF-NSE are preferable to those of existing adaptive system identification methods in the sense that optimal echo reduction is available immediately upon initiation and is subsequently unaffected by either double-talk or by room changes, without the need for a separate DTD algorithm and concomitant problems. NMF-NSE however also introduces more perceptible distortion to the speech signal during double-talk than the conventional methods tested.

In chapter 5, doubletalk detection for block-based AEC was also addressed in a reverberant MSSS framework. For this approach, the magnitude spectral echo estimate that is produced by NMF-NSE was utilized to compute a DT detection variable, which is compared to a threshold to control the adaptation of a paired conventional block based AEC. By virtue of the robustness of this echo estimate to both DT and room change, it was shown that NMF-

DTD enables its paired block-based adaptive filter to converge at a rate approaching that of an identical adaptive filter without a DTD, including during initiation and after room changes. In contrast, the adaptation rate of an identical adaptive filter under the control of a conventional DTD method was severely impeded by false positives during such events. A drawback of NMF-DTD however, is that it requires a higher hardware resource requirement than the conventional DTD method that was used as a reference.

In Chapter 6, we investigated and then addressed a specific problem related to dereverberation, namely, the single channel RIR inversion problem. We comprehensively described how all-pass phase distortion arises in this context, elucidated a number of its properties, and described how some existing inversion techniques manage to mitigate it. The resulting insights motivated two novel single channel inversion schemes, both of which aim to invert a single RIR while precisely mitigating perceivable all-pass phase distortion for low-delay. The first proposed scheme follows from more recent inversion approaches in that it selectively modifies the inverse of the minimum phase sequence of an RIR so that perceivable phase distortion is not introduced into the processed speech signal. It was shown, using a perceptually relevent objective measure of phase distortion, that this approach prevents the introduction of perceivable all-pass phase distortion into the processed speech. In addition, subjective listening tests revealed that the proposed approach yielded better quality speech relative to a comparable inversion technique; albeit, with some loss of intelligibility for RIRS with a large number of maximum phase zeros due to magnitude distortion. For the second inversion approach the all-pass phase distortion problem was framed as a speech enhancement problem, suited to the application of NMF and the techniques of MSSS; similar to chapters 4 and 5. Subjective listening tests demonstrated that this approach yields more highly rated speech to a comparable algorithm for RIRs with a relatively small number of high-Q maximum phase zeros; however, for RIRs with a large number of such zeros its performance degrades rather sharply. A benefit of these two algorithms is that by addressing all-pass phase distortion without introducing excessive delay, they are suited to applications that require low delay, as may arise in real-time application of inverse filtering algorithms.

7.2 Future Work

In the course of performing the work presented in this thesis a number of issues arose that we contend deserve further work. One direction of further work that applies to all the contributions made in this thesis is real-time implementation of the proposed algorithms. It is anticipated that the transition from simulation, as demonstrated herein, to DSP hardware, will involve addressing a number of issues, many of which are related to the iterative nature of the NMF updates and the associated numerical issues. Specific areas of further work for each individual contributory chapter follow.

7.2.1 NMF-NSE

While we are content in chapter 4 to have shown in the comparative study that the level of distortion introduced by NMF-NSE during DT is at least invariant, being unaffected by abrupt

room change, the reported level of distortion during DT however, may be reduced by further innovation to the speaker basis. The generality of the low-rank speaker-independent basis $\mathbf{B}_v$ that was used in the comparative study causes the separation procedure to erroneously select vectors from $\mathbf{B}_d(k)$ rather than $\mathbf{B}_v$ during DT, i.e. speaker matching. As was shown in section 4.3.2.7 however, employing a speaker-dependent $\mathbf{B}_v$ basis reduces the frequency with which this error occurs. Such a basis could be trained or compiled, either upon initiation or opportunely whenever $\mathbf{x}(k) = \mathbf{0}$ and $\mathbf{y}(k) \neq \mathbf{0}$. The latter approach represents a departure from existing approaches to echo mitigation, such as AEC, which generally ignore $\mathbf{v}(k)$, and as such, is an interesting direction for further work.

The choice of R_v and R_d was shown in section 4.3.2 to be a trade-off, with R_d being inversely proportional to echo-matching error and proportional to speaker-matching error, and visa versa R_v. In the subsequent comparison study, R_v was kept small in favor of echo reduction, but this tradeoff might be alleviated, without resorting to a speaker-dependent basis, by employing a larger $\mathbf{B}_v$ in conjunction with an additional sparsity constraint. A larger $\mathbf{B}_v$ would be better able to represent any subsequent unknown speaker, while the imposition of the sparsity constraint on $\mathbf{g}_v(k)$ would ensure that during any individual frame only a small number of columns from $\mathbf{B}_v$ could be selected during the separation stage, and only those columns would be optimized to the current speaker during the post-separation NMF procedure and used as the final estimate of the near-end speaker spectrum for that frame. By this way a large $\mathbf{B}_v$ multiplied by a sparse $\mathbf{g}_v(k)$ might provide better separation fidelity and less cross-matching error to the existing low-rank $\mathbf{B}_v$ approach; the additional computational load may be controlled by a mechanism that discards impertinent vectors in the speaker basis early on in the optimization procedure for each frame, similar to that proposed in [357]. As described in chapter 2, sparse NMF techniques already exist and have been employed in MSSS studies [24]–[30].

An aspect of the near-end environment that has not been incorporated into this study is the magnitude of the frequency response of the loudspeaker, which has been tacitly assumed to be flat. As described in chapter 2, this is an active area of AEC and DTD research, and it would be interesting to see how the NMF-NSE separation fidelity is affected by the presence of an unknown frequency response. A known frequency response, of any type, could be readily incorporated into NMF-NSE by multiplying the columns of the basis $\mathbf{x}(k)$ by that response.

In section 4.3.3 of chapter 4, NMF-NSE and AEC-DTD where shown to have different performance characteristics, with NMF-NSE being shown to provide a relatively fixed level of performance during all states of the acoustic echo problem, and with AEC-DTD being shown to provide a more variable level of performance that can be superior to NMF-NSE during certain conditions. Future work could involve developing a hybrid of these two approaches, one that compliments the two techniques such that an algorithm with better overall performance characteristics is realised. One straightforward approach would be to run both algorithms in parallel and employ a control algorithm to switch between both algorithms

depending on the state; one rule would be to switch the output to NMF-NSE should a room change be detected. In addition, as demonstrated in chapter 5, the output of NMF-NSE can be used to provide doubletalk detection for the AEC.

Wideband speech recovery is another interesting research topic that could be addressed using NMF-NSE. Currently, public telephones constrain the speech bandwidth to be between 300-3400 Hz, which limits the quality of the resulting speech signal. Wideband speech recovery algorithms seek to reconstruct this lost bandwidth such that perceptually better quality speech is attained. In [433], the techniques of model-based MSSS where applied to the more general problem of bandwidth expansion. In this application, a wideband spectral basis is learned using example wideband audio a priori. The input narrowband audio signal is then decomposed onto the narrowband frequency range of the basis. The audio signal is then reconstructed using the full bandwidth of the basis, resulting in a wideband estimate of the input. Following a similar procedure, where the speaker basis is trained on wideband speech rather than narrowband speech as employed in this work, NMF-NSE could be extended to expand bandwidth of the near-end speaker signal.

Sparsity of speech signals in the STFT domain was discussed regularly throughout this thesis. However, it is noteworthy that this property of speech signals has hitherto not been explicitly exploited for the AEC-DTD problem, while sparsity related to the RIR, as discussed in section 2.2.1, has been extensively explored, see for example [30]. Given the successful use of sparsity, albeit indirectly, in this work, we contend that investigating the applications of sparsity for AEC-DTD is an interesting avenue for future research.

Finally, it is worth contemplating further work in the context of infinte hardware resources. In such a context, a Graphical source model approach to the Acoustic echo problem and/or the Doubletalk Detection problem becomes interesting. Graphical models, as employed in [319], can incorporate higher level features such as semantic or grammar cues in addition to spectro-temporal cues, and have been shown to improve model-based MSSS [319], for a fixed grammar. In the acoustic echo context a graphical model could be dynamically trained for the echo signal, incorporating information at the semantic or grammar level; and a static near-end speaker model incorporating generic semantic and grammar rules could be employed for the near-end speaker. It is envisioned that, given sufficent modeling power, such an approach could lead to complete separation of the echo and near-end speaker.

7.2.2 NMF-DTD

The experimental results in section 5.4.2 of chapter 4 indicated that the false positives generated by NMF-DTD, which are induced by echo-matching, are somewhat tolerable, as that they occur intermittingly over time and are independent of the adaptive enclosure model. Also in section 5.4.2, it was shown that the P_m performance of NMF-DTD matches that of conventional DTD, implying that speaker-matching is similarly unproblematic. Given these results, we contend that future work to improve the performance of NMF-DTD by optimizing NMF-NSE i.e. by reducing cross-matching via a speaker dependent $\mathbf{B}_v$, would have limited utility. However, an issue that is worth addressing is the relatively high hardware resource

requirement of NMF-DTD. For example, since NMF-DTD is not concerned with recreating the near-end speaker in each frame, it being instead tasked with generating a DT decision for each frame, it is feasible that its hardware resource requirement could be reduced by processing a subset of frequency bins of $\mathbf{y}(k)$ rather than the whole of $\mathbf{y}(k)$. It is conceivable moreover, that by choosing these frequency bins such that their combined SNR is higher than that of $\mathbf{y}(k)$ on average i.e. omit high frequencies and favor low frequencies where speech dominates, the classification of DT may also improve; this approach has precedence in the DTD literature [88].

Due to its fitness for NMF-DTD, the GMDFα algorithm was chosen as the foreground adaptive filter in the experiments in chapter 5. However, the utility of NMF-DTD is not restricted to this algorithm. For example, NMF-DTD could be paired straightforwardly with STFT-based system identification techniques [434], including AES techniques [130, 131, 135], whereby both the STFT-based system identification technique and the NMF-DTD algorithm would share the same window size and stepsize. NMF-DTD could also be applied to the multi-delay adaptive algorithm (MDF), which is a member of the FDAF class and whose input blocks are overlapped by 50 %, but unlike GMDFα, does not allow for overlap between output frames. To integrate these algorithms, the block size of MDF could be set to the NMF-DTD frame length or visa versa, by which each computed value of $\hat{\xi}^2(k)$ and the derived DT decision would correspond to an output block of MDF, rather than to an output frame as it did for GMDFα-MDF-DTD. It is envisioned that hardware resource savings would accrue from these AEC/AES-NMF-DTD pairings, by sharing common FFT operations and memory; likewise, for AES input frames.

7.2.3 Single Channel RIR inversion

In chapter 6, our two proposed inversion schemes were evaluated in a simulation environment using pre-recorded RIRs and speech signals, which is the norm for newly proposed DSP algorithms. As mentioned at the top of this section, the use of these algorithms in real time on both measured and estimated RIRs should be investigated in future work. However, apart from the obvious need to evaluate these algorithms under real-time constraints it is particularly important in the audio equalization context, where it was has been shown that full or ideal RIR inverse filters perform considerable worse in real-time than what is expected based on results attained in offline simulations [403]. This simulation-practice performance differential was ascribed to the inverse RIRs, which were described as being weakly non-stationary [403]. However, the inverses of complex smoothed RIRs were shown to perform as anticipated in real-time [435], which suggests that appropriately modified RIRs, such as produced by the proposed method, may perform as expected in such environments.

In chapter 6, we assessed the performance of the proposed inversion techniques using standard comparative listening tests. Future work may comprise of conducting more sophisticated listening tests where the processed speech is evaluated over several different categories; this generally involves trained listeners and listening tests undertaken over a

number of days, which is expensive to perform. As described in the evaluation section the proposed partial inversion technique leaves magnitude distortion in the processed speech, which reportedly affects speech intelligibility; in some cases, the proposed algorithm introduces additional magnitude distortion. Therefore, future work should involve more closely examining this trade-off between magnitude and all-pass phase distortion. One possible modification to the proposed approach may involve setting the pole replacement radius for each indentified pole rather with an offset that applies to all poles.

A novel aspect of the partial minimum phase inversion algorithm presented in chapter 6 was the use of peak finding in the group delay function to estimate the high-Q maximum phase poles of the room transfer functions. We anticipate future work in applying this approach for other tasks in RIR inversion, and perhaps, even for the single channel RIR estimation problem. Moreover, on the single channel RIR estimation problem, which as a research topic is still in its infancy [389-391]; since we assumed the ideal case, that of a measured RIR, we contend that the results presented in chapter 6 may be considered as being representative of a performance upper bound for single channel inverse filter based dereverberation; at least, without introducing delay or artifacts. Additionally, the descriptions given in chapter 6 may provide insight for researchers in the blind single channel dereverberation field.

