Robust Phase-based Speech Signal Processing

From Source-Filter Separation to Model-Based Robust ASR

Erfan Loweimi

Supervisors: Professor Jon Barker and Professor Thomas Hain

Faculty of Engineering
The University of Sheffield

This dissertation is submitted for the degree of
Doctor of Philosophy

Speech and Hearing Research Group February 2018

I would like to dedicate this thesis to my loving parents ...

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. The length of this thesis including footnotes, appendices and references is approximately 100000 words. This thesis contains 90 figures and 21 tables.

Erfan Loweimi
February 2018

Acknowledgements

First and foremost, I would like express my utmost gratitude to my supervisors, Professor Jon Barker and Professor Thomas Hain, for their mentorship, guidance and support throughout my time studying in Sheffield. Without their help this research would never have come to fruition. I learned a lot from the regular discussion every week and their wisdom, insight, diligence and passion in research. It has been an honour and privilege to work with them.

Secondly, I would like to thank the University of Sheffield for granting me the faculty scholarship, which has made it possible for me to complete this PhD. Also I would like to thank the Department of Computer Science in the Faculty of Engineering for the financial support, allowing me to attend many international conferences and workshops.

I am grateful to many people in the Speech and Hearing Research Group (SPandH) and Machine Intelligence for Natural Intelligence (MINI) group for their help during my time here. Special thanks go to Oscar Saz Torralba and Mortaza Doulaty for their early help, unending advice and assistance whenever I ran into problems. There are many people who have made our group an interesting place to work. I would like to thank Madina Hasan, José González, Raymond Ng, Ning Ma, Sarah Al-Sharif, Yulan Liu, Donya Yazdani, Mauro Nicolao, Charles Fox, Amy Beeston, Rosanna Milner, Iñigo Casanueva, Bhusan Chettri, Asif Jalal, Maryam Al Dabel, Chenhao Wu, Rabab Algadhy, Atiqah Masrani, Bahman Mirheidari, Sadeen Alharbi, Mashael AlSaleh, Saeid Mokaram, Salil Deena, Fariba Yousefi, Lubna Al-Hinti and Hanan Halawani for their help and friendship. Also I would like to thank the SPandH academics: Professor Phil Green, Professor Roger Moore, Professor Guy Brown, Dr Yoshi Gotoh and Dr Heidi Christensen.

In addition, I would like to thank my examiners Professor Yannis Stylianou and Professor Guy Brown for their insightful comments and valuable suggestions to improve the thesis.

The biggest thanks go to my parents. For many years, they have offered everything possible to support me. Without their encouragements, I would not be here. This thesis is dedicated to them. Last but not least, I would like to specially thank my dear sister and brother-in-law for their help and taking care of my parents during my long absence.

Abstract

The Fourier analysis plays a key role in speech signal processing. As a complex quantity, it can be expressed in the polar form using the magnitude and phase spectra. The magnitude spectrum is widely used in almost every corner of speech processing. However, the phase spectrum is not an obviously appealing start point for processing the speech signal. In contrast to the magnitude spectrum whose fine and coarse structures have a clear relation to speech perception, the phase spectrum is difficult to interpret and manipulate. In fact, there is not a meaningful trend or extrema which may facilitate the modelling process. Nonetheless, the speech phase spectrum has recently gained renewed attention. An expanding body of work is showing that it can be usefully employed in a multitude of speech processing applications. Now that the potential for the phase-based speech processing has been established, there is a need for a fundamental model to help understand the way in which phase encodes speech information.

In this thesis a novel phase-domain source-filter model is proposed that allows for deconvolution of the speech vocal tract (filter) and excitation (source) components through phase processing. This model utilises the Hilbert transform, shows how the excitation and vocal tract elements mix in the phase domain and provides a framework for efficiently segregating the source and filter components through phase manipulation. To investigate the efficacy of the suggested approach, a set of features is extracted from the phase filter part for automatic speech recognition (ASR) and the source part of the phase is utilised for fundamental frequency estimation. Accuracy and robustness in both cases are illustrated and discussed. In addition, the proposed approach is improved by replacing the log with the generalised logarithmic function in the Hilbert transform and also by computing the group delay via regression filter.

Furthermore, statistical distribution of the phase spectrum and its representations along the feature extraction pipeline are studied. It is illustrated that the phase spectrum has a bell-shaped distribution. Some statistical normalisation methods such as mean-variance normalisation, Laplacianisation, Gaussianisation and Histogram equalisation are successfully applied to the phase-based features and lead to a significant robustness improvement.

The robustness gain achieved through using statistical normalisation and generalised logarithmic function encouraged the use of more advanced model-based statistical techniques such as vector Taylor Series (VTS). VTS in its original formulation assumes usage of the log function for compression. In order to simultaneously take advantage of the VTS and generalised logarithmic function, a new formulation is first developed to merge both into a unified framework called generalised VTS (gVTS). Also in order to leverage the gVTS framework, a novel channel noise estimation method is developed. The extensions of the gVTS framework and the proposed channel estimation to the group delay domain are then explored. The problems it presents are analysed and discussed, some solutions are proposed and finally the corresponding formulae are derived. Moreover, the effect of additive noise and channel distortion in the phase and group delay domains are scrutinised and the results are utilised in deriving the gVTS equations. Experimental results in the Aurora-4 ASR task in an HMM/GMM set up along with a DNN-based bottleneck system in the clean and multi-style training modes confirmed the efficacy of the proposed approach in dealing with both additive and channel noise.

