MULTI-MICROPHONE SIGNAL PROCESSING FOR AUTOMATIC SPEECH RECOGNITION IN MEETING ROOMS

By

Marc Ferràs Font

Als que més estimo.

Als que més m'estimen.

Table of Contents

Table of Contents v

List of Tables viii

List of Figures x

Abstract xv

Acknowledgements xvi

1 Introduction 1

2 An Overview of Automatic Speech Recognition 4
- 2.1 Introduction . . . 4
- 2.2 Preprocessing . . . 7
- 2.3 Feature extraction . . . 11
- 2.4 Decoding . . . 15
- 2.5 Speaker Adaptation . . . 17
- 2.6 Evaluation of Speech Recognition Systems . . . 18

3 System Evaluation Test-Beds 20
- 3.1 Corpora and Tasks . . . 20
  - 3.1.1 The ICSI Meeting Project . . . 21
  - 3.1.2 The NIST Meeting Room Project . . . 22
- 3.2 Experimental Test-beds . . . 23
  - 3.2.1 Mismatched Conditions Digit Test-Bed (MMCDT) . . . 23
  - 3.2.2 Matched Conditions Digit Test-Bed (MCDT) . . . 24
  - 3.2.3 Mismatched Conditions Conversational Test-Bed (MMCCT) . . . 26
- 3.3 Matched-Pairs Significance Testing . . . 27

4 Reverberation. Equalization Techniques **30**
4.1 Reverberation 30
4.2 Speaker-to-receiver impulse response 32
4.3 Measuring Reverberation 33
4.4 Impulse Response Inversion 35
4.5 Multiple-Channel Impulse Response Inversion. The Multiple Input-Output Inversion Theorem (MINT) 39
4.6 Equalization-based Dereverberation Techniques 40
4.6.1 Single-channel Linear Least Squares Equalization 40
4.6.2 Multi-channel Linear Least Squares Equalization 43
4.6.3 Mutually Referenced Equalizers 45

5 Multi-Channel Dereverberation Techniques Based On Time-Delay Estimation **48**
5.1 Time-Delay Estimation Techniques 48
5.1.1 Implementation and Test 50
5.2 Delay-and-Sum 52
5.2.1 Implementation 53
5.2.2 Evaluation 53
5.3 Delay-and-Feature-Domain-Sum 56
5.3.1 Implementation 59
5.3.2 Evaluation 60
5.4 Time-Frequency Masking 61
5.4.1 Time-Frequency Representation of Speech Signals 61
5.4.2 Dual-Microphone Phase-Error Based Filtering 62
5.4.3 Multiple-Microphone Phase-Error Based Filtering 64
5.4.4 Implementation 65
5.4.5 Evaluation 66

6 Dereverberation Techniques Based On Linear Prediction **74**
6.1 Speech production, Autoregressive Modelling and Linear Prediction . 74
6.2 Linear Prediction in adverse environments 76
6.2.1 Noise and reverberation 76
6.2.2 Linear Prediction-based Dereverberation 78
6.3 Correlation Shaping 79
6.3.1 Weighted Correlation Shaping 82
6.3.2 Don't Care Region 82
6.3.3 Multi-channel Correlation Shaping 83
6.3.4 Implementation 84

6.3.5 Evaluation 85

7 Conclusions **96**

A Mathematical background **100**
A.1 Linear Least Squares Equalization 100
A.2 Minimum-norm Matrix Inversion 102
A.3 Correlation Shaping Gradient Derivation 104

Bibliography **106**

List of Tables

3.1 Meeting corpora contributions for RT04S NIST evaluation. 22
3.2 Number of distant microphones provided in RT04 development data. 23
3.3 Training and test conditions for the mismatched conditions digit test-bed (MMCDT) 24
3.4 WER of single distant microphones on the mismatched conditions digit test-bed (MMCDT). 25
3.5 Training and test conditions for the matched conditions digit test-bed (MCDT) 27
3.6 WER of single distant microphones on the matched conditions digit test-bed (MCDT). 28
3.7 Training and test conditions for the mismatched conditions conversational test-bed (MMCCT) 29
3.8 WER of a single distant microphone on the mismatched conditions conversational test-bed (MMCCT). 29

5.1 WER of multiple distant microphones processed with delay-and-sum on the Mismatched Conditions Digit Test-bed (MMCDT). 54
5.2 WER of multiple distant microphones processed with delay-and-sum on the matched conditions digit test-bed (MCDT). 55
5.3 WER of multiple distant microphones processed with delay-and-sum on the Mismatched Conditions Conversational Test-bed (MMCCDT). 56

5.4 WER of multiple distant microphones processed with delay-and-feature-domain-sum on the Mismatched Conditions Digit Test-bed (MMCDT). 69
5.5 WER of multiple distant microphones processed with delay-and-feature-domain-sum on the matched conditions digit test-bed (MCDT). . . . 70
5.6 WER of multiple distant microphones processed with Phase-Error Based Filtering on the Mismatched Conditions Digit Test-bed (MMCDT). . 71
5.7 WER of multiple distant microphones processed with Phase-Error Based Filtering on the matched conditions digit test-bed (MCDT). 72
5.8 WER of multiple distant microphones processed with Phase-Error Based Filtering on the Mismatched Conditions Conversational Test-bed (MMCCDT). 73

6.1 WER of multiple distant microphones processed with 4-channel correlation shaping on the Mismatched Conditions Digit Test-bed (MMCDT). 92
6.2 WER of multiple distant microphones processed with 4-channel correlation shaping on the matched conditions digit test-bed (MCDT). . . 93
6.3 WER of multiple distant microphones processed with 4-channel correlation shaping on the Mismatched Conditions Conversational Test-bed (MMCCDT). 95

7.1 Relative WER improvement of the explored multiple-channel techniques over a single distant microphone on the proposed test-beds. . . 98

List of Figures

2.1 Black box diagram of a speech recognition system 5
2.2 A typical speech recognition framework. 5
2.3 8-band Mel-scaled triangular filterbank 12
2.4 Block diagram of PLP feature extraction 13

3.1 3-partition training-test non-speaker-overlapped Meeting Digits split (Number of speakers in brackets) 26

4.1 A speaker-to-receiver impulse response. The three types of reflections are identified in the graph. 33
4.2 1-channel speaker-to-receiver impulse response inversion block diagram. 35
4.3 Speaker-to-receiver impulse response inversion. Truncation issue. (a) Impulse response to be inverted. (b) Truncated theoretical inverse impulse response. (c) Equalized impulse response. 37
4.4 Pole-zero plot for (a) direct impulse response and (b) its corresponding inverse filter 38
4.5 Block diagram representation of Bezout identity. 39
4.6 Single-channel LLS Equalizer. (a) Speaker-to-receiver impulse response. (b) LLS-Equalized impulse response. (c) Log-scaled LLS-Equalized impulse response. 42
4.7 2-channel LLS Equalizer. (a) Speaker-to-receiver impulse responses. (b) LLS-Equalized impulse response. (c) Log-scaled LLS-Equalized impulse response. 44

4.8 Mismatched order 2-channel LLS Equalizer. (a) Speaker-to-receiver impulse responses. (b) LLS-Equalized impulse response. (c) Log-scaled LLS-Equalized impulse response. 45

5.1 Non-weighted vs. PHAT-weighted time-delay estimation (TDE). (a) Non-weighted cross-correlation function (CCF) for non-processed speech. (b) PHAT-weighted CCF for non-processed speech. (c) Non-weighted CCF for ICSI-OGI Wiener-filtered speech . (d) PHAT-weighted CCF for ICSI-OGI Wiener-filtered speech. (e) Non-weighted CCF for speech corrupted with WGN. (d) PHAT-weighted CCF for speech corrupted with WGN. 51
5.2 Delay-and-sum block diagram. 52
5.3 Delay-and-feature-domain-sum block diagram. 57
5.4 Normalized MSE of feature vectors of a speech signal corrupted with white gaussian noise, and processed using delay-and-sum (DS), delay-and-feature-domain-sum (DFDS). Single distant microphone (SDM) features were also included for further comparison. 59
5.5 Analysis and synthesis by means of the short-time Fourier transform. 62
5.6 Dual-microphone phase-error based filtering block diagram. 63
5.7 Magnitude spectrum masking function in phase-error based filtering. . 64
5.8 Masking process in phase-error based filtering. (a) Squared phase-error spectrum. (b) Mask. (c) Original amplitude spectrum. (d) Masked amplitude spectrum. 67

6.1 Source-filter model for speech signal production. 74
6.2 Linear prediction analysis. (a) Voiced segment. (b) Unvoiced segment. (d) Autocorrelation function (AF) of (a). (d) AF of (b). (e) LP residual of a voiced segment. (f) LP residual of an unvoiced segment. (g) AF of (e). (h) AF of (f). 77

6.3 Isolating reverberation using linear prediction analysis. (a) A close-talking microphone utterance. (b) Linear prediction residual of (a). (c) A simple speaker-to-receiver impulse response. (d) Autocorrelation function (AF) of (c). (e) AF of a 30ms long prediction residual unvoiced segment in (b). (f) AF of the whole prediction residual in (b). 79
6.4 Correlation shaping block diagram. 80
6.5 Single-channel correlation shaping block diagram. 82
6.6 2-channel Correlation shaping block diagram. 83
6.7 Single-channel correlation shaping technique using a simple speaker-to-receiver impulse response as the input signal. 86
6.7.1 (a) Speaker-to-receiver impulse response. (b) Equalizer impulse response found through correlation shaping. (c) Equalized impulse response. 86
6.7.2 Linear-scaled (a) and log-scaled (b) autocorrelation function (AF) of the speaker-to-receiver impulse response in 6.7.1(a). Linear-scaled (c) and log-scaled (d) AF of the output. 86
6.8 Single-channel correlation shaping (CS) technique using white noise convolved with a real truncated speaker-to-receiver impulse response as the input signal. 88
6.8.1 Linear-scaled (a) and log-scaled (b) autocorrelation function (AF) of the input signal. 88
6.8.2 Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with no don't care region. 88
6.8.3 Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with 18.7ms long don't care region. 88
6.8.4 (a) Speaker-to-receiver impulse response. (b) Resulting equalizer using CS with no don't care region. (c) Equalized impulse response. 88

6.8.5 (a) Speaker-to-receiver impulse response. (b) Resulting equalizer using CS with 18.7ms long don't care region. (c) Equalized impulse response. 88
6.9 Single-channel correlation shaping (CS) technique using white noise convolved with a real speaker-to-receiver impulse response as the input signal. 89
6.9.1 Linear-scaled (a) and log-scaled (b) autocorrelation function (AF) of the input signal. 89
6.9.2 Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with no don't care region. 89
6.9.3 Linear-scaled (a) and Log-scaled (b) AF of the output signal, using CS with 18.7ms long don't care region. 89
6.9.4 (a) Speaker-to-receiver impulse response. (b) Resulting equalizer using CS with no don't care region. (c) Equalized impulse response. 89
6.9.5 (a) Speaker-to-receiver impulse response. (b) Resulting equalizer using CS with 18.7ms long don't care region. (c) Equalized impulse response. 89
6.10 4-channel correlation shaping (CS) technique using white noise convolved with real speaker-to-receiver impulse responses as the input signal. 91
6.10.1 Linear-scaled (a) and log-scaled (b) autocorrelation function (AF) of the output signal, using CS with no don't care region. 91
6.10.2 Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with 18.7ms long don't care region. 91
6.10.3 Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with 18.7ms long don't care region and exponential weighting. 91
6.10.4 Linear-scaled (a) and log-scaled (b) equalized speaker-to-receiver impulse response using CS with no don't care region. (c) Equalized impulse response. 91

6.10.5 Linear-scaled (a) and log-scaled (b) equalized speaker-to-receiver impulse response using CS with 18.7ms long don't care region. . 91
6.10.6 Linear-scaled (a) and log-scaled (b) equalized speaker-to-receiver impulse response using CS with 18.7ms long don't care region and exponential weighting. 91
6.10.7 Exponential weighting function used in Figures 6.10.3 and 6.10.6. 91

Abstract

Performance of current speech recognition systems severely degrades in the presence of noise and reverberation. While rather simple and effective noise reduction techniques have been extensively applied, coping with reverberation still remains as one of the toughest problems in speech recognition and signal processing.

Single-microphone dereverberation algorithms typically result in very low speech recognition performance. Taking advantage of multiple-microphones popularity and theoretical achievements such as the MINT theorem, blind multiple-microphone signal processing techniques are explored and evaluated in several speech recognition test-beds. These include connected digits recognition in meeting rooms and matched and mismatched training-test conditions as well as conversational speech recognition in meetings in mismatched training-test conditions.

Multi-channel equalization techniques are shown to be effective under explicit knowledge of either speaker-to-receiver impulse response or order determination, but not in real situations. Blind dereverberation techniques based on time-delay estimation, such as delay-and-sum, delay-and-feature-domain-sum and phase-error based filtering are shown to significantly improve recognition accuracy over a single distant microphone, while being robust enough for real noisy and reverberant speech. Dereverberation techniques based on linear prediction are introduced and multi-channel correlation shaping is further explored for speech recognition. It is shown to improve recognition accuracy of a single distant microphone only in some situations.

Acknowledgements

The work exposed in this Masters thesis has been developed within the AMI (Augmented Multi-Party Interaction) training program, coordinated by the Institut Dalle Molle d'Intelligence Artificielle Perceptive (IDIAP, Switzerland) and University of Edimburgh (United Kingdom). The internship itself consisted in a one year long visit at the International Computer Science Institute (ICSI, Berkeley, California) with the purpose of acquiring knowledge and researching multi-microphone signal processing algorithms for Automatic Speech Recognition (ASR) in the context of meeting rooms.

First, I would like to thank the Augmented Multiparty Interaction (AMI) project team for giving me the opportunity to visit ICSI, as well as to ICSI itself, for hosting me and, specially, to Barbara Peskin and Nelson Morgan.

Loads of thanks to Dave Gelbart who I could exchange so many thoughts about research, but also life, with. Thanks for being such a nice and uncommonly fair person. Please, sleep a little bit more. Thanks to all the ICSI Speech Group, for being always so supportive and in such a good mood. Thanks for letting me in the nourishing Novel Approaches meetings. Thanks to Frantisek Grezl for letting me bother him once in a while, Qifeng Zhu, for his quick technical support, Barry Chen, for his everlasting smile, Andreas Stolcke, for his patience and, last but not least, Xavier Anguera and Alberto Amengual for encouraging and cheering me up when I needed it... cause I needed it.

I also want to thank Dusan Macho, for becoming my supervisor at Universitat Politècnica de Catalunya (UPC, Spain), and the coordinator and director of the European Masters in Language and Speech (EMLS) at UPC, Climent Nadeu and

Nuria Castell.

Here, a whole paragraph to Climent Nadeu. Thanks for your uniquely patient and supportive e-mails from the far far away. Also for exchanging our thoughts about life. Also, so many other "also"s.

No need to mention, either, my everlasting friends in Barcelona.

Finally, my parents... I am endlessly grateful to my parents, for giving me the opportunity to open my eyes in one of the most beautiful planets I have ever known.

Berkeley, California Marc Ferràs Font

June 15, 2005

Chapter 1

Introduction

Current performance of automatic speech recognition (ASR) systems relies on the quality of the speech input and, in practice, only reasonable accuracy is achieved when close-talking microphones are used. Getting rid of the annoying close-talking microphones would definitely mean a significant step forward towards a more user-friendly speech recognition. Taking advantage of multiple distant microphones to further process speech and audio signals is, therefore, becoming more and more popular in the last years, although the cost of replicated hardware is still prohibitive for home users. Nonetheless, some PDA devices are already including low-quality microphone arrays. For meeting rooms, it seems that this cost could be assumed effortlessly.

Multiple distant microphone speech streams allow space and time structure to be exploited, as opposed to only time-domain sampling for a single microphone. This can be used to design spatial filters that select or block certain directions of arrival, speech and noise sources, for instance, using array signal processing techniques. Unfortunately, these typically rely on a speaker (or speakers) direction of arrival (DOA) estimate and on a priori knowledge of the microphone lay-out in space which, sometimes, is not available, as it is the case of our corpora. Therefore, these approaches will not be further mentioned in this work. The techniques explored in this work only require the multiple-microphone speech signals to be processed, and no other explicit knowledge.

On the other hand, meetings is a specially challenging domain for ASR. Interaction level is very high, usually involving more than one speaker, overlapping speech and disfluencies. Any number of topics can be covered by a wide enough collection of meetings, ensuring the use of large vocabulary. Speech recognition can benefit from the combination of multi-modal information such as video and speech. In a similar way, the use of multiple microphones is also expected to improve robustness on one side, but also bring a more natural feeling to the meeting participants.

This thesis explores several signal processing approaches for blindly combining speech information from several distant microphones at the front-end level for ASR, which are to be evaluated on several meeting room corpora across several tasks.

Chapter 2 sets out a brief overview of automatic speech recognition, from preprocessing to speaker adaptation techniques, as well as evaluation metrics. It is, though, specially focused on the front-end stage, where signal processing techniques operate.

Aiming to evaluate the explored algorithms explored, Chapter 3 describes the several speech corpora that are dealt with. Three ASR test-beds, covering both digits and conversational tasks, are further introduced for a more complete evaluation. A brief explanation on the matched pairs significance test (MPST) is also included.

Chapter 4 is focused on reverberation, which plays an important role in this work. Reverberation itself is first introduced, and modelling and characterization are further described, focusing on dereverberation issues. Single-microphone and multiple-microphone equalization techniques are explored, although not evaluated on speech recognition.

Chapter 5 presents multiple-microphone TDOA[1]-based dereverberation techniques. A brief section on time-delay estimation (TDE) is also included. Delay-and-sum, delay-and-feature-domain-sum and time-frequency masking are described and evaluated on the test-beds proposed in Chapter 3.

Chapter 6 explores how linear prediction can be taken advantage of in dereverberation techniques. Correlation Shaping is presented as an example of these type of algorithm. Evaluation on the three proposed ASR test-beds is also carried out.

[1]Time Difference Of Arrival.

To end with, conclusions and future work are exposed in Chapter 7.

Chapter 2

An Overview of Automatic Speech Recognition

This chapter gives a quick overview on the current speech recognition technology. Front-end processing, that is, preprocessing and feature extraction, is emphasized, since it is here where most of the work on robustness is done. Due to the vast literature available on speech recognition, decoding and speaker adaptation are not reviewed in depth.

2.1 Introduction

An automatic speech recognizer is a system which outputs words or sequences of words from speech signals. Current systems are implemented as computer software and, thus, speech signals first need to be transduced by a microphone and digitized prior to any digital processing. Typically, high quality speech is sampled at 16Ksps, since most of the speech information is confined below 8kHz, and quantized at a sample depth which ensures a low quantization noise floor. 16bit/sample are enough for linear coding schemes[1].

This black-box is a rather simple idea and, in practice, it involves processing at

[1]Dynamic range of speech signals can be quite high since energy can vary depending on whether a voiced or unvoiced sound is being uttered, for instance. A 16-bit quantization grid is found to be fine enough for acquiring weakest signals properly while ensuring a high enough dynamic range for the loudest ones.

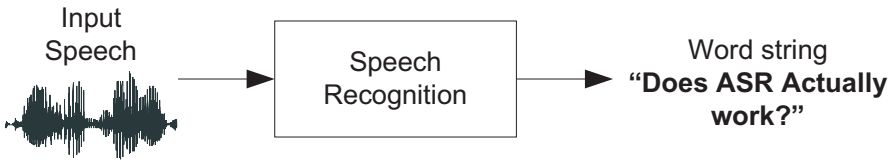


Figure 2.1: Black box diagram of a speech recognition system

the physical, acoustic and linguistic levels. It's not yet well understood how these different levels of processing should interact for proper speech recognition, and usually a reasonable and feasible approach, such as the one shown in Figure 2.2, is used instead.

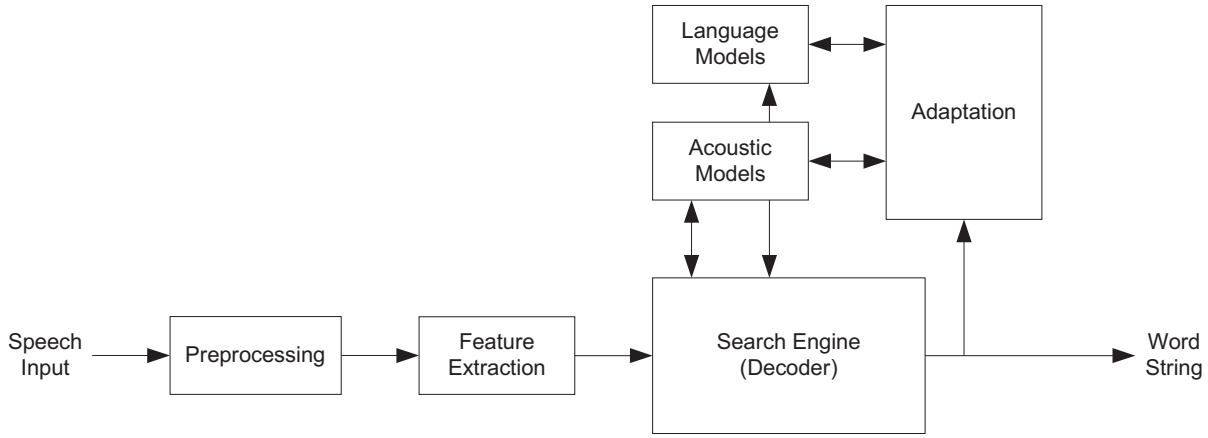


Figure 2.2: A typical speech recognition framework.

First, signal preprocessing is performed to enhance speech. This enhancements range from a simple pre-emphasis filter to more sophisticated noise reduction and dereverberation algorithms.

The feature extraction module manipulates speech data so that further stages can use a more compact, though meaningful and tractable representation.

The decoder seeks for the best match between a sequence of features and every possible sequence of words, using the available information from the acoustic and language models. Usually, some of the information in the decoding process is used as

feedback to adapt the acoustic and/or language models to improve performance as new speakers, environments or tasks are introduced.

The acoustic and language models include most of the knowledge of the recognition system and they must be ready before running the recognizer. If statistical approaches are used, as it is typically done, a training phase is required prior to the recognition step.

ASR systems can fall into several categories according to the nature of the utterances they are thought to recognize:

- Isolated words: Only one word can be recognized at a time and stops are required among words. Unless direct template matching techniques are used, an acoustic model for each word is typically used. The language model can be as simple as a list of possible words.

- Connected words: A pause among words is not required. Acoustic models are linked versions of isolated word acoustic models.

- Continuous speech: Utterance boundaries are determined by the recognizer itself, since sub-word units are used as part of acoustic modelling. Language modelling becomes more and more important as vocabulary grows.

- Spontaneous speech: The recognizer must be able to handle joined words as well as disfluencies such as "ums" and "ahs". The same approach as in continuous speech recognition is adopted for the acoustic and language models.

Speech recognition systems can be classified based on other criteria, such as the

number of speakers[2], vocabulary size[3] or language model complexity[4].

2.2 Preprocessing

Speech signals can be first enhanced so that recognition performs better on successive steps. It is understood that preprocessing is performed at the signal level. The following list summarizes some of these techniques.

1. Pre-emphasis filtering is the very first enhancement step for speech signals. It is aimed to compensate for lip radiation and inherent attenuation of high frequencies in the sampling process. High frequency components are emphasized and low frequency components are attenuated. This is quite a standard preprocessing step and is typically performed by means of a simple high-pass FIR filter, such as $H(z) = 1 - az^{-1}$, being a close to 1.

2. Speech enhancement techniques are usually aimed for channel and noise compensation in adverse environments. Robustness can be addressed either at the signal or at the feature level, or both. Only the former is explored at this point.

   - Noisy environments, such as streets or car interiors, can severely degrade accuracy of speech recognition systems, sometimes rendering them useless. Current techniques are not yet able to properly cope with non-stationary and impulsive noise whereas quite successful techniques have been developed for stationary or slow-varying noise.

     The most popular techniques for noise reduction are the so-called spectral methods, mostly because of their simplicity and effectiveness. In this approach, both short-time magnitude (or energy) noise and noisy speech spectrums are estimated first. According to a suppression rule, a spectral gain function is applied to the noisy speech amplitude spectrum by

[2]Speaker independence is usually desirable for speech recognition.

[3]Up to tens of thousands of words for large vocabulary. Speech recognition accuracy decreases significantly as vocabulary size grows.

[4]Syntactic and semantic constraints help continuous speech recognition by limiting the number of utterances that can be actually recognized.

means of a data-dependant time-varying filter. The enhanced magnitude and noisy phase spectrums are then combined to produce a clean short-time spectrum estimate. For time-domain resynthesis, overlap-add (OLA) methods are typically used.

A wide range of suppression rules have been studied in the past. In spectral substraction [6], the short-time clean spectrum is estimated as a linear substraction of the noise spectrum. In the well-known Wiener filtering technique the gain function is made to depend directly on the SNR, so that the mean square error (MSE) between the noisy and the estimated speech is minimized. Spectral methods rely on a good estimate of the noise spectrum which is not always available. Typically, it is taken from non-speech segments from the speech signal. Other methods such as quantile-based noise estimation [43] can be used to estimate the noise spectrum for those situations in which speech detection is specially difficult. Another issue regarding spectral methods is the so-called musical noise. This is an annoying artifact caused by deficient estimation of noise and noisy speech spectrums. Variances at each spectral bin are not allowed to decrease by means of averaging over long segments as this would result in speech distortion. Thus, spectral peaks may appear on both spectrum estimates, yielding a spiky gain function. As a consequence, sinusoids of random frequencies, amplitude and phase arise in time-domain. [10] and [11] propose spectral estimators to minimize this phenomenon. Although musical noise is a very important issue for speech enhancement it does not seem to affect seriously speech recognition accuracy.

Several other techniques for speech enhancement in noisy environments can be found in the literature. They mainly involve the use of Kalman filters, neural networks or suppression rules in transformed domains other than Fourier, such as cepstrum or Karhunen-Loeve, but they are far less popular than spectral methods.

- Dereverberation refers to the removal or attenuation of reverberation at

the signal level, that is, as a speech enhancement technique. As commonly opposed to noise, reverberation is correlated with the clean speech signal. This fact makes enhancement specially difficult as different behavior cannot be assumed for clean and reverberated speech. Although many techniques are available in the literature, up to now, none of these algorithms have been extensively used as a preprocessing step for speech recognition, mostly because of their little recognition accuracy improvement but also their high computational cost.

According to the nature of the problem to solve, two different views of dereverberation stand out:

- Non-blind techniques assume some a priori knowledge about the underlying degradation process, usually the speaker-to-receiver impulse response (see Section 4.2).
- Blind techniques do not assume any knowledge other than the speech signals themselves. This can be achieved by a single step or by using a speaker-to-receiver impulse response estimation procedure in combination with a non-blind equalization algorithm.