The description of how all-pass phase distortion arises in chapter 6 originally motivated us to extract the casual component of the Least Squares inverse of the RIR and use it for inverse filtering; this component corresponds to the inverse of the minimum phase zeros of the RIR. However, while in general the resulting inverse filter did not introduce all-pass phase distortion into the processed speech as expected, it was found to have modest reverberation performance. This description also prompted us to use the complex cepstrum to decompose the RIR into its minimum and maximum phase filters; in an attempt to invert only the minimum phase filter thereby preventing all-pass phase distortion. However, it was found that the maximum phase filters obtained by this approach invariable contained large gains in frequency bands containing clusters of maximum phase zeros; accordingly, the minimum phase filters contained significant attenuation at these same bands, rendering this filters unusable. A future topic of work is to investigate, in more detail, why such seemingly straightforward approaches for single channel RIR inversion, without introducing all-pass phase distortion, are inadequate.

7.2.4 Publications arising from this work

N. Cahill and R. Lawlor, "A novel approach to mixed phase room impulse response inversion for speech dereverberation," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 4593-4596.

N. Cahill and R. Lawlor, " A novel approach to acoustic echo cancellation," in *16th European Signal Processing Conference EUSIPCO-08*, Lausanne, Switzerland.

N. Cahill and R. Lawlor, "An Approach to Doubletalk Detection Based on Non-Negative Matrix Factorization," in *Signal Processing and Information Technology, 2008. ISSPIT 2008. IEEE International Symposium on*, 2008, pp. 497-501.

8 REFERENCES

[1] E. Hänsler, "The hands-free telephone problem- An annotated bibliography," *Signal Processing,* vol. 27, pp. 259-271, 1992.

[2] P. A. Naylor and N. D. E. Gaubitch, *Speech Dereverberation*, 1 ed.: Springer, 2010.

[3] S. L. Gay and J. E. Benesty, *Acoustic Signal Processing for Telecommunication* vol. 511. Boston Kluwer Academic, 2000.

[4] E. Hänsler and G. Schmidt, *Acoustic Echo and Noise Control: A Practical Approach*. New York: Wiley 2004.

[5] J. Benesty, Gänsler, T., Morgan, D.R., Sondhi, M.M., Gay, S.L (eds), *Advances in Network and Acoustic Echo Cancellation*. New York: Springer-Verlag, 2001.

[6] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature,* vol. 401, pp. 788-791, Oct 1999.

[7] B. D. Radlovic and R. A. Kennedy, "Nonminimum-phase equalization and its subjective importance in room acoustics," *Speech and Audio Processing, IEEE Transactions on,* vol. 8, pp. 728-737, 2000.

[8] E. A. P. Habets, "Single- and Multi-Microphone Speech Dereverberation using Spectral Enhancement," Technische Universiteit Eindhoven, 2007.

[9] E. Zwicker and H. Fastl, *Psychoacoustics - Facts and Models*, 2 ed. Heidelberg.: Springer, 1999.

[10] P. C. Loizou, *Speech enhancement: theory and practice*: CRC Press, 2007.

[11] R. H. Bolt and A. D. MacDonald, "Theory of Speech Masking by Reverberation," *J. Acoust. Soc. Am.,* vol. 21, 1949.

[12] A. K. Nábělek, T. R. Letowski, and F. M. Tucker, "Reverberant overlap- and self-masking in consonant identification," *J. Acoust. Soc. Am.,* vol. 86, pp. 1259-1265 1989.

[13] M. M. Sondhi and D. A. Berkley, "Silencing echoes on the telephone network," *Proceedings of the IEEE,* vol. 68, pp. 948-963, 1980.

[14] O. M. M. Mitchell and D. A. Berkley, "A full-duplex echo suppressor using center clipping," *Bell Syst. Tech. J.,* vol. 50, pp. 1619-1630, 1971.

[15] C. Breining, P. Dreiscitel, E. Hansler, A. Mader, B. Nitsch, H. Puder*, et al.*, "Acoustic echo control. An application of very-high-order adaptive filters," *Signal Processing Magazine, IEEE,* vol. 16, pp. 42-69, 1999.

[16] Chao, J., Tsujii, and S., "A stable and distortion-free echo and howling canceller," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP-88)*, 1988, pp. 1620–1623.

[17] *Least mean-square adaptive filters*. Hoboken N. J.: Wiley-Interscience, 2003.

[18] J. K. Soh and S. C. Douglas, "Efficient Implementations Of The NLMS Algorithm With Decorrelation Filters For Acoustic Echo Cancellation," in *International Workshop on Acoustic Echo and Noise Control* Pocono Manor, PA, 1999, pp. 164-167.

[19] L. Rrtveit and J. H. Husy, "A new prewhitening-based adaptive filter which converges to the Wiener-solution," in *Signals, Systems and Computers, 2009 Conference Record of the Forty-Third Asilomar Conference on*, 2009, pp. 1360-1364.

[20] R. Frenzel and M. E. Hennecke, "Using prewhitening and stepsize control to improve the performance of the LMS algorithm for acoustic echo compensation," in *Circuits and Systems, 1992. ISCAS '92. Proceedings., 1992 IEEE International Symposium on*, 1992, pp. 1930-1932 vol.4.

[21] H. Yasukawa and S. Shimada, "An acoustic echo canceller using subband sampling and decorrelation methods," *Signal Processing, IEEE Transactions on,* vol. 41, pp. 926-930, 1993.

[22] S. Yamamoto, S. Kitayama, J. Tamura, and H. Ishigami, "An adaptive echo canceler with linear predictor," *Trans. IEICE of Japan,* vol. 62, pp. 851 - 857, 1979.

[23] C. Paleologu, J. Benesty, and S. Ciochina, "A variable step-size proportionate NLMS algorithm for Echo Cancellation," *Revue Roumaine Des Sciences Techniques-Serie Electrotechnique Et Energetique,* vol. 53, pp. 309-317, Jul-Sep 2008.

[24] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *Speech and Audio Processing, IEEE Transactions on,* vol. 8, pp. 508-518, 2000.

[25] P. Loganathan, E. A. P. Habets, and P. A. Naylor, "A proportionate adaptive algorithm with variable partitioned block length for acoustic echo cancellation," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 73-76.

[26] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, 2002, pp. II-1881-II-1884.

[27] S. L. Gay, "An efficient, fast converging adaptive filter for network echo cancellation," in *Signals, Systems & Computers, 1998. Conference Record of the Thirty-Second Asilomar Conference on*, 1998, pp. 394-398 vol.1.

[28] S. Makino, Y. Kaneda, and N. Koizumi, "Exponentially weighted stepsize NLMS adaptive filter based on the statistics of a room impulse response," *Speech and Audio Processing, IEEE Transactions on,* vol. 1, pp. 101-108, 1993.

[29] P. Loganathan, A. W. H. Khong, and P. A. Naylor, "A Class of Sparseness-Controlled Algorithms for Echo Cancellation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 17, pp. 1591-1601, 2009.

[30] J. Benesty, Y. A. Huang, J. Chen, and P. A. Naylor, "Adaptive algorithms for the identification of sparse impulse responses " in *Selected methods for acoustic echo and noise control*, ed: Springer, 2006, pp. 125-153.

[31] R. Harris, D. Chabries, and F. Bishop, "A variable step (VS) adaptive filter algorithm," *Acoustics, Speech and Signal Processing, IEEE Transactions on,* vol. 34, pp. 309-316, 1986.

[32] R. H. Kwong and E. W. Johnston, "A variable step size LMS algorithm," *Signal Processing, IEEE Transactions on,* vol. 40, pp. 1633-1642, 1992.

[33] V. J. Mathews and Z. Xie, "A stochastic gradient adaptive filter with gradient adaptive step size," *Signal Processing, IEEE Transactions on,* vol. 41, pp. 2075-2087, 1993.

[34] J. B. Evans, P. Xue, and B. Liu, "Analysis and implementation of variable step size adaptive algorithms," *Signal Processing, IEEE Transactions on,* vol. 41, pp. 2517-2535, 1993.

[35] T. Aboulnasr and K. Mayyas, "A robust variable step-size LMS-type algorithm: analysis and simulations," *Signal Processing, IEEE Transactions on,* vol. 45, pp. 631-639, 1997.

[36] S. Hyun-Chool, A. H. Sayed, and S. Woo-Jin, "Variable step-size NLMS and affine projection algorithms," *Signal Processing Letters, IEEE,* vol. 11, pp. 132-135, 2004.

[37] A. Mader, H. Puder, and G. U. Schmidt, "Step-size control for acoustic echo cancellation filter - an overview," *Signal Process.,* vol. 80, pp. 1697-1719, 2000.

[38] H. Buchner, J. Benesty, T. Gansler, and W. Kellermann, "Robust extended multidelay filter and double-talk detector for acoustic echo cancellation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 14, pp. 1633-1644, 2006.

[39] M. Asif Iqbal and S. L. Grant, "Novel variable step size nlms algorithms for echo cancellation," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 241-244.

[40] J. Benesty, C. Paleologu, and S. Ciochina, "On Regularization in Adaptive Filtering," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 19, pp. 1734-1742, 2011.

[41] S. Haykin, "Adaptive Filter Theory," 3rd ed. Englewood Cliffs, N. J.: Prentice Hall, 1996.
[42] G. O. Glentis, K. Berberidis, and S. Theodoridis, "Efficient least squares adaptive algorithms for FIR transversal filtering," *Signal Processing Magazine, IEEE,* vol. 16, pp. 13-41, 1999.
[43] Ozeki, K., Umeda, and T., "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electron. Commun. Jpn.,* vol. 67-A, p. 19, 1984.
[44] S. L. Gay and S. Tavathia, "The fast affine projection algorithm," in *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, 1995, pp. 3023-3026 vol.5.
[45] C. Paleologu, S. Ciochina, and J. Benesty, "An Efficient Proportionate Affine Projection Algorithm for Echo Cancellation," *Signal Processing Letters, IEEE,* vol. 17, pp. 165-168, 2010.
[46] T. v. Waterschoot, G. Rombouts, and M. Moonen, "Optimally regularized adaptive filtering algorithms for room acoustic signal enhancement," *Signal Process.,* vol. 88, pp. 594-611, 2008.
[47] A. Tikhonov and V. Arsenin, *Solutions of Ill-Posed Problems*. New York: Wiley, 1977.
[48] L. Yuanqing and D. D. Lee, "Bayesian regularization and nonnegative deconvolution for room impulse response estimation," *Signal Processing, IEEE Transactions on,* vol. 54, pp. 839-847, 2006.
[49] F. Sha, L. K. Saul, and D. Lee, "Multiplicative updates for nonnegative quadratic programming in support vector machines " in *Advances in Neural Information Processing Systems*, 2002.
[50] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J.Acoust.Soc.Am.,* vol. 65, pp. 943–950, 1979.
[51] J. J. Shynk, "Frequency-domain and multirate adaptive filtering," *Signal Processing Magazine, IEEE,* vol. 9, pp. 14-37, 1992.
[52] D. Mansour and A. Gray, Jr., "Unconstrained frequency-domain adaptive filter," *Acoustics, Speech and Signal Processing, IEEE Transactions on,* vol. 30, pp. 726-734, 1982.
[53] M. Dentino, J. McCool, and B. Widrow, "Adaptive filtering in the frequency domain," *Proceedings of the IEEE,* vol. 66, pp. 1658-1659, 1978.
[54] E. Ferrara, "Fast implementations of LMS adaptive filters," *Acoustics, Speech and Signal Processing, IEEE Transactions on,* vol. 28, pp. 474-475, 1980.
[55] J. S. Soo and K. K. Pang, "Multidelay block frequency domain adaptive filter," *Acoustics, Speech and Signal Processing, IEEE Transactions on,* vol. 38, pp. 373-376, 1990.
[56] J. Benesty and T. Gansler, "A multidelay double-talk detector combined with the MDF adaptive filter," *Eurasip Journal on Applied Signal Processing,* vol. 2003, pp. 1056-1063, Oct 2003.
[57] E. Moulines, O. Ait Amrane, and Y. Grenier, "The generalized multidelay adaptive filter: structure and convergence analysis," *Signal Processing, IEEE Transactions on,* vol. 43, pp. 14-28, 1995.
[58] N. Bernhard H, "A frequency-selective stepfactor control for an adaptive filter algorithm working in the frequency domain," *Signal Processing,* vol. 80, pp. 1733-1745, 2000.
[59] Lee, Junghsi, C. Huang, and Hsu, "Step-Size Bounds Analysis of the Generalized Multidelay Adaptive Filter " in *Proceedings of the World Congress on Engineering WCE 2007*, London, UK., 2007.
[60] J. Prado and E. Moulines, "Frequency-domain adaptive filtering with applications to acoustic echo cancellation," *Annales des Télécommunications,* vol. 49, pp. 414--428, 1994.
[61] W. L. B. Jeannes, P. Scalart, G. Faucon, and C. Beaugeant, "Combined noise and echo reduction in hands-free systems: a survey," *Speech and Audio Processing, IEEE Transactions on,* vol. 9, pp. 808-820, 2001.
[62] J. Lariviere and R. Goubran, "GMDF for noise reduction and echo cancellation," *Signal Processing Letters, IEEE,* vol. 7, pp. 230-232, 2000.
[63] M. Zeller and W. Kellermann, "Fast and Robust Adaptation of DFT-Domain Volterra Filters in Diagonal Coordinates Using Iterated Coefficient Updates," *Signal Processing, IEEE Transactions on,* vol. 58, pp. 1589-1604, 2010.
[64] K. Eneman and M. Moonen, "Iterated partitioned block frequency-domain adaptive filtering for acoustic echo cancellation," *Speech and Audio Processing, IEEE Transactions on,* vol. 11, pp. 143-158, 2003.
[65] B. J. and G. T., "On data-reuse adaptive algorithms," in *8th IEEE International Workshop on Acoustic Echo and Noise Control,*, 2003.