Keywords: Phase spectrum, group delay, source-filter separation, Hilbert transform, robust ASR, statistical normalisation, generalised vector Taylor series, channel estimation

Table of contents

List of figures

List of tables

Nomenclature

The beginning of wisdom is the definition of terms. – Socrates

It was attempted to maintain symbol compatibility and avoid ambiguity over the mathematical notation throughout the thesis. However, in some cases, a symbol is inevitably reused to represent a different parameter, in order to be consistent with the reference or the well-established symbols and notations. In such cases, it is made clear by the context and extra explanations. In general, the following symbols, variable and acronyms are used over the thesis with the presented explanation.

Variables, Symbols and Operations

$*$	convolution in $x*h$ and conjugate in x^* (assuming x is complex)
$\approx$	approximately equal to
$\breve{X}$	X^{α}
$\gg$	much greater than
$\hat{x}$	estimate of the true value of variable x
$\ll$	much less than
$\mathbb{E}\{f(x)\}$	the expected value of f(x), x is a random variable
$\prod_{n=1}^{N} x_n$	production from $n=1$ to N, that is, $x_1 x_2 ... x_N$
$\propto$	proportional to
$\sum_{n=1}^{N} x_n$	summation from $n=1$ to N, that is, $x_1 + x_2 + ... + x_N$
$\tilde{X}$	$log(X)$
$argmax\, f(x)$	the value of x that maximises the value of $f(x)$

$argmin\ f(x)$	the value of x that minimises the value of $f(x)$
$exp(x)$	exponential of x
$log(x)$	logarithm of x
$sign(x)$	sign(um) function
$\mathbb{R}^d$	d-dimensional Euclidean space
μ	mean vector in multivariate Gaussian distribution
μ_m^x	mean vector of the m^{th} component of the GMM of x
$\odot$	Hadamard (element-wise) matrix multiplication
Σ	covariance matrix in multivariate Gaussian distribution
σ^2	variance
Σ_m^x	covariance matrix of the m^{th} component of the GMM of x
$\mathbf{X}$	an arbitrary matrix
$\mathbf{x}$	vector of an arbitrary dimension
$\mathbf{X}^T$	transpose of matrix $\mathbf{X}$
D	number of dimensions of a vector or a square matrix
d	dimension index
$diag\{\mathbf{x}\}$	a diagonal matrix whose main diagonal consists of the vector $\mathbf{x}$
J_x	Jacobian matrix (partial derivative with respect to x)
M	number of components (Gaussians) in a mixture (GMM)
p_m^x	weight of the m^{th} component of the GMM of x
x	scalar quantity
X^{-1}	inverse of the square matrix $\mathbf{X}$
x_{ij}	scalar value that is the element in row i and column j of matrix $\mathbf{X}$
(n)	n is a continuous independent variable

$[n]$	n is a discrete independent variable
$\check{x}[q]$	generalised cepstrum of signal $x[n]$ (q denotes quefrency)
ω	radial frequency (rad/s)
$\tau_X(\omega)$	group delay of x, $-\frac{d}{d\omega}arg\{X(\omega)\}$
$\tilde{x}[q]$	cepstrum of signal $x[n]$ (q denotes quefrency)
$ARG\{X(\omega)\}$	wrapped or principal phase spectrum of x
$arg\{X(\omega)\}$	unwrapped (continuous) phase spectrum of x
C	Discrete Cosine Transform (DCT) matrix
C^{-1}	Inverse Discrete Cosine Transform (DCT) matrix
$h[n]$	impulse response of the channel h
k	discrete frequency index
N	number of samples in a frame
n	discrete time index
N_{FFT}	size of FFT
q	discrete quefrency index
$r_x[l]$	autocorrelation of x at l^{th} lag
T	number of frames of an utterance or a sequence of observation
t	time frame index
W	power spectrum (periodogram) of additive noise w
$w[n]$	additive noise in the time domain or the window function
$x[n]$	clean speech in the time domain
X_{Im}	Imaginary part of the Fourier transform
X_{Re}	Real part of the Fourier transform
$y[n]$	noisy observation in the time domain

Acronyms

AM	Acoustic Model or Amplitude Modulation
AMS	Analysis-Modification-Synthesis
ARMA	AutoRegressive Moving Average
ASR	Automatic Speech Recognition
BN	Bottle Neck (feature)
CDF	Cumulative Distribution Function
$CGD(F)$	Chirp Group Delay (Function)
CMN	Cepstral Mean Normalisation
CMVN	Cepstral Mean Variance Normalisation
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DNN	Deep Neural Network
FBE	Filter Bank Energy
FFT	Fast Fourier Transform
FM	Frequency Modulation
FT	Fourier Transform
$GD(F)$	Group Delay (Function)
GenLog	Generalised Logarithmic Function
GMM	Gaussian Mixture Model
gVTS	generalised Vector Taylor Series
HEQ	Histogram EQualisation
HMM	Hidden Markov Model
HPS	Harmonic Product Spectrum