The most straightforward approach to dereverberation is inverting the speaker-to-receiver impulse response. In this line, [15] proposed a linear least squares (LLS) fitting of the equalized impulse response[5] as well as the inclusion of weighting and a "don't care" region.. Its single microphone version showed a high speech recognition accuracy improvement for the "don't care" approach. The multi-microphone version of LLS achieved almost perfect dereverberation and, correspondingly, a huge improvement in both speech recognition accuracy and audible quality. In [5], the blind adaptive mutually referenced equalizers (MRE) technique is applied to speech signals which results in successful dereverberation for multiple-microphone signals on simulated reverberation. In HERB [34], voiced structure of speech segments is exploited to estimate a more robust dereverberation

[5]Note that the speaker-to-receiver impulse response must be known a priori or estimated.

filter by means of an iterative procedure. Speech recognition experiments show little accuracy loss as more severe reverberation is introduced.

[16] proposes a maximum-kurtosis approach to dereverberation. Here, the linear prediction residual of reverberated speech is assumed more Gaussian than that of clean speech[6] and, therefore, it shows lower kurtosis. A least mean squares (LMS) gradient descent approach is chosen to maximize kurtosis in an adaptive manner. One of the main drawbacks of this algorithm is its sensitivity to outliers, since kurtosis involves fourth order statistics. In [44], a maximum-likelihood approach is presented to shape the maximum-kurtosis output probability density function in order to have both a high kurtosis and bounded derivative, thus reducing sensitivity to outliers. Although this approach makes maximum-kurtosis dereverberation more practical and robust for real recordings, it seems that the algorithm can converge to non-desired states, which make no sense as dereverberated speech.

In [15], correlation shaping is introduced, aiming to achieve whitening of the linear prediction residual of speech signals by shaping the output autocorrelation function, while still allowing short-term correlation by including a "don't care" region.

Most of the approaches for dereverberation are evaluated on simulated reverberant speech, partly for better algorithm analysis, but also due to a lack of robustness for processing real reverberant speech. In any case, though, multiple-microphone techniques are becoming more and more popular, partly thanks to the multiple input-output inversion theorem (MINT) (see Section 4.5) and partly to the unsuccessful behavior of single-microphone techniques. Currently available hardware and computing power allow multiple-microphone techniques to be boosted further.

[6] By the Central Limit Theorem.

2.3 Feature extraction

After preprocessing speech at the signal level, salient features are extracted before performing word-sequence search. By salient features we understand any features that are relevant for the speech recognition process. This is a crucial step since only some of the speech information is passed through to subsequent modules to perform classification.

Down-sampling is also embedded into feature extraction. Typically, speech signals are grouped into frames which are processed as a whole. For a more smooth evolution they are taken every certain time, say, 10ms. Conventional feature extraction techniques typically capture the short-term spectral envelope information of the speech signal[7,8]. The most common way to proceed is by means of cepstrum, which results from an homomorphic transformation of the input speech signal [23]. Mel-frequency cepstral coefficients (MFCC) and perceptual linear prediction (PLP) features have been extensively used, although other speech representations have been and are being explored as well [23] [7] [3].

- To extract MFCC features, the energy spectrum is first estimated by means of a discrete Fourier transform (DFT) for every input frame.

$$X(k) = \sum_{n=0}^{N-1} x(n) \mathrm{e}^{-j2\pi \frac{kn}{N}} \qquad 0 \leq k \leq N-1 \tag{2.3.1}$$

$$|X(k)|^2 = X(k)X^*(k) \tag{2.3.2}$$

  where $x(n)$ is the input speech signal, $X(k)$ is the corresponding DFT, $|X(k)|^2$ is its energy spectrum, and N is the frame size in samples.

[7] The spectral envelope is closely related to the resonant frequencies (formants) that produced the speech, which are determined by the pose of the human active articulators at a certain instant.

[8] Some novel approaches for feature extraction, such as HATS or TRAPS, also exploit long-term acoustic context [7].

The spectrum is then warped on a mel-frequency scale[9], according to the following triangular M-band filterbank:

$$H_m(k) = \begin{cases} 0 & k < f(m-1) \\ \frac{2(k-f(m-1))}{(f(m)-f(m-1))} & f(m-1) \leq k \leq f(m) \\ \frac{2(f(m+1)-k)}{(f(m+1)-f(m))} & f(m) \leq k \leq f(m+1) \\ 0 & k > f(m+1) \end{cases} \tag{2.3.3}$$

with m ranging from 0 to $M-1$. Here, $H_m(k)$ is the weight given to the kth energy spectrum bin contributing to the mth output band. An 8-band mel-scaled triangular filterbank is shown in Figure 2.3.

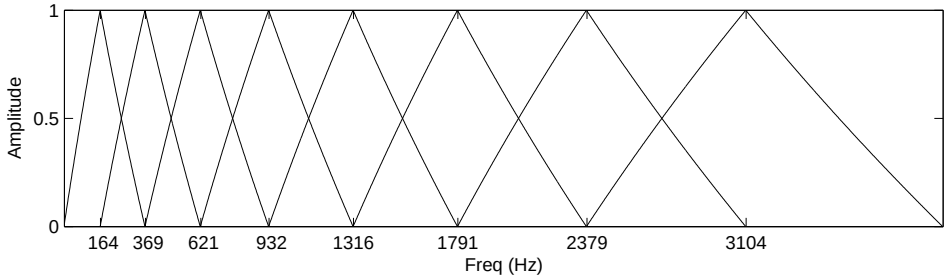


Figure 2.3: 8-band Mel-scaled triangular filterbank

Logarithm filterbank energies (logFBE) are next obtained as

$$S(m) = \ln \sum_{n=0}^{N-1} H_m(k)|X(k)|^2 \tag{2.3.4}$$

A discrete cosine transform (DCT) is typically performed to get the final cepstral domain representation, but also to achieve a high degree of decorrelation[10] of the output features.

$$c(n) = \sum_{m=0}^{M-1} S(m) \cos \frac{\pi n}{M}(\frac{m+1}{2}) \tag{2.3.5}$$

[9]The linear-to-mel frequency transformation is given by the formula $\text{mel}(f) = 2595 \log_{10}(1 + \frac{f}{700})$, where f is given in Hz.

[10]Feature decorrelation is desirable to improve the performance of the classification stage.

LogFBEs can also be significantly decorrelated by means of frequency filtering (FF) [37] [33], which makes use of very simple linear filters to derive a new set of features. Thus, the last step in MFCC feature extraction, the discrete cosine transform, is replaced by a much simpler processing step. Two filters, which only require a few subtractions,

$$\begin{aligned} H_1(z) &= 1 - z^{-1} \\ H_2(z) &= z - z^{-1} \end{aligned} \tag{2.3.6}$$

were shown to improve recognition performance over standard MFCC features in several noisy speech conditions [37].

- Perceptual linear prediction (PLP) features are derived from a psycoacoustically-motivated version of the well-known linear prediction (LP) analysis method [23]. Here, the linear predictor that minimizes the MSE prediction error over every frame is found by solving the normal equations [9]. To build these equations, the autocorrelation function must be estimated from the input signal. In PLP, though, the autocorrelation function is computed as the inverse Fourier transform of its power spectrum estimate after several perceptual spectral transformations[11] are performed. These are shown in Figure 2.4.

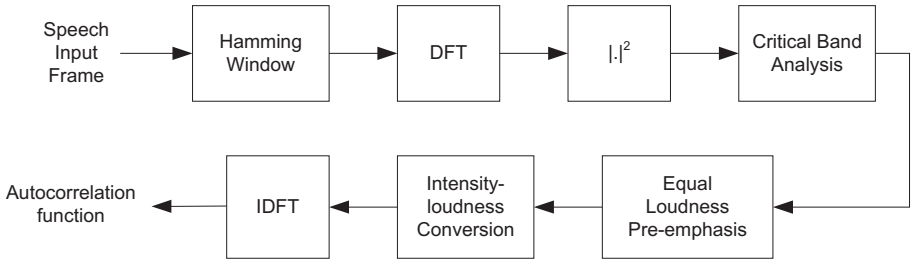


Figure 2.4: Block diagram of PLP feature extraction

The linear predictor obtained with this method is then transformed to cepstral

[11] The energy spectrum is first warped on a bark scale (critical-band analysis) which is a non-linear warping of the frequency axis. An equal pre-emphasis curve based on human hearing sensitivity is next applied. As the last step, sound intensity is mapped into perceived loudness by means of a cube root transformation, an operation that achieves compression, as the logarithm does in MFCC.

coefficients [23]. As [20] reports, using PLP features improved MFCC recognition accuracy in several noisy conditions.

Either using MFCC or PLP features, first (Δ) and even second order ($\Delta\Delta$) time derivatives of the features vectors are commonly included as extra features to capture longer acoustic context.

For further reading on feature extraction please refer to [23].

Features can be post-processed to improve recognition performance in adverse environments using additional techniques:

- Cepstral mean substraction (CMS) [2][42] is a widely-used method for removing short-term invariant linear channel distortion in speech signals[12]. In the spectral domain, this distortion affects as an additional transfer function, i.e., a constant multiplicative factor for each frequency. In the cepstral domain, though, this is translated into an additive effect[13], which can be cancelled by the substraction of the overall average of the cepstral coefficients over a speech segment. Unfortunately, this kind of processing is only done within a single frame span and reverberation can not be properly handled.

- Cepstral mean and variance normalization (CMVN) [46] is a simple and extensively used technique for improving robustness in speech recognition. For each utterance, mean and variance for each feature component are estimated[14]. The mean is first substracted from each of the feature vectors in the utterance, as in CMS, resulting in a zero-mean feature set. Next, each component is scaled independently in order to have unity variance.

  This simple correction helps acoustic classes, phones, for instance, have more invariant position and size in the feature space. CMVN, thus, reduces mismatch between training and test conditions, since the first and second moments of the

[12] Short-term linear distortion includes microphone distortion but also reverberation color.

[13] Thanks to properties of the logarithm function.

[14] As well as for CMS, only speech frames should be considered, since feature vectors in non-speech sections might affect mean and variance estimates significantly.

feature distribution are forced to be the same for both situations. Feature vectors of noisy and reverberated speech signals typically result in shifted mean and lower variance [35] for the former, and only a mean shift for the latter. CMVN is well-suited for these situations and robustness can be considerably improved.

- Relative spectral transformation (RASTA) [21] takes advantage of band-pass filtering in the feature domain to get rid of most of the non-linguistic information. Vocal tract shape variations are constrained by articulation physics within a certain range. Thus, RASTA uses this band-pass approach to reject either fast or slow fluctuations in the feature vectors which are not feasible when speech is uttered.

2.4 Decoding

The decoding step is aimed to find the optimal sequence of words given a sequence of observed features. For this purpose, additional knowledge, which is stored in the form of acoustic and language models, is required. These models must be known prior to recognition, for instance, in stochastic approaches, estimated from transcribed speech corpora in the training phase.

The speech recognition problem can be formulated as

$$\begin{aligned} \hat{\mathbf{W}} &= \arg\max_{\mathbf{W}} P(\mathbf{W}|\mathbf{O}) && (2.4.1)\\ &= \arg\max_{\mathbf{W}} \frac{P(\mathbf{O}|\mathbf{W})P(\mathbf{W})}{P(\mathbf{O})} && (2.4.2)\\ &= \arg\max_{\mathbf{W}} P(\mathbf{O}|\mathbf{W})P(\mathbf{W}) && (2.4.3)\\ & && (2.4.4) \end{aligned}$$

which corresponds to the maximum a posteriori (MAP) criterion decision rule [9], based on Bayesian theory. Here, $\mathbf{O}$ is the observed feature sequence, $\mathbf{W}$ is the word sequence under test and $\hat{\mathbf{W}}$ is the optimal word sequence. Therefore, the most likely

word sequence is chosen based on the observed data and the modelled probability distributions[15]. Training strategies other than MAP, such as maximum mutual information estimation (MMIE) [45][39] or minimum classification error (MCE) [39] have also been successfully applied, although not extensively.

For small vocabulary speech recognition, computing $P(\mathbf{O}|\mathbf{W})$ from a separate model for each of the words is still feasible whereas, to alleviate the computational requirements and to reliably train all word models in large vocabulary ASR, $P(\mathbf{O}|\mathbf{W})$ is split into two separate problems. First, the observed feature sequence is mapped into sub-word units by means of sub-word acoustic modelling. Second, every possible word or word sequence is mapped in terms of these sub-word units by means of pronunciation modelling. Hidden Markov models (HMM) [38] are typically used for acoustic modelling. In such case, every sub-word unit is modelled using an HMM in terms of observed feature vectors, and every word is modelled by an HMM as a network of sub-word units, as

$$P(\mathbf{O}|\mathbf{W}) = P(\mathbf{O}|\mathbf{S})P(\mathbf{S}|\mathbf{W}) \tag{2.4.5}$$

where, $P(\mathbf{O}|\mathbf{S})$ is the probability of observing the sequence $\mathbf{O}$ given the sub-word unit sequence $\mathbf{S}$ and $P(\mathbf{S}|\mathbf{W})$ is the probability of the sub-word unit sequence $\mathbf{S}$ given the word sequence $\mathbf{W}$.

To model the prior probability $P(\mathbf{W})$ over all possible word sequences $\mathbf{W} = w_1, w_2, \ldots, w_n$, independence is typically assumed and conditional probabilities, such as $P(w_i|w_1, w_2, \ldots, w_{i-1})$ are further used to model context[16], as

$$P(\mathbf{W}) = P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_1, w_2, \ldots, w_{i-1}) \tag{2.4.6}$$

[15]For continuous speech recognition, observation probability is usually in the form of a continuous Gaussian mixture, $P(\mathbf{y_t}|s_t) = \sum_k P(w_k|s_t)\mathcal{N}(\mathbf{y_t}; \mu_\mathbf{i}, \mathbf{\Sigma_i})$, with $\mathbf{y_t}$ being the observed feature vector at time t, s_t, the state at time t, $P(w_k|s_t)$, the weighting factor for the ith Gaussian, given the state at time t and $\mathcal{N}(\mathbf{y_t}; \mu_\mathbf{i}, \mathbf{\Sigma_i})$, the ith Gaussian in the mixture, with mean vector $\mu_\mathbf{i}$ and covariance matrix $\mathbf{\Sigma_i}$).

[16]Very large corpora are required to statistically estimate long-term context. Typically, language modelling can take advantage of no more than 4-word context.

where w_i is the ith word in the word sequence, $P(w_i|w_1, w_2, \ldots, w_n)$ is the probability of observing the word w_i given its context.

Statistical approaches, such as N-grams, are among the most widely used, both because of simplicity and recognition performance. Context-free grammars and other rule-based systems have not yet reached as high a performance as statistical approaches achieve, although it is thought that, as speech recognition technology evolves, they are to become more and more relevant.

For the search process, the Viterbi [23] algorithm is extensively used although, for large vocabulary ASR, sub-optimal solutions are usually taken due to the enormous computational cost related to optimal search.

Please refer to [23][9] for further reading on acoustic, pronunciation and language modelling, as well as on search strategies.

2.5 Speaker Adaptation

In order to address speaker independent speech recognition, one single acoustic model[17] is typically trained over as many data as possible to cover variability over gender, speaking style or accent. This model is thought to work optimally overall, but it is not optimized for any particular speaker. Speaker adaptation is aimed to specialize the overall acoustic model for every speaker. Recognition accuracy is expected to improve since only variability from one speaker is present during adaptation. This specialization is accomplished using speech and transcription data for every speaker. For supervised speaker adaptation, the transcriptions are required whereas to adapt the acoustic models in an unsupervised manner, the decoded hypotheses suffice.

To accomplish this specialization, speech and transcription data for every speaker are required, for supervised speaker adaptation or, if transcripts are not available, the decoded hypotheses can be used for unsupervised adaptation.

[17] Or separate male and female acoustic models.

Two main approaches for speaker adaptation stand out:

- Maximum-likelihood linear regression (MLLR) [12][29] seeks a linear transformation of the model parameters,

$$\hat{\mu} = \mathbf{W}\dot{\mu} \tag{2.5.1}$$

  in order to maximize likelihood for the given adaptation data. Here, $\dot{\mu} = [1\ \mu_1\ \mu_2\ \ldots\ \mu_n]^T$ is the extended mean vector[18], $\mathbf{W} \in \mathbb{R}^{n \times n+1}$, the transformation matrix, $\hat{\mu}$, the adapted mean vector, and n is the number of means to be adapted for this particular speaker[19]. Although, variance adaptation can also be performed, it is understood that most of the variability among speakers is explained by phone position in the acoustic space and, therefore, most of the improvement can be addressed by mean adaptation, only.

- Maximum a posteriori (MAP) seeks direct estimation of the adapted model parameters as

$$\hat{\mu}_{jm} = \alpha\tilde{\mu}_{jm} + (1 - \alpha)\mu_{jm} \tag{2.5.2}$$

  where $\tilde{\mu}_{jm}$ is the mean for state j and mixture component m[20], MAP-estimated from adaptation data, μ_{jm} is the corresponding mean, MAP-estimated from speaker-independent data, and $\hat{\mu}_{jm}$ is the newly adapted mean.

2.6 Evaluation of Speech Recognition Systems

ASR systems are evaluated in terms of the number of errors that are made. Several types of errors can be identified:

[18] The mean vector is extended for the transformation to be able to adapt mean offsets as well.

[19] Each transformation matrix, $\mathbf{W}$, is tied across a set of Gaussians, typically determined by a regression tree classifier at a previous step.

[20] When Gaussian mixtures are used for estimation of observation probability.

- Insertions: Certain words are present in the output hypotheses[21] which were not present in the transcription.
- Deletions: Certain words are not present in the hypotheses while they are in the transcription.
- Substitutions: Certain words in the hypotheses don't match the corresponding words in the transcription.

Combining these three types of errors, word error rate (WER) can be taken as a measure of recognition accuracy as

$$WER = \frac{\sum_{s=1}^{S} I + D + S}{\sum_{s=1}^{S} W} \tag{2.6.1}$$

where I is the number of insertions, D, the number of deletions, S, the number of substitutions and W, the total number of words in the transcription.

Other metrics, such as sentence error rate (SER) [9], have also been proposed, but WER is, by far, the most widely accepted criterion in the speech recognition community.

[21] Word guesses output by the recognition system.

Chapter 3

System Evaluation Test-Beds

In the previous chapter, speech recognition technology was quickly reviewed. Before describing any multiple-microphone enhancement technique, though, the corpora and the test-beds over which they are to be evaluated are first presented. At the end of the chapter, the matched pairs significance test is described for more reliable performance comparison.

3.1 Corpora and Tasks

The algorithms presented in this thesis are aimed at improving recognition accuracy. Word error rate (WER), the most extensively used metric for speech recognition, is chosen to measure algorithm performance. For this purpose, several test-beds are set out across various corpora, all of them involving only recordings in real meeting rooms. Two different complexity tasks, conversational speech and digits, are also explored, on one hand, to ease algorithm tuning and development, but also for a more complete system evaluation.

These corpora are comprised within two big Meeting Projects: the ICSI Meeting Project and the NIST Meeting Room Project[1], which are described more in depth in Sections 3.1.1 and 3.1.2, correspondingly.

[1]The ICSI meeting corpus is among the corpora made available by the Linguistic Data Consortium (LDC). It is also part of the corpora used for annual National Institute of Standards and Technology (NIST) meeting evaluations.

Several sources of variability are involved in these corpora, such as

- Speech recording quality.
- Acoustic characteristics of the meeting rooms.
- Number of microphones and microphone lay-out.
- Required speech recognition complexity.
- Speech overlap level.
- Meeting interaction level.
- Meeting topic.
- Meeting naturalness.

For speech enhancement, the use of data collected in several meeting rooms, at different sites, using different equipment and over different microphone lay-outs is indeed very valuable, since algorithms' performance can be more reliably contrasted.

3.1.1 The ICSI Meeting Project

The ICSI Meeting Project [24][8] addresses recent research on recognition and understanding of meetings. It provides a collection of about 75 hours of publicly available meeting room data along with multi-level annotation. Most of the collected data consists in regular research meetings at ICSI of about 1 hour long which involve from 3 to 10, both native and non-native, speakers. Overlap among speakers is naturally present in the database, and is also annotated in the transcription.

Close-talking and up to 6 table-top microphones are available, 2 of them being low-quality PDA microphones and the rest being high-quality PZM[2] microphones. Thus,

[2]Pressure zone microphones use omnidirectional microphone capsules pointed down on a flat surface, such as a table, which results in an hemispherical directivity pattern. Another distinctive feature is picking up pressure field close to the microphone while attenuating furthest acoustic sources.

a total of 4 high-quality omnidirectional microphones can be used for multi-channel signal processing algorithm development under real reverberant and noisy conditions. For more rapid algorithm development and tuning for ASR, task complexity can be reduced in order to avoid large-vocabulary, spontaneous and multi-party interaction. Thereby, a subset of the database, the Meeting Digits corpus, consisting of manually transcribed connected digit utterances, is available for the digits recognition task. In this corpus, room acoustics, microphone lay-out and speaker variability remain the same as in the conversational speech part.

3.1.2 The NIST Meeting Room Project

The NIST Meeting Room Project [36] is aimed to support audio and video recognition technology in meetings. The so-called Rich Transcription (RT) evaluations, focused on speech-to-text transcription, speaker segmentation and video extraction technologies, are periodically scheduled. Several data collection sites, such as ICSI[3], NIST[4], CMU[5] and LDC[6], have collaborated to provide meeting corpora for this purpose.

The meeting training data provided for the RT04[7] evaluation are shown in Table 3.2. Any other publicly available source might be used as well.

	Duration (hours)	**Meetings**
CMU Meeting Corpus	10	18
ICSI Meeting Corpus	72	75
NIST Pilot Meeting Corpus	13	17

Table 3.1: Meeting corpora contributions for RT04S NIST evaluation.

The RT04 development data consisted of the 80-minute long RT02 test set. A total of 8 excerpts of about 11 minutes each were used. 8 other 10 minutes long excerpts were included as RT04 evaluation data. Both development and evaluation data were collected at ICSI, CMU, NIST and LDC, and all of them included distant microphone data, as summarized in Table 3.2.

[3]International Computer Science Institute.
[4]National Institute of Standards and Technology.
[5]Carneggie Melon University.
[6]Linguistic Data Consortium.
[7]2004 Rich Transcription evaluation.

Number of microphones for RT04 data

	Development	Evaluation
CMU	1	1
ICSI	4^a+2^b	4 + 2
NIST	7	7-8
LDC	7-8	7-10

[a] Table-top PZMs.
[b] Low-quality PDA microphones.

Table 3.2: Number of distant microphones provided in RT04 development data.

Microphone lay-out was either not available or not accurately specified in the provided meeting data. This fact is especially important if array signal processing techniques were to be developed or evaluated on it.

3.2 Experimental Test-beds

Three different recognition test-beds were used for algorithm evaluation, being, all of them, based on the HMM-based SRI[8] speech recognizer [32]. They are aimed to cover several algorithm development needs and task complexity.

3.2.1 Mismatched Conditions Digit Test-Bed (MMCDT)

At a first stage, evaluations were run on the Meeting Digits task using a basic SRI speech recognizer setup. Algorithms that work at the signal level output yield an enhanced waveform, which can be fed into the recognizer front-end in a straightforward way. In the mismatched conditions digit test-bed (MMCDT) algorithm performance was evaluated without retraining the recognizer, that is, using close-talking microphone speech data in the training phase and Meeting Digits in the test phase. This is indeed a specially realistic situation for practical recognition systems, since a regular user does not have either access or time to train the recognizer. Here, the mission of enhancement algorithms is, thus, to reduce mismatch between train and test conditions.

[8] Stanford Research Institute at Stanford University.

Input waveforms were sampled at 16ksps and processed by the corresponding enhancement algorithm. The SRI recognizer was trained on switchboard conversational speech data, the input waveforms of which are sampled at 8ksps and, therefore, a downsampling step was required. This was performed by the SRI front-end itself, as a built-in feature.

For this test-bed only native speakers (15 out of 29) were considered, in order to avoid bias due to further training-test mismatch. Table 3.3 summarizes the conditions for the MMCDT.

NR[a]	Non-processed & ICSI-OGI Wiener-filtered
Features	39 MFCC (including energy, Δ and $\Delta\Delta$)
GDAM[b]	Yes
Training Data	8ksps Switchboard Conversational Telephone Speech (CTS)
Test Speakers	Male(13), Female(2), native
Test Data	Meeting Digits, 1910 utterances, 6239 digits
MVN[c]	Yes
VTLN[d]	No
Speaker Adaptation	MLLR[e]

[a] Noise-reduced speech waveforms
[b] Gender Dependant Acoustic Models
[c] Mean and Variance Feature Normalization
[d] Vocal Tract Length Normalization
[e] Maximum-Likelihood Linear Regression [12]

Table 3.3: Training and test conditions for the mismatched conditions digit test-bed (MMCDT)

Single distant microphone (SDM) experiments were run on this test-bed to define the baseline for further algorithm comparison. Independent baselines were chosen for noise-reduced and non-processed set-ups based on WER. Tables 3.4(a) and (b) summarize these results. SDM Channel 6 and SDM Channel F were set as the baselines for noise-reduced and non noise-reduced data sets, respectively.

3.2.2 Matched Conditions Digit Test-Bed (MCDT)

Speech enhancement algorithms tend to distort speech as well, which probably results in a bias in the acoustic feature distribution. In this line, the matched conditions digit

Noise-Reduced

	No MLLR	MLLR[a]
SDM Channel 6[b]	5.2%	2.9%
SDM Channel 7	8.0%	4.4%
SDM Channel E	6.5%	4.0%
SDM Channel F	5.3%	3.3%

Non Noise-Reduced

	No MLLR	MLLR
SDM Channel 6	6.3%	3.8%
SDM Channel 7	10.1%	5.5%
SDM Channel E	8.1%	4.7%
SDM Channel F[c]	6.1%	3.8%

[a] Maximum-Likelihood Linear Regression Speaker Adaptation.
[b] Noise-reduced MMCDT baseline.
[c] Non Noise-Reduced MMCDT baseline.

Table 3.4: WER of single distant microphones on the mismatched conditions digit test-bed (MMCDT).

test-bed (MCDT) is proposed to avoid this mismatch issue.

Here, the same speech recognizer setup as in MMCDT was used. Both training and test waveforms were sampled at 16ksps, but they were still downsampled by the SRI front-end to 8ksps. Since only 2 native female speakers were available in the Meeting Digits corpus, only native male speakers were considered[9]. Furthermore, the corpus was split into three training-test partitions for cross-validation sampling, aimed to improve the experiments' significance, with no speaker overlap between train and test splits, as shown in Figure 3.1. Table 3.5 summarizes train and test conditions for the matched conditions digit test-bed.

Just as in MMCDT, independent baselines were set for MCDT based on the best SDM word accuracy results. These are shown in Tables 3.6(a) and (b). SDM Channel 6 was chosen as the baseline for both noise-reduced and non noise-reduced waveforms.

[9] Performing train and test on only 2 female speakers separately could not provide the required speaker variability for further generalization.