[66] S. M. Kuo, K.-A. Lee, and S. G. Woon, *Subband Adaptive Filtering: Theory and Implementation*: Wiley, 2009.
[67] Y. Avargel and I. Cohen, "System Identification in the Short-Time Fourier Transform Domain With Crossband Filtering," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 1305-1319, 2007.
[68] Y. Avargel and I. Cohen, "Adaptive System Identification in the Short-Time Fourier Transform Domain Using Cross-Multiplicative Transfer Function Approximation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 16, pp. 162-173, 2008.
[69] A. Gilloire and M. Vetterli, "Adaptive filtering in subbands with critical sampling: analysis, experiments, and application to acoustic echo cancellation," *Signal Processing, IEEE Transactions on,* vol. 40, pp. 1862-1875, 1992.
[70] D. Duttweiler, "A Twelve-Channel Digital Echo Canceler," *Communications, IEEE Transactions on,* vol. 26, pp. 647-653, 1978.
[71] G. Szwoch, A. Czyzewski, and M. Kulesza, "A low complexity double-talk detector based on the signal envelope," *Signal Processing,* vol. 88, pp. 2856-2862, 2008.
[72] H. Ezzaidi, I. Bourmeyster, and J. Rouat, "A new algorithm for double talk detection and separation in the context of digital mobile radio telephone," in *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, 1997, pp. 1897-1900 vol.3.
[73] T. A. Vu, "Double talk detection using a psychoacoustic auditory model," University of Ottawa (Canada), 2007.
[74] G. Szwoch, A. Czyzewski, and A. Ciarkowski, "A Double-Talk Detector Using Audio Watermarking," *J. Audio Eng. Soc,* vol. 57, pp. 916--926, 2009.
[75] Y. Hua and W. Bo-Xiu, "A new double-talk detection algorithm based on the orthogonality theorem," *Communications, IEEE Transactions on,* vol. 39, pp. 1542-1545, 1991.
[76] R. D. Wesel, "Cross-correlation vectors and double-talk control for echo cancellation," Unpublished, 1994.
[77] J. Benesty, D. R. Morgan, and J. H. Cho, "A new class of doubletalk detectors based on cross-correlation," *Speech and Audio Processing, IEEE Transactions on,* vol. 8, pp. 168-172, 2000.
[78] T. Gansler, "The fast normalized cross-correlation double-talk detector," *Signal Processing,* vol. 86, pp. 1124-1139, Jun 2006.
[79] J. H. Cho, D. R. Morgan, and J. Benesty, "An objective technique for evaluating doubletalk detectors in acoustic echo cancelers," *Speech and Audio Processing, IEEE Transactions on,* vol. 7, pp. 718-724, 1999.
[80] A. Per and A. Jakobsson, "A study of doubletalk detection performance in the presence of acoustic echo path changes," *Consumer Electronics, IEEE Transactions on,* vol. 52, pp. 515-522, 2006.
[81] F. Lindstrom, C. Schuldt, and I. Claesson, "An Improvement of the Two-Path Algorithm Transfer Logic for Acoustic Echo Cancellation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 1320-1326, 2007.
[82] C. Schuldt, F. Lindstrom, and I. Claesson, "An improved deviation measure for two-path echo cancellation," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, 2010, pp. 305-308.
[83] A. Sugiyama, J. Berclaz, and M. Sato, "Noise-robust double-talk detection based on normalized cross correlation and a noise offset," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, 2005, pp. iii/153-iii/156 Vol. 3.
[84] D. G. James and A. G. Rafik, "Statistical Analysis of Doubletalk Detection for Calibration and Performance Evaluation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 1035-1043, 2007.
[85] M. A. lqbal, J. W. Stokes, and S. L. Grant, "Normalized Double-Talk Detection Based on Microphone and AEC Error Cross-Correlation," in *Multimedia and Expo, 2007 IEEE International Conference on*, 2007, pp. 360-363.
[86] K. Ghose and V. U. Reddy, "A double-talk detector for acoustic echo cancellation applications," *Signal Processing,* vol. 80, pp. 1459-1467, 2000.
[87] M. S. Kumar, "Low delay nearend speech detector for acoustic echo cancellation," in *TENCON 2008 - 2008 IEEE Region 10 Conference*, 2008, pp. 1-6.
[88] T. Gansler, M. Hansson, C. J. Ivarsson, and G. Salomonsson, "A double-talk detector based on coherence," *Communications, IEEE Transactions on,* vol. 44, pp. 1421-1427, 1996.

[89] J. Vogel, M. Heckmann, and K. Kroschel, "Frequency domain step-size control in non-stationary environments," in *Signals, Systems and Computers, 2000. Conference Record of the Thirty-Fourth Asilomar Conference on*, 2000, pp. 212-216 vol.1.

[90] T. Gansler and J. Benesty, "A frequency-domain double-talk detector based on a normalized cross-correlation vector," *Signal Processing,* vol. 81, pp. 1783-1787, Aug 2001.

[91] M. A. Iqbal, S. L. Grant, and J. W. Stokes, "A frequency domain doubletalk detector based on cross-correlation and extension to multi-channel case," in *Signals, Systems and Computers, 2009 Conference Record of the Forty-Third Asilomar Conference on*, 2009, pp. 638-641.

[92] P. Ahgren, "Acoustic echo cancellation and doubletalk detection using estimated loudspeaker impulse responses," *Speech and Audio Processing, IEEE Transactions on,* vol. 13, pp. 1231-1237, 2005.

[93] K. Ochiai, T. Araseki, and T. Ogihara, "Echo Canceler with Two Echo Path Models," *Communications, IEEE Transactions on,* vol. 25, pp. 589-595, 1977.

[94] M. A. Iqbal and S. L. Grant, "Novel and Efficient Download Test for Two Path Echo Canceller," in *Applications of Signal Processing to Audio and Acoustics, 2007 IEEE Workshop on*, 2007, pp. 167-170.

[95] Diethorn and E. J, "Improved decision logic for two-path echo cancelers," in *Proc. Int. Workshop Acoust. Echo Noise Control, IWAENC*, 2001

[96] H. K. Jung, N. S. Kim, and T. Kim, "A new double-talk detector using echo path estimation," *Speech Communication,* vol. 45, pp. 41-48, 2005.

[97] H. Jiaquan, S. Nordholm, and Z. Zhuquan, "An echo path variation detector based on coherence," in *Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on*, 2003, pp. 124-128 Vol.1.

[98] M. A. Iqbal and S. L. Grant, "A Novel Normalized Cross-Correlation Based Echo-Path Change Detector," in *Region 5 Technical Conference, 2007 IEEE*, 2007, pp. 249-251.

[99] A. I. Mohammad, "Simple and efficient solutions to the problems associated with acoustic echo cancellation," Ph.D, University of Missouri-Rolla, 2007.

[100] C. Breining, "A robust fuzzy logic-based step-gain control for adaptive filters in acoustic echo cancellation," *Speech and Audio Processing, IEEE Transactions on,* vol. 9, pp. 162-167, 2001.

[101] M. A. Iqbal, J. Stokes, J. C. Platt, A. Surendran, and G. S. L., "Doubletalk Detection Using Real Time Recurrent Learning," in *International workshop on acoustic echo and noise control (IWAENC)* Paris, France, 2006.

[102] T. Gansler, S. L. Gay, M. M. Sondhi, and J. Benesty, "Double-talk robust fast converging algorithms for network echo cancellation," *IEEE Transactions on Speech and Audio Processing,* vol. 8, pp. 656-663, Nov 2000.

[103] L. R. Vega, H. Rey, J. Benesty, and S. Tressens, "A Fast Robust Recursive Least-Squares Algorithm," *IEEE Transactions on Signal Processing,* vol. 57, pp. 1209-1216, Mar 2009.

[104] T. Gansler, J. Benesty, S. L. Gay, and M. M. Sondhi, "A robust proportionate affine projection algorithm for network echo cancellation," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, 2000, pp. II793-II796 vol.2.

[105] L. R. Vega, H. Rey, J. Benesty, and S. Tressens, "A Family of Robust Algorithms Exploiting Sparsity in Adaptive Filters," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 17, pp. 572-581, 2009.

[106] L. R. Vega, H. Rey, J. Benesty, and S. Tressens, "A New Robust Variable Step-Size NLMS Algorithm," *Signal Processing, IEEE Transactions on,* vol. 56, pp. 1878-1893, 2008.

[107] J. Jang-Chyuan and H. Shih-Fu, "Acoustic echo cancellation using iterative-maximal-length correlation and double-talk detection," *Speech and Audio Processing, IEEE Transactions on,* vol. 9, pp. 932-942, 2001.

[108] J. F. Doherty and R. Porayath, "A robust echo canceler for acoustic environments," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on,* vol. 44, pp. 389-396, 1997.

[109] J. C. Jenq and S. F. Hsieh, "Decision of double-talk and time-variant echo path for acoustic echo cancellation," *Signal Processing Letters, IEEE,* vol. 10, pp. 317-319, 2003.

[110] W. Toon van, R. Geert, V. Piet, and M. Marc, "Double-Talk-Robust Prediction Error Identification Algorithms for Acoustic Echo Cancellation," *Signal Processing, IEEE Transactions on,* vol. 55, pp. 846-858, 2007.

[111] S. Yamamoto and S. Kitayama, "An adaptive echo canceler with variable step gain method," *Tran. IEICE of Japan,* vol. E.65, pp. 1-8, 1982.
[112] J. Benesty, H. Rey, L. R. Vega, and S. Tressens, "A nonparametric VSSNLMS algorithm," *IEEE Signal Processing Letters,* vol. 13, pp. 581-584, Oct 2006.
[113] C. Paleologu, J. Benesty, and S. Ciochina, "A Variable Step-Size Affine Projection Algorithm Designed for Acoustic Echo Cancellation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 16, pp. 1466-1478, 2008.
[114] V. Jean-Marc, "On Adjusting the Learning Rate in Frequency Domain Echo Cancellation With Double-Talk," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 1030-1034, 2007.
[115] P. O'Grady, B. Pearlmutter, and S. Rickard, "Survey of sparse and non-sparse methods in source separation," *International Journal of Imaging Systems and Technology,* vol. 15, pp. 18-33, 2005.
[116] J.-M. Yang and H. Sakai, "A New Adaptive Filter Algorithm for System Identification Using Independent Component Analysis.," *IEICE Transactions* vol. E90-A pp. 1549-1554 2007.
[117] T. S. Wada and J. Biing-Hwang, "Acoustic echo cancellation based on independent component analysis and integrated residual echo enhancement," in *Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA '09. IEEE Workshop on*, 2009, pp. 205-208.
[118] D. W. E. Schobben and P. W. Sommen, "A frequency domain blind signal separation method based on decorrelation," *Signal Processing, IEEE Transactions on,* vol. 50, pp. 1855-1865, 2002.
[119] J. Gunther and T. Moon, "Adaptive cancellation of acoustic echoes during double-talk based on an information theoretic criteria," in *Signals, Systems and Computers, 2009 Conference Record of the Forty-Third Asilomar Conference on*, 2009, pp. 650-654.
[120] J. Gunther, "Learning Echo Paths During Continuous Double-Talk using Semi-Blind Source Separation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. PP, pp. 1-1, 2011.
[121] Lee, E. Intae, A. C. M. Kalker, C. Kinney, B. Lee, and A. Kalker, "Solving the Acoustic Echo Cancellation Problem in Double-Talk Scenario Using Non-Gaussianity of the Near-End Signal.," in *ICA 2009*, 2009, pp. 589-596.
[122] F. Nesta, T. S. Wada, and J. Biing-Hwang, "Batch-Online Semi-Blind Source Separation Applied to Multi-Channel Acoustic Echo Cancellation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 19, pp. 583-599, 2011.
[123] B. S. Nollett and D. L. Jones, "Nonlinear echo cancellation for hands-free speakerphones," in *IEEE Workshop on Nonlinear Signal and Image Processing (NSIP)*, 1997.
[124] A. Stenger and W. Kellermann, "Adaptation of a memoryless preprocessor for nonlinear acoustic echo cancelling," *Signal Processing,* vol. 80, pp. 1747-1760, 2000.
[125] A. Guerin, G. Faucon, and R. Le Bouquin-Jeannes, "Nonlinear acoustic echo cancellation based on Volterra filters," *Speech and Audio Processing, IEEE Transactions on,* vol. 11, pp. 672-683, 2003.
[126] Y. J. Park and H. M. Park, "DTD-free nonlinear acoustic echo cancellation based on independent component analysis," *Electronics Letters,* vol. 46, pp. 866-868, 2010.
[127] Y. Avargel and I. Cohen, "Modeling and Identification of Nonlinear Systems in the Short-Time Fourier Transform Domain," *Signal Processing, IEEE Transactions on,* vol. 58, pp. 291-304, 2010.
[128] F. Kuech and W. Kellermann, "A novel multidelay adaptive algorithm for Volterra filters in diagonal coordinate representation [nonlinear acoustic echo cancellation example]," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on*, 2004, pp. ii-869-72 vol.2.
[129] T. Gupta, S. B. Suppappola, and A. Spanias, "Nonlinear acoustic echo control using an accelerometer," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 2009, pp. 1313-1316.
[130] C. Avendano, "Acoustic echo suppression in the STFT domain," in *Applications of Signal Processing to Audio and Acoustics, 2001 IEEE Workshop on the*, 2001, pp. 175-178.
[131] C. Faller and J. Chen, "Suppressing Acoustic Echo in a Spectral Envelope Space," *Speech and Audio Processing, IEEE Transactions on,* vol. 13, pp. 1048-1062, 2005.
[132] C. Faller and C. Tournery, "Robust Acoustic ECHO Control using a Simple ECHO Path Model," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, 2006, pp. V-V.

[133] Y.-S. Park and J.-H. Chang, "Double-talk detection based on soft decision for acoustic echo suppression," *Signal Processing,* vol. 90, pp. 1737-1741, 2010.
[134] P. Seon Joon, C. Chom Gun, L. Chungyong, and Y. Dae Hee, "Integrated echo and noise canceler for hands-free applications," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on,* vol. 49, pp. 188-195, 2002.
[135] E. A. P. Habets, S. Gannot, and I. Cohen, "Robust early echo cancellation and late echo suppression in the STFT domain," presented at the International Workshop on Acoustic Echo and Noise Control (IWAENC), Seattle, Washington, USA, 2008.
[136] V. Turbin, A. Gilloire, and P. Scalart, "Comparison of three post-filtering algorithms for residual acoustic echo reduction," in *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, 1997, pp. 307-310 vol.1.
[137] S. Gustafsson, R. Martin, P. Jax, and P. Vary, "A psychoacoustic approach to combined acoustic echo cancellation and noise reduction," *Speech and Audio Processing, IEEE Transactions on,* vol. 10, pp. 245-256, 2002.
[138] L. Seung Yeol and K. Nam Soo, "A Statistical Model-Based Residual Echo Suppression," *Signal Processing Letters, IEEE,* vol. 14, pp. 758-761, 2007.
[139] A. S. Chhetri, A. C. Surendran, J. W. Stokes, and J. C. Platt, "Regression based residual acoustic echo suppression," in *IWAENC*, 2005.
[140] N. Madhu, I. Tashev, and A. Acero, "AN EM-based probabilistic approach for Acoustic Echo Suppression," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 265-268.
[141] Hoshuyama, Osamu, Sugiyama, and Akihiko, "Nonlinear Acoustic Echo Suppressor Based on Spectral Correlation between Residual Echo and Echo Replica," *IEICE Transactions,* vol. 89-A, pp. 3254-3259, 2006.
[142] G. Enzner and P. Vary, "Frequency-domain adaptive Kalman filter for acoustic echo control in hands-free telephones," *Signal Processing,* vol. 86, pp. 1140-1156, 2006.
[143] G. Enzner, "Model-based interrelations of adaptive filter algorithms in acoustic echo control," in *Communications, Computers and Signal Processing, 2009. PacRim 2009. IEEE Pacific Rim Conference on*, 2009, pp. 909-913.
[144] G. Enzner, "Signal Models, Filter Structures, and Adaptive Algorithms for Acoustic Echo Control," *Voice Communication (SprachKommunikation), 2008 ITG Conference on,* pp. 1-4, 2008.
[145] E. Hänsler and G. U. Schmidt, "Hands-free telephones – joint control of echo cancellation and postfiltering," *Signal Processing,* vol. 80, pp. 2295-2305, 2000.
[146] G. Enzner and P. Vary, "Robust and elegant, purely statistical adaptation of acoustic echo canceler and postfilter," in *International Workshop on Acoustic Echo and Noise Control (IWAENC)*, 2003.
[147] J. Herault and C. Jutten, "Space or time adaptive signal processing by neural network models," in *AIP Conference Proceedings 151 on Neural Networks for Computing*, Snowbird, Utah, United States, 1987, pp. 206-211.
[148] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent component analysis*: J. Wiley, 2001.
[149] P. Comon, "Independent component analysis, a new concept?," *Signal Process.,* vol. 36, pp. 287-314, 1994.
[150] P. Comon and C. Jutten, *Handbook of blind source separation: independent component analysis and applications*: Elsevier, 2010.
[151] M. Zibulevsky and B. A. Pearlmutter, "Blind Source Separation by Sparse Decomposition in a Signal Dictionary," *Neural Computation,* vol. 13, pp. 863-882, 2001/04/01 2001.
[152] S. Roweis, "One microphone sound source separation," in *Advances in Neural Information Processing (NIPS)*, 2001, pp. 793-799.
[153] S. Rickard and O. Yilmaz, "On the approximate W-disjoint orthogonality of speech," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, 2002, pp. I-529-I-532.
[154] P. Bofill and M. Zibulevsky, "Underdetermined blind source separation using sparse representations," *Signal Processing,* vol. 81, pp. 2353-2362, 2001.
[155] O. Yilmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *Signal Processing, IEEE Transactions on,* vol. 52, pp. 1830-1847, 2004.
[156] P. D. O'Grady and B. A. Pearlmutter, "The LOST Algorithm: Finding Lines and Separating Speech Mixtures," *EURASIP Journal on Advances in Signal Processing,* vol. 2008, p. 17 2008.

[157] D. Barry, E. Coyle, and B. Lawlor, "Sound Source Separation: Azimuth Discrimination and Resynthesis," in *7th International Conference on Digital Audio Effects, DAFX 04,* , Naples, Italy, 2004.

[158] D. Barry, E. Coyle, and B. Lawlor, "Real-time Sound Source Separation using Azimuth Discrimination and Resynthesis," in *17th Audio Engineering Society Convention*, Moscone Centre, San Francisco, CA, USA,, 2004.

[159] B. A. Pearlmutter and A. M. Zador, "*Monaural Source Separation Using Spectral Cues*," in *ICA2004*, Granada, Spain, , 2004, pp. 478-485.

[160] G. Hu and D. Wang, "Auditory Segmentation Based on Onset and Offset Analysis," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 396-405, 2007.

[161] H. Guoning and W. DeLiang, "Monaural speech segregation based on pitch tracking and amplitude modulation," *Neural Networks, IEEE Transactions on,* vol. 15, pp. 1135-1150, 2004.

[162] W. Mingyang, W. DeLiang, and G. J. Brown, "A multipitch tracking algorithm for noisy speech," *Speech and Audio Processing, IEEE Transactions on,* vol. 11, pp. 229-241, 2003.

[163] G. J. Brown and M. Cooke, "Computational auditory scene analysis," *Computer Speech & Language,* vol. 8, pp. 297-336, 1994.

[164] D. F. Rosenthal and H. G. Okuno, *Computational auditory scene analysis*: Lawrence Erlbaum Associates, 1998.

[165] D. L. Wang and G. J. Brown, *Computational auditory scene analysis: principles, algorithms and applications*: Wiley interscience, 2006.

[166] L. Benaroya, F. Bimbot, and R. Gribonval, "Audio source separation with a single sensor," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 14, pp. 191-199, 2006.

[167] M. N. Schmidt, "Single-channel source separation using non-negative matrix factorization," Ph.D., Technical University of Denmark, 2008.

[168] P. Paatero and U. Tapper, "Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values," *Environmetrics,* vol. 5, pp. 111-126, 1994.

[169] D. Donoho and V. Stodden, "When Does Non-Negative Matrix Factorization Give Correct Decomposition into Parts?," in *Advances in Neural Information Processing Systems (NIPS)*, 2003.

[170] H. Laurberg, "Uniqueness of Non-Negative Matrix Factorization," in *Statistical Signal Processing, 2007. SSP '07. IEEE/SP 14th Workshop on*, 2007, pp. 44-48.

[171] A. Cichocki, R. Zdunek, and S. Amari, "New Algorithms for Non-Negative Matrix Factorization in Applications to Blind Source Separation," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, 2006, pp. V-V.

[172] I. T. Jolliffe, *Principal component analysis*: Springer-Verlag, 2002.

[173] F. Tao, S. Z. Li, S. Heung-Yeung, and Z. HongJiang, "Local non-negative matrix factorization as a visual representation," in *Development and Learning, 2002. Proceedings. The 2nd International Conference on*, 2002, pp. 178-183.

[174] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*: John Wiley & Sons, 2009.

[175] L. Tao and C. Ding, "The Relationships Among Various Nonnegative Matrix Factorization Methods for Clustering," in *Data Mining, 2006. ICDM '06. Sixth International Conference on*, 2006, pp. 362-371.

[176] D. Chris, "On the Equivalence of (Convex-) Nonegative Matrix Factorization and K-meaning Clustering," in *Stanford Workshop on Algorithms for Modern Massive Data Sets (MMDS 2006)*, 2006.

[177] C. Ding, X. He, and H. D. Simon, "On the equivalence of nonnegative matrix factorization and spectral clustering " in *SIAM Data Mining Conf*, 2005, pp. 606--610.

[178] C. Ding, T. Li, and W. Peng, "On the equivalence between Non-negative Matrix Factorization and Probabilistic Latent Semantic Indexing," *Comput. Stat. Data Anal.,* vol. 52, pp. 3913-3927, 2008.

[179] E. Gaussier and C. Goutte, "Relation between PLSA and NMF and implications," in *28th annual international ACM SIGIR conference on Research and development in information retrieval*, Salvador, Brazil, 2005, pp. 601-602.

[180] Hofmann and Thomas, "Probabilistic Latent Semantic Analysis," in *Uncertainty in Artificial Intelligence, UAI'99*, 1999, pp. 289--296.