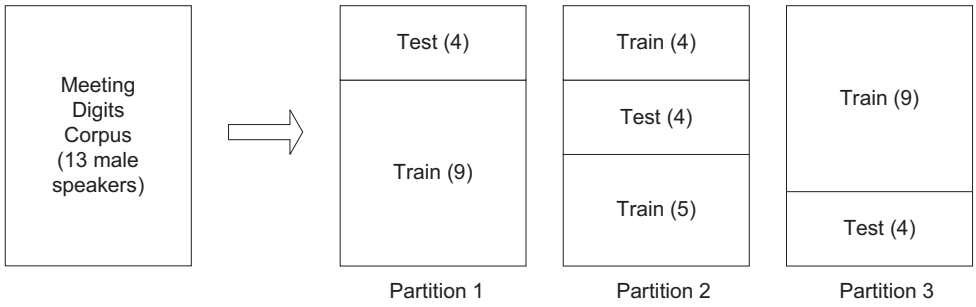


Figure 3.1: 3-partition training-test non-speaker-overlapped Meeting Digits split (Number of speakers in brackets)

3.2.3 Mismatched Conditions Conversational Test-Bed (MMCCT)

Recognition accuracy of the SRI recognizer on Meeting Digits is quite high[10], which usually results in difficult comparison among systems, since very few errors are made[11]. Furthermore, they tend to be insuperable errors, usually not related to speech quality, but to articulation clarity of the utterances. Thus, the recognizer is not working at an appropriate range for noticing enhancements at its input.

Therefore, to complete algorithm evaluation, a conversational speech and large vocabulary speech recognizer, the 2004 ICSI-SRI-UW[12] meeting recognition system [32], was set up and ran over the RT04S NIST evaluation development meeting data, consisting of 8 10-minute long pieces (see Section 3.1.2). Each of these meeting excerpts was first segmented by the SRI system. The resulting segmentation was kept the same across all experiments in order to minimize other sources that could affect algorithm performance comparison. Decoding was based on the 5xRT SRI recognition engine. Preliminary hypotheses were first obtained using with-in word phone-loop MLLR, bi-gram scoring, and 4-gram re-scoring. Table 3.7 summarizes the mismatched conditions conversational test-bed (MMCCT) conditions. Central

[10]Up to 2% WER.

[11]The less tokens differ when comparing performance between two systems, the more difficult it is to get significance (see Section 3.3).

[12]International Computer Science Institute (ICSI), Stanford Research Institute (SRI), University of Washington (UW).

NR[a]	Non-processed & ICSI-OGI Wiener-filtered
Features	39 MFCC (including energy, Δ and $\Delta\Delta$)
GDAM[b]	No (Male models only)
Training Speakers	Male (13), native, 9 speakers/part.
Training Data	Meeting Digits
	P1[c]: 1040 utterances (3377 digits)
	P2: 1030 utterances (3394 digits)
	P3: 950 utterances (3081 digits)
Test Speakers	Male (13), native, 4-5 speakers/part.
Test Data	Meeting Digits
	P1: 470 utterances (1549 digits)
	P2: 480 utterances (1030 digits)
	P3: 560 utterances (1845 digits)
MVN[d]	Yes
VTLN[e]	No
Speaker Adaptation	MLLR[f]

[a] Noise-reduced speech waveforms
[b] Gender Dependant Acoustic Models
[c] P1=Partition 1, P2=Partition 2 and P3=Partition 3
[d] Mean and Variance Feature Normalization
[e] Vocal Tract Length Normalization
[f] Maximum-Likelihood Linear Regression [12]

Table 3.5: Training and test conditions for the matched conditions digit test-bed (MCDT)

microphones, previously specified by NIST, were set as the baselines for each meeting independently. Single distant microphone WER is shown in Table 3.8.

3.3 Matched-Pairs Significance Testing

In order to more reliably compare performance among systems, significance tests are run. The matched pairs significance test (MPST) is chosen for this task to compare a baseline system versus a candidate system. Here, the transcriptions and the hypotheses from the outputs for both experiments are first aligned to compute word-level differences. Later, significance is calculated based on how likely it is for chance to explain those differences.

To proceed, the number of times the hypotheses differ between the two systems,

Noise-Reduced

	No MLLR	MLLR[a]
SDM Channel 6[b]	3.1%	2.2%
SDM Channel 7	4.7%	3.0%
SDM Channel E	4.9%	3.2%
SDM Channel F	4.0%	2.9%

Non Noise-Reduced

	No MLLR	MLLR
SDM Channel 6[c]	3.3%	2.2%
SDM Channel 7	4.9%	3.0%
SDM Channel E	5.1%	3.3%
SDM Channel F	4.0%	2.8%

[a] Maximum-Likelihood Linear Regression Speaker Adaptation.
[b] Noise-Reduced MCDT baseline.
[c] Non Noise-Reduced MCDT baseline.

Table 3.6: WER of single distant microphones on the matched conditions digit testbed (MCDT).

say M, and the number of times a candidate system is better, say N, are computed[13]. A binomial density distribution[14] is next evaluated to get the probability that each of these differences were produced randomly under an uniform distribution. Depending on the chosen significance threshold, the significance criterion would be

$$P_{bin}(k \geq N|M) < 0.01 \tag{3.3.1}$$

for significance, and,

$$P_{bin}(k \geq N|M) < 0.05 \tag{3.3.2}$$

for weak significance, as usually used. This means that the probability that the improvement resulted by chance is very low, or low enough, respectively. Conversely, the improvement would be explained by the true better performance of the candidate system.

[13] To compute N, the references are required too.
[14] A binomial distribution results from successive independent fair coin toss experiments.

NR[a]	Non-processed & ICSI-OGI Wiener-filtered
Features	62-component extended multiframe features
GDAM[b]	Yes
Training Data	RT04s NIST Evaluation's Meeting Development Data[c]
	MAP[d] adaptation from 420h MMIE[e]
	CTS[f]-trained acoustic models
Test Data	10 min. segments[g] for each meeting
MVN[h]	Yes
VTLN[i]	Yes
Speaker Adaptation	MLLR[j]
LM[k]	Bi-gram + 4-gram re-scoring

[a] Noise-reduced speech waveforms
[b] Gender Dependant Acoustic Models
[c] ICSI, NIST, LDC and CMU meeting corpora
[d] Maximum A Posteriori
[e] Maximum Mutual Information Estimation [45]
[f] Conversational Telephone Speech
[g] Specified for each meeting in the evaluation procedure by NIST
[h] Mean and Variance Feature Normalization
[i] Vocal Tract Length Normalization
[j] Maximum-Likelihood Linear Regression [12]
[k] Language Modelling

Table 3.7: Training and test conditions for the mismatched conditions conversational test-bed (MMCCT)

Noise-Reduced

	Overall	**ICSI**	**NIST**	**LDC**	**CMU**
SDM	48.4%	34.7%	48.2%	56.2%	62.1%

Non Noise-Reduced

	Overall	**ICSI**	**NIST**	**LDC**	**CMU**
SDM	50.1%	37.4%	49.4%	56.2%	64.8%

Table 3.8: WER of a single distant microphone on the mismatched conditions conversational test-bed (MMCCT).

Chapter 4

Reverberation. Equalization Techniques

The previous chapters were focused on speech recognition. In this chapter, reverberation itself, as well as modelling, characterization and metrics, are introduced, thus focusing on its physical side. Speaker-to-receiver impulse response inversion, be it single-channel or multi-channel, is presented as a first approach for dereverberation. Its drawbacks are also set out, aiming to clarify why dereverberation is still an unsolved and difficult field. In the rest of the chapter, the Single-Channel and Multi-Channel Linear Least Squares (LLS) equalizers are explored as non-blind equalization techniques. To conclude with, the mutually referenced equalizers (MRE) technique is overviewed as a blind equalization approach.

4.1 Reverberation

Sound propagation in enclosed environments is especially hard to address, mainly due to complexity of enclosures[1], but also complexity of meaningful acoustic signals, e.g., audio or speech. Its characterization is subject to phenomena which strongly depend on every particular situation, that is, size, geometry and materials of the enclosure, as well as size, position and shape of objects and obstacles placed into it, source and receiver poses, etc... all of them being parameters hardly under control in real situations.

[1]This results in arbitrarily complex boundary conditions for sound propagation.

A wide range of acoustic field wavelengths[2] and obstacle sizes can be found, and different phenomena are derived from their interaction. For long wavelengths and relatively small obstacles, sound is diffracted[3]. For shorter wavelengths and larger obstacles, waves are mostly reflected, ending up bouncing once and again inside the enclosure, resulting in the reverberation phenomenon.

A ray approximation is usually adopted to model reverberation. In this model sound waves are propagated straight until they find an obstacle, for example, a wall. At this point, part of the energy of the wave is transmitted and the rest is reflected according to a transmission and a reflection coefficient[4]. Thus, if the source wave is a sinusoid, after one reflection, the reflected wave is a complex-scaled version of it, that is, for more complex sounds, the reflected wave would be a filtered and delayed version of the original one. For more than one reflection, assuming the superposition principle to hold, reverberation can be modelled as an impulse response which gathers all individual reflection delays and filters. Thus,

$$x(n) = h(n) * s(n) = \sum_{m=0}^{M-1} h(m)s(n-m) \tag{4.1.1}$$

where $*$ is the convolution operator, $h(n)$ is the source-to-receiver, or speaker-to-receiver, impulse response, $s(n)$ is the source sound signal and $x(n)$ is the reverberated signal, all of them being sequences sampled in the time domain. M is the length of $h(n)$. Using vector notation, (4.1.1) can be rewritten as

$$x(n) = \mathbf{h}^T \cdot \mathbf{s}(n) \tag{4.1.2}$$

where $\mathbf{h} = [h(0)\ h(1)\ \ldots\ h(M-1)]^T$ and $\mathbf{s}(n) = [s(n)\ s(n-1)\ \ldots\ s(n-M+1)]^T$.

A model such as the one in (4.1.1) does not account for any non-linear propagation-related phenomenon, e.g., time evolution but, nonetheless, it is an attractive, simple

[2]Human audible frequency range starts at about 20Hz and extends up to 20kHz. Their associated wavelengths are 17m and 17mm, respectively, which differ in 3 orders of magnitude.

[3]Diffraction is the change in the directions and intensities of waves when passing by an obstacle of about the same size as the their wavelengths

[4]The transmission and reflection coefficients sum up to 1 and depend on the acoustic impedance of the material the wall is made of and the characteristic impedance of the air.

and effective enough way of modelling reverberation.

4.2 Speaker-to-receiver impulse response

As seen in the previous section, reverberation behavior is characterized by the speaker-to-receiver impulse response. Reflections can be classified depending on the time they take to reach the receiver. The fastest possible acoustic path from the speaker to the receiver yields a spike in the impulse response which is called *direct path*. It carries most of the energy of the impulse response, since it only depends on the attenuation of the medium, e.g. air, which is typically much lower than for reflected waves. The so-called *early reflections* arrive after the direct path, coming from the very first reflections on large flat surfaces, such as windows or walls in a room. They are discrete echoes of considerable amplitude and, in small or medium enclosures, they may not yet perceived as reverberation but as kind of coloring[5]. Finally, a large amount of closely spaced replicas coming from recurrent reflections inside the enclosure, make the reverberation up. Figure 4.1 shows a speaker-to-receiver impulse response estimated from real data.

Real reverberation is not static. Usually, the speaker moves its head as it is speaking which may be, indeed, a source of information for the listener. For reverberation simulations, the speaker-to-receiver impulse response can be taken from real data, but typically assuming both stationarity and linearity. Since the speaker-to-receiver impulse response depends on many different acoustic paths, slight changes in the speaker's pose can result in qualitative changes in the interference patterns, highly depending on the geometry of the enclosure. Applying impulse responses taken from real data on clean speech signals results in noiseless reverberant speech whereas in real situations, for distant microphones, for instance, noise and reverberation phenomena are typically present at the same time.

[5]For speech signals, reflections up to about 50ms after the direct path are not perceived separately from the direct sound. Spectral coloration may be perceived instead.

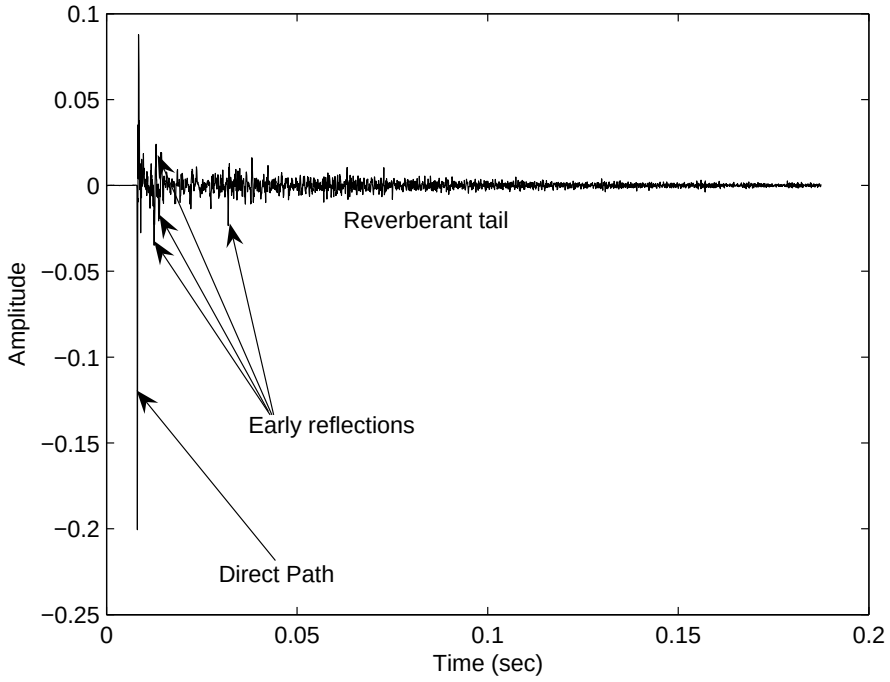


Figure 4.1: A speaker-to-receiver impulse response. The three types of reflections are identified in the graph.

4.3 Measuring Reverberation

Several measures can be taken to characterize reverberation and several methods can be used to take such measures. Signal-to-reverberation ratio (SRR) and reverberation time (RT) [40] are, perhaps, the most important parameters to account for in reverberant environments.

SRR measures how much the direct path signal is corrupted by the rest of the reverberation effects. In fact, this measure is rather analogous to the popular signal-to-noise ratio (SNR) to account for extremely correlated noise (reverberation). SRR can be estimated from the speaker-to-receiver impulse response as

$$SRR(dB) = 10 \log_{10} \frac{h^2(\delta)}{\sum_{l=0\ (l\neq\delta)}^{M-1} h^2(l)} \tag{4.3.1}$$

where $h(n)$ is the speaker-to-receiver impulse response, M, its length in samples, and δ the time-index of the direct path, in samples.

SRR is a ratio between the energy of the direct path versus the rest of the energy in the speaker-to-receiver impulse response and, thus, it does not involve any information about the duration of the impulse response.

On the other hand, reverberation time focuses on the determination of the duration of the speaker-to-receiver impulse response. Many approaches exist for its estimation. In Shroeder's method, only the sampled speaker-to-receiver impulse response is required [41]. Decay in the impulse response energy at time-index m is first calculated as

$$D(m) = 10\log_{10} T\sum_{l=m}^{M-1} h^2(l) - 10\log_{10} T\sum_{l=0}^{M-1} h^2(l) \tag{4.3.2}$$

where $T = 1/fs$ is the sampling period.

Usually, reverberation time is referred to a reference decay level. For a RT_{60}[6], decay $D(m)$ must be -60dB. Thus, by solving

$$D(fsRT_{60}) = -60 \tag{4.3.3}$$

an RT_{60} estimate can be obtained, in seconds.

Although both SRR and RT are valuable for the evaluation of reverberation, their relationship to speech recognition accuracy is not clear yet. In [15], it is found that RT is an important parameter to be minimized for improving both audible quality and speech recognition accuracy. Nonetheless, WER is the only metric used for performance evaluation in this thesis.

[6]Time needed for the reverberation energy to be 60dB below the total impulse response energy.

4.4 Impulse Response Inversion

Perhaps the most straightforward idea for combating reverberation is inverting the speaker-to-receiver impulse response. For such inversion, a filter which, when convolved with the speaker-to-receiver impulse response a perfectly equalized response results, is sought. Thus, as Figure 4.2 shows, an equalizer filter, $g(n)$, would recover the original speech signal, $s(n)$, from the reverberation waveform.

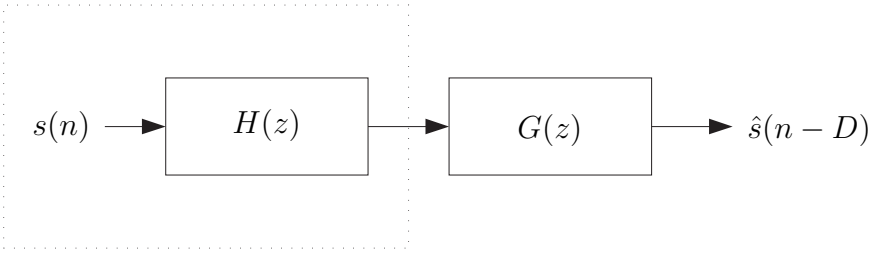


Figure 4.2: 1-channel speaker-to-receiver impulse response inversion block diagram.

Therefore,

$$\delta(n-D) = h(n) * g(n) = \sum_{m} h(m)g(n-m) \tag{4.4.1}$$

where $h(n)$ is the speaker-to-receiver impulse response, $g(n)$, the inverse filter impulse response and $\delta(n)$ is the Kronecker delta function[7].

Despite its conceptual simplicity, this approach has several important drawbacks:

1. The speaker-to-receiver impulse response must be known a priori. In laboratory conditions this can be achieved through system identification techniques but in real world situations such as those found in this work, i.e., in meeting rooms, one cannot expect it to be known. Nonetheless, blind reverberant impulse response identification techniques have been developed for this purpose [18] [47]. One of

[7]Note that a delay, D, is allowed in the equalized impulse response. Not considering this delay might cause the impulse response not to be invertible for causal systems, since an equalizing causal filter will only be able to delay its input.

these techniques could be used in conjunction with an impulse response inversion module to perform dereverberation.

2. To achieve complete equalization an infinite equalizing impulse response is commonly required. To illustrate, a very simple impulse response is shown next. The impulse response to invert corresponds to the formula,

$$h(n) = \delta(n) - 0.8\delta(n - 100) \tag{4.4.2}$$

which is a FIR filter with just two non-zero values, that is, the direct path and an echo. In such a simple case, the impulse response can be analytically inverted and it is found to be[8]

$$g(n) = \sum_{k=0}^{\infty} 0.8^k \delta(n - 100k) \tag{4.4.3}$$

which is an infinite series, and $g(n)$, thus, has infinite extent. This means that if the inverse filter is truncated a perfectly equalized impulse response is only possible within the filter length span, since boundary effects arise. Conversely, if boundary effects were expected to be minimized by means of any optimization procedure, complete equalization would not be achievable within the equalizer time span.

Ideally, infinite impulse response (IIR) equalizers should be used. Nonetheless, as described in the next point, the speaker-to-receiver impulse response inverse can be unstable. This is a major concern in adaptive filter design. Thereby, finite impulse response (FIR) filters which approximate IIR behavior are typically used. In order to minimize boundary effects in the resulting equalized impulse response, though, the length of these filters must be kept very long. Furthermore, if the equalizer is to be designed adaptively, convergence issues also may arise due to the amount of parameters that need to be estimated.

[8]The z-transform of $h(n)$ is $H(z) = 1 - 0.8z^{-100}$. Thus, the inverse filter will be $G(z) = \frac{1}{1-0.8z^{-100}}$ which corresponds to $g(n) = 0.8g(n - 100) + \delta(n) = \sum_{k=0}^{\infty} 0.8^k \delta(n - 100k)$.

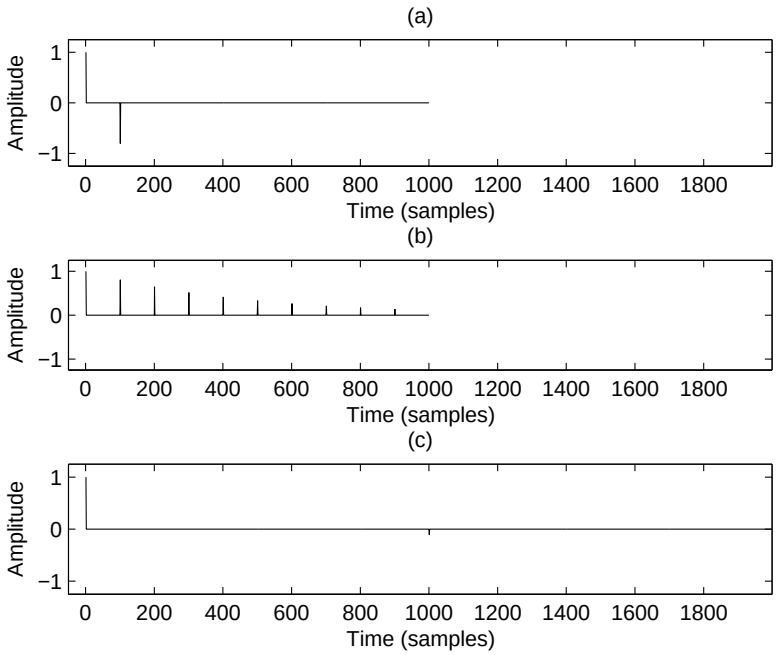


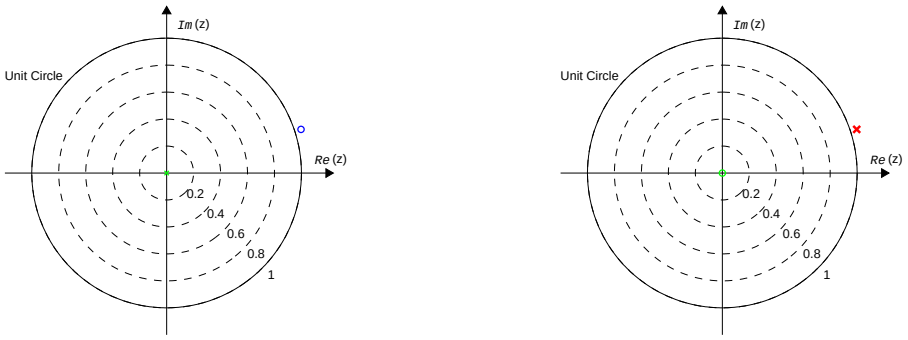
Figure 4.3: Speaker-to-receiver impulse response inversion. Truncation issue. (a) Impulse response to be inverted. (b) Truncated theoretical inverse impulse response. (c) Equalized impulse response.

3. The speaker-to-receiver impulse response can be non-invertible by a causal system, that is, even in the simplest case of just one reflection, a causal system can fail to invert it. This is due to a stability issue. No stable equalizer can be found unless the speaker-to-receiver impulse response has all of its zeros either inside the unit circle[9], for a causal equalizer, or outside the unit circle, for a non-causal one. Therefore, using FIR filters for equalization ensures stability, at the expense of not being able to achieve complete equalization.

   In (4.4.2), if the delay and the coefficients for each of the echoes are chosen carefully, such that it is no longer a minimum-phase impulse response,

[9]Filters for which all zeros lie inside the unit circle in the z-transform domain are called minimum-phase.

$$h(n) = 0.5\delta(n) - 0.8\delta(n - 10) \tag{4.4.4}$$



(a) (b)

Figure 4.4: Pole-zero plot for (a) direct impulse response and (b) its corresponding inverse filter

it can be seen that one of its zeros falls outside the unit circle (see Figure 4.4 (a)). It is clear, thus, that its inverse filter can be unstable for a causal system, since its pole lies outside the unit circle. For this particular type of impulse responses, this happens whenever the direct path gain is lower than the echo gain[10].

4. As a practical issue, impulse responses can exhibit deep valleys in their spectrum. After equalization, these result in high amplification gains in these frequencies and, thus, long and resonant equalization filters, too. This is a main source of artifact in the resulting waveform.

[10]Impulse responses of the form $h(n) = a\delta(n) + b\delta(n - D)$ have a zero (or a pole for its inverse filter) at $z = \sqrt[D]{\frac{b}{a}}$.

4.5 Multiple-Channel Impulse Response Inversion. The Multiple Input-Output Inversion Theorem (MINT)

As described in Section 4.4, perfect impulse response inversion cannot be generally achieved using a single microphone signal. Thanks to the so-called Multiple Input Output Inversion Theorem (MINT), though, complete equalization is possible when combining multiple-microphone information under certain conditions. The so-called Bezout identity [5] [22] states

$$H_1(z)G_1(z) + H_2(z)G_2(z) = z^{-D} \tag{4.5.1}$$

where $H_1(z)$ and $H_2(z)$ are the z-domain polynomials associated with the speaker-to-receiver FIR impulse responses of each channel, and $G_1(z)$ and $G_2(z)$ are the polynomials associated with the inverse FIR equalization filters. D is a delay allowed to avoid causality issues. Figure 4.5 shows identity (4.5.1) in terms of a block diagram.

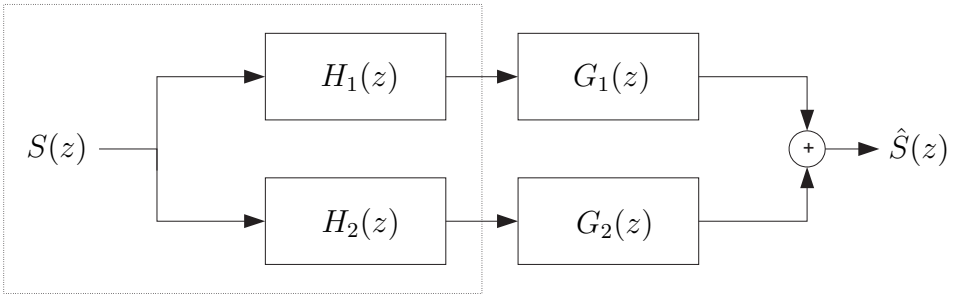


Figure 4.5: Block diagram representation of Bezout identity.

(4.5.1) holds under the following conditions:

1. The order of $G_1(z)$ and $G_2(z)$ must be, at least, $M-1$ if the order of the speaker-to-receiver impulse responses $H_1(z)$ and $H_2(z)$ are M, or the largest of them.

2. $H_1(z)$ and $H_2(z)$ cannot share any common zero. They must be co-prime polynomials.

This can be translated into perfect dereverberation in equalization terms, if the order of the channels and the speaker-to-receiver impulse responses are known. Regarding the second condition, sharing of common zeros for impulse responses obtained from real data would certainly be a chance situation.

If $G_1(z)$ and $G_2(z)$ are polynomials of higher order than $M-1$, multiple solutions, i.e., multiple multiple-microphone equalizers, would exist.