[181] C. D. Sigg and J. M. Buhmann, "Expectation-maximization for sparse and non-negative PCA," in *Proceedings of the 25th international conference on Machine learning*, Helsinki, Finland, 2008, pp. 960-967.
[182] M. D. Plumbley, "Algorithms for nonnegative independent component analysis," *Neural Networks, IEEE Transactions on,* vol. 14, pp. 534-543, 2003.
[183] C. H. Q. Ding, L. Tao, and M. I. Jordan, "Convex and Semi-Nonnegative Matrix Factorizations," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 32, pp. 45-55, 2010.
[184] I. Dhillon and S. Sra, "Generalized nonnegative matrix approximations with Bregman divergences," in *In: Neural Information Proc. Systems*, 2005, pp. 283-290.
[185] A. Garrido Frenich, M. Martınez Galera, J. L. Martınez Vidal, D. L. Massart, J. R. Torres-Lapasió, K. De Braekeleer*, et al.*, "Resolution of multicomponent peaks by orthogonal projection approach, positive matrix factorization and alternating least squares," *Analytica Chimica Acta,* vol. 411, pp. 145-155, 2000.
[186] D. Guillamet and J. Vitri, "Non-negative Matrix Factorization for Face Recognition," in *Proceedings of the 5th Catalonian Conference on AI: Topics in Artificial Intelligence*, 2002, pp. 336-344.
[187] I. Buciu and I. Pitas, "Application of non-negative and local non negative matrix factorization to facial expression recognition," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, 2004, pp. 288-291 Vol.1.
[188] M. Morup, M. N. Schmidt, and L. K. Hansen, "Invariant sparse coding of image and music data," Technical University of Denmark2006.
[189] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization," *Computational Statistics and Data Analysis,* vol. 52, pp. 155-173, 2007.
[190] F. Shahnaz, M. W. Berry, V. P. Pauca, and R. J. Plemmons, "Document clustering using nonnegative matrix factorization," *Information Processing and Management,* vol. 42, pp. 373-386, 2006.
[191] K. Drakakis, S. Rickard, R. Frein, and A. Cichocki, "Analysis of financial data using non-negative matrix factorization," in *International Mathematical Forum*, 2008, pp. 1853-1870.
[192] L. Tang, "Non-Negative Matrix Factorization for Stock Market Pricing," in *Biomedical Engineering and Informatics, 2009. BMEI '09. 2nd International Conference on*, 2009, pp. 1-5.
[193] P. Smaragdis, "Discovering auditory objects through non-negativity constraints," in *Statistical and Perceptual Audio Processing, (SAPA-04)*, 2004.
[194] C. Yong-Choon, C. Seungjin, and B. Sung-Yang, "Non-negative component parts of sound for classification," in *Signal Processing and Information Technology, 2003. ISSPIT 2003. Proceedings of the 3rd IEEE International Symposium on*, 2003, pp. 633-636.
[195] Y.-C. Cho and S. Choi, "Nonnegative features of spectro-temporal sounds for classification," *Pattern Recognition Letters,* vol. 26, pp. 1327-1336, 2005.
[196] S. A. Abdallah and P. M. D., "Polyphonic transcription by non-negative sparse coding of power spectra. ," in *International Conference on Music Information Retrieval (ISMIR 2004).* Barcelona, Spain, , 2004, pp. 318-325.
[197] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, 2003, pp. 177-180.
[198] E. Benetos, M. Kotti, and C. Kotropoulos, "Musical Instrument Classification using Non-Negative Matrix Factorization Algorithms and Subset Feature Selection," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, 2006, pp. V-V.
[199] A. Bertrand, K. Demuynck, V. Stouten, and H. Van hamme, "Unsupervised learning of auditory filter banks using non-negative matrix factorisation," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 4713-4716.
[200] F. Sha and L. Saul, "Real-time pitch determination of one or more voices by nonnegative matrix factorization," in *Advances in Neural Information Processing Systems 17 (NIPS)*, 2005, pp. 1233-1240.
[201] K. Kumar, R. Singh, B. Raj, and R. Stern, "Gammatone sub-band magnitude-domain dereverberation for ASR," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 4604-4607.

[202] H. Kameoka, T. Nakatani, and T. Yoshioka, "Robust speech dereverberation based on non-negativity and sparse nature of speech spectrograms," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 2009, pp. 45-48.

[203] H. Asari, "Non-negative matrix factorization: A possible way to learn sound dictionaries," Tony Zador Lab, Watson School of Biological Sciences, Cold Spring Harbor Laboratory. 2005.

[204] D. Lee and S. Seung, "Algorithms for Non-negative Matrix Factorization," in *NIPS*, 2000, pp. 556-562.

[205] P. O'Grady, "Sparse Separation of Underdetermined Speech Mixtures (Dissertation)," Ph.D., Hamilton Institute, National University of Ireland Maynooth, 2007.

[206] C. Fevotte, N. Bertin, and J. L. Durrieu, "Nonnegative Matrix Factorization with the Itakura-Saito Divergence: With Application to Music Analysis," *Neural Computation,* vol. 21, pp. 793-830, Mar 2009.

[207] R. Kompass, "A Generalized Divergence Measure for Nonnegative Matrix Factorization," *Neural Comput.,* vol. 19, pp. 780-791, 2007.

[208] R. Hennequin, B. David, and R. Badeau, "Beta-Divergence as a Subclass of Bregman Divergence," *Signal Processing Letters, IEEE,* vol. 18, pp. 83-86, 2011.

[209] A. Cichocki, R. Zdunek, and S.-i. Amari, "Csiszár's Divergences for Non-negative Matrix Factorization: Family of New Algorithms," in *Independent Component Analysis and Blind Signal Separation*. vol. 3889, J. Rosca, D. Erdogmus, J. Príncipe, and S. Haykin, Eds., ed: Springer Berlin / Heidelberg, 2006, pp. 32-39.

[210] A. Cichocki, H. Lee, Y.-D. Kim, and S. Choi, "Non-negative matrix factorization with α-divergence," *Pattern Recognition Letters,* vol. 29, pp. 1433-1440, 2008.

[211] R. de Frein and S. T. Rickard, "Learning speech features in the presence of noise: Sparse convolutive robust non-negative matrix factorization," in *Digital Signal Processing, 2009 16th International Conference on*, 2009, pp. 1-6.

[212] T. Virtanen, "Monaural Sound Source Separation by Nonnegative Matrix Factorization With Temporal Continuity and Sparseness Criteria," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 1066-1074, 2007.

[213] M. S. Lewicki and T. J. Sejnowski, "Learning Overcomplete Representations," *Neural Computation,* vol. 12, pp. 337-365, 2000/02/01 2000.

[214] P. O. Hoyer, "Non-negative Matrix Factorization with Sparseness Constraints," *J. Mach. Learn. Res.,* vol. 5, pp. 1457-1469, 2004.

[215] P. O. Hoyer, "Non-negative sparse coding," in *Neural Networks for Signal Processing XII (Proc. IEEE Workshop on Neural Networks for Signal Processing)*, Martigny, Switzerland, 2002, pp. 557-565.

[216] J. Eggert and E. Korner, "Sparse coding and NMF," in *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, 2004, pp. 2529-2533 vol.4.

[217] P. Hoyer. (2004). *Non-negative matrix factorization with sparseness constraints*. Available: http://arxiv.org/abs/cs.LG/0408058

[218] R. Tandon and S. Sra, "Sparse nonnegative matrix approximation: new formulations and algorithms," *Max Planck Institute for Biological Cybernetics,*2010.

[219] N. Hurley and S. Rickard, "Comparing Measures of Sparsity," *Information Theory, IEEE Transactions on,* vol. 55, pp. 4723-4741, 2009.

[220] T. Virtanen, "Sound source separation using sparse coding with temporal continuity objective," in *International Computer Music Conference, ICMC*, 2003.

[221] Z. Chen and A. Cichocki, "Nonnegative matrix factorization with temporal smoothness and/or spatial decorrelation constraints," Laboratory for Advanced Brain Signal Processing, RIKEN.2005.

[222] L. Hualiang, T. Adali, W. Wei, and D. Emge, "Non-Negative Matrix Factorization with Orthogonality Constraints for Chemical Agent Detection in Raman Spectra," in *Machine Learning for Signal Processing, 2005 IEEE Workshop on*, 2005, pp. 253-258.

[223] C. Seungjin, "Algorithms for orthogonal nonnegative matrix factorization," in *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, 2008, pp. 1828-1832.

[224] P. Smaragdis, "Convolutive Speech Bases and Their Application to Supervised Speech Separation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 1-12, 2007.

[225] T. Virtanen, "Separation of sound sources by convolutive sparse coding," in *Statistical and Perceptual Audio Processing (SAPA)*, 2004.

[226] W. Wenwu, "Squared Euclidean Distance Based Convolutive Non-Negative Matrix Factorization with Multiplicative Learning Rules For Audio Pattern Separation," in *Signal Processing and Information Technology, 2007 IEEE International Symposium on*, 2007, pp. 347-352.

[227] P. D. O'Grady and B. A. Pearlmutter, "Convolutive Non-Negative Matrix Factorisation with a Sparseness Constraint," in *Machine Learning for Signal Processing, 2006. Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on*, 2006, pp. 427-432.

[228] W. Wenwu, A. Cichocki, and J. A. Chambers, "A Multiplicative Algorithm for Convolutive Non-Negative Matrix Factorization Based on Squared Euclidean Distance," *Signal Processing, IEEE Transactions on,* vol. 57, pp. 2858-2864, 2009.

[229] D. FitzGerald, M. Cranitch, and E. Coyle, "Shifted 2D Non-negative Tensor Factorisation," in *Irish Signals and Systems Conference, 2006. IET*, 2006, pp. 509-513.

[230] C.-J. Lin, "Projected Gradient Methods for Nonnegative Matrix Factorization," *Neural Computation,* vol. 19, pp. 2756-2779, 2007/10/01 2007.

[231] J. Kim and H. Park, "Fast Nonnegative Matrix Factorization: An Active-Set-like method and comparisons," *Siam Journal on Scientific Computing,* vol. 33, pp. 3261-3281, 2011.

[232] A. Cichocki, R. Zdunek, and S. i. Amari, "Nonnegative Matrix and Tensor Factorization [Lecture Notes]," *Signal Processing Magazine, IEEE,* vol. 25, pp. 142-145, 2008.

[233] A. Cichocki and R. Zdunek. *NMFLAB – MATLAB Toolbox for Non-Negative Matrix Factorization,* . Available: http://www.bsp.brain.riken.jp/ICALAB/nmflab.html

[234] A. Cichocki, S.-I. Amari, R. Zdunek, R. Kompass, G. Hori, and Z. He, "Extended SMART Algorithms for Non-negative Matrix Factorization," in *Artificial Intelligence and Soft Computing, ICAISC 2006*, Zakopane, Poland, 2006.

[235] L. Chih-Jen, "On the Convergence of Multiplicative Update Algorithms for Nonnegative Matrix Factorization," *Neural Networks, IEEE Transactions on,* vol. 18, pp. 1589-1596, 2007.

[236] L. Finesso and P. Spreij, "Nonnegative matrix factorization and I-divergence alternating minimization," *Linear Algebra and its Applications,* vol. 416, pp. 270-287, 2006.

[237] R. Badeau, N. Bertin, and E. Vincent, "Stability Analysis of Multiplicative Update Algorithms and Application to Nonnegative Matrix Factorization," *Neural Networks, IEEE Transactions on,* vol. 21, pp. 1869-1881, 2010.

[238] E. Gonzalez and Y. Zhang, "Accelerating the Lee-Seung algorithm for nonnegative matrix factorization," Dept. Comput. & Appl. Math.

[239] A. Cichocki, R. Zdunek, S. Choi, R. Plemmons, and S.-I. Amari, "Novel Multi-layer Non-negative Tensor Factorization with Sparsity Constraints," presented at the Proceedings of the 8th international conference on Adaptive and Natural Computing Algorithms, Part II, Warsaw, Poland, 2007.

[240] M. N. Schmidt and M. Mørup, "Nonnegative matrix factor 2-d deconvolution for blind single chanel source separation," in *Symposium on Independent Component Analysis and Blind Signal Separation (ICA"2006)*, 2006.

[241] C. L. Lawson and R. J. Hanson, "Solving Least Squares Problems," *Society for Industrial Mathematics,* 1995.

[242] K. Jingu and P. Haesun, "Toward Faster Nonnegative Matrix Factorization: A New Algorithm and Comparisons," in *Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on*, 2008, pp. 353-362.

[243] D. Kim, S. Sra, and I. S. Dhillon, "Fast newton-type methods for the least squares nonnegative matrix approximation problem," in *Data Mining, Proceedings of SIAM Conference on*, 2007, pp. 343-354.

[244] R. Zdunek and A. Cichocki, "Fast nonnegative matrix factorization algorithms using projected gradient approaches for large-scale problems," *Intell. Neuroscience,* vol. 2008, pp. 1-13, 2008.

[245] A. Cichocki, R. Zdunek, and S.-i. Amari, "Hierarchical ALS algorithms for nonnegative matrix and 3D tensor factorization," presented at the Proceedings of the 7th international conference on Independent component analysis and signal separation, London, UK, 2007.

[246] N. Gillis and F. Glineur, "Accelerated multiplicative updates and hierarchical als algorithms for nonnegative matrix factorization," Université catholique de Louvain, Center for Operations Research and Econometrics (CORE)Jul 2011.

[247] R. Zdunek and A. Cichocki, "Nonnegative matrix factorization with constrained second-order optimization," *Signal Process.,* vol. 87, pp. 1904-1916, 2007.

[248] A. Cichocki, R. Zdunek, A. H. Phan, and S.-I. Amari, "Quasi-Newton Algorithms for Nonnegative Matrix Factorization," in *Nonnegative Matrix and Tensor Factorizations*, ed: John Wiley & Sons, Ltd, 2009, pp. 295-335.
[249] A. Cichocki and R. Zdunek, "Multilayer nonnegative matrix factorisation," *Electronics Letters,* vol. 42, pp. 947-948, 2006.
[250] A. Cichocki and R. Zdunek, "Regularized Alternating Least Squares Algorithms for Non-negative Matrix/Tensor Factorization," in *Proceedings of the 4th international symposium on Neural Networks: Advances in Neural Networks, Part III*, Nanjing, China, 2007, pp. 793-802.
[251] C. Févotte, N. Bertin, and J.-L. Durrieu, "Nonnegative Matrix Factorization with the Itakura-Saito Divergence: With Application to Music Analysis," *Neural Computation,* vol. 21, pp. 793-830, 2009/03/01 2008.
[252] M. N. Schmidt, O. Winther, and L. K. Hansen, "Bayesian non-negative matrix factorization," in *Independent Component Analysis and Signal Separation*, 2009.
[253] R. Albright, J. Cox, D. Duling, A. N. Langville, and C. D. Meyer, "Algorithms, initializations, and convergence for the nonnegative matrix factorization," in *12th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2006.
[254] S. Wild, J. Curry, and A. Dougherty, "Improving non-negative matrix factorizations through structured initialization," *Pattern Recognition,* vol. 37, pp. 2217-2232, 2004.
[255] A. Langville, C. Meyer, and R. Albright, "Initializations for the nonnegative matrix factorization," in *Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.
[256] Z. Zheng, J. Yang, and Y. Zhu, "Initialization enhancer for non-negative matrix factorization," *Engineering Applications of Artificial Intelligence,* vol. 20, pp. 101-110, 2007.
[257] S. Yu, Y. Zhang, W. Liu, N. Zhao, X. Xiao, and G. Yin, "A novel initialization method for nonnegative matrix factorization and its application in component recognition with three-dimensional fluorescence spectra," *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy,* vol. 86, pp. 315-319, 2012.
[258] C. Boutsidis and E. Gallopoulos, "SVD based initialization: A head start for nonnegative matrix factorization," *Pattern Recogn.,* vol. 41, pp. 1350-1362, 2008.
[259] M. Rezaei and R. Boostani, "Using Genetic algorithm to enhance nonnegative matrix factorization initialization," in *Machine Vision and Image Processing (MVIP), 2010 6th Iranian*, 2010, pp. 1-5.
[260] P. D. O'Grady and B. A. Pearlmutter, "Discovering speech phones using convolutive non-negative matrix factorisation with a sparseness constraint," *Neurocomputing,* vol. 72, pp. 88-101, Dec 2008.
[261] H. Kim and H. Park, "Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis," *Bioinformatics,* vol. 23, pp. 1495-1502, June 15, 2007 2007.
[262] N. Mohammadiha and A. Leijon, "Nonnegative matrix factorization using projected gradient algorithms with sparseness constraints," in *Signal Processing and Information Technology (ISSPIT), 2009 IEEE International Symposium on*, 2009, pp. 418-423.
[263] Cherry and E. Colin, "Some Experiments on the Recognition of Speech, with One and with Two Ears," *J. Acoust. Soc. Am.,* vol. 25, pp. 975-979, 1953.
[264] A. S. Bregman, *Auditory scene analysis: the perceptual organization of sound*: MIT Press, 1994.
[265] L. W. DeLiang and G. J. Brown, "Separation of speech from interfering sounds based on oscillatory correlation," *Neural Networks, IEEE Transactions on,* vol. 10, pp. 684-697, 1999.
[266] P. Mowlaee, "New strategies for single-channel speech separation," Ph.D., Aalborg University, Denmark, 2010.
[267] F. R. Bach and M. I. Jordan, "Learning Spectral Clustering, With Application To Speech Separation," *J. Mach. Learn. Res.,* vol. 7, pp. 1963-2001, 2006.
[268] F. R. Bach and M. Jordon, "Blind one-microphone speech separation: A spectral learning approach," in *Adv. Neural Inf. Process. Syst. (NIPS)*, 2004.
[269] R. J. Weiss and D. P. W. Ellis, "Estimating Single-Channel Source Separation Masks: Relevance Vector Machine Classifiers vs. Pitch-Based Masking " in *Workshop on Statistical and Perceptual Audition SAPA-06 (Oct 2006)*, 2006, pp. 31-36.
[270] M. H. Radfar, R. M. Dansereau, and A. Sayadiyan, "Speaker-independent model-based single channel speech separation," *Neurocomputing,* vol. 72, pp. 71-78, 2008.

[271] M. Radfar, R. Dansereau, and A. Sayadiyan, "A maximum likelihood estimation of vocal-tract-related filter characteristics for single channel speech separation," *EURASIP J. Audio Speech Music Process.,* vol. 2007, pp. 6-6, 2007.

[272] M. Stark, M. Wohlmayr, and F. Pernkopf, "Source-Filter Based Single-Channel Speech Separation Using Pitch Information," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 19, pp. 242-255, 2011.

[273] M. H. Radfar, R. M. Dansereau, and A. Sayadiyan, "Monaural speech segregation based on fusion of source-driven with model-driven techniques," *Speech Communication,* vol. 49, pp. 464-476, 2007.

[274] M. Stark, M. Wohlmayr, and F. Pernkopf, "Single Channel Speech Separation Using Source-Filter Representation," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, 2010, pp. 826-829.

[275] D. Ellis, "Model-based scene analysis," in *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*, D. Wang and G. Brown, Eds., ed New York, NY, USA,: Wiley/IEEE Press, 2006.

[276] D. Wang and L. Jae, "The unimportance of phase in speech enhancement," *Acoustics, Speech and Signal Processing, IEEE Transactions on,* vol. 30, pp. 679-681, 1982.

[277] H. Pobloth and W. B. Kleijn, "On phase perception in speech," in *Acoustics, Speech, and Signal Processing, 1999. ICASSP '99. Proceedings., 1999 IEEE International Conference on*, 1999, pp. 29-32 vol.1.

[278] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *Acoustics, Speech and Signal Processing, IEEE Transactions on,* vol. 32, pp. 1109-1121, 1984.

[279] K. K. Paliwal and L. Alsteris, "Usefulness of phase spectrum in human speech perception," in *Eur. Conf. Speech Communication and Technology (Eurospeech)*, 2003, pp. 2117 - 2120

[280] S. Guangji, M. M. Shanechi, and P. Aarabi, "On the importance of phase in human speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 14, pp. 1867-1874, 2006.

[281] R. M. Parry and I. Essa, "Incorporating Phase Information for Source Separation via Spectrogram Factorization," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, 2007, pp. II-661-II-664.

[282] R. Parry and I. Essa, "Phase-Aware Non-negative Spectrogram Factorization," in *Independent Component Analysis and Signal Separation*. vol. 4666, M. Davies, C. James, S. Abdallah, and M. Plumbley, Eds., ed: Springer Berlin / Heidelberg, 2007, pp. 536-543.

[283] H. Kameoka, N. Ono, K. Kashino, and S. Sagayama, "Complex NMF: A new sparse representation for acoustic signals," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 2009, pp. 3437-3440.

[284] B. J. King and L. Atlas, "Single-Channel Source Separation Using Complex Matrix Factorization," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 19, pp. 2591-2597, 2011.

[285] J. Le Roux, E. Vincent, Y. Mizuno, H. Kameoka, N. Ono, and S. Sagayama, "Consistent Wiener Filtering: Generalized Time-Frequency Masking Respecting Spectrogram Consistency," in *Latent Variable Analysis and Signal Separation*. vol. 6365, V. Vigneron, V. Zarzoso, E. Moreau, R. Gribonval, and E. Vincent, Eds., ed: Springer Berlin / Heidelberg, 2010, pp. 89-96.

[286] L. Benaroya and F. Bimbot, "Wiener based source separation with hmm/gmm using a single sensor," in *4th Int. Symp. on Independent Component Anal. and Blind Signal Separation (2003)* 2003.

[287] C. Jingdong, J. Benesty, H. Yiteng, and S. Doclo, "New insights into the noise reduction Wiener filter," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 14, pp. 1218-1234, 2006.

[288] D. P. W. Ellis and R. J. Weiss, "Model-Based Monaural Source Separation Using a Vector-Quantized Phase-Vocoder Representation," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, 2006, pp. V-V.

[289] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *Acoustics, Speech and Signal Processing, IEEE Transactions on,* vol. 27, pp. 113-120, 1979.

[290] M. H. Radfar, A. H. Banihashemi, R. M. Dansereau, and A. Sayadiyan, "Nonlinear minimum mean square error estimator for mixture-maximisation approximation," *Electronics Letters,* vol. 42, pp. 724-725, 2006.

[291] S. T. Roweis, "Factorial models and refiltering for speech separation and denoising," in *Speech Communication and Technology, European Conference on, EUROSPEECH*, 2003, pp. 1009-1012.
[292] D. Wang, "On Ideal Binary Mask As the Computational Goal of Auditory Scene Analysis," in *Speech Separation by Humans and Machines*, P. Divenyi, Ed., ed: Springer US, 2005, pp. 181-197.
[293] S. A. Raki, S. Makino, H. Sawada, and R. Mukai, "Reducing musical noise by a fine-shift overlap-add method applied to source separation using a time-frequency mask," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, 2005, pp. iii/81-iii/84 Vol. 3.
[294] A. M. Reddy and B. Raj, "Soft Mask Methods for Single-Channel Speaker Separation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 1766-1776, 2007.
[295] A. M. Reddy and B. Raj, "Soft mask estimation for single channel speaker separation," in *ISCA Tutorial Res. Workshop Statist. Percept. Audio Process* 2004.
[296] M. H. Radfar and R. M. Dansereau, "Single-Channel Speech Separation Using Soft Mask Filtering," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 2299-2310, 2007.
[297] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *International Conference on Spoken Language Processing, INTERSPEECH*, Pittsburgh, USA, 2006.
[298] P. Mowlaee, A. Sayadiyan, and H. Sheikhzadeh, "Evaluating single-channel speech separation performance in transform-domain," *Journal of Zhejiang University - Science C,* vol. 11, pp. 160-174, 2010.
[299] Y. Litvin and I. Cohen, "Single-channel source separation of audio signals using Bark Scale Wavelet Packet Decomposition," in *Machine Learning for Signal Processing, 2009. MLSP 2009. IEEE International Workshop on*, 2009, pp. 1-4.
[300] L. Benaroya, R. Blouet, C. Fevotte, and I. Cohen, " Single sensor source separation using multiple-window STFT representation " in *International Workshop on Acoustic Echo and Noise Control (IWAENC'06)* Paris, France, 2006.
[301] M. N. Schmidt, "Single-channel source separation using non-negative matrix factorization," Technical University of Denmark, 2008.
[302] G. Rapaport, "Codebook-based Single Channel Blind Source Separation of Audio Signals.," M.Sc, Department of Electrical Engineering, Israel Institute of Technology,Technion, Israel Institute of Technology, 2011.
[303] A. Ozerov, P. Philippe, F. Bimbot, and R. Gribonval, "Adaptation of Bayesian Models for Single-Channel Source Separation and its Application to Voice/Music Separation in Popular Songs," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 1564-1578, 2007.
[304] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society. Series B (Methodological),* vol. 39, pp. 1-38, 1977.
[305] Z. Ghahramani and M. I. Jordan, "Factorial Hidden Markov Models," *Machine Learning,* vol. 29, pp. 245-273, 1997.
[306] R. M. Neal, "Probabilistic inference using markov chain monte carlo methods," Dept. of Computer Science, University of Toronto, 1993.
[307] L. Benaroya, L. M. Donagh, F. Bimbot, and R. Gribonval, "Non negative sparse representation for Wiener based source separation with a single sensor," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*, 2003, pp. VI-613-16 vol.6.
[308] A. M. Reddy and B. Raj, "A minimum mean squared error estimator for single channel speaker separation " in *Interspeech,*, 2004, pp. 2445–2448.
[309] T. Beierholm, B. D. Pedersen, and O. Winther, "Low complexity Bayesian single channel source separation," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on*, 2004, pp. V-529-32 vol.5.
[310] M. Radfar, R. Dansereau, and W. Y. Chan, "Monaural Speech Separation Based on Gain Adapted Minimum Mean Square Error Estimation," *Journal of Signal Processing Systems,* vol. 61, pp. 21-37, 2010.

[311] A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot, "One microphone singing voice separation using source-adapted models," in *Applications of Signal Processing to Audio and Acoustics, 2005. IEEE Workshop on*, 2005, pp. 90-93.

[312] A. Ozerov, E. Vincent, and F. Bimbot, "A General Flexible Framework for the Handling of Prior Information in Audio Source Separation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. PP, pp. 1-1, 2011.