For more than 2 channels, identity (4.5.1) takes the form of

$$\sum_{k=0}^{K} H_k(z)G_k(z) = z^{-D} \tag{4.5.2}$$

The conditions for (4.5.2) to hold are the analogous to those in the 2-channel case:

- The order of $G_i(z), \forall i$ must be, at least, $M-1$ if M is the highest order of $H_i(z), \forall i$.
- $H_i(z), \forall i$ can not have any common zero

4.6 Equalization-based Dereverberation Techniques

Three dereverberation techniques focused on speaker-to-receiver impulse response equalization are presented in the next sections. Two non-blind techniques, Single-Channel and Multiple-Channel Linear Least Squares (LLS) equalizers are explored, implemented and tested to illustrate the theoretical content described in Sections 4.4 and 4.5 in more practical situations. As an example of a blind equalization technique, Mutually Reference Equalizers (MRE) is described as well.

4.6.1 Single-channel Linear Least Squares Equalization

As shown in 4.4, FIR filters are typically used for impulse response inversion, in order to avoid the inherent instability issues related to equalization. Thus, a truncated and

approximate version of the true equalizer is used instead. This approximation can be performed by means of an optimization procedure, such as a least squares fitting of the desired impulse response in terms of $g(n)$ [15]. Such Linear Least Squares (LLS) problem can be stated, in vector notation, as

$$\arg\min_{\mathbf{g}}((\mathbf{Hg}-\mathbf{d})^T(\mathbf{Hg}-\mathbf{d})) \tag{4.6.1}$$

, where $\mathbf{g} = [g(0)\ g(1)\ \cdots\ g(L-1)]^T$ is the inverse impulse response, $\mathbf{d} = [0\ \cdots\ 0\ 1\ 0\ \cdots\ 0]^T$ is the desired equalized impulse response, $\mathbf{H}$ is the convolution matrix, defined as

$$\mathbf{H} = \begin{pmatrix} h(0) & 0 & 0 & \cdots & & 0 \\ h(1) & h(0) & 0 & \cdots & & 0 \\ \vdots & & & & & \vdots \\ h(M-1) & \cdots & h(1) & h(0) & 0 & 0 \\ 0 & h(M-1) & \cdots & h(1) & h(0) & 0 \\ \vdots & & & & & \vdots \\ 0 & \cdots & h(M-1) & \cdots & \cdots & h(1) & h(0) \\ 0 & & \cdots & & 0 & 0 & h(M-1) \end{pmatrix} \tag{4.6.2}$$

and L and M are $\mathbf{g}$ and $\mathbf{h}$ filter lengths, respectively.

The solution to (4.6.1) (see Appendix A.1) is

$$\mathbf{g} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{d} \tag{4.6.3}$$

Implementation and Test

Single-channel LLS equalizer system was implemented in MATLAB. A 50ms long truncated speaker-to-receiver impulse response[11] from the varechoic chamber at Bell Labs was taken and inverted as described in (4.6.3). The equalizer length was also

[11]The impulse response had a SRR of about 1.3dB and a RT60 of 0.5s.

set at 50ms. Figure 4.6 shows equalization results for this particular impulse response.

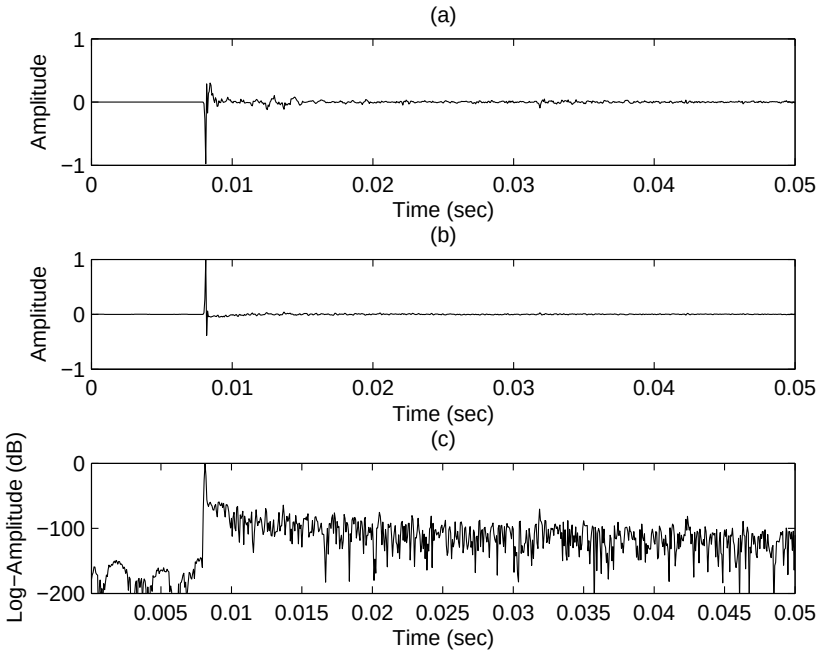


Figure 4.6: Single-channel LLS Equalizer. (a) Speaker-to-receiver impulse response. (b) LLS-Equalized impulse response. (c) Log-scaled LLS-Equalized impulse response.

Although Figure 4.6 (b)(c) indicates that the LLS equalizer works properly, the obtained optimal $\mathbf{g}$ was not able to completely invert the speaker-to-receiver impulse response, as discussed in Section 4.4. The residual for the fitting problem was spread evenly along the equalized impulse response length (this is shown clearer in Figure 4.6 (c)). This effect may cause, for example, an echo which, at first, was not perceived as such, an early reflection, for instance, to be spread in time and be perceived as more reverberant than the original one. As stated in [15] this may also impair speech recognition accuracy since it may have the effect of lengthening reverberation time. [15] proposes a weighted-LLS (WLLS) procedure to overcome this problem so that error for further samples in the equalized responses are given more relevance, for example. This would result in a shortening of the overall reverberation time.

The system was not evaluated for speech recognition. In this direction, detailed speech recognition experiments were run in [15].

4.6.2 Multi-channel Linear Least Squares Equalization

Following from Section 4.5, complete speaker-to-receiver impulse response inversion can be achieved using several microphones.

Single-channel LLS equalization can be effortlessly ported to a multiple-microphone version extending $\mathbf{H}$ and $\mathbf{g}$ in (4.6.3) [15] as

$$\begin{aligned} \mathbf{H} &= [\mathbf{H_1}\ \mathbf{H_2}\ \cdots\ \mathbf{H_C}] \\ \mathbf{g} &= [\mathbf{g_1}^T\ \mathbf{g_2}^T\ \cdots\ \mathbf{g_C}^T]^T \end{aligned} \tag{4.6.4}$$

where $\mathbf{H_i}$ is the convolution matrix for channel i, $\mathbf{g_i}$ is the equalizer for channel i and C is the number of microphones in the system.

Since the underlying linear system of equations is

$$\mathbf{Hg} = \mathbf{d} \tag{4.6.5}$$

, the more channels are added, the more variables are involved[12], whereas the number of equations[13] remains the same. The system of equations can eventually become underdetermined yielding $\mathbf{H}^T\mathbf{H}$ non full-rank and, thus, not invertible. Minimum-norm matrix inversion (see Appendix A.2) was performed to overcome this.

Implementation and Test

This system was implemented in MATLAB. As for the 1-microphone LLS equalizer, two speaker-to-receiver impulse responses[14] from the varechoic chamber at Bell Labs were taken and truncated at 50ms in order to reverberate a single-channel speech waveform. Next, the 2-channel LLS equalizer was run using 50ms long filters. The

[12] As many as the dimension of $\mathbf{g}$.

[13] As many as the dimension of $\mathbf{d}$.

[14] The impulse responses had a SRR of about 1.3dB and a RT60 of 0.5s, as in the single channel LLS equalizer test.

reverberated waveforms were fed through the 2-channel equalizer to eventually get a dereverberated speech waveform. Informal listening and visual inspection showed the equalized waveform to be indistinguishable from the original waveform. Figures 4.7 (b) and (c) show almost perfect equalization.

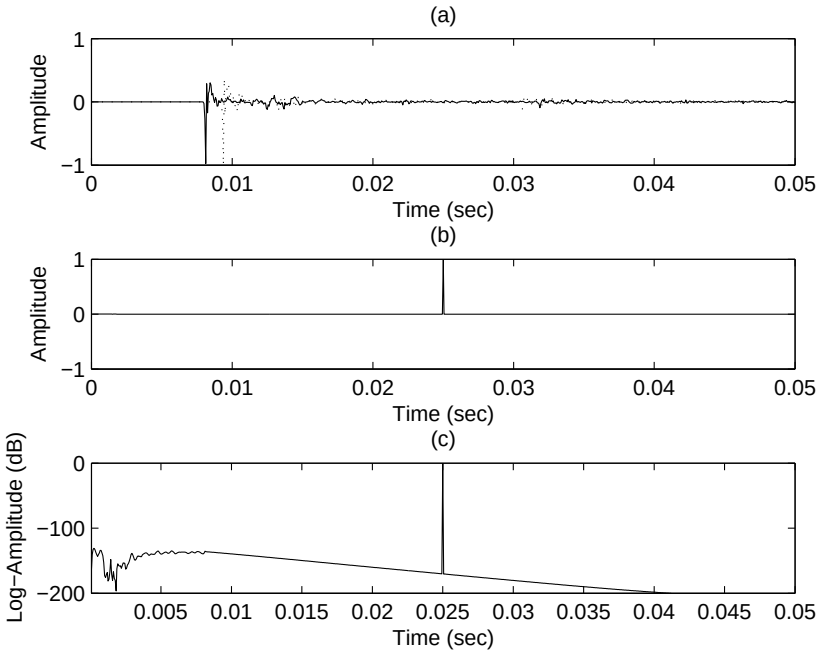


Figure 4.7: 2-channel LLS Equalizer. (a) Speaker-to-receiver impulse responses. (b) LLS-Equalized impulse response. (c) Log-scaled LLS-Equalized impulse response.

For MINT to hold, channel order must also be known a priori. In order to check what the effects of channel order mismatch are, the previous experiment was run on 150ms long speaker-to-receiver impulse responses while keeping the equalizer order at 50ms. Since the computational cost for equalization is usually very high, using short filters is a practical constraint. On the other hand, real speaker-to-receiver impulse responses are very long. Thus, this experiment was indeed a more realistic equalization situation. As shown in Figure 4.8 (b), the effects of channel order mismatch can be catastrophic, in terms of impulse response equalization. The 2-channel LLS equalizer is able to perfectly invert the speaker-to-receiver impulse responses only up to 50ms. Informal listening confirmed a high level of artifact, mostly high frequency

ringing, for this equalization test. It is worth noting how the energy of the equalized impulse response grows enourmously after the first 50ms, compared to the slowly decaying speaker-to-receiver impulse response.

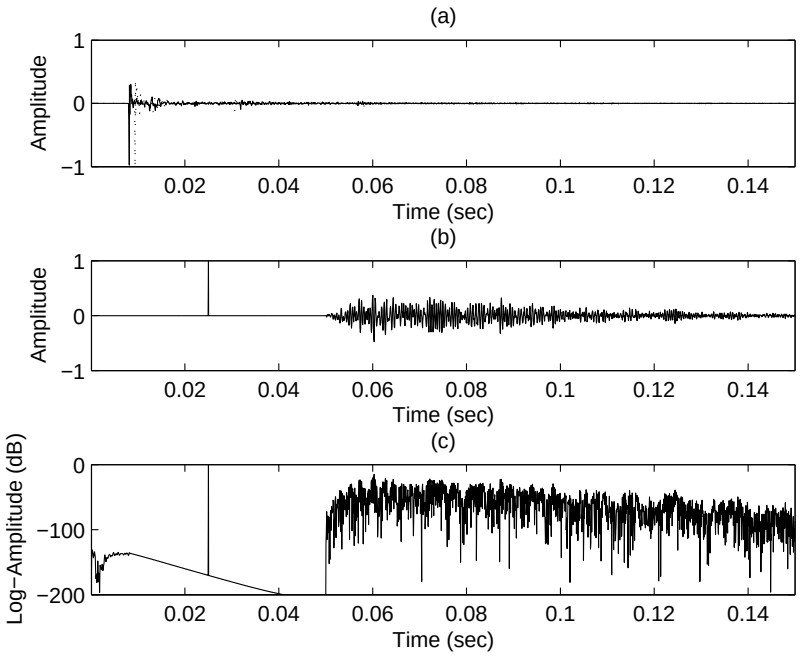


Figure 4.8: Mismatched order 2-channel LLS Equalizer. (a) Speaker-to-receiver impulse responses. (b) LLS-Equalized impulse response. (c) Log-scaled LLS-Equalized impulse response.

4.6.3 Mutually Referenced Equalizers

So-called Mutually Referenced Equalizers [14] is a blind equalization technique which was first developed for communications systems. Here, second order statistics of each channel's output are used to find a set of equalizers, one for each possible delays. In the following model for the observed signal,

$$\mathbf{x}(n) = \mathbf{H}\,\mathbf{s}(n) \tag{4.6.6}$$

$\mathbf{x}(n) = [\mathbf{x_1}(n) \ \cdots \ \mathbf{x_C}(n)]^T$, $\mathbf{x_i}(n)$ and $\mathbf{s}(n)$ are the tap delay line vectors at instant n, and $\mathbf{H} = [\mathbf{H_1} \ \mathbf{H_2} \ \cdots \ \mathbf{H_C}]^T$ is the multiple-channel convolution matrix which yields the observed vector $\mathbf{x}(n)$.

For each possible delay D in the equalized impulse response, that is, $\delta(n - D)$, a different relationship can be derived involving its corresponding equalizer $\mathbf{v_D}$ and recovering the original signal $s(n)$. To illustrate this, for $D = 1$, the desired equalized impulse response is $\delta(n - 1)$. Equalizer $\mathbf{v_1}$ is thought to achieve equalization for this particular delay. To compensate for it and recover the non-delayed original signal $s(n)$, the input signal can be advanced 1 sample, $x(n + 1)$. Thus, proceeding in an analogous way for all delays

$$\begin{aligned} \mathbf{v_0}^T\mathbf{x_n} &= \mathbf{v_0}^T \ \mathbf{H} \ \mathbf{s}(n) = [1 \ 0 \ \cdots \ 0] \ \mathbf{s}(n) = \mathbf{s}(n) \\ \mathbf{v_1}^T\mathbf{x_{n+1}} &= \mathbf{v_1}^T \ \mathbf{H} \ \mathbf{s}(n+1) = [0 \ 1 \ 0 \ \cdots \ 0] \ \mathbf{s}(n+1) = \mathbf{s}(n) \\ \mathbf{v_2}^T\mathbf{x_{n+2}} &= \mathbf{v_2}^T \ \mathbf{H} \ \mathbf{s}(n+2) = [0 \ 0 \ 1 \ 0 \ \cdots \ 0] \ \mathbf{s}(n+2) = \mathbf{s}(n) \end{aligned} \tag{4.6.7}$$

and equating pairs of equations from 4.6.7,

$$\begin{aligned} \mathbf{v_0}^T\mathbf{x_n} - \mathbf{v_1}^T\mathbf{x_{n+1}} &= 0 \\ \mathbf{v_1}^T\mathbf{x_n} - \mathbf{v_2}^T\mathbf{x_{n+1}} &= 0 \\ &\vdots \\ \mathbf{v_{i-1}}^T\mathbf{x_n} - \mathbf{v_i}^T\mathbf{x_{n+1}} &= 0 \end{aligned} \tag{4.6.8}$$

At this point, an overall error function can be built by combining the previous ones as

$$J_{MRE}(\mathbf{v_1}, \mathbf{v_2}, \mathbf{v_3}, \mathbf{v_4}) = E\left[\left|\mathbf{v_0}^T\mathbf{x_n} - \mathbf{v_1}^T\mathbf{x_{n+1}}\right|^2\right] + \ldots + E\left[\left|\mathbf{v_3}^T\mathbf{x_n} - \mathbf{v_4}^T\mathbf{x_{n+1}}\right|^2\right] \tag{4.6.9}$$

Overall minimization of this cost function would lead to trivial solutions $\mathbf{v_i} = \mathbf{0}$, $\forall i$. Therefore, constrained minimization must be performed. [14] explores several ways to cope with this, either through linear system solving or by means of LMS and RLS [19] adaptive filtering.

For speech and audio applications a reduced MRE criterion is proposed in [5]. A 2-channel and 3-point delay MRE error function is minimized through RLS adaptive filtering. The algorithm performed successfully for artificially reverberated speech waveforms.

For its application in real reverberant speech two relevant points arise, though. First, a 3-point delay criterion may not be robust enough to noise in real environments[15]. Introducing more delays would solve this problem, but at the expense of increasing computation time enormously. Second, and more important, the sensitivity of second order blind methods to channel order mismatch, as shown in the example in Figure 4.8, renders this method unpractical for real situations. This system was, thus, not implemented due to the exposed reasons.

[15] A 2-point criterion was shown not to be robust in [5].

Chapter 5

Multi-Channel Dereverberation Techniques Based On Time-Delay Estimation

The previous chapter explored equalization as a means to dereverberate speech signals. In this chapter, multi-channel techniques that are based on cross-channel time alignment, are presented. A brief overview on time-delay estimation (TDE) is set out in the first part of the chapter, since this is a crucial step in these type of systems. Delay-and-sum (DS) and delay-and-feature-domain-sum (DFDS) are next described as two examples of quite straightforward ways to combine multi-channel speech signals. To end the chapter with, a multi-channel time-frequency masking approach is explored. All these techniques were evaluated on the test-beds presented in Section 3.2.

5.1 Time-Delay Estimation Techniques

Estimating relative time-delay between two waveforms is an old and still common problem found in signal processing. However, no optimal solution has yet been found. Even under strong statistical assumptions, maximum-likelihood (ML) [25] or maximum a posteriori (MAP) estimation fails due to time non-differentiability[1] and, thus,

[1]ML, MAP or any other cost functions are neither continuous nor differentiable with respect to time, for sampled signals. This is not related to the cost function itself, but to quantization of time. In practice, a search must be performed over a set of discrete delays and optimize a certain cost

approximate solutions, i.e., search-based approaches, are adopted.

TDE is typically based on cross-correlation methods. From stochastic process theory, the cross-correlation function for two stationary processes[2] is defined as

$$r_{x_1x_2}(m) = \mathrm{E}\left[x_1(n)x_2(n-m)\right] \tag{5.1.1}$$

where $x_1(n)$ and $x_2(n)$ are the processes for which relative time-delay is to be estimated and $\mathrm{E}[\cdot]$ denotes the expectation operator which, in practice, since only realization of these processes are available, is replaced by the time average

$$\hat{r}_{x_1x_2}(m) = \sum_{n=0}^{N-1} x_1(n)x_2(n-m) \tag{5.1.2}$$

where $x_1(n)$ and $x_2(n)$ are the waveforms that are to be compared, and N is the length of $x_1(n)$.

To proceed, the lag with maximum cross-correlation is taken as the best time-delay estimate, as

$$\delta_{x_1x_2} = \arg\max_m \hat{r}_{x_1x_2}(m) \tag{5.1.3}$$

and $\delta_{x_1x_2}$ is, thus, the relative time-delay between waveforms.

$\hat{r}_{x_1x_2}(m)$ is an optimal estimate of the cross-correlation function only in the presence of white gaussian noise. In this sense, it is not especially well-suited for reverberant speech, since the latter is specially correlated with with non-reverberant speech.

Generalized cross-correlation (GCC) methods [26] are typically used to improve time-delay estimation in quite a straightforward way, since the cross-correlation function can also be obtained through inverse Fourier transformation of the power spectral density (PSD). In this process, arbitrary weighting can be applied in the frequency

function, which could as well be ML or MAP. In this sense, the finer the sampling grid, the better estimates can be achieved.

[2] A sampled waveform is understood to be stationary, since it is a realization of the underlying stochastic process.

domain, by which a filtered cross-correlation function results. Thus, generalized GCC takes the form of

$$r_{x_1x_2}(m) = \frac{1}{2\pi}\int_{-\pi}^{\pi} W(\omega)G_{x_1x_2}(\omega)e^{j2\pi\omega m}d\omega \tag{5.1.4}$$

where $W(\omega)$ is a weighting function and $G_{x_1x_2}(\omega)$ is the cross-PSD of $x_1(n)$ and $x_2(n)$.

In practical situations, a discrete Fourier transform (DFT) is used and $W(\omega)$ becomes a weighting sequence. Several weighting approaches can be found in [26], being the phase transform (PHAT) one of the most popular ones. In PHAT weighting,

$$W(\omega) = \frac{1}{|G_{x_1x_2}(\omega)|} \tag{5.1.5}$$

Therefore, the PHAT-weighted cross-PSD is expected to have unity amplitude and its corresponding cross-correlation, a delta function. In real situations, though, a phase noise in the cross-PSD will result in a non-perfect delta function.

5.1.1 Implementation and Test

Both non-weighted and PHAT-weighted generalized cross-correlation methods were implemented in MATLAB. Two real reverberant waveforms[3] were used to evaluate their behavior. Figures 5.1 (a) and (b) show the resulting cross-correlation functions. Both methods yielded 54 samples as time-delay estimates.

ICSI-OGI Wiener filtering was run on the non-processed waveforms to get an enhanced pair of speech signals. Their cross-correlation functions are shown in Figure 5.1 (c) and (d). Both techniques gave 54 samples as the time-delay estimates, as in the previous case.

Therefore, both techniques achieve similar or identical[4] performance. Similar situations are expected to be encountered in our speech databases.

[3]Taken from PZM distant-microphones E and F in the ICSI Meeting Digits corpus.
[4]For these particular waveforms.

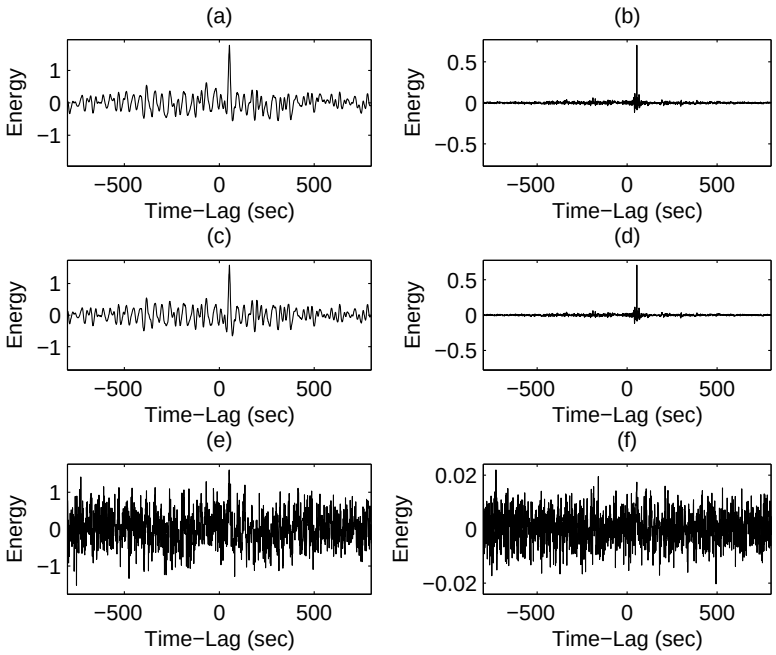


Figure 5.1: Non-weighted vs. PHAT-weighted time-delay estimation (TDE). (a) Non-weighted cross-correlation function (CCF) for non-processed speech. (b) PHAT-weighted CCF for non-processed speech. (c) Non-weighted CCF for ICSI-OGI Wiener-filtered speech . (d) PHAT-weighted CCF for ICSI-OGI Wiener-filtered speech. (e) Non-weighted CCF for speech corrupted with WGN. (d) PHAT-weighted CCF for speech corrupted with WGN.

To further evaluate both methods in a more critical situation, a great amount of uncorrelated white gaussian noise (WGN) was added to the non-processed files. Figures 5.1 (e) and (f) show their cross-correlation functions. Time-delay estimates turned out to be 53 vs. -729, for non-weighted and PHAT-weighted methods, respectively, showing a spreading effect that non-weighted cross-correlation did not. Under strong noisy conditions, thus, PHAT weighting can behave worse than the non-weighted method. However, when cross-correlation is estimated over long waveforms, that is, assuming the pose of the speaker to be stationary, (**??**) should be reliable enough and a large amount of noise should be introduced to reach this state.

On the other hand, when PHAT weights are close to 0 for certain frequency bins,

their phase estimates are nearly undefined and, thus, their contribution should be minimized by means of further weighting [26]. This effect was not accounted for in our implementation.

5.2 Delay-and-Sum

Delay-and-sum is a simple and popular dereverberation technique based on a strong simplification of the speaker-to-receiver inverse impulse response, to the extent of being considered as just a simple delta, $\delta(n)$, function. To proceed, one channel is chosen as a reference[5] and the time-difference of arrival (TDOA) for the rest of the channels is estimated using any TDE technique (see Section 5.1). Next, the time-aligned[6] speech signals are summed up as,

$$\hat{s}(n) = \frac{1}{C} \sum_{c=1}^{C} x_c(n - \delta_c) \tag{5.2.1}$$

where C is the number of channels, $x_c(n)$ is the cth channel waveform and $\hat{s}(n)$ the enhanced waveform. A block diagram for delay-and-sum is shown in Figure 5.2.

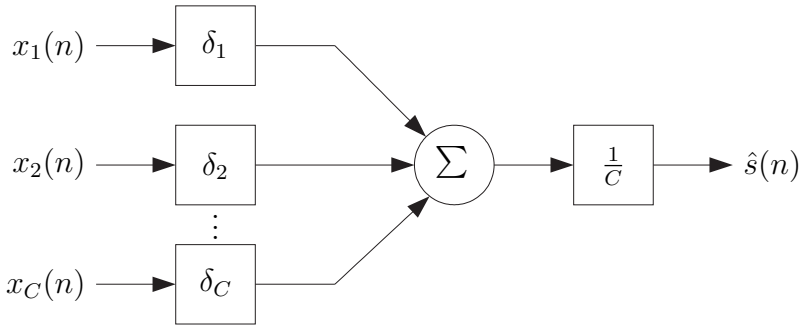


Figure 5.2: Delay-and-sum block diagram.

Time alignments are, thus, thought to normalize the speaker-to-receiver delays. This correction has the effect of focusing on the "wanted" source once the channels are summed up, since they should have similar waveforms. The chosen TDOAs hopefully

[5] More robustness against TDOA errors can be achieved taking multiple reference channels.
[6] Using the corresponding TDOAs.

attenuate sources coming from other directions[7] by means of interference. This is certainly true for spatially-uncorrelated noise acoustic fields, for example, but, in general, no warranty is given in this line.

5.2.1 Implementation

A non-weighted cross-correlation based delay-and-sum system was implemented by Tuomo Pirinen in Spring 2004 at ICSI. Informal listening on the multiple distant microphone ICSI Meeting Digits corpus shows a slight improvement in speech quality as well as in background noise reduction, mostly from HVAC systems. PHAT-weighted delay estimation was integrated into delay-and-sum and compared to its non-weighted counterpart[8]. No audible improvement over the non-weighted case was noticed on informal listenings. In fact, cross-channel delays differed marginally, mostly due to the length of the segmented speech waveforms[9].

5.2.2 Evaluation

Delay-and-sum was evaluated on the three proposed test-beds using both non-weighted (NW-DS) and PHAT-weighted (PHAT-DS) cross-correlation for time-delay estimation.