[313] M. J. Reyes-Gomez, D. P. W. Ellis, and N. Jojic, "Multiband audio modeling for single-channel acoustic source separation," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on*, 2004, pp. V-641-4 vol.5.

[314] M. H. Radfar, W. Wong, R. M. Dansereau, and W. Y. Chan, "Scaled factorial hidden Markov models: A new technique for compensating gain differences in model-based single channel speech separation," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, 2010, pp. 1918-1921.

[315] M. H. Radfar and R. M. Dansereau, "Single channel speech separation using maximum a posteriori estimation," in *Spoken Language Processing, ISCA International Conference on (INTERSPEECH)* 2007.

[316] M. Cooke, J. R. Hershey, and S. J. Rennie, "Monaural speech separation and recognition challenge," *Computer Speech & Language,* vol. 24, pp. 1-15, 2010.

[317] J. R. Hershey, T. T. Kristjansson, S. J. Rennie, and P. A. Olsen, "Single channel speech separation using factorial dynamics.," in *Advances in Neural Information Processing Systems (NIPS)*, Vancouver, British Columbia, Canada., 2006.

[318] J. Hershey, T. Kristjansson, S. Rennie, and P. Olsen, "Single channel speech separation using layered hidden Markov models," in *NIPS*, 2006, pp. 593-600.

[319] J. R. Hershey, S. J. Rennie, P. A. Olsen, and T. T. Kristjansson, "Super-human multi-talker speech recognition: A graphical modeling approach," *Computer Speech & Language,* vol. 24, pp. 45-66, 2010.

[320] T. Kristjansson, J. R. Hershey, P. A. Olsen, S. Rennie, and R. Gopinath, "Super-human multi-talker speech recognition: The IBM 2006 speech separation challenge system," in *ICSLP*, 2006.

[321] T. Virtanen, "Speech recognition using factorial hidden Markov models for separation in the feature space," in *Interspeech06*, 2006, pp. 173-176.

[322] R. Weiss and D. Ellis, "Monaural Speech Separation using Source-Adapted Models," in *Proceedings of the {IEEE} International Conference on Acoustics, Speech, and Signal Processing*, 2007, pp. 114-117.

[323] R. J. Weiss and D. Ellis, "A variational EM algorithm for learning eigenvoice parameters in mixed signals," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 2009, pp. 113-116.

[324] N. H. Pontoppidan and M. Dyrholm, "Fast Monaural Separation of Speech," in *Audio Engineering Society Conference: 23rd International Conference: Signal Processing in Audio Recording and Reproduction*, 2003.

[325] M. Stark and F. Pernkopf, "On optimizing the computational complexity for VQ-based single channel source separation," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, 2010, pp. 237-240.

[326] P. Li, Y. Guan, S. Wang, B. Xu, and W. Liu, "Monaural speech separation based on MAXVQ and CASA for robust speech recognition," *Computer Speech & Language,* vol. 24, pp. 30-44, 2010.

[327] J. Nix, M. Kleinschmidt, and V. Hohmann, "Computational auditory scene analysis by using statistics of high-dimensional speech dynamics and sound source direction," in *Eurospeech*, 2003, pp. 1441-1444.

[328] S. Kammi and M. R. Karami, "An efficient VQ-based method for monaural speech separation," in *Systems, Signals and Image Processing (IWSSIP), 2011 18th International Conference on*, 2011, pp. 1-4.

[329] P. Mowlaee, M. G. Christensen, and S. H. Jensen, "New Results on Single-Channel Speech Separation Using Sinusoidal Modeling," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 19, pp. 1265-1277, 2011.

[330] M. G. Christensen and P. Mowlaee, "A new metric for VQ-based speech enhancement and separation," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 4764-4767.

[331] P. Mowlaee and A. Saiadiyan, "Model-based Monaural Sound Separation by Split-VQ of Sinusoidal Parameters," in *European Conference on Signal Processing 2008*, 2008.

[332] P. Mowlaee, A. Sayadian, M. Sheikhan, and M. Fallah, "Single-channel music/speech separation using non-linear masks," in *Telecommunications, 2008. IST 2008. International Symposium on*, 2008, pp. 782-786.
[333] M. H. Radfar, R. M. Dansereau, and A. Sayadiyan, "Performance Evaluation of Three Features for Model-Based Single Channel Speech Separation Problem," in *INTERSPEECH-2006*, 2006.
[334] M. H. Radfar, R. M. Dansereau, and A. Sayadiyan, "A Novel Low Complexity VQ-Based Single Channel Speech Separation Technique," in *Signal Processing and Information Technology, 2006 IEEE International Symposium on*, 2006, pp. 572-577.
[335] M. H. Radfar, R. M. Dansereau, and A. Sayadiyan, "On the Choice of Window Size in Model-Based Single Channel Speech Separation," in *Electrical and Computer Engineering, 2006. CCECE '06. Canadian Conference on*, 2006, pp. 298-301.
[336] S. Srinivasan, J. Samuelsson, and W. Kleijn, "Speech enhancement using a-priori information with classified noise codebooks " in *EUSIPCO* 2004 pp. 1461–1464.
[337] S. Srinivasan, J. Samuelsson, and W. B. Kleijn, "Codebook driven short-term predictor parameter estimation for speech enhancement," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 14, pp. 163-176, 2006.
[338] S. Srinivasan, J. Samuelsson, and W. B. Kleijn, "Codebook-Based Bayesian Speech Enhancement," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, 2005, pp. 1077-1080.
[339] S. Srinivasan, J. Samuelsson, and W. B. Kleijn, "Codebook-Based Bayesian Speech Enhancement for Nonstationary Environments," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 15, pp. 441-452, 2007.
[340] R. Blouet, G. Rapaport, and C. Fevotte, "Evaluation of several strategies for single sensor speech/music separation," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 37-40.
[341] R. Balan, A. Jourjine, and J. Rosca, "Ar process and sources can be reconstructed from degenerate mixtures," in *Independent Component Analysis and Blind Signal Separation, International Conference on (ICA)* 1999, pp. 467–472.
[342] A. Schutz and D. Slock, "Blind audio source separation using short+long term AR source models and spectrum matching," in *Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE), 2011 IEEE*, 2011, pp. 112-115.
[343] G. Mysore, P. Smaragdis, and B. Raj, "Non-negative Hidden Markov Modeling of Audio with Application to Source Separation," in *Latent Variable Analysis and Signal Separation*. vol. 6365, V. Vigneron, V. Zarzoso, E. Moreau, R. Gribonval, and E. Vincent, Eds., ed: Springer Berlin / Heidelberg, 2010, pp. 140-148.
[344] B. Wang and M. D. Plumbley, "Single Channel Audio Separation by Non-negative Matrix Factorization," in *Digital Music Research Network One-day Workshop 2006 (DMRN+1)*, London, UK, 2006.
[345] B. Wang and M. D. Plumbley, "Investigating single channel audio source separation methods based on nonnegative matrix factorization," in *CA Research Network Int'l Workshop*, 2006, pp. 17-20.
[346] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 14, pp. 1462-1469, 2006.
[347] E. Vincent, R. Gribonval, and M. D. Plumbley, "Oracle estimators for the benchmarking of source separation algorithms," *Signal Process.,* vol. 87, pp. 1933-1950, 2007.
[348] T. Virtanen, "Monaural sound source separation by perceptually weighted nonnegative matrix factorization ", Tampere University of Technology, 2007.
[349] T. Virtanen, A. T. Cemgil, and S. Godsill, "Bayesian extensions to non-negative matrix factorisation for audio signal modelling," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 1825-1828.
[350] S. Kirbiz, A. T. Cemgil, Gu, x, and B. nsel, "Bayesian Inference for Nonnegative Matrix Factor Deconvolution Models," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, 2010, pp. 2812-2815.
[351] S. Kırbız and B. Günsel, "A Perceptually Enhanced Blind Single-Channel Audio Source Separation by Non-negative Matrix Factorization," in *18th European Signal Processing Conference (EUSIPCO) 2010,*, Aalborg, Denmark, 2010.

[352] B. Raj and P. Smaragdis, "Latent variable decomposition of spectrograms for single channel speaker separation," in *Applications of Signal Processing to Audio and Acoustics, 2005. IEEE Workshop on*, 2005, pp. 17-20.

[353] P. Smaragdis, B. Raj, and M. Shashanka, "Supervised and Semi-supervised Separation of Sounds from Single-Channel Mixtures," in *Independent Component Analysis and Signal Separation*. vol. 4666, M. Davies, C. James, S. Abdallah, and M. Plumbley, Eds., ed: Springer Berlin / Heidelberg, 2007, pp. 414-421.

[354] M. Shashanka, B. Raj, and P. Smaragdis, "Probabilistic Latent Variable Models as Nonnegative Factorizations," *Computational Intelligence and Neuroscience,* vol. 2008, 2008.

[355] P. Smaragdis, B. Raj, and M. Shashanka, "Sparse and shift-invariant feature extraction from non-negative data," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 2069-2072.

[356] M. V. S. Shashanka, B. Raj, and P. Smaragdis, "Sparse Overcomplete Decomposition for Single Channel Speaker Separation," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, 2007, pp. II-641-II-644.

[357] P. Smaragdis, M. Shashanka, and B. Raj, "A Sparse Non-Parametric Approach for Single Channel Separation of Known Sounds," in *Advances in Neural Information Processing Systems 22*, Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, Eds., ed, 2009, pp. 1705-1713.

[358] A. Ozerov, C. Fevotte, and M. Charbit, "Factorial Scaled Hidden Markov Model for polyphonic audio representation and source separation," in *Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA '09. IEEE Workshop on*, 2009, pp. 121-124.

[359] M. N. Schmidt, J. Larsen, and H. Fu-Tien, "Wind Noise Reduction using Non-Negative Sparse Coding," in *Machine Learning for Signal Processing, 2007 IEEE Workshop on*, 2007, pp. 431-436.

[360] X. Lai, S. Li, and J. Yang, "Convolutive Sparse Non-negative Matrix Factorization for windy speech," in *Signal Processing (ICSP), 2010 IEEE 10th International Conference on*, 2010, pp. 494-497.

[361] K. W. Wilson, B. Raj, P. Smaragdis, and A. Divakaran, "Speech denoising using nonnegative matrix factorization with priors," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 4029-4032.

[362] K. W. Wilson, R. Bhiksha, and P. Smaragdis, "Regularized non-negative matrix factorization with temporal dependencies for speech denoising," in *INTERSPEECH-2008,* , 2008, pp. 411-414.

[363] C. D. Sigg, T. Dikk, and J. M. Buhmann, "Speech enhancement with sparse coding in learned dictionaries," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, 2010, pp. 4758-4761.

[364] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on],* vol. 54, pp. 4311-4322, 2006.

[365] M. N. Schmidt and J. Larsen, "Reduction of non-stationary noise using a non-negative latent variable decomposition," in *Machine Learning for Signal Processing, 2008. MLSP 2008. IEEE Workshop on*, 2008, pp. 486-491.

[366] J. So-Young, K. Kyuhong, J. Jae-Hoon, and O. Kwang-Cheol, "Semi-blind disjoint non-negative matrix factorization for extracting target source from single channel noisy mixture," in *Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA '09. IEEE Workshop on*, 2009, pp. 73-76.

[367] M. Casey and A. Westner, "Separation of Mixed Audio Sources by Independent Subspace Analysis," in *International Computer Music Conference, ICMA*, Berlin, 2000.

[368] J. C. Brown and P. Smaragdis, "Independent component analysis for automatic note extraction from musical trills," *J. Acoust. Soc. Amer, ,* vol. 115, pp. 2295-2306, 2004.

[369] D. FitzGerald, "Automatic drum transcription and source separation," Dublin Inst. Technol., Dublin, Ireland, , 2004.

[370] C. Uhle, C. Dittmar, and T. Sporer, "Extraction of drum tracks from polyphonic music using independent subspace analysis " in *Int. Symp. Independent Compon. Anal. Blind Signal Separation*, Nara, Japan, , pp. 843–848.

[371] D. FitzGerald, "Automatic drum transcription and source separation," Ph.D, Dublin Inst. Technol., Dublin, Ireland, , 2004.