As shown in Tables 5.1 and 5.2 and 5.3, combining multiple-microphone signals using delay-and-sum improves word accuracy over a single distant microphone for the three proposed test-beds. This improvement is larger in non-matched conditions experiments. Thereby, DS, besides enhancing audible speech quality, is reducing mismatch between train and test conditions. Considerable improvements are also due to speaker adaptation, since acoustic models are specialized for each speaker, getting

[7] Or, in general, positions as delay estimation doesn't make any near-field or far-field assumption (a near-field source result in a non-flat wavefront).

[8] MATLAB source code for both NW-DS and PHAT-DS is available on-line at http://www.icsi.berkeley.edu/Speech/papers/multimic/ .

[9] As an example, for the ICSI Meeting Digits corpus the utterance mean length was 4.26 digits, which would correspond to about 1 second of speech waveform. In the delay estimation process, cross-correlation product terms would be strongly averaged, yielding quite reliable delay estimates, even in the non-weighted case.

rid of most of the inter-speaker variance. In a similar line, SDM word accuracy improves significantly by matching train and test conditions (see Table 5.2). Here, the difference in WER of DS over SDM is reduced, but it is still significant.

Regarding accuracy across delay estimation techniques, NW-DS and PHAT-DS achieve similar WER, although the latter behaves slightly better than the former, specially on non noise-reduced waveforms and using speaker adaptation. This improvement is also visible in the noise-reduced conditions.

Noise-Reduced		
	No MLLR	**MLLR**[a]
SDM Channel 6[b]	5.2%	2.9%
4-channel NW-DS[c]	2.4%	1.8%
4-channel PHAT-DS[d]	2.4%	1.7%
Significance Testing		
SDM Channel 6 ↔ 4-channel NW-DS	3.4e-29	1.43e-9
SDM Channel 6 ↔ 4-channel PHAT-DS	6.68e-27	3.46e-10
4-channel NW-DS ↔ 4-channel PHAT-DS	0.422	0.240
Non Noise-Reduced		
	No MLLR	**MLLR**
SDM Channel F	6.1%	3.8%
4-channel NW-DS	3.3%	2.1%
4-channel PHAT-DS	2.6%	1.9%
Significance Testing		
SDM Channel F ↔ 4-channel NW-DS	1.11e-23	1.20e-14
SDM Channel F ↔ 4-channel PHAT-DS	2.14e-37	3.18e-9
4-channel NW-DS ↔ 4-channel PHAT-DS	1.28e-7	3.2e-2

[a] Maximum-Likehood Linear Regression Speaker Adaptation.
[b] Single distant microphone with lower error-rate.
[c] Delay-and-sum using non-weighted cross-correlation for TDOA estimation.
[d] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.

Table 5.1: WER of multiple distant microphones processed with delay-and-sum on the Mismatched Conditions Digit Test-bed (MMCDT).

For the conversational test-bed (see Table 5.3), NW-DS achieves most of the

Noise-Reduced

	No MLLR	MLLR[a]
SDM Channel 6[b]	3.1%	2.2%
4-channel NW-DS[c]	2.33%	1.73%
4-channel PHAT-DS[d]	2.33%	1.63%
Significance Testing		
SDM Channel 6 ↔ 4-channel NW-DS	3e-4	2.1e-2
SDM Channel 6 ↔ 4-channel PHAT-DS	4.8e-4	1.6e-3
4-channel NW-DS ↔ 4-channel PHAT-DS	0.43	0.26

Non Noise-Reduced

	No MLLR	MLLR
SDM Channel 6	3.3%	2.2%
4-channel NW-DS	2.53%	1.8%
4-channel PHAT-DS	2.4%	1.63%
Significance Testing		
SDM Channel 6 ↔ 4-channel NW-DS	6.45e-10	1.74e-7
SDM Channel 6 ↔ 4-channel PHAT-DS	5.99e-13	3.40e-10
4-channel NW-DS ↔ 4-channel PHAT-DS	0.19	0.22

[a] MLLR Speaker Adaptation.
[b] Single distant microphone with lower error-rate.
[c] Delay-and-sum using non-weighted cross-correlation for TDOA estimation.
[d] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.

Table 5.2: WER of multiple distant microphones processed with delay-and-sum on the matched conditions digit test-bed (MCDT).

improvement for the noise-reduced data over a single distant microphone, although PHAT-DS gets an even lower WER. For noisy speech, NW-DS accuracy drops down, nearly reaching the same WER as SDM. Here, PHAT-DS only gets slightly higher WER.

It is interesting to note the almost non-existent improvement for NIST meetings, which include an important amount of overlapping speech. If more than one speaker is present in the utterance, more than one set of TDOAs along with their start and end time should be estimated. For this technique, enhancement is only effective for one of the speakers, since one set of TDOAs is accounted for.

Noise-Reduced

	Overall	ICSI	NIST	LDC	CMU
SDM[a]	48.4%	34.7%	48.2%	56.2%	62.1%
NW-DS[b]	44.8%	28.0%	44.2%	54.2%	62.1%
PHAT-DS[c]	43.2%	25.9%	45.5%	50.2%	62.1%
Significance Testing					
SDM ↔ NW-DS	9.32e-8				
SDM ↔ PHAT-DS	8.18e-11				
NW-DS ↔ PHAT-DS	0.03				

Non Noise-Reduced

	Overall	ICSI	NIST	LDC	CMU
SDM	50.1%	37.4%	49.4%	56.2%	64.8%
NW-DS	49.2%	35.1%	49.3%	56.3%	64.8%
PHAT-DS	45.7%	28.8%	49.2%	52.0%	64.8%
Significance Testing					
SDM ↔ NW-DS	0.37				
SDM ↔ PHAT-DS	3.12e-7				
NW-DS ↔ PHAT-DS	1.03e-5				

[a] Single distant microphone.
[b] Delay-and-sum using non-weighted cross-correlation for TDOA estimation.
[c] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.

Table 5.3: WER of multiple distant microphones processed with delay-and-sum on the Mismatched Conditions Conversational Test-bed (MMCCDT).

5.3 Delay-and-Feature-Domain-Sum

As a straightforward extension of delay-and-sum processing, delay-and-feature-domain-sum performs averaging after feature extraction, instead of at the signal level,

$$\begin{aligned} \mathbf{f_c}(m) &= FE_m\{x_c(n-\delta_c)\} \\ \hat{\mathbf{f}}(m) &= \tfrac{1}{C}\sum_{c=1}^{C}\mathbf{f_c}(m) \end{aligned} \tag{5.3.1}$$

where $m, 0 \leq m \leq M-1$ is the frame index, FE_m is the feature extraction operator for the mth frame, $\mathbf{f_c}(m)$ is the corresponding feature vector for channel c, $\hat{\mathbf{f}}(m)$ is the enhanced feature vector at frame m and C is the number of channels. This process is shown in Figure 5.3.

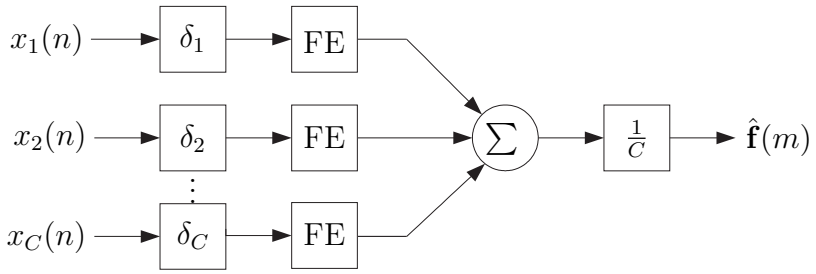


Figure 5.3: Delay-and-feature-domain-sum block diagram.

In conventional delay-and-sum, non-focused sources[10] are attenuated by means of inter-channel signal interference. This can be an effective approach when spatially uncorrelated noise is present, but it may result in poor enhancement in the presence of reverberation or distributed noise sources.

Feature extraction, on the other hand, usually lacks phase information (see Section 2.3) and, thus, interference seems not as feasible as for delay-and-sum. Noise and reverberation can be thought of as fluctuations of feature vectors around a desired mean[11] value, the statistics of which are not known a priori, either, as they depend on the type of reverberation and noise. Assuming spatial diversity, though, feature averaging should reduce some of the spatial variance of the feature vectors.

MFCC feature extraction can be compactly written as

$$\mathbf{f} = \mathbf{D^{-1}} \log(\mathbf{Mx}) \tag{5.3.2}$$

where $\mathbf{x}$ is the amplitude spectrum of an input speech frame, $\mathbf{M}$ is the Mel-filterbank transformation matrix, log is the component-by-component logarithm vectorial function, $\mathbf{D^{-1}}$ is the inverse DCT transformation matrix, and $\mathbf{f}$ is the extracted feature vector. For two-channel DFDS feature averaging, assuming $\mathbf{x_1}$ and $\mathbf{x_2}$ to come from time-aligned waveforms,

$$\hat{\mathbf{f}} = \frac{1}{2}\mathbf{D^{-1}} \log(\mathbf{Mx_1}) + \frac{1}{2}\mathbf{D^{-1}} \log(\mathbf{Mx_2}) \tag{5.3.3}$$

[10]Sources not specified by the time-differences Of arrival (TDOA).
[11]Here, in the spatial sense.

Since $\mathbf{D^{-1}}$ and averaging are both linear operators, they can be rearranged as

$$\hat{\mathbf{f}} = \mathbf{D^{-1}}\frac{1}{2}(\log(\mathbf{Mx_1}) + \log(\mathbf{Mx_2})) \tag{5.3.4}$$

and taking advantage of logarithm properties,

$$\begin{aligned} \hat{\mathbf{f}} &= \mathbf{D^{-1}}\log(\mathbf{Mx_1} \odot \mathbf{Mx_2})^{\frac{1}{2}} \\ \hat{\mathbf{f}} &= \mathbf{D^{-1}}\log(\mathrm{gm}(\mathbf{Mx_1}, \mathbf{Mx_2})) \end{aligned} \tag{5.3.5}$$

, with $\odot$ meaning component-by-component product of vectors and gm, component-by-component geometric mean. Thus, averaging in the feature domain is equivalent to performing the geometric average over channels of the mel-warped amplitude spectrums.

To compare DFDS versus DS behavior, a simple experiment was carried out. A speech utterance was added two white gaussian noise signals, yielding two corrupted input speech waveforms. Next, these were processed by both DS and DFDS, and their feature vectors were compared to the clean and noisy features of a single distant microphones. Feature extraction included log-energy, 1st to 12th cepstral coefficients, along with deltas and double deltas, to eventually make up 39-dimensional extended feature vectors[12]. Mean square error (MSE) was computed by averaging feature vector error[13] over frames and feature components for each system, over several SNRs. The MSE was normalized to the clean feature set energy for this particular speech waveform. The results of these experiments, shown in Figure 5.4, show lower MSE error for DFDS than for DS. This difference gets larger as more noise is introduced. It must be noted that similar experiments which did not include delta and double delta coefficients, resulted in worst performance for DFDS at high SNR values.

Since ASR does not typically make use of Euclidean distance, feature vector MSE may not be relevant for explaining its performance in speech recognition. However, this experiment highlights two major points:

[12]This feature extraction set-up is the same the SRI recognizer is using in the digits (MMCDT and MCDT) evaluation test-beds.

[13]With respect to the clean feature vectors.

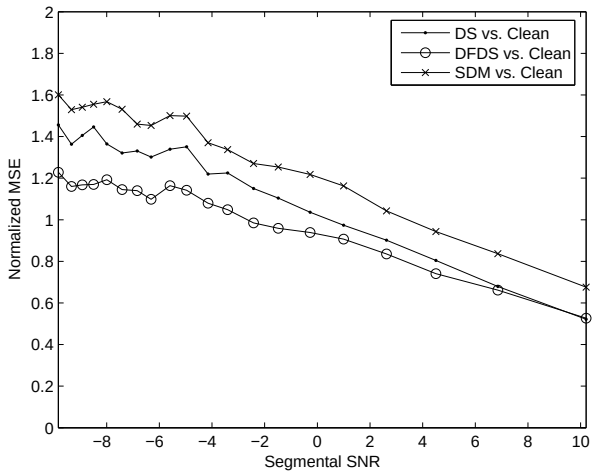


Figure 5.4: Normalized MSE of feature vectors of a speech signal corrupted with white gaussian noise, and processed using delay-and-sum (DS), delay-and-feature-domain-sum (DFDS). Single distant microphone (SDM) features were also included for further comparison.

- MFCC feature averaging seems to behave properly, yielding comparable, or better, MSE error than that of delay-and-sum, and improving single distant microphone performance as well.
- The more white gaussian noise is added, the higher performance for DFDS, compared to that of DS and SDM.

5.3.1 Implementation

Implementation of DFDS was quite straightforward taking advantage of the time-delay estimation and delay-and-sum previous implementations.

Perhaps the most important issue is normalization across several microphones. DFDS performs geometric average of the amplitude spectrum over channels. If the overall energies for each of the channels differ considerably, their amplitude spectrum mean would yield very biased estimates[14]. To overcome this, mean and variance

[14]In geometric mean, if one of the values to be averaged is very low, it will dominate the averaged output.

feature normalization was performed before averaging.

5.3.2 Evaluation

DFDS was evaluated on two of the three proposed test-beds, using non-weighted and PHAT-weighted cross-correlation for time-delay estimation.

In mismatched conditions for connected digits recognition (see Table 5.4), DFDS outperforms only a single distant microphone. DS is consistently better than DFDS achieving significant differences in WER, which is actually hard to achieve at these low WER. Although the feature MSE experiment (see Figure 5.4), shows better performance for DFDS than DS processing, extrapolation to ASR performance is shown to be dangerous, since metrics other than Euclidean distance[15] are typically used in speech recognition at the feature level. Other factors, such as the pronunciation and language model, may have a large effect on recognition accuracy. The considerable improvement after MLLR speaker adaptation suggests an important training-test mismatch.

For the matched conditions digits test-bed (MCDT), results of which are shown in Table 5.5, DFDS performance is much closer to that of DS. It is worth noting the important role of speaker adaptation for DFDS processing. In noisy conditions, the ASR performance of DFDS seems to outperform DS, achieving weak significance for non-weighted TDE and noisy speech. For noise-reduced data this difference is not as noticeable and significance is not reached, either. Nonetheless, here, DFDS behaves considerably better than in MMCDT (see Table 5.4) compared to standard delay-and-sum processing, further supporting a mismatched training-test conditions hypothesis for MMCDT. Interestingly, the lowest WER for digits recognition, across techniques and across test-beds is achieved when speaker adaptation is used along with non noise-reduced waveforms. In this case, thus, speaker adaptation takes advantage of information that is not present after noise reduction, which is probably in the form of artifacts or distortion in the speech signals.

[15]For instance, Mahalanobis distance, $(\mathbf{f_1}-\mathbf{f_2})^T\mathbf{C}^{-1}(\mathbf{f_1}-\mathbf{f_2})$, being $\mathbf{f_1}$ and $\mathbf{f_2}$ two feature vectors.

The MMCCDT test-bed was not evaluated for DFDS due to technical issues. MMCCDT includes several steps, such as segmentation, gender detection, speaker clustering and recognition. These steps use different feature extraction set-ups. Averaging features only for recognition would mean either keeping fixed the rest of steps, in which case accuracy would not be comparable to the other techniques described in the thesis, or averaging features for each of the steps, in which case different types of features would be mixed up and where the accuracy results come from would not be clear, since DFDS performance depends on the type of feature used.

5.4 Time-Frequency Masking

5.4.1 Time-Frequency Representation of Speech Signals

An extensively-used representation for speech signals is the spectrogram, which sets time and frequency information out together. Formant evolution over time and pitch contours, both crucial in understanding speech signals, are clearly displayed on this type of representation.

In a similar way, it is common to perform short-time fourier transform (STFT) [30] time-frequency analysis. Here, the input signal, $x(n)$ is decomposed into overlapping frames each of which is transformed into the frequency domain by means of the discrete Fourier transform (DFT) as

$$X_m(\omega) = \sum_{n=0}^{N-1} x_m(n) w_1(n) \mathrm{e}^{j\omega n} \tag{5.4.1}$$

where $X_m(\omega)$ is the corresponding amplitude and phase at frequency ω for the mth frame and $w_a(n)$ is a windowing sequence. This way, $x(n)$ can be understood as a collection of time-frequency dependent complex-valued cells from which the time-domain signal can be later resynthesized. To proceed, every frequency-domain frame is inverse transformed to the time-domain using an IDFT, optionally windowed, say by $w_s(n)$, to avoid time aliasing, and added to the previously overlapped resynthesized frames. Figure 5.5 shows a block diagram for a typical time-frequency domain processing system.

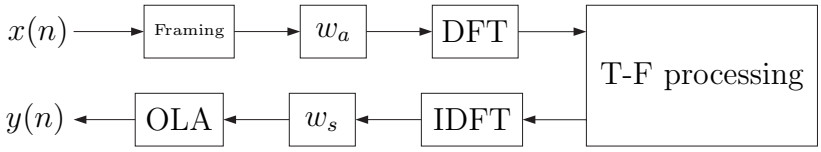


Figure 5.5: Analysis and synthesis by means of the short-time Fourier transform.

The spectrogram, $S_m(\omega)$ is related to the STFT as

$$S_m(\omega) = |X_m(\omega)|^2 \tag{5.4.2}$$

5.4.2 Dual-Microphone Phase-Error Based Filtering

Based on the time-frequency processing framework described in Section 5.4.1, phase-error based filtering (PBF) [1] processes time-frequency input cells by means of a masking approach.

As stated in [1], the mean phase error (MPV) between two speech signals, defined as

$$MPV = \sum_{m=1}^{M} \sum_{\omega=-\omega_s}^{\omega_s} \theta_{\beta,m}^2(\omega) \tag{5.4.3}$$

with

$$\theta_{\beta,m}(\omega) = \angle X_{1,m}(\omega) - \angle X_{2,m}(\omega) - \omega\beta \tag{5.4.4}$$

is a good indicator of the amount of noise and reverberation that is present in the speech signals. Here, $\angle X_{1,m}(\omega)$ and $\angle X_{2,m}(\omega)$ are the phase spectrums of the input signals at frame m, respectively, $\theta_{\beta,m}(\omega)$ is the phase-error assuming β as their TDOA, N is the number of frames in the speech segment, and $(\omega_s, -\omega_s)$ the frequency limits of the DFT analysis. Thus, phase-error is a measure of time-misalignment for each frequency bin. If the input signals are time-aligned, overall phase-error can be reduced to

$$\theta_{\beta,m}(\omega) = \angle X_{1,m}(\omega) - \angle X_{2,m}(\omega) \tag{5.4.5}$$

Phase error is, thereby, used as a time-varying criterion to enhance multi-microphone speech signals. Frequency bins with high phase-error[16] are assigned lower magnitude, to minimize its contribution to the final spectrum estimate. Zero phase-error yields, thus, non-processed amplitude spectrum bins. A block diagram of PBF is shown in Figure 5.6. Phase-error is first computed from the two phase spectrums. A masking function is then derived to weight the amplitude spectrum for each channel. Spectrums are later converted to cartesian form and summed up in a similar way to delay-and-sum and, actually, when $\gamma = 0$ delay-and-sum is obtained.

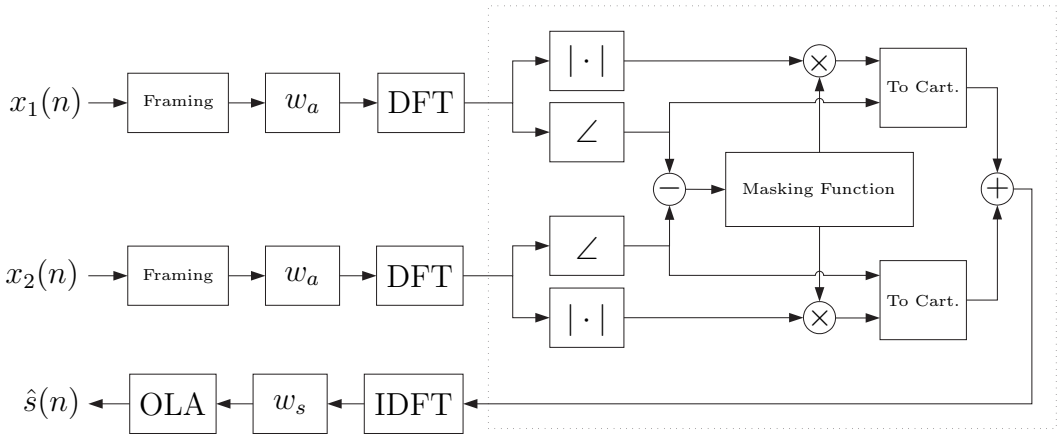


Figure 5.6: Dual-microphone phase-error based filtering block diagram.

The masking function,

$$M(\omega) = \frac{1}{1 + \gamma\theta^2_{\beta,m}(\omega)} \tag{5.4.6}$$

was proposed in [1]. Here, γ allows for phase-error weighting. This function is shown in Figure 5.7 for several values of γ. Larger values for γ yield higher attenuation after masking.

[16]Non time-aligned sinusoids result in higher phase-error, suggesting they arrive from different spatial directions, such as different speakers or reverberation reflections.

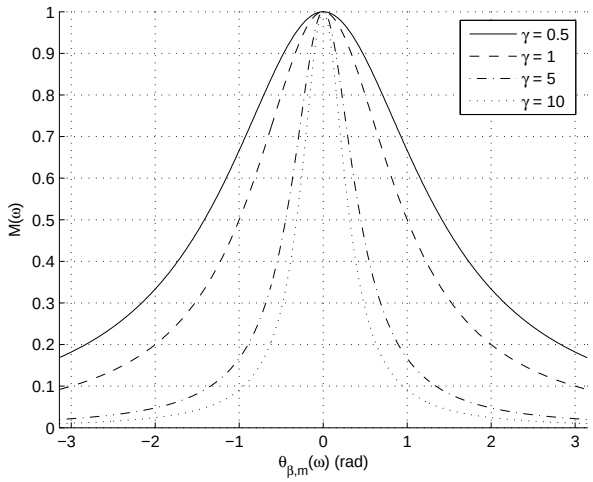


Figure 5.7: Magnitude spectrum masking function in phase-error based filtering.

5.4.3 Multiple-Microphone Phase-Error Based Filtering

Dual-microphone phase-error based filtering can be extended to more than two channels by performing masking on all possible pairs of microphones, yielding the family of masking functions

$$M_{ij}(\omega) = \frac{1}{1 + \gamma\theta_{ij}^2(\omega)} \tag{5.4.7}$$

for microphone pair i and j, which is readily extended from (5.4.6).

For each channel, several masking functions must be, thus, combined. As analyzed and proposed in [28] and [27], modified geometric mean is used for this purpose as

$$\Phi_i(\omega) = \left(\prod_{j=1, j\neq i}^{C} M_{ij}(\omega) \right)^{\frac{1}{k}} \tag{5.4.8}$$

where C is the number of microphones and k is a factor affecting the aggressiveness of the algorithm, as γ does. Geometric mean in upper-bounded by the arithmetic mean and lower-bounded by the smallest value that is being averaged. Using this approach, very high phase-error estimates dominate the mask averaging process, i.e., when a pair of microphones results in a very large phase-error for a certain frequency

bin, its corresponding masking value is close to zero. Geometrically averaging this value with the masking values for other pairs of microphones results in a masking value close to 0 anyway. For a pure geometric mean, $k = M$.

The enhanced spectrum is, thus, obtained by using the multi-channel mask $\Phi_i(\omega)$ by summing up the enhanced spectrums for each channel as

$$\hat{S}(\omega) = \sum_{i=1}^{C} \Phi_i(\omega) X_i(\omega) \tag{5.4.9}$$

5.4.4 Implementation

Multiple-microphone phase-error based filtering was implemented in MATLAB assuming time-aligned input speech signals. Since using wrong TDOA estimates may result in distortion of the desired signal, only PHAT-weighted generalized cross-correlation was used[17,18].

The frame size was set to 1024 samples at 16ksps as suggested in [1]. Smaller frame sizes resulted in stronger musical noise, probably due to less reliable phase estimates. The frame shift was set to 10ms to match the shift in the feature extraction process.

Although phase-error, as defined in (5.4.4), follows formally from the spectrum representation of the input signals, it is a non-consistent criterion for measuring phase difference. Since phase can be understood as circular, two different phase-error measures can be derived, given two phase values. Considering (5.4.4) yields a random choice between them, depending on the particular values. To cope with this, both distance measures were calculated, and the minimum was taken for less aggressive processing.

[17] As lower WER was achieved for PHAT-DS in the previous experiments.

[18] MATLAB source code for multi-channel phase-based filtering is available on-line at http://www.icsi.berkeley.edu/Speech/papers/multimic/ .

5.4.5 Evaluation

The performance of the algorithm was first evaluated by means of informal listening, which showed comparable performance to that of delay-and-sum. Being the phase-error spectrum quite spiky (see Figure 5.8), the larger γ is, the spikier the mask is, which impacts directly on the amplitude spectrum. This results in artifacts similar to musical noise in other spectral enhancement techniques such as spectral substraction. Aggressiveness increased as γ and k were set larger, but specially k resulted in severe speech distortion, although stronger dereverberation. Since speech signals in the proposed test-beds are not highly corrupted, k was set to C to perform standard geometric mean, for less aggressive behavior.

Figure 5.8 shows the masking process for a female voiced frame in the noise-reduced Meeting Digits corpus. As shown in (c) and (d), some of the spectral peaks are attenuated due to phase mismatch after global time-alignment, breaking its formant-like structure.

Phase-error filtering was evaluated on the three proposed test-beds described in Chapter 3. Several choices for γ were tried to explore the aggressiveness of the technique under noisy and less-noisy conditions.

As summarized in Table 5.6 the phase-error based technique performs significantly better than a single distant microphone (SDM). WER, though, degrades rapidly as γ is set larger, suggesting that the algorithm is behaving too aggressive for the speech signal quality in this database, that is, the benefit obtained by dereverberation is not enough to compensate for artifacts and speech distortion. Compared to PHAT-DS, it performs worse in all cases, which indicates that masking is not beneficial for this test-bed. Even for $\gamma = 0$, where PHAT-PBF should be equal to PHAT-DS, a slight difference is found. This may be due to the STFT processing performed by PHAT-PBF, as opposed to time-domain operation of PHAT-DS. Although using MLLR speaker adaptation a considerable improvement is obtained, it is still dominated by the degradation at the signal level.

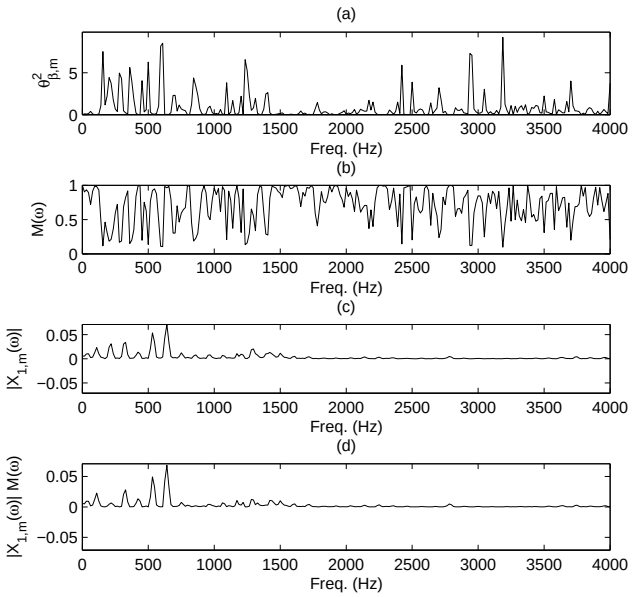


Figure 5.8: Masking process in phase-error based filtering. (a) Squared phase-error spectrum. (b) Mask. (c) Original amplitude spectrum. (d) Masked amplitude spectrum.

As for the matched training test-bed (MCDT) results shown in Table 5.7, PHAT-PBF is consistently superior to a single distant microphone and only comparable to PHAT-DS. Small values of γ resulted in lower WER than that of PHAT-DS, although the significance level was not reached.

In the mismatched conditions conversational test-bed (MMCCDT), PHAT-PBF outperforms a single distant microphone for small values of γ. For non noise-reduced waveforms, PHAT-PBF $\gamma = 0.5$ showed a very significant loss of performance compared to PHAT-PBF $\gamma = 0$, for instance, which, to our understanding, could be attributed to technical problems while running the experiment.

Compared to other multi-channel techniques, PHAT-PBF performance is only comparable to that of PHAT-DS when $\gamma = 0$, in which case PHAT-PBF should correspond to PHAT-DS. Therefore, again, phase-based masking does not seem to be beneficial, not even in the noisy case, where slightly more distortion could be allowed.

As pointed out in Section 5.4.4, non-accurate time-delay estimates might explain the poor performance obtained using phase-based filtering, since they affect the phase-error spectrum by adding an offset to (5.4.5), which could be large in terms of radians for certain frequencies. This might eventually attenuate desired speech information in its amplitude spectrum and, therefore, show a special sensitivity to time-delay estimation. Microphone lay-out, sampling rate and, especially, speech signal quality[19] might aggravate this further, since time-delay estimation accuracy depends strongly on these factors.

[19] In real noisy and reverberant speech signals very accurate delay estimates should not be expected.

Noise-Reduced		
	No MLLR	**MLLR**[a]
SDM Channel 6[b]	5.2%	2.9%
4-channel NW-DS[c]	2.4%	1.8%
4-channel PHAT-DS[d]	2.4%	1.7%
4-channel NW-DFDS[e]	3.5%	2.1%
4-channel PHAT-DFDS[f]	3.4%	2.1%
Significance Testing		
SDM Channel 6 ↔ 4-channel NW-DFDS	7.39e-11	8.65e-6
SDM Channel 6 ↔ 4-channel PHAT-DFDS	9.98e-12	3.23e-5
4-channel NW-DFDS ↔ 4-channel PHAT-DFDS	0.06	0.37
4-channel NW-DS ↔ 4-channel NW-DFDS	1.12e-11	5.48e-3
4-channel PHAT-DS ↔ 4-channel PHAT-DFDS	1.43e-9	3.58e-4

Non Noise-Reduced		
	No MLLR	**MLLR**
SDM Channel F	6.1%	3.8%
4-channel NW-DS	3.3%	2.1%
4-channel PHAT-DS	2.6%	1.9%
4-channel NW-DFDS	4.3%	2.4%
4-channel PHAT-DFDS	4.1%	2.4%
Significance Testing		
SDM Channel F ↔ 4-channel NW-DFDS	1.09e-7	2.84e-8
SDM Channel F ↔ 4-channel PHAT-DFDS	4.39e-10	1.36e-9
4-channel NW-DFDS ↔ 4-channel PHAT-DFDS	0.09	0.5
4-channel NW-DS ↔ 4-channel NW-DFDS	8.16e-6	0.04
4-channel PHAT-DS ↔ 4-channel PHAT-DFDS	2.46e-6	2.48e-4

[a] Maximum-Likehood Linear Regression Speaker Adaptation.
[b] Single distant microphone with lower error-rate.
[c] Delay-and-sum using non-weighted cross-correlation for TDOA estimation.
[d] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.
[e] Delay-and-feature-domain-sum using non-weighted cross-correlation for TDOA estimation.
[f] Delay-and-feature-domain-um using PHAT-weighted cross-correlation for TDOA estimation.

Table 5.4: WER of multiple distant microphones processed with delay-and-feature-domain-sum on the Mismatched Conditions Digit Test-bed (MMCDT).

Noise-Reduced

	No MLLR	**MLLR**[a]
SDM Channel 6[b]	3.1%	2.2%
4-channel NW-DS[c]	2.33%	1.73%
4-channel PHAT-DS[d]	2.33%	1.63%
4-channel NW-DFDS[e]	2.53%	1.7%
4-channel PHAT-DFDS[f]	2.46%	1.6%
Significance Testing		
SDM Channel 6 ↔ 4-channel NW-DFDS	1.27e-3	2.64e-6
SDM Channel 6 ↔ 4-channel PHAT-DFDS	2.08e-3	9.20e-4
4-channel NW-DFDS ↔ 4-channel PHAT-DFDS	0.36	0.17
4-channel NW-DS ↔ 4-channel NW-DFDS	0.14	0.50
4-channel PHAT-DS ↔ 4-channel PHAT-DFDS	0.21	0.49

Non Noise-Reduced

	No MLLR	**MLLR**
SDM Channel 6	3.3%	2.2%
4-channel NW-DS	2.53%	1.8%
4-channel PHAT-DS	2.4%	1.63%
4-channel NW-DFDS	2.6%	1.43%
4-channel PHAT-DFDS	2.7%	1.56%
Significance Testing		
SDM Channel 6 ↔ 4-channel NW-DFDS	1.05e-9	2.12e-8
SDM Channel 6 ↔ 4-channel PHAT-DFDS	1.65e-8	2.13e-10
4-channel NW-DFDS ↔ 4-channel PHAT-DFDS	0.27	0.19
4-channel NW-DS ↔ 4-channel NW-DFDS	0.27	0.04
4-channel PHAT-DS ↔ 4-channel PHAT-DFDS	0.08	0.38

[a] Maximum-Likehood Linear Regression Speaker Adaptation.
[b] Single distant microphone with lower error-rate.
[c] Delay-and-sum using non-weighted cross-correlation for TDOA estimation.
[d] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.
[e] Delay-and-feature-domain-sum using non-weighted cross-correlation for TDOA estimation.
[f] Delay-and-feature-domain-sum using PHAT-weighted cross-correlation for TDOA estimation.

Table 5.5: WER of multiple distant microphones processed with delay-and-feature-domain-sum on the matched conditions digit test-bed (MCDT).

Noise-Reduced

	No MLLR	MLLR[a]
SDM Channel 6[b]	5.2%	2.9%
4-channel PHAT-DS[c]	2.4%	1.7%
4-channel PHAT-PBF[d] $\gamma = 0$	2.2%	1.7%
4-channel PHAT-PBF $\gamma = 0.5$	2.4%	1.9%
4-channel PHAT-PBF $\gamma = 1$	2.7%	2.1%
4-channel PHAT-PBF $\gamma = 3$	3.4%	2.3%
4-channel PHAT-PBF $\gamma = 5$	3.9%	2.7%
Significance Testing		
SDM Channel 6 $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0$	2.34e-29	3.10e-11
4-channel PHAT-DS $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0$	0.09	0.5

Non Noise-Reduced

	No MLLR	MLLR
SDM Channel F	6.1%	3.8%
4-channel PHAT-DS	2.6%	1.9%
4-channel PHAT-PBF $\gamma = 0$	2.5%	1.8%
4-channel PHAT-PBF $\gamma = 0.5$	2.7%	2.0%
4-channel PHAT-PBF $\gamma = 1$	2.9%	2.1%
4-channel PHAT-PBF $\gamma = 3$	3.8%	2.6%
4-channel PHAT-PBF $\gamma = 5$	4.4%	3.0%
Significance Testing		
SDM Channel F $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0$	6.25e-38	1.03e-19
4-channel PHAT-DS $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0$	9.60e-3	0.09

[a] Maximum-Likehood Linear Regression Speaker Adaptation.
[b] Single distant microphone with lower error-rate.
[c] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.
[d] Phase-Error Based Filtering using PHAT-weighted cross-correlation for TDOA estimation.

Table 5.6: WER of multiple distant microphones processed with Phase-Error Based Filtering on the Mismatched Conditions Digit Test-bed (MMCDT).

Noise-Reduced

	No MLLR	**MLLR**[a]
SDM Channel 6[b]	3.1%	2.2%
4-channel PHAT-DS[c]	2.33%	1.63%
4-channel PHAT-PBF[d] $\gamma = 0$	2.43%	1.63%
4-channel PHAT-PBF $\gamma = 0.5$	2.4%	1.66%
4-channel PHAT-PBF $\gamma = 1$	2.5%	1.73%
4-channel PHAT-PBF $\gamma = 3$	2.66%	1.73%
4-channel PHAT-PBF $\gamma = 5$	2.93%	1.86%
Significance Testing		
SDM Channel 6 $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0$	1.23e-3	3.10e-3
4-channel PHAT-DS $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0$	0.31	0.5

Non Noise-Reduced

	No MLLR	**MLLR**
SDM Channel 6	3.3%	2.2%
4-channel PHAT-DS	2.4%	1.63%
4-channel PHAT-PBF $\gamma = 0$	2.3%	1.56%
4-channel PHAT-PBF $\gamma = 0.5$	2.33%	1.5%
4-channel PHAT-PBF $\gamma = 1$	2.43%	1.5%
4-channel PHAT-PBF $\gamma = 3$	2.73%	1.86%
4-channel PHAT-PBF $\gamma = 5$	2.93%	2.13%
Significance Testing		
SDM Channel 6 $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0$	5.57e-6	1.77e-4
SDM Channel 6 $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0.5$	8.01e-6	1.13e-4
SDM Channel 6 $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 1$	2.04e-4	4.41e-5
4-channel PHAT-DS $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0$	0.18	0.31
4-channel PHAT-DS $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 0.5$	0.25	0.1
4-channel PHAT-DS $\leftrightarrow$ **4-channel PHAT-PBF** $\gamma = 1$	0.5	0.11

[a] Maximum-Likehood Linear Regression Speaker Adaptation.
[b] Single distant microphone with lower error-rate.
[c] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.
[d] Phase-Error Based Filtering using PHAT-weighted cross-correlation for TDOA estimation.

Table 5.7: WER of multiple distant microphones processed with Phase-Error Based Filtering on the matched conditions digit test-bed (MCDT).

Noise-Reduced

	Overall	ICSI	NIST	LDC	CMU
SDM[a]	48.4%	34.7%	48.2%	56.2%	62.1%
PHAT-DS[b]	43.2%	25.9%	45.5%	50.2%	62.1%
PHAT-PBF[c] $\gamma = 0$	43.5%	26.1%	46.6%	50.6%	62.1%
PHAT-PBF $\gamma = 0.5$	46.5%	29.8%	50.6%	54.8%	62.1%
Significance Testing					
SDM $\leftrightarrow$ **PHAT-PBF** $\gamma = 0$	1.14e-8				
SDM $\leftrightarrow$ **PHAT-PBF** $\gamma = 0.5$	0.09				
PHAT-DS $\leftrightarrow$ **PHAT-PBF** $\gamma = 0$	0.36				
PHAT-DS $\leftrightarrow$ **PHAT-PBF** $\gamma = 0.5$	3.22e-4				

Non Noise-Reduced

	Overall	ICSI	NIST	LDC	CMU
SDM	50.1%	37.4%	49.4%	56.2%	64.8%
PHAT-DS	45.7%	28.8%	49.2%	52.0%	64.8%
PHAT-PBF $\gamma = 0$	45.4%	27.8%	47.8%	53.1%	64.8%
PHAT-PBF $\gamma = 0.5$	55.0%	47.9%	52.3%	59.0%	64.8%
Significance Testing					
SDM $\leftrightarrow$ **PHAT-PBF** $\gamma = 0$	1.25e-6				
SDM $\leftrightarrow$ **PHAT-PBF** $\gamma = 0.5$	9.71e-4				
PHAT-DS $\leftrightarrow$ **PHAT-PBF** $\gamma = 0$	0.36				
PHAT-DS $\leftrightarrow$ **PHAT-PBF** $\gamma = 0.5$	1.96e-11				

[a] Single distant microphone.
[b] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.
[c] Phase-Error Based Filtering using PHAT-weighted cross-correlation for TDOA estimation.

Table 5.8: WER of multiple distant microphones processed with Phase-Error Based Filtering on the Mismatched Conditions Conversational Test-bed (MMCCDT).

Chapter 6

Dereverberation Techniques Based On Linear Prediction

In the previous chapter, TDE-based dereverberation techniques made a strong simplification of the reverberating process. This chapter focuses on the use of linear prediction for dereverberation purposes. An overview of linear prediction and autoregressive modelling is first set out. Next, how dereverberation techniques can benefit from linear prediction is described. Finally, correlation shaping is explored, as well as evaluated on the proposed ASR test-beds, as an example of this type of techniques.

6.1 Speech production, Autoregressive Modelling and Linear Prediction

Speech signal production can be explained by the source-filter model, by which an excitation source signal, intended to be related to the lungs and the vocal chords, passes through a filter, which approximates vocal tract behavior, to eventually produce the observed speech signal.

Figure 6.1: Source-filter model for speech signal production.

The vocal tract can also be thought of as a varying cross-section area lossless tube. These changes in section area result in transmitted and reflected waves[1], creating a reverberant environment, which is overall modelled by the filter previously described. An autoregressive (AR) rational model such as

$$x(n) = \sum_{p=1}^{P} a(p)x(n-p) + e(n) \tag{6.1.1}$$

in the time domain, or

$$X(z) = \frac{E(z)}{A(z)} = \frac{1}{1 + \sum_{p=1}^{P} a(p)z^{-p}} E(z) \tag{6.1.2}$$

in z-transformed domain, is specially well-suited to this kind of phenomenon. Here $x(n)$, $X(z)$ is the observed signal, $e(n)$, $E(z)$ is the input excitation signal, $1/A(z)$ is the production filter, and P is the order of the AR model. As it can be noted from (6.1.1), the output of the model at time n is explained by the excitation $e(n)$, but also by the past outputs $x(n-p)$, which model reverberant behavior.

Linear prediction analysis [31] is a procedure to optimally estimate $a(p)$ or, equivalently $A(z)$, only using the observed waveform. The AR production model in (6.1.1) can be rearranged in the form of a linear predictor, as

$$e(n) = x(n) - \sum_{p=1}^{P} a(p)x(n-p) \tag{6.1.3}$$

in which, $x(n)$ is predicted from past $x(n-p)$ values, and $a(n)$ is chosen minimize the prediction error $e(n)$, which corresponds to the excitation sequence. Mean-square error (MSE) is the most accepted criterion for prediction error minimization although other approaches have been successfully applied [4]. For MSE minimization, the cost function

$$\xi = \sum_{n=0}^{N-1} e^2(n) \tag{6.1.4}$$

[1] Acoustic energy is not propagated in a straight line along the vocal tract. Instead, changes in section area result in mismatch in acoustic impedance and, thus, reflected and transmitted waves. Thus, energy is not propagated straight along the vocal tract.

is minimized, where N is the time span over which the prediction error is to be minimized. Speech is not stationary, since linguistic information is produced by the variations over time of the articulator organs. For this reason, the analysis must be carried out over short-time segments of speech[2]. Minimization is achieved by solving a set of linear equations [23] [4], typically by means of the Levinson-Durbin recursion [4].

For a true AR process, if the order P is chosen accurately, $e(n)$ is necessarily to be white gaussian noise. For speech signals, the order is determined so that $a(p)$ captures most of the vocal tract articulation information[3]. The linear prediction residual of clean speech signals, $e(n)$, lacks most of the articulation-related correlation. For unvoiced sounds (see Figure 6.2 (g)), it can be considered roughly white. For voiced sounds (see Figure 6.2 (h)), the glottal closures add a spiky character to it. Nonetheless, the linear prediction residual signal is much whiter than the input waveform.

6.2 Linear Prediction in adverse environments

6.2.1 Noise and reverberation

Linear prediction analysis in (6.1.3) assumes $e(n)$ and $a(p)$ to be the source excitation and the prediction filter associated with an AR model, but no other components. The way it operates is such that everything that can not be modelled as an AR process, that is, anything that can not be linearly predicted, is left to $e(n)$. Therefore, linear prediction analysis has no special mechanism to robustly estimate its parameters.

A complementary way to understand what happens when noisy and reverberant speech is analyzed using linear prediction, is through spectral envelope reasoning. Linear prediction approximates the spectral envelope of the input waveform as the spectrum of an AR model, so any disturbance is fitted into the model as well. For speech signals, the poles of the AR model are usually close to the unit circle, yielding

[2]Short enough to be considered to have stationary statistics, about 20-30ms.

[3]Up to 4-6 resonant frequencies (formants) can be identified in wide-band spectrograms of speech signals. Assuming each formant to be identified with a pair of complex poles in (6.1.2), an order of 8 to 12 would be required.

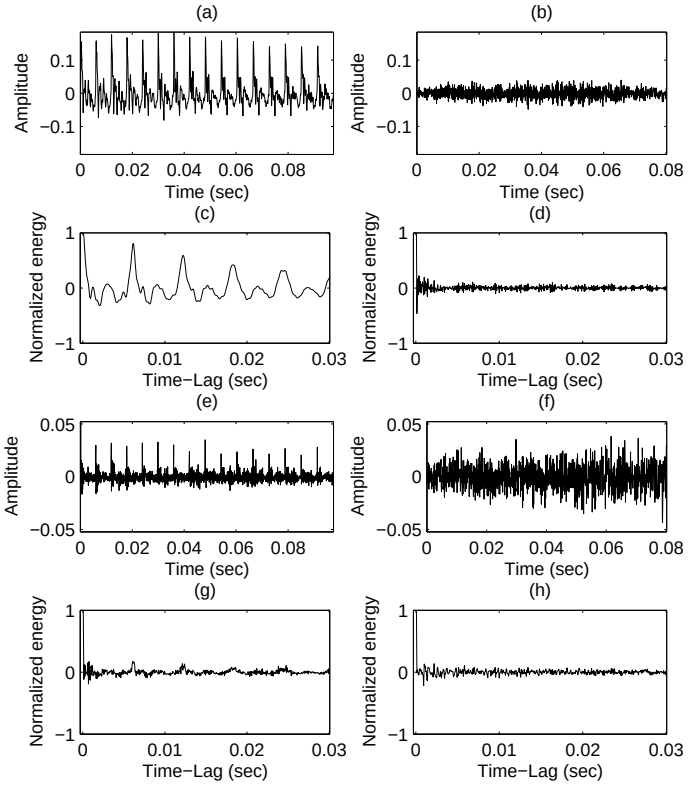


Figure 6.2: Linear prediction analysis. (a) Voiced segment. (b) Unvoiced segment. (d) Autocorrelation function (AF) of (a). (d) AF of (b). (e) LP residual of a voiced segment. (f) LP residual of an unvoiced segment. (g) AF of (e). (h) AF of (f).

high Q-factors for its second order sections[4]. Slight deviations in $a(n)$ can bias the center frequency of the formants, significantly affecting the accuracy of the analysis procedure.

On the other hand, reverberation, as well as speech, can also be explained by an AR model. When reverberant speech is analyzed, thus, the observed signal can be overall modelled by just an AR filter. In principle, there is no way to split speech AR

[4] $A(z)$ can be factored into second order sections, each of which corresponds to a band-pass filter. These are typically related to the formants in speech production.

information from AR reverberation information. In practice, due to the physical constraints involved in production of speech and production of reverberation, a carefully chosen predictor order can get rid of most of the reverberation. This fact is used in dereverberation techniques based on linear prediction analysis.

For further discussion on linear prediction in reverberant environments please refer to [13].

6.2.2 Linear Prediction-based Dereverberation

Although, as discussed in Section 6.2.1, reverberation can impair linear prediction accuracy of speech signals, short-term linear prediction is becoming a popular preprocessing step for dereverberation algorithms. When the prediction order is low enough, linear prediction only accounts for short-term correlation (only a 12 samples span for $P = 12$) and, thus, it is not able to deal with long-term correlation properly. The prediction residual roughly contains the speech source excitation signal and reverberation. At this point, if the prediction residual of clean speech was assumed to be white, the autocorrelation function of the reverberant residual would coincide with the autocorrelation function of the speaker-to-receiver impulse response and, thus, reverberation would have been isolated. Unfortunately, this assumption is always violated, strictly speaking.

Speech articulation has a much higher variation rate than any speaker-to-receiver impulse response, for which changes can be considered as very slowly varying. The latter depends mostly on the room characteristics, and the poses of the speaker and the receiver. In practical situations, only changes in the pose of the speaker affect, since rooms and receivers are supposed to be static. This can be taken advantage of to further isolate reverberation. In Figure 6.3 the whole utterance is averaged for computing the autocorrelation function of the prediction residual since stationarity of the impulse response is assumed. Here, speech correlation is averaged over all the utterance resulting in severe distortion and even cancellation, depending on the uttered speech. Figure 6.3 (d) and (f) show that using the whole utterance to estimate its autocorrelation function most of the reverberation information is preserved. On

the other hand, Figure 6.3 (e) shows the 30ms-long autocorrelation function of the prediction residual in which reverberation structure is not clear due to high short-term speech correlation interference.

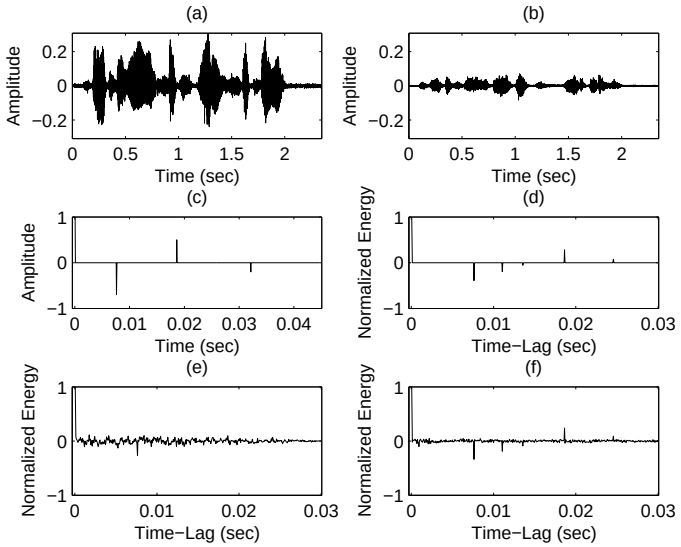


Figure 6.3: Isolating reverberation using linear prediction analysis. (a) A close-talking microphone utterance. (b) Linear prediction residual of (a). (c) A simple speaker-to-receiver impulse response. (d) Autocorrelation function (AF) of (c). (e) AF of a 30ms long prediction residual unvoiced segment in (b). (f) AF of the whole prediction residual in (b).

Furthermore, perceptual linear prediction (PLP), typically used for feature extraction in speech recognition, could also be used to obtain a perceptual linear prediction residual on which dereverberation algorithms could operate.

6.3 Correlation Shaping

Correlation shaping (CS) [15] is a technique aimed to reshape the autocorrelation function of a signal by means of linear filtering. Linear prediction residual signals, on the other hand, can be used as inputs for speech dereverberation algorithms for not interfering with speech articulation, as shown in Section 6.2.2. Dereverberation can, therefore, be achieved by reshaping the linear prediction residual to a Kronecker delta

function, $\delta(n)$, that is, whitening it. Such decorrelation process is achieved through adaptive filtering as shown in Figure 6.4.

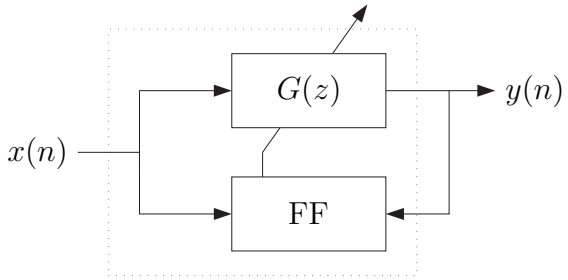


Figure 6.4: Correlation shaping block diagram.

As Figure 6.4 shows, a forward pass filters the input signal according to the current filter, $G(z)$, to make up the output signal. A feedback function, FF, takes both input and output to adapt the filter. The adaptation criterion is that of reshaping the output correlation function, $r_{yy}(\tau)$, to a certain desired shape, $r_{dd}(\tau)$. One way of achieving this is to minimize the MSE between the current and target output correlation function for each lag as

$$e(\tau) = (r_{yy}(\tau) - r_{dd}(\tau))^2 \tag{6.3.1}$$

where $e(\tau)$ is the error corresponding to autocorrelation lag τ. Since, $r_{yy}(\tau)$ depends on the shaping filter, $G(z)$, with impulse response $g(n)$, as

$$y(n) = \sum_{m=0}^{M-1} g(m)x(n-m) \tag{6.3.2}$$

and

$$r_{yy}(\tau) = \sum_{n=0}^{N-1} y(n)y(n-\tau) \tag{6.3.3}$$

where N is the number of samples over which autocorrelation[5] is computed, τ is the correlation lag, M is the equalizer length, the Least-Mean Squares LMS gradient can be found (see Appendix A.3) to be

[5]Here, $r_{yy}(\tau)$, which is strictly $\mathrm{E}[y(n)y(n-\tau)]$, has been replaced by its time averaged estimate.

$$\nabla(l) = \sum_{\tau} \frac{\partial e(\tau)}{\partial g(l)} = \sum_{\tau} \left(r_{yy}(\tau) - r_{dd}(\tau)\right) \left(r_{yx}(l-\tau) + r_{yx}(l+\tau)\right) \tag{6.3.4}$$

and the gradient descent update equation becomes

$$g(l, n+1) = g(l, n) - \mu \nabla(l) \tag{6.3.5}$$

where μ is the learning rate parameter.

For dereverberation purposes, the linear prediction residual is fed into the correlation shaping processor, as shown in Figure 6.5, and the target output correlation is set to be $r_{dd}(\tau) = \delta(\tau)$.

By further exploiting autocorrelation symmetry, (6.3.4) can be simplified as

$$\nabla(l) = \sum_{\tau>0} r_{yy}(\tau) \left(r_{yx}(l-\tau) + r_{yx}(l+\tau)\right) \tag{6.3.6}$$

In (6.3.6), error for lag 0 is specified as r_{yy} instead of $r_{yy} - r_{dd}$. This yields a different energy for the output signal which is not critical unless it vanishes. To cope with it, [15] proposed a gradient normalization as

$$\nabla'(l) = \frac{\nabla(l)}{\sqrt{\sum_j \nabla^2(j)}} \tag{6.3.7}$$

and, therefore, the filter update equation becomes eventually

$$g(l, n+1) = g(l, n) - \mu \nabla'(l) \tag{6.3.8}$$

As shown in Figure 6.5, the dereverberated speech signal can be obtained by directly applying the equalizer $g(l, n)$ onto the input signal in order to avoid linear prediction reconstruction artifacts[6].