[372] M. E. Davies and C. J. James, "Source separation using single channel ICA," *Signal Processing,* vol. 87, pp. 1819-1832, 2007.
[373] G.-J. Jang and T.-W. Lee, "A Maximum likelihood approach to single-channel source separation," *Journal of Machine Learning Research,* vol. 4, pp. 1365-1392, 2003.
[374] G.-J. Jang and T.-W. Lee, "A probabilistic approach to single channel source separation," in *Neural Information Processing Systems, Advances in (NIPS).* , 2003.
[375] G. J. Jang, L. Te-Won, and O. Yung-Hwan, "Single-channel signal separation using time-domain basis functions," *Signal Processing Letters, IEEE,* vol. 10, pp. 168-171, 2003.
[376] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature,* vol. 381, pp. 607-609, 1996.
[377] B. A. Pearlmutter and R. K. Olsson, "Linear Program Differentiation for Single-Channel Speech Separation," in *Machine Learning for Signal Processing, 2006. Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on*, 2006, pp. 421-426.
[378] H. Asari, B. A. Pearlmutter, and A. M. Zador, "Sparse Representations for the Cocktail Party Problem," *The Journal of Neuroscience,* vol. 26, pp. 7477-7490, July 12, 2006 2006.
[379] W. M. Fisher, G. R. Doddington, and K. M. Goudie-Marshall, "The DARPA Speech Recognition Research Database: Specifications and Status," presented at the Proceedings of DARPA Workshop on Speech Recognition, 1986.
[380] J. Y. C. Wen, N. D. Gaubitch, A. P. Habets, T. Myatt, and P. A. Naylor, "Evaluation of speech dereverberation algorithms using the MARDY database," presented at the International Workshop on Acoustic Echo and Noise Control (IWAENC), Paris, France, 2006.
[381] P. Smaragdis, "From learning music to learning to separate," in *Forum Acusticum*, 2005.
[382] P. O'Grady, "Sparse Separation of Underdetermined Speech Mixtures," Ph.D., Hamilton Institute, National University of Ireland Maynooth, 2007.
[383] P. Yun-Sik and C. Joon-Hyuk, "Frequency Domain Acoustic Echo Suppression Based on Soft Decision," *Signal Processing Letters, IEEE,* vol. 16, pp. 53-56, 2009.
[384] A. Gray, Jr. and J. Markel, "Distance measures for speech processing," *Acoustics, Speech and Signal Processing, IEEE Transactions on,* vol. 24, pp. 380-391, 1976.
[385] C. Févotte, "BSS_EVAL toolbox user guide," R. Gribonval, Ed., Tech. Rep. 1706 ed. IRISA, Rennes, France, : [Online]. Available: http://www.irisa.fr/metiss/bss_eval. , 2005.
[386] M. Miyoshi and Y. Kaneda, "Inverse filtering of room acoustics," *Acoustics, Speech and Signal Processing, IEEE Transactions on,* vol. 36, pp. 145-152, 1988.
[387] P. A. Nelson, F. Orduna-Bustamante, and H. Hamada, "Inverse filter design and equalization zones in multichannel sound reproduction," *Speech and Audio Processing, IEEE Transactions on,* vol. 3, pp. 185-192, 1995.
[388] N. D. Gaubitch and P. A. Naylor, "Equalization of Multichannel Acoustic Systems in Oversampled Subbands," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 17, pp. 1061-1070, 2009.
[389] J. R. Hopgood and P. J. W. Rayner, "Blind single channel deconvolution using nonstationary signal processing," *Speech and Audio Processing, IEEE Transactions on,* vol. 11, pp. 476-488, 2003.
[390] C. Evers and J. R. Hopgood, "Parametric modelling for single-channel blind dereverberation of speech from a moving speaker," *Signal Processing, IET,* vol. 2, pp. 59-74, 2008.
[391] S. Mosayyebpour, A. Sayyadiyan, M. Zareian, and A. Shahbazi, "Single Channel Inverse Filtering of Room Impulse Response by Maximizing Skewness of LP Residual," in *Signal Acquisition and Processing, 2010. ICSAP '10. International Conference on*, 2010, pp. 130-134.
[392] M. Karjalainen, T. Paatero, J. N. Mourjopoulos, and P. D. Hatziantoniou, "About room response equalization and dereverberation," in *Applications of Signal Processing to Audio and Acoustics, 2005. IEEE Workshop on*, 2005, pp. 183-186.
[393] S. T. Neely and J. B. Allen, "Invertibility of a room impulse response," *J. Acoust. Soc. Am. ,* vol. 66, pp. 65-169, 1979.
[394] J. N. Mourjopoulos, "Digital Equalization of Room Acoustics," *J. Audio Eng. Soc,* vol. 42, pp. 884-900, 1994.
[395] J. Mourjopoulos, "On the variation and Invertibility of Room Impulse Response Functions," *Journal of Sound and Vibration,* vol. 102, pp. 217-228, 1985.
[396] J. Vanderkooy, "Aspects of MLS Measuring Systems," *J. Audio Eng. Soc,* vol. 42, pp. 219-231, 1994.

[397] J. Usher, "Acoustic Impulse Response Measurement Using Speech and Music Signals," in *Audio Engineering Society Convention 128*, 2010.
[398] D. Preis, "Phase Distortion and Phase Equalization in Audio Signal Processing-A Tutorial Review," *J. Audio Eng. Soc,* vol. 30, pp. 774--794, 1982.
[399] J. N. Mourjopoulos, "Comments on 'Analysis of Traditional and Reverberation-Reducing Methods of Room Equalization'," *J. Audio Eng. Soc,* vol. 51, pp. 1186--1188, 2003.
[400] J. Mourjopoulos, P. Clarkson, and J. Hammond, "A comparative study of least-squares and homomorphic techniques for the inversion of mixed phase signals," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '82.*, 1982, pp. 1858-1861.
[401] L. D. Fielder, "Analysis of Traditional and Reverberation-Reducing Methods of Room Equalization," *J. Audio Eng. Soc,* vol. 51, pp. 3--261, 2003.
[402] Y. Haneda, S. Makino, and Y. Kaneda, "Multiple-point equalization of room transfer functions by using common acoustical poles," *Speech and Audio Processing, IEEE Transactions on,* vol. 5, pp. 325-333, 1997.
[403] P. D. Hatziantoniou and J. N. Mourjopoulos, "Errors in Real-Time Room Acoustics Dereverberation," *J. Audio Eng. Soc,* vol. 52, pp. 883--899, 2004.
[404] S. G. Norcross, G. A. Soulodre, and M. C. Lavoie, "Subjective Investigations of Inverse Filtering," *J. Audio Eng. Soc,* vol. 52, pp. 1003-1028, 2004.
[405] B. D. Radlovic, R. C. Williamson, and R. A. Kennedy, "Equalization in an acoustic reverberant environment: robustness results," *Speech and Audio Processing, IEEE Transactions on,* vol. 8, pp. 311-319, 2000.
[406] A. V. Oppenheim and R. W. Schafer, *Digital signal processing*. Englewood Cliffs, N.J.: Prentice-Hall, 1975.
[407] J. O. Smith, *Introduction to Digital Filters with Audio Applications*: \htmladdnormallink\texttthttp:http://ccrma.stanford.edu1 jos/filters/\texttt/\-ccrma.stanford.edu/\-\~jos/\-filters/, 2011.
[408] A. Maamar, I. Kale, A. Krukowski, and B. Daoud, "Partial Equalization of Non-Minimum-Phase Impulse Responses," *EURASIP Journal on Applied Signal Processing,* vol. 2006, 2006.
[409] R. A. Kennedy and B. D. Radlovic, "Iterative cepstrum-based approach for speech dereverberation," in *Signal Processing and Its Applications, 1999. ISSPA '99. Proceedings of the Fifth International Symposium on*, 1999, pp. 55-58 vol.1.
[410] P. D. Hatziantoniou and J. N. Mourjopoulos, "Results for Room Acoustics Equalisation Based on Smoothed Responses," in *Audio Engineering Society Convention 114*, 2003.
[411] P. D. Hatziantoniou and J. N. Mourjopoulos, "Generalized Fractional-Octave Smoothing of Audio and Acoustic Responses," *J. Audio Eng. Soc,* vol. 48, pp. 259-280, 2000.
[412] P. M. Mourjopoulos J., "Pole and Zero Modeling of the Room Transfer Function," *Journal of Sound and Vibration,* vol. 46(2), pp. 281-302, 1991.
[413] B. Bank, "Audio Equalization with Fixed-Pole Parallel Filters: An Efficient Alternative to Complex Smoothing," in *Audio Engineering Society Convention 128*, 2010.
[414] B. Bank, "Warped IIR Filter Design with Custom Warping Profiles and Its Application to Room Response Modeling and Equalization," in *Audio Engineering Society Convention 130*, 2011.
[415] M. Karjalainen and T. Paatero, "Equalization of Loudspeaker and Room Responses Using Kautz Filters: Direct Least Squares Design," *EURASIP Journal on Advances in Signal Processing,* vol. 2007, 2007.
[416] M. Karjalainen and T. Paatero, "Equalization of Audio Systems using Kautz Filters with Log-like Frequency Resolution," in *Audio Engineering Society Convention 120*, 2006.
[417] M. Karjalainen, A. Makivirta, P. Antsalo, and V. Valimaki, "Low-Frequency Modal Equalization of Loudspeaker-Room Responses," in *Audio Engineering Society Convention 111*, 2001.
[418] F. E. Toole and S. E. Olive, "The Perception of Sound Coloration Due to Resonances in Loudspeakers and Other Audio Components," in *81st Convention of the Audio Engineering Society* 1986, p. 231.
[419] F. E. Toole and S. E. Olive, "The Modification of Timbre by Resonances: Perception and Measurement," in *83rd Convention of the Audio Engineering Society*, 1987
[420] A. Härmä, M. Karjalainen, L. Savioja, V. Välimäki, U. K. Laine, and J. Huopaniemi, "Frequency-Warped Signal Processing for Audio Applications," in *Audio Engineering Society Convention 108*, 2000.

[421] M. Karjalainen, P. Antsalo, and A. Makivirta, "Modal Equalization by Temporal Shaping of Room Response," in *Audio Engineering Society proceedings: 23rd International proceedings: Signal Processing in Audio Recording and Reproduction*, 2003.

[422] P. Antsalo, M. Karjalainen, A. Makivirta, and V. Valimaki, "Comparison of Modal Equalizer Design Methods," in *Audio Engineering Society Convention 114*, 2003.

[423] S. Bharitkar and C. Kyriakakis, "A comparison between multi-channel audio equalization filters using warping," in *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, 2003, pp. 63-66.

[424] P. Rubak and L. G. Johansen, "Design and Evaluation of Digital Filters Applied to Loudspeaker/Room Equalization," in *Audio Engineering Society Convention 108*, 2000.

[425] O. Kirkeby and P. A. Nelson, "Digital Filter Design for Inversion Problems in Sound Reproduction," *J. Audio Eng. Soc,* vol. 47, pp. 583-595, 1999.

[426] A. Mertins, M. Tiemin, and M. Kallinger, "Room Impulse Response Shortening/Reshaping With Infinity- and *p*-Norm Optimization," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 18, pp. 249-259, 2010.

[427] M. Kallinger and A. Mertins, "Room Impulse Response Shortening by Channel Shortening Concepts," in *Signals, Systems and Computers, 2005. Conference Record of the Thirty-Ninth Asilomar Conference on*, 2005, pp. 898-902.

[428] A. Farina and E. Ugolotti, "Spatial Equalization of Sound Systems in Cars," in *Audio Engineering Society proceedings: 15th International proceedings: Audio, Acoustics & Small Spaces*, 1998.

[429] J. S. Abel and J. O. Smith, "Robust Design of Very High-Order Allpass Dispersion Filters," in *Proc. of the Int. Conf. on Digital Audio Effects (DAFx-06)*, ed. Montreal, Quebec, Canada, 2006, pp. 13-18.

[430] M. Karjalainen, P. Ansalo, A. Mäkivirta, T. Peltonen, and V. Välimäki, "Estimation of Modal Decay Parameters from Noisy Response Measurements," *J. Audio Eng. Soc,* vol. 50, pp. 867-878, 2002.

[431] A. Mäkivirta, P. Antsalo, M. Karjalainen, and V. Välimäki, "Modal Equalization of Loudspeaker - Room Responses at Low Frequencies," *J. Audio Eng. Soc,* vol. 51, pp. 324-343, 2003.

[432] J. D. Reiss, "Design of Audio Parametric Equalizer Filters Directly in the Digital Domain," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 19, pp. 1843-1848, 2011.

[433] P. Smaragdis and B. Raj, "Example-Driven Bandwidth Expansion," in *Applications of Signal Processing to Audio and Acoustics, 2007 IEEE Workshop on*, 2007, pp. 135-138.

[434] Y. Avargel and I. Cohen, "On Multiplicative Transfer Function Approximation in the Short-Time Fourier Transform Domain," *Signal Processing Letters, IEEE,* vol. 14, pp. 337-340, 2007.

[435] J. N. Mourjopoulos and P. D. Hatziantoniou, "Real-Time Room Equalization Based on Complex Smoothing: Robustness Results," in *Audio Engineering Society Convention 116*, 2004.