[6]If the equalizer is assumed to be very slow time-varying, it can be approximated by a linear operator. Thus, equalizer and linear prediction systems could be swapped and the dereverberated speech signal could be obtained by directly applying the equalizer onto the input speech.

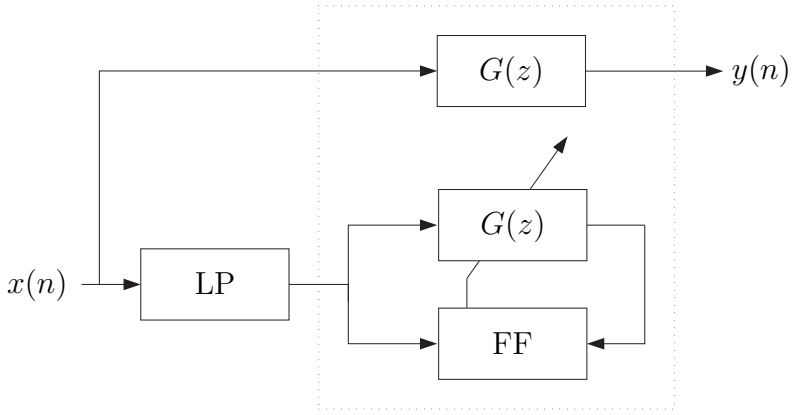


Figure 6.5: Single-channel correlation shaping block diagram.

6.3.1 Weighted Correlation Shaping

Correlation shaping can be easily modified to allow for error weighting independently for each autocorrelation lag, as

$$e(\tau) = w(\tau)(r_{yy}(\tau) - r_{dd}(\tau))^2 \tag{6.3.9}$$

and its corresponding gradient (see Appendix A.3),

$$\nabla(l) = \sum_{\tau>0} w(\tau) r_{yy}(\tau) \left(r_{yx}(l-\tau) + r_{yx}(l+\tau)\right) \tag{6.3.10}$$

where $w(\tau)$ is the weighting sequence.

For dereverberation purposes, larger weights can be applied to furthest autocorrelation lags, thus aiming for shortening reverberation time.

6.3.2 Don't Care Region

Based on the assumption that reverberation time is specially harmful [15], both from the human perception point of view and for speech recognition, a Don't Care region in the desired output autocorrelation function can be included to improve the whitening process for long lags at the expense of allowing higher level short-term correlation. This is achieved by not including the first autocorrelation lags in the gradient computation, as

$$\nabla(l) = \sum_{\tau \geq \tau_0, \tau > 0} r_{yy}(\tau) \left(r_{yx}(l-\tau) + r_{yx}(l+\tau) \right) \tag{6.3.11}$$

where lags from 1 to τ_0 don't contribute to gradient calculation. (6.3.11) is equivalent to applying

$$w(\tau) = \begin{cases} 1 & \text{for } \tau = 0 \\ 0 & \text{for } 0 < \tau < \tau_0 \\ 1 & \text{for } \tau \geq \tau_0 \end{cases} \tag{6.3.12}$$

as a weighting function for the weighted correlation shaping in Section 6.3.1.

6.3.3 Multi-channel Correlation Shaping

Correlation shaping can be easily extended to multiple channels. As stated by the MINT theorem (see Section 4.5), equalization is improved by using multiple equalizers, under certain conditions. In a similar way, whitening is expected to improve when combining multiple-microphone speech signals. Figure 6.6 shows a 2-channel version of this technique.

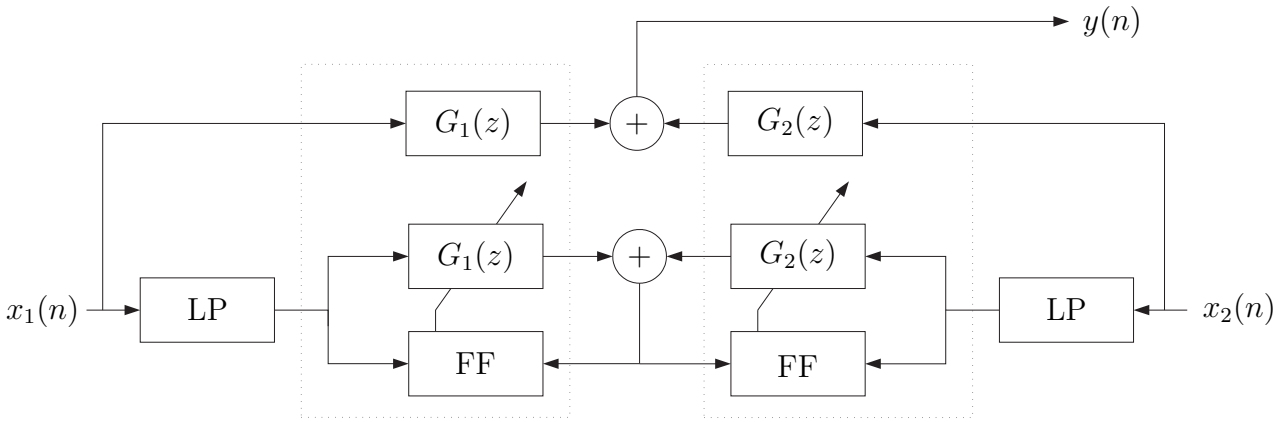


Figure 6.6: 2-channel Correlation shaping block diagram.

Since the output is now

$$y(n) = \sum_{c=1}^{C} \sum_{m=0}^{M-1} g_c(m) x_c(n-m) \tag{6.3.13}$$

the LMS gradient, for the don't care and weighted case, becomes (see Appendix A.3)

$$\nabla_c(l) = \sum_{\tau \geq \tau_0, \tau > 0} w(\tau) r_{yy}(\tau) \left(r_{yx_c}(l-\tau) + r_{yx_c}(l+\tau) \right) \tag{6.3.14}$$

and the filter update equation,

$$g_c(l, n+1) = g_c(l, n) - \mu \nabla'_c(l) \tag{6.3.15}$$

with

$$\nabla'_c(l) = \frac{\nabla_c(l)}{\sqrt{\sum_{c=1}^{C} \sum_j \nabla_c^2(j)}} \tag{6.3.16}$$

6.3.4 Implementation

Both single-channel and multi-channel versions of correlation shaping were implemented in MATLAB[7]. On-line adaptive correlation shaping, as proposed in [15], was first tested, but serious issues, such as no convergence and very low shaping performance, were encountered. Since in the proposed test-beds utterances are already segmented aimed at including only one speaker, stationarity of the speaker-to-receiver impulse response was assumed and, therefore, the whole speech segment was used to estimate autocorrelation functions. The algorithm showed to be significantly more stable, although convergence speed was still judged to be slow. This approach results in severe distortion for time-varying signals, such as linear prediction residuals of speech, but in a more reliable estimation for stationary components, such as the speaker-to-receiver impulse response. On the other hand, when stationarity is violated, for instance, when the speaker is changing his/her pose or when overlapping speech is present, dereverberation decreases its expected performance or may even

[7]MATLAB source code for single-channel and multi-channel correlation shaping processing is available on-line at http://www.icsi.berkeley.edu/Speech/papers/multimic/ .

fail. We understand that blind dereverberation is a tough enough problem to further cope with dynamic behavior.

Correlation shaping is very demanding in terms of computational cost. For each filter update iteration, several autocorrelation and cross-correlation functions must be estimated. Thus, computation time becomes an important issue when they need to be evaluated for several channels and for many lags, say, up to τ_{max}, which should raise up to hundreds of ms for real reverberation. To lower the computational load of the algorithm, $r_{yx}(\tau)$ and $r_{yy}(\tau)$ are estimated from the input cross-correlation functions as

$$r_{yx_c}(\tau) = \sum_{k=1}^{C} \sum_{m=0}^{M-1} g_k(l) r_{x_k x_c}(\tau - l) \tag{6.3.17}$$

and

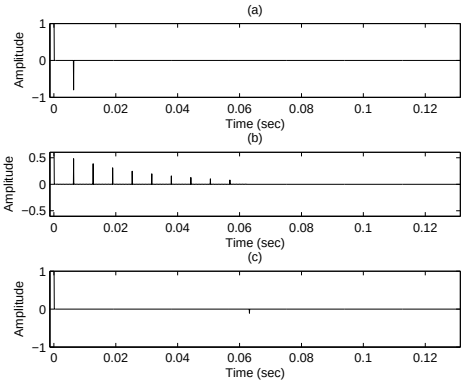
$$r_{yy}(\tau) = \sum_{k=1}^{C} \sum_{m=0}^{M-1} g_k(l) r_{yx_c}(\tau + l) \tag{6.3.18}$$

from the set $r_{x_i x_j}(\tau), \forall i, j$ of autocorrelation function pairs, as proposed in [15]. Therefore, input autocorrelation functions are just estimated once, as an initialization procedure. In our case, this approach is far more efficient than estimating cross-correlation and output autocorrelation directly from filtering the input signal[8]. However, care must be taken at the autocorrelation function boundaries, since transients appear due to the filtering procedure. To cope with this issue, the filtering procedure was performed on extended-lag auto and cross-correlation functions, followed by a trimming step to discard the transients.

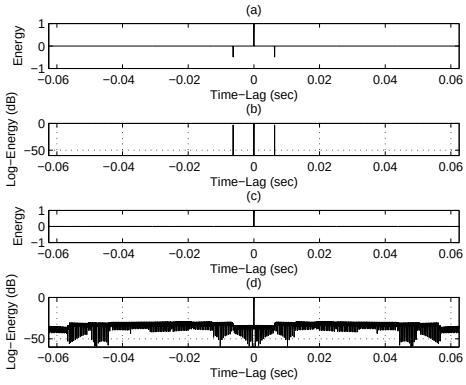
6.3.5 Evaluation

To first validate our implementation, a simple speaker-to-receiver impulse response, the one shown in Figure 4.3 (a), was used as the input signal for the correlation

[8]Direct autocorrelation estimation involves a cost of $O(N\tau_{max})$ MAC operations, where N is the length of the signal over which autocorrelation is calculated. For autocorrelation filtering a cost of $O(M\tau_{max})$ is needed, where M is the filter length. Therefore, if the whole speech segment is used to compute autocorrelation, $M \ll N$.



6.7.1: (a) Speaker-to-receiver impulse response. (b) Equalizer impulse response found through correlation shaping. (c) Equalized impulse response.



6.7.2: Linear-scaled (a) and log-scaled (b) autocorrelation function (AF) of the speaker-to-receiver impulse response in 6.7.1(a). Linear-scaled (c) and log-scaled (d) AF of the output.

Figure 6.7: Single-channel correlation shaping technique using a simple speaker-to-receiver impulse response as the input signal.

shaping processor. Thus, it is ensured that the autocorrelation function of the input is exactly the same as the speaker-to-receiver one. To be able to compare results with those in Figure 4.3, 62.5ms (or 1000 taps at 16ksps) equalizer length was used too. The learning rate was set to 1e-2 and the autocorrelation function was estimated up to $\tau = 1000$ samples (62.5ms). Convergence time was judged to be fast. After 300 filter updates, convergence is thought to be achieved. Figure 6.7.1 (c) shows perfect equalization[9] within the first 50ms, just as in the truncated theoretical impulse response inversion example in Figure 4.3. Regarding its correlation behavior, Figure 6.7.2 shows how the whitening process has successfully removed echo energy. Nonetheless, this energy is now evenly spread along the correlation function at the level of -30dB, instead of -∞dB at the input, as shown in Figures 6.7.2 (b) and (d).

To further evaluate correlation shaping, white noise was convolved with a real and non minimum-phase truncated speaker-to-receiver impulse response[10], signal that was

[9]This simple speaker-to-receiver impulse response is minimum-phase and, therefore, invertible by a causal system.

[10]Taken from the varechoic chamber at Bell Labs. 100% open panels were used. Only the first 62.5ms were used in the experiment.

used as the input[11] to the single-channel correlation shaping processor. The equalizer filter length was set at 1000 taps (62.5ms@16ksps) and initialized with delay-and-sum derived impulse responses[12]. 0ms and 18.7ms were used for the don't care region. For both approaches, convergence was judged to be guaranteed after 2000 iterations using 2.5e-4 as the learning rate, although the don't care approach showed much slower convergence.

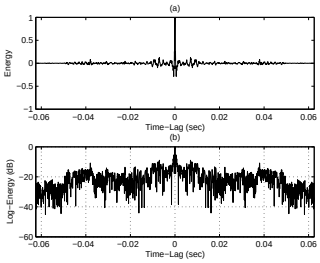
Figures 6.8.1 6.8.2 and 6.8.3 show the autocorrelation functions of the input signal, and output signals without and with don't care region, respectively. Figure 6.8.3 shows that long-term whitening can be improved by about 10dB by using the don't care approach. This is, of course, at expense of introducing short-term correlation up to 18.7ms. Figures 6.8.4 and 6.8.5 show impulse response behavior. Using the don't care approach modifies the equalized impulse response in a similar way as for autocorrelation, that is, not equalizing properly within the don't care region span, but improving long-term behavior.

Similar experiments are shown in Figure 6.9. Here, the speaker-to-receiver impulse response was not truncated, having a SRR of 1dB and a RT60 of 0.3s. The equalizer length was set at 62.5ms and correlation shaping was performed up to 125ms. Autocorrelation functions are similar to those in the previous experiment. Nonetheless, correlation shaping was not achieved beyond the equalizer span, 62.5ms. This is clearly shown in Figure 6.9.3 where, after 62.5ms, the whitening level becomes about 7 dB higher, being even comparable to the original non-shaped autocorrelation (see Figure 6.9.1).

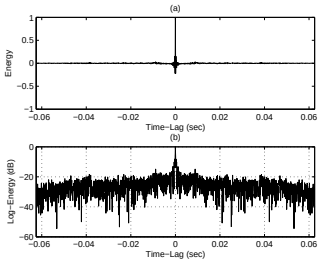
A 4-channel version of correlation shaping was also implemented. Figure 6.10 shows the results of these experiments. Three approaches, no don't care (DC) region, 18.7ms long DC region and 18.7ms long DC region plus exponential weighting were explored. Figures 6.9.2 and 6.10.1 show a very slight improvement, about 3-4dB for long-term autocorrelation, on the whitening level when using 4-channel CS. For the

[11] The autocorrelation function of white noise is asymptotically $\delta(\tau)$ function and, therefore, the autocorrelation function of the input is nearly the speaker-to-receiver impulse response's.

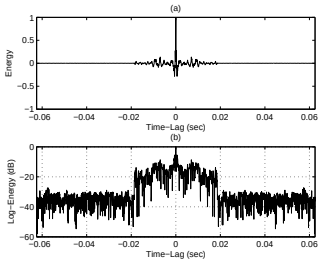
[12] A delta function centered at the maximum correlation delay.



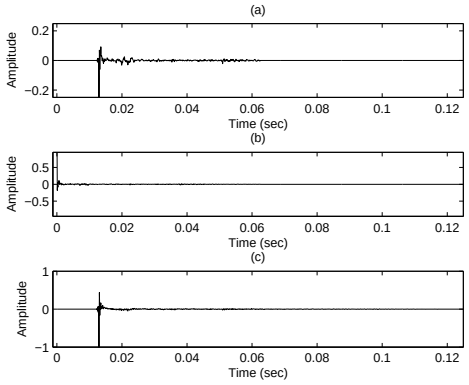
6.8.1: Linear-scaled (a) and log-scaled (b) autocorrelation function (AF) of the input signal.



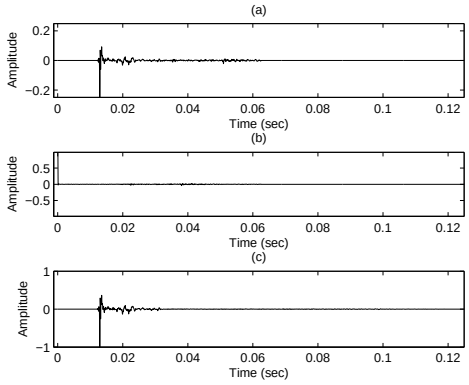
6.8.2: Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with no don't care region.



6.8.3: Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with 18.7ms long don't care region.

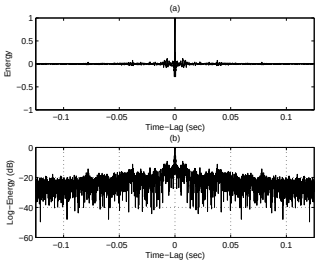


6.8.4: (a) Speaker-to-receiver impulse response. (b) Resulting equalizer using CS with no don't care region. (c) Equalized impulse response.

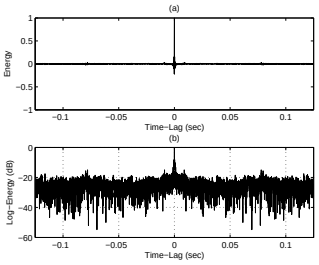


6.8.5: (a) Speaker-to-receiver impulse response. (b) Resulting equalizer using CS with 18.7ms long don't care region. (c) Equalized impulse response.

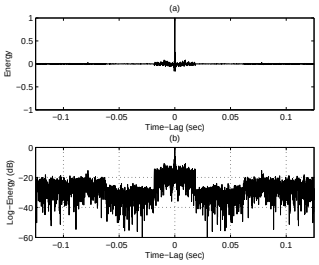
Figure 6.8: Single-channel correlation shaping (CS) technique using white noise convolved with a real truncated speaker-to-receiver impulse response as the input signal.



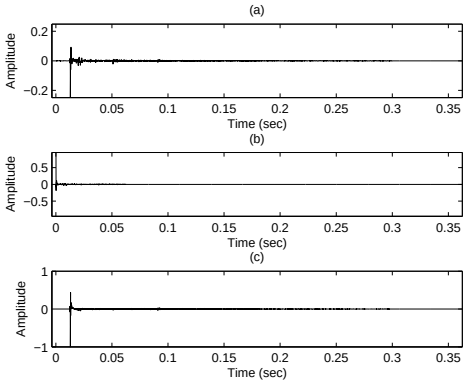
6.9.1: Linear-scaled (a) and log-scaled (b) autocorrelation function (AF) of the input signal.



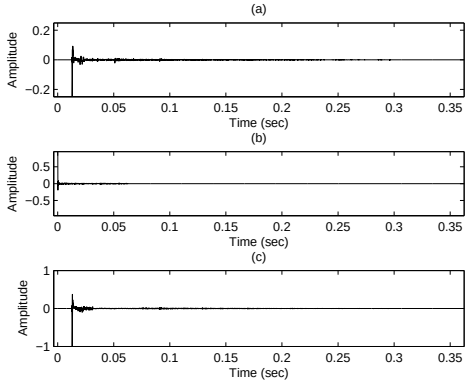
6.9.2: Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with no don't care region.



6.9.3: Linear-scaled (a) and Log-scaled (b) AF of the output signal, using CS with 18.7ms long don't care region.



6.9.4: (a) Speaker-to-receiver impulse response. (b) Resulting equalizer using CS with no don't care region. (c) Equalized impulse response.



6.9.5: (a) Speaker-to-receiver impulse response. (b) Resulting equalizer using CS with 18.7ms long don't care region. (c) Equalized impulse response.

Figure 6.9: Single-channel correlation shaping (CS) technique using white noise convolved with a real speaker-to-receiver impulse response as the input signal.

DC approach, Figures 6.9.3 and 6.10.2 shown a more noticeable improvement, about 4-5dB for lags further than 18.7ms. Nonetheless, a considerable drop in whitening performance is still present after 62.5ms, the length of the equalizers. Using the weighting function shown in Figure 6.10.7 showed to be effective for reducing this difference (see Figure 6.10.3).

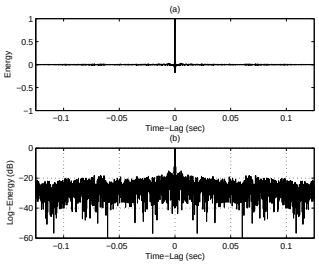
As far as impulse responses are concerned, using the don't care approach results in compacting impulse response energy towards the direct path (see Figures 6.10.4 and 6.10.5). Exponential weighting also helps in this line, as shown in Figure 6.10.6.

[15] explained the don't care approach of multi-channel correlation shaping in terms of reverberation time shortening. These measures were not carried out in this work, since, intuitively, not very good performance was expected, from inspection of Figures 6.10.4, 6.10.5 and 6.10.6. This might be due to some implementation issue or bug, although proper correlation shaping behavior is judged. Informal listening showed no better quality than delay-and-sum. Furthermore, using filters longer than 62.5ms still resulted in waveforms which were perceived as more reverberant, regardless of the approach used.

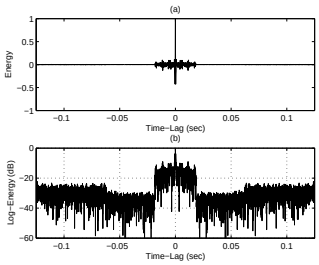
Nonetheless, speech recognition evaluations were run on the three proposed testbeds. Since the computational load of correlation shaping is very high, the amount of channels to be used was limited to 4, based on a maximum energy criterion[13]. Furthermore, only a limited set of set-ups, those that were expected to work best, were evaluated. In this line, 62.5ms long equalizers, a 18.7ms long don't care region and exponential weighting were always used. Two computationally reasonable values for τ_{max}, 62.5ms and 125ms, were taken. Higher values required an unpractical amount of computation time.

In mismatched digits recognition experiments, shown in Table 6.1, correlation shaping outperforms a single distant microphone in terms of WER for both correlation shaping spans, $\tau_{max} = 62.5ms$ and $\tau_{max} = 125ms$. Nonetheless, this difference gets

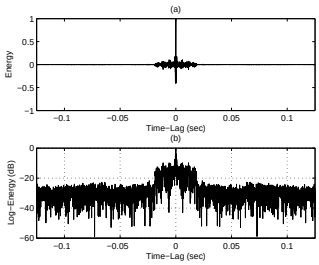
[13]For distant microphone signals, energy is typically correlated with distance, and distance with reverberation. Selecting the most energetic signals is a very simple test aimed at getting the less reverberant ones.



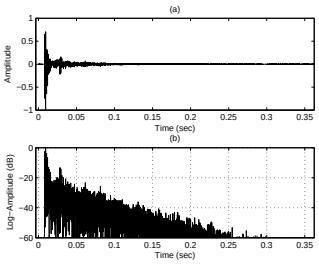
6.10.1: Linear-scaled (a) and log-scaled (b) autocorrelation function (AF) of the output signal, using CS with no don't care region.



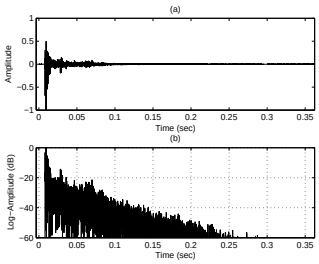
6.10.2: Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with 18.7ms long don't care region.



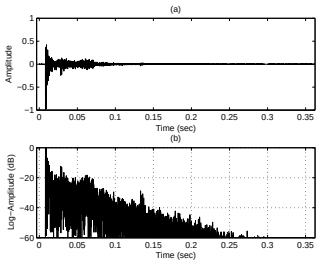
6.10.3: Linear-scaled (a) and log-scaled (b) AF of the output signal, using CS with 18.7ms long don't care region and exponential weighting.



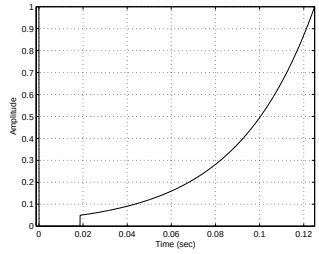
6.10.4: Linear-scaled (a) and log-scaled (b) equalized speaker-to-receiver impulse response using CS with no don't care region. (c) Equalized impulse response.



6.10.5: Linear-scaled (a) and log-scaled (b) equalized speaker-to-receiver impulse response using CS with 18.7ms long don't care region.



6.10.6: Linear-scaled (a) and log-scaled (b) equalized speaker-to-receiver impulse response using CS with 18.7ms long don't care region and exponential weighting.



6.10.7: Exponential weighting function used in Figures 6.10.3 and 6.10.6.

Figure 6.10: 4-channel correlation shaping (CS) technique using white noise convolved with real speaker-to-receiver impulse responses as the input signal.

lower for $\tau_{max} = 125ms$, which might be related to the loss of whitening performance shown in Figure 6.9. This behavior is consistent across noise-reduced and non noise-reduced data sets. Furthermore, informal listening on waveforms shaped with $\tau_{max} = 125ms$ were judged to be more reverberant than with $\tau_{max} = 62.5ms$.

Correlation shaping achieved considerably lower recognition accuracy than PHAT-DS processing, reaching significance in almost all cases. Thus, PHAT-DS reduces training-test mismatch more efficiently than CS.

Noise-Reduced

	No MLLR	**MLLR**[a]
SDM Channel 6[b]	5.2%	2.9%
4-channel PHAT-DS[c]	2.4%	1.7%
4-channel CS[d] $\tau_{max} = 62.5ms$	3.0%	2.0%
4-channel CS $\tau_{max} = 125ms$	3.6%	2.3%
Significance Testing		
SDM Channel 6 $\leftrightarrow$ **4-channel CS** $\tau_{max} = 62.5ms$	2.07e-14	2.37e-7
SDM Channel 6 $\leftrightarrow$ **4-channel CS** $\tau_{max} = 125ms$	9.14e-7	3.00e-3
4-channel PHAT-DS $\leftrightarrow$ **4-channel CS** $\tau_{max} = 62.5ms$	3.86e-5	0.02
4-channel PHAT-DS $\leftrightarrow$ **4-channel CS** $\tau_{max} = 125ms$	1.86e-11	9.82e-7

Non Noise-Reduced

	No MLLR	**MLLR**
SDM Channel F	6.1%	3.8%
4-channel PHAT-DS	2.6%	1.9%
4-channel CS $\tau_{max} = 62.5ms$	3.4%	2.3%
4-channel CS $\tau_{max} = 125ms$	3.7%	2.6%
Significance Testing		
SDM Channel F $\leftrightarrow$ **4-channel CS** $\tau_{max} = 62.5ms$	1.86e-17	2.51e-12
SDM Channel F $\leftrightarrow$ **4-channel CS** $\tau_{max} = 125ms$	1.10e-15	2.14e-8
4-channel PHAT-DS $\leftrightarrow$ **4-channel CS** $\tau_{max} = 62.5ms$	8.88e-8	5.44e-4
4-channel PHAT-DS $\leftrightarrow$ **4-channel CS** $\tau_{max} = 125ms$	8.29e-11	1.35e-8

[a] Maximum-Likehood Linear Regression Speaker Adaptation.
[b] Single distant microphone with lower error-rate.
[c] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.
[d] Correlation shaping using PHAT TDE for equalizer initialization.

Table 6.1: WER of multiple distant microphones processed with 4-channel correlation shaping on the Mismatched Conditions Digit Test-bed (MMCDT).

For the matched training experiments, which are shown in Table 6.2, correlation shaping behaves in the same line as for MMCDT. PHAT-DS is consistently better than CS. Interestingly, WER of non-noise reduced CS-processed waveforms are lower than for noise-reduced ones. This might be explained by the fact that speech distortion and artifacts related to Wiener filtering are not present in the non-processed waves. Thus, in these experiments, Wiener filtering and correlation shaping seem to interact negatively.

Noise-Reduced		
	No MLLR	**MLLR**[a]
SDM Channel 6[b]	3.1%	2.2%
4-channel PHAT-DS[c]	2.33%	1.63%
4-channel CS[d] $\tau_{max} = 62.5ms$	2.86%	2.0%
4-channel CS $\tau_{max} = 125ms$	2.83%	1.8%
Significance Testing		
SDM Channel 6 $\leftrightarrow$ **4-channel CS** $\tau_{max} = 62.5ms$	0.17	0.26
SDM Channel 6 $\leftrightarrow$ **4-channel CS** $\tau_{max} = 125ms$	0.27	0.08
4-channel PHAT-DS $\leftrightarrow$ **4-channel CS** $\tau_{max} = 62.5ms$	5.16e-3	6.92e-3
4-channel PHAT-DS $\leftrightarrow$ **4-channel CS** $\tau_{max} = 125ms$	5.31e-3	0.11
Non Noise-Reduced		
	No MLLR	**MLLR**
SDM Channel 6	3.3%	2.2%
4-channel PHAT-DS	2.4%	1.63%
4-channel CS $\tau_{max} = 62.5ms$	2.6%	1.73%
4-channel CS $\tau_{max} = 125ms$	2.53%	1.73%
Significance Testing		
SDM Channel 6 $\leftrightarrow$ **4-channel CS** $\tau_{max} = 62.5ms$	3.20e-4	9.91e-3
SDM Channel 6 $\leftrightarrow$ **4-channel CS** $\tau_{max} = 125ms$	1.27e-4	4.32e-3
4-channel PHAT-DS $\leftrightarrow$ **4-channel CS** $\tau_{max} = 62.5ms$	0.27	0.30
4-channel PHAT-DS $\leftrightarrow$ **4-channel CS** $\tau_{max} = 125ms$	0.25	0.37

[a] Maximum-Likehood Linear Regression Speaker Adaptation.
[b] Single distant microphone with lower error-rate.
[c] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.
[d] Correlation shaping using PHAT TDE for equalizer initialization.

Table 6.2: WER of multiple distant microphones processed with 4-channel correlation shaping on the matched conditions digit test-bed (MCDT).

Regarding the conversational test-bed, only the 4 channels with more energy were used for correlation shaping. In the evaluation results, shown in Table 6.3, CS $\tau_{max} = 62.5ms$ improves WER for a single distant microphone (SDM), although PHAT-DS outperforms CS, both for noisy and noise-reduced waveforms. Significance is reached in both cases. On the other hand, recognition accuracy for CS $\tau_{max} = 125ms$ drops down, hardly achieving lower WER than that of SDM, in the noise-reduced data set. In the noisy test-bed, CS results in even lower accuracy than for SDM, although significance is not reached. This low performance is thought to be related to the poor whitening performance of CS beyond the time span of the equalizing filters, as shown in Figure 6.10, as well as the loss of control of the algorithm on the output autocorrelation function for lags larger than τ_{max}.

Noise-Reduced

	Overall	ICSI	NIST	LDC	CMU
SDM[a]	48.4%	34.7%	48.2%	56.2%	62.1%
PHAT-DS[b]	43.2%	25.9%	45.5%	50.2%	62.1%
4-channel CS[c] $\tau_{max} = 62.5ms$	45.8%	32.4%	45.4%	51.9%	62.1%
4-channel CS $\tau_{max} = 125ms$	47.4%	34.3%	48.0%	53.8%	62.1%
Significance Testing					
SDM $\leftrightarrow$ **CS** $\tau_{max} = 62.5ms$	7.93e-6				
SDM $\leftrightarrow$ **CS** $\tau_{max} = 125ms$	0.02				
PHAT-DS $\leftrightarrow$ **CS** $\tau_{max} = 62.5ms$	8.68e-4				
PHAT-DS $\leftrightarrow$ **CS** $\tau_{max} = 125ms$	1.42e-7				

Non Noise-Reduced

	Overall	ICSI	NIST	LDC	CMU
SDM	50.1%	37.4%	49.4%	56.2%	64.8%
PHAT-DS	45.7%	28.8%	49.2%	52.0%	64.8%
4-channel CS $\tau_{max} = 62.5ms$	48.7%	35.4%	49.2%	54.2%	64.8%
4-channel CS $\tau_{max} = 125ms$	55.2%	45.9%	66.0%	53.7%	64.8%
Significance Testing					
SDM $\leftrightarrow$ **CS** $\tau_{max} = 62.5ms$	5.14e-3				
SDM $\leftrightarrow$ **CS** $\tau_{max} = 125ms$	0.07				
PHAT-DS $\leftrightarrow$ **CS** $\tau_{max} = 62.5ms$	3.15e-4				
PHAT-DS $\leftrightarrow$ **CS** $\tau_{max} = 125ms$	4.48e-9				

[a] Single distant microphone.

[b] Delay-and-sum using PHAT-weighted cross-correlation for TDOA estimation.

[c] Correlation shaping using PHAT delay estimates for equalizer initialization.

Table 6.3: WER of multiple distant microphones processed with 4-channel correlation shaping on the Mismatched Conditions Conversational Test-bed (MMCCDT).

Chapter 7

Conclusions

Accuracy of current speech recognition systems drops down in the presence of noise and/or reverberation. These are typical conditions found in speech signals acquired from distant microphones. On the other hand, the meeting recognition task is taking advantage of multiple streams of speech data, aiming at achieving a more natural feeling for speech recognition users.

In this work several multiple-microphone speech enhancement techniques were explored, primarily focusing on blind signal processing algorithms for combating reverberation.

Regarding equalization techniques it was shown that:

- Speaker-to-receiver impulse response inversion is inherently problematic due to instability issues. Direct impulse response inversion fails even in the most scented scenarios, e.g., with no noise and unrealistically simple and short impulse responses.

- Linear least squares (LLS) equalization lacks instability issues, although an inverse impulse response approximation is made. The single-channel LLS equalizer was shown to spread speaker-to-receiver impulse response energy along the equalizer time span, which actually resulted in more reverberant speech signals. A simple graphical study and informal listening showed that multiple-channel LLS equalization performs almost perfect dereverberation under certain hypotheses. Nonetheless, impulse response order mismatch was shown be decisive, highlighting the importance of this hypothesis in the MINT theorem and

rendering it not practical in real situations.

Several time-delay and linear prediction based dereverberation techniques were explored and evaluated on speech recognition test-beds, covering several meeting rooms, tasks, and training-test and speaker adaptation conditions. It was shown that:

- Delay-and-sum processing improves speech recognition accuracy over a single distant microphone (SDM) in all proposed test-beds and conditions. Furthermore, most of the benefit is achieved in mismatched training-test conditions, as shown in Table 7.1. For the conversational speech test-bed a 10.7% and 8.8% WER relative improvement was achieved using PHAT-DS, on noise-reduced and non noise-reduced data sets respectively.

  Combining DS with PHAT-weighted cross-correlation for time-delay estimation achieves lower WER than its non-weighted cross-correlation counterpart, NW-DS, although significance is not guaranteed. Thus, further work on delay-and-sum should explore more reliable time-delay estimation techniques. To cope with instability of real speaker-to-receiver impulse responses, time-delay estimation can be performed in a time-varying manner.

- Delay-and-feature-domain-sum (DFDS) improves recognition accuracy over a single distant microphone in all tested conditions. In noisy and matched training-test conditions (see Table 7.1), DFDS outperforms all other systems across noise, training-test and speaker adaptation conditions, achieving the lowest WER for digits recognition in the proposed test-beds. Nonetheless, only weak significance was reached for DS vs DFDS WER comparisons. Acoustic model retraining was shown to be specially beneficial for this technique.

  Regarding time-delay estimation techniques, PHAT-DFDS outperformed NW-DFDS in most cases, but not significantly.

  Further work on DFDS would require a deeper exploration for conversational speech recognition to confirm the trends shown on the digits test-beds. In another direction, its behavior in noisier environments, such as car interiors, should be explored as well.

Noise-Reduced

System	MMCDT[a]	MCDT[b]	MMCCDT[c]
NW-DS	53.8%	24.8%	7.4%
PHAT-DS	53.8%	25.9%	10.7%
NW-DFDS	32.6%	22.7%	—
PHAT-DFDS	34.6%	27.3%	—
PHAT-PBF $\gamma = 0$	57.6%	25.9%	10.1%
PHAT-PBF $\gamma = 0.5$	53.8%	24.5%	3.9%
CS $\tau_{max} = 62.5ms$	42.3%	9.0%**[d]	5.4%
CS $\tau_{max} = 125ms$	30.7%	18.2%*	2.1%*[e]

Non Noise-Reduced

System	MMCDT	MCDT	MMCCDT
NW-DS	45.9%	23.3%	1.8%
PHAT-DS	57.4%	27.2%	8.8%
NW-DFDS	36.8%	35.0%	—
PHAT-DFDS	36.8%	29.1%	—
PHAT-PBF $\gamma = 0$	59.0%	30.3%	9.4%
PHAT-PBF $\gamma = 0.5$	55.7%	31.8%	-8.9%
CS $\tau_{max} = 62.5ms$	44.2%	21.4%	2.8%
CS $\tau_{max} = 125ms$	39.3%	23.3%	-9.23%**

[a] Mismatched training-test conditions digits test-bed.
[b] Matched training-test conditions digits test-bed.
[c] Mismatched training-test conditions conversational test-bed.
[d] In **, significance was not reached.
[e] In *, only weak significance was reached.

Table 7.1: Relative WER improvement of the explored multiple-channel techniques over a single distant microphone on the proposed test-beds.

- Multi-channel time-frequency masking improves a single distant microphone WER in all proposed conditions. Compared to DS, it results in poor recognition performance, unless $\gamma = 0$ (see Table 7.1), case in which PHAT-PBF and PHAT-DS would coincide. For conversational speech, it only achieved a 3.9% and 8.9% WER relative improvement over a single distant microphone, for noise-reduced and non-noise-reduced data sets, respectively. Recognition accuracy degraded rapidly as γ was set larger, showing to be quite an aggressive algorithm.

Further work should include a deep study on sensitivity to time-delay estimation errors, since the algorithm seems to be very sensitive to them. Specially, how the TDOA stationarity and speaker-to-receiver impulse response over whole utterances assumptions affect its performance should be researched.

- Multi-channel correlation shaping outperforms a single distant microphone when long-term shaping is not performed, i.e., setting $\tau_{max} = 62.5ms$, which corresponds to the equalizer length. Nonetheless, this improvement is not as big as for DS, DFDS or PHAT-PBF processing. For mismatched training-test conditions, CS $\tau_{max} = 125ms$ improvements are modest even not being able to outperform SDM in some case. Nonetheless, in matched training-test conditions, CS $\tau_{max} = 125ms$ gets a lower WER than CS $\tau_{max} = 62.5ms$.

To complete this exploration of correlation shaping processing, a deep study of its convergence properties should be carried out. Serious convergence issues arose in its on-line implementation. However, assuming the speaker-to-receiver impulse response to be stationary and performing adaptive filtering as a batch training technique did not show these problems, but still a very slow convergence rate. On the other hand, the effect of speaker-to-receiver impulse response shortening claimed in [15] was not noticeable in our experiments, suggesting that some issues could be present in our implementation.

Appendix A

Mathematical background

A.1 Linear Least Squares Equalization

Direct equalization involves a linear system of equations,

$$\mathbf{Hg} = \mathbf{d} \tag{A.1.1}$$

to be solved. Since matrix H is usually estimated from real data, the system might not be compatible. Furthermore, if **H** is not square, it could be overdetermined or underdetermined (see Appendix lAppendixMNMInversion for undertermined system solution).

Thus, (A.1.1) can be rearranged as

$$\mathbf{e} = (\mathbf{Hg} - \mathbf{d}) \tag{A.1.2}$$

yielding an error vector for its solution. The norm of this vector, or its square, can be used as a cost function to be minimized for Least Squares solving.

In linear least squares (LLS) equalization, therefore, is formulated in terms of the cost function

$$\xi = (\mathbf{Hg} - \mathbf{d})^T(\mathbf{Hg} - \mathbf{d}) \tag{A.1.3}$$

, where $\mathbf{H} \in \mathbb{R}^{m \times n}$, $\mathbf{g} \in \mathbb{R}^n$ and $\mathbf{d} \in \mathbb{R}^m$, which needs to be minimized by choosing the right equalizer $\mathbf{g}$.

Finding the optimal $\mathbf{g}$,

$$\frac{\partial \xi}{\partial \mathbf{g}^T} = \mathbf{H}^T(\mathbf{Hg} - \mathbf{d}) \tag{A.1.4}$$

$$= \mathbf{H}^T\mathbf{Hg} - \mathbf{H}^T\mathbf{d} = 0 \tag{A.1.5}$$

and, then,

$$\mathbf{g_{opt}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{d} \tag{A.1.6}$$

A.2 Minimum-norm Matrix Inversion

Least Squares solving of underdetermined linear systems of equations such as, in matrix form,

$$\mathbf{Ax} = \mathbf{b} \tag{A.2.1}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $m < n$, that is,

$$\mathbf{A^T Ax} = \mathbf{A^T b} \tag{A.2.2}$$

where $\mathbf{A^T A} \in \mathbb{R}^{n \times n}$, requires the inversion of $\mathbf{A^T A}$, which is not a full-rank matrix. Thereby, an infinite set of solutions satisfies (A.2.2) and, consequently, several inverse matrices exist. Finding the minimum-norm inverse matrix [17], which is unique, is one way to overcome this problem.

In singular value decomposition (SVD), a positive-definite matrix, say $\mathbf{A} \in \mathbb{R}^{m \times n}$, can be factored as

$$\mathbf{A} = \mathbf{U \Lambda V}^T \tag{A.2.3}$$

where $\mathbf{U} = [\mathbf{u_1}\ \mathbf{u_2}\ \cdots\ \mathbf{u_m}] \in \mathbb{R}^{m \times m}$, $\mathbf{V} = [\mathbf{v_1}\ \mathbf{v_2}\ \cdots\ \mathbf{v_n}] \in \mathbb{R}^{n \times n}$ and $\mathbf{\Lambda} = diag(\lambda_1, ..., \lambda_p)$, with $p = \min(m, n)$ and $\lambda_i > 0, \forall i : 1 \leq i \leq p$ or, in series form

$$\mathbf{A} = \sum_{i=1}^{p} \lambda_i \mathbf{u_i v_i}^T \tag{A.2.4}$$

Using SVD, its full-rank inverse matrix can be found by just inverting its corresponding singular values in the series expansion. Thus,

$$\mathbf{A^{-1}} = \sum_{i=1}^{p} \lambda_i^{-1} \mathbf{u_i v_i}^T \tag{A.2.5}$$

If $\mathbf{A}$ is rank-deficient, though,

$$\lambda_i > 0, \forall i : 1 \leq i \leq k, k < p \tag{A.2.6}$$

and, thus, some singular values are 0[1].

To get the minimum-norm inverse of $\mathbf{A}$ only the non-zero singular values are used in the series expansion. They are ordered decreasingly, and only the first kth are used for the inverse matrix reconstruction, and, thus

$$\mathbf{A}^{-\mathbf{1}}_{\mathbf{mn}} = \sum_{i=1}^{k} \lambda_i^{-1} \mathbf{u_i} \mathbf{v_i}^T \tag{A.2.7}$$

For a more detailed discussion about inversion of rank-deficient matrices, please refer to [17].

[1] In practical situations very small singular values are found instead of zeros. To identify them, a threshold, either fixed or a function of the largest singular value, may be used.

A.3 Correlation Shaping Gradient Derivation

In mathematical terms, Correlation Shaping aims to find a vector $\mathbf{g} = [g(0)\ g(1)\ \ldots\ g(M-1)]^T$ such that

$$e(\tau) = (r_{yy}(\tau) - r_{dd}(\tau))^2 \tag{A.3.1}$$

or rather

$$e(\tau) = w(\tau)(r_{yy}(\tau) - r_{dd}(\tau))^2 \tag{A.3.2}$$

for the more general weighted case, are minimized. The additional constraints

$$r_{yy}(\tau) = \mathrm{E}[y(n)y(n-\tau)] \tag{A.3.3}$$

,with E denoting the expectation operator, and

$$y(n) = \sum_{c=1}^{C} \sum_{m=0}^{M-1} g_c(m) x_c(n-m) \tag{A.3.4}$$

bond $r_{yy}(\tau)$ to $g(m)$ and, thus, to $\mathbf{g}$.

To ease the optimization procedure, a Least Mean Squares (LMS) gradient descent approach is adopted [15]. Differentiation of A.3.2 yields

$$\frac{\partial e(\tau)}{\partial g_c(l)} = 2w(\tau)\left(r_{yy}(\tau) - r_{dd}(\tau)\right)\frac{\partial r_{yy}(\tau)}{\partial g_c(l)} \tag{A.3.5}$$

On the other hand, from (A.3.3)

$$\frac{\partial r_{yy}(\tau)}{\partial g_c(l)} = \frac{\partial}{\partial g_c(l)}\mathrm{E}[y(n)y(n-\tau)] \tag{A.3.6}$$

which, since $\mathrm{E}[\cdot]$ and $\frac{\partial}{\partial g_c(l)}$ are both linear operators, can be rearranged as

$$\mathrm{E}[\frac{\partial}{\partial g_c(l)}\left(y(n)y(n-\tau)\right)] \tag{A.3.7}$$

Proceeding with differentiation,

$$\begin{aligned}
\frac{\partial r_{yy}(\tau)}{\partial g_c(l)} &= \mathrm{E}[y(n-\tau)\frac{\partial y(n)}{\partial g_c(l)} + y(n)\frac{\partial y(n-\tau)}{\partial g_c(l)}] && \text{(A.3.8)}\\
&= \mathrm{E}[(y(n-\tau)x_c(n-l) + y(n)x_c(n-l-\tau))] && \text{(A.3.9)}\\
&= \mathrm{E}[y(n-\tau)x_c(n-l)] + \mathrm{E}[y(n)x_c(n-l-\tau)] && \text{(A.3.10)}\\
&= \mathrm{E}[y(n)x_c(n-(l-\tau))] + \mathrm{E}[y(n)x_c(n-(l+\tau))] && \text{(A.3.11)}\\
&= r_{yx}(l-\tau) + r_{yx}(l+\tau) && \text{(A.3.12)}
\end{aligned}$$

Therefore, after dropping constant factors

$$\frac{\partial e(\tau)}{\partial g_c(l)} = w(\tau)\left(r_{yy}(\tau) - r_{dd}(\tau)\right)\left(r_{yx}(l-\tau) + r_{yx}(l+\tau)\right) \qquad \text{(A.3.13)}$$

and, since the error must be minimized overall, the gradient is averaged for all lags as

$$\nabla_c(l) = \sum_{\tau} w(\tau)\left(r_{yy}(\tau) - r_{dd}(\tau)\right)\left(r_{yx}(l-\tau) + r_{yx}(l+\tau)\right) \qquad \text{(A.3.14)}$$

Bibliography

[1] P. Aarabi and G. Shi, *Phase-based dual-microphone robust speech enhancement*, IEEE Transactions on Systems, Man and Cybernetics, vol. 34, August 2004.

[2] B. S. Atal, *Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification*, Journal of the Acoustic Society of America, vol. 55, June 1974.

[3] M. Athineos and D. Ellis, *Frequency-domain linear prediction for temporal features*, Automatic Speech Recognition and Understanding Workshop, 2003.

[4] T. Backstrom, *Linear predictive modelling of speech - constraints and line spectrum pair decomposition*, Ph.D. thesis, Electrical and Communications Engineering. Helsinki University of Technology, 2004.

[5] T. S. Bakir and R. M. Mersereau, *Blind adaptive dereverberation of speech signals using a microphone array*, Proc. ASAP, 2003.

[6] S. F. Boll, *Suppression of acoustic noise in speech using spectral substraction*, IEEE Trans. Acoustics, Speech and Signal Processing, vol. 27, 1979, pp. 113–120.

[7] B. Chen, S. Chang, and S. Sivadas, *Learning discriminative temporal patterns in speech: Development of novel traps-like classifiers*, Proc. Eurospeech, 2003.

[8] The ICSI Meeting Corpus, *http://www.icsi.berkeley.edu/speech/mr/*.

[9] L. Deng and D. O'Shaughnessy, *Speech processing. a dynamic and optimization-oriented approach*, Marcel Dekker, Inc., 2003.

[10] Y. Ephraim and D. Malah, *Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator*, IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 32, December 1984, pp. 1109–1121.

[11] ———, *Speech enhancement using a minimum mean-square error log-spectral amplitude estimator*, IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 33, April 1985, pp. 443–445.

[12] M. J. F. Gales and P. C. Woodland, *Mean and variance adaptation within the mllr framework*, Computer Speech and Language, vol. 10, 1996, pp. 249–264.

[13] N. D. Gaubitch, P. A. Naylor, and D. B. Ward, *On the use of linear prediction for dereverberation of speech*, International Workshop on Acoustic, Echo and Noise Control (Kyoto, Japan), September 2003.

[14] D. Gesbert, P. Duhamel, and S. Mayrargue, *On-line multichannel equalization based on mutually referenced filters*, IEEE Transactions on Signal Processing, vol. 45, September 1997, pp. 2307–2317.

[15] B. W. Gillespie, *Strategies for improving audible quality and speech recognition accuracy of reverberant speech*, Ph.D. thesis, University of Washington, 2002.

[16] B. W. Gillespie and H. Malvar, *Speech dereverberation via maximum-kurtosis subband adaptive filtering*, Proc. ICASSP, 2001.

[17] G. H. Golub and C.F. Van Loan, *Matrix computations*, The Johns Hopkins University Press, 1996.

[18] M. Gurelli and C. L. Nikias, *An eigenvector-based algorithm for multichannel blind deconvolution of input colored signals*, IEEE Trans. Signal Processing, vol. 43, January 1995, pp. 134–149.

[19] S. Haykin, *Adaptive filtering theory*, 4th ed., Prentice Hall, 2001.

[20] H. Hermansky, *Perceptual linear predictive (plp) analysis of speech*, Journal of the Acoustic Society of America, vol. 87, April 1990, pp. 1738–1752.

[21] H. Hermansky and N. Morgan, *Rasta processing of speech*, IEEE Transactions on Speech and Audio Processing, vol. 2, October 1994, pp. 578–589.

[22] M. Hofbauer and H. Loeliger, *Limitations of fir multimicrophone speech dereverberation in the low-delay case*, Proc. IWAENC, 2003.

[23] X. Huang, A. Acero, and H. Hon, *Spoken language processing. a guide to theory, algorithm and system development*, Prentice-Hall, 2001.

[24] A. Janin, J. Ang, S. Bhagat, R.Dhillon, J.Edwards, J. Macias-Guarasa, N. Morgan, B. Peskin, E. Shriberg, A. Stolcke, C. Wooters, and B. Wrede, *The icsi meeting corpus: Resources and research*, NIST ICASSP, Meeting Recognition Workshop (Montreal, Canada), May 2004.

[25] Steven M. Kay, *Fundamentals of statistical signal processing Estimation theory*, Prentice Hall, 1993.

[26] C. H. Knapp and C. Carter, *The generalized correlation method for estimation of time delay*, IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 24, August 1976.

[27] C. Y. Lai and P. Aarabi, *Multiple-microphone time-varying filters for robus speech recognition*, Proc. ICASSP, 2004.

[28] C. Y. K. Lai, *Analysis and extension of time-frequency masking*, Master's thesis, Department of Electrical and Computer Engineering, University of Toronto, 2003.

[29] C. J. Leggetter, *Improved acoustic modeling for hmms using linear transformations*, Ph.D. thesis, Department of Engineering, University of Cambridge, February 1995.

[30] J. S. Lim and A.V. Oppenheim, *Advanced topics in signal processing*, Prentice Hall.

[31] J. Makhoul, *Linear prediction. a tutorial review*, IEEE Proceedings, vol. 63, 1975, pp. 561–580.

[32] N. Mirghafori, A. Stolcke, C. Wooter, T. Pirinen, I. Bulyko, D. Gelbart, M. Graciarena, S. Otterson, B. Peskin, and M. Ostendorf, *From switchboard to meetings: Development of the 2004 icsi-sri-uw meeting recognition system*, Proceedings of International Conference on Spoken Language Processing (Jeju, Korea), October 2004.

[33] Climent Nadeu, Javier Hernando, and Monica Gorricho, *On the decorrelation of filter-bank energies in speech recognition*, Proc. Eurospeech, 1995.

[34] T. Nakatani, M. Miyoshi, and K. Kinoshita, *Implementation and effects of single channel dereverberation based on the harmonic structure of speech*, Proc. IWAENC, September 2003.

[35] J. P. Openshaw and J. S. Mason, *On the limitations of cepstral features in noise*, Proc. ICASSP, 1994.

[36] The NIST Meeting Room Project, *http://www.nist.gov/speech/test_beds/mr_proj/*.

[37] Pere Pujol, Susagna Pol, Climent Nadeu, Astrid Hagen, and Hervé Bourlard, *Comparison and combination of features in a hybrid hmm/mlp and a hmm/gmm speech recognition system*, IEEE Transactions in Speech and Audio Signal Processing, vol. 13, January 2005.

[38] L. R. Rabiner, *A tutorial on hidden markov models and selected applications in speech recognition*, Proceedings of the IEEE, vol. 77, February 1989, pp. 257–285.

[39] W. Reichl and G. Ruske, *Discriminative training for continuous speech recognition*, Proceedings of Eurospeech, vol. 1, 1995, pp. 537–540.

[40] W. C. Sabine, *Collected papers on acoustics*, Peninsula Publishing, Los Altos, CA, 1993.

[41] M. R. Schroeder, *New method of measuring reverberation time*, Journal of the Acoustical Society of America, vol. 37, 1965, pp. 409–412.

[42] R. Schwarz, *Comparative experiments on large vocabulary speech recognition*, Proc. ARPA Workshop Human Language Technology, 1993.

[43] V. Stahl, A. Fischer, and R. Bippus, *Quantile-based noise estimation for spectral substraction and wiener filtering*, Proc. ICASSP, vol. 3, 2000, pp. 1875–1878.

[44] M. Tonelli, N. Mitianoudis, and M. Davies, *A maximum-likelihood approach to blind audio de-reverberation*, Proc. DAFX, 2004.

[45] V. Valtchev, J. J. Odell, P. C. Woodland, and S. J. Young, *Mmie training of large vocabulary recognition systems*, Speech Communication, September 1997, pp. 303–314.

[46] O. Viikki and L. Laurila, *Cepstral domain segmental feature vector normalization for noise robust speech recognition*, Speech Communication 25, 1998, pp. 133–147.

[47] P. Zhao and J. P. Reilly, *Exponentially decaying time-recursive blind deconvolution algorithm for speech dereverberation*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (ASSP), January 1995, pp. 127–130.