HTK	HMM Toolkit
IDCT	Inverse Discrete Cosine Transform
IFD	Instantaneous Frequency Deviation
IGDD	Inverse Group Delay Deviation
LPC	Linear Predictive Coding
MAP	Maximum A Posteriori
MFCC	Mel-Frequency Cepstral Coefficient
MinPh	Minimum-phase component of a mixed-phase signal
MMSE	Minimum Mean Squared Error
MODGD(F)	MODified Group Delay (Function)
MVN	Mean Variance Normalisation
PDF	Probability Density Function
PLP	Perceptual Linear Prediction
PMF	Probability Mass Function
PS	Product Spectrum
RV	Random Variable
SNR	Signal-to-Noise Ratio
SPLICE	Stereo Piece-wise Linear Compensation for Environment
SRH	Summation Residual Harmonic
STFT	Short-time Fourier Transform
VAD	Voice Activity Detection
VT	Vocal Tract
VTS	Vector Taylor Series
WER	Word Error Rate
WSJ	Wall Street Journal

Chapter 1

Introduction

"Where shall I begin, please your Majesty?" he asked.
"Begin at the beginning," the King said gravely,
"and go on till you come to the end: then stop."
– Lewis Carroll, *Alice's Adventures in Wonderland & Through the Looking-Glass*

Three things should be considered: problems, theorems and applications.
– Gottfried Wilhelm Leibniz, *Dissertatio de Arte Combinatoria*

1.1 Unimportance of the Phase Spectrum in Speech Processing

The Fourier analysis plays a key role in speech signal processing. As a complex quantity, it can be expressed in the polar form using the magnitude and phase spectra. The magnitude spectrum is widely used in almost every corner of speech processing. However, the phase spectrum is not an obviously appealing start point for speech signal processing. The majority of the algorithms in this field put the magnitude spectrum at the centre of attention and either directly pass the phase spectrum to the output without any processing, e.g. in speech enhancement or totally discard it, for example in the automatic speech recognition (ASR) front-ends such as the MFCC[1] [1] which serves as the Swiss Army knife of the speech processing. There are three main reasons which discourage applying the phase spectrum:

- historical bias that considers phase devoid of perceptually important information

[1] Mel-Frequency Cepstral Coefficient

- complicated shape due to the phase wrapping, hindering physical interpretation and/or mathematical modelling
- shown to be useful only in long-term analysis whereas due to the non-stationarity of speech, it should be processed in short-term

1.1.1 Historical Bias Against Phase

Historical bias dates back to the mid of 19^{th} century and originates from Ohm's acoustic Law [2] which states that the human ear acts as a spectral analyser and the phase spectrum has no effect on how the ear perceives the sound. In other words, human auditory system is *phase deaf*. This law was not welcomed at first (1843) and there were bitter debates between Seebeck and Ohm [3]. Seebeck discredited Ohm's phase law and forced him to withdraw it. However, in 1863 Helmholtz sided with Ohm [4] and convinced the community that Ohm's acoustic law is correct. Seebeck passed away in 1849 but later studies in the 20th century by Schouten [5] verified Seebeck's idea. Regardless, Ohm and Helmholtz's work lead to a bias against the usefulness of the phase in terms of carrying no perceptually important information.

1.1.2 Physical Interpretation and Mathematical Modelling

The properties of the human speech production system from both physiologic and physical points of view are clear to a great extent. For instance, for voiced sounds the vocal cords vibrate and this results in a quasi-periodic behaviour in the time domain, characterised by a fundamental periodicity which has a clear implication in the frequency domain. Also, the vocal tract could be modelled as a set of tubes with different cross sections. Based on Physics laws, such system has some resonant frequencies. These properties can be easily observed in the magnitude spectrum. However, due to the phase wrapping phenomenon, the phase spectrum takes a noise-like shape which hinders the physical interpretation and mathematical modelling.

1.1.3 Phase Usefulness in Long-term Speech Processing

It was observed that when the speech signal is decomposed into long frames (e.g. as long as 1 second), speech phase spectrum becomes important in the sense that the phase-only reconstructed speech becomes intelligible [6]. This was verified in other studies such as [7] and was shown that by frame length extension, the significance of the phase spectrum increases. However, based on the current paradigms for analysing non-stationary signals,

before analysis, they should be decomposed into short frames in which stationarity holds. For speech signal, typical frame length in which stationarity holds is in the range of 20-40 ms. This is far less than long frames in which the phase spectrum obtains some significance.

1.1.4 Magnitude Spectrum Works Well

On the other hand, the magnitude spectrum, suffers from neither of the aforementioned problems. From a perceptual standpoint, there is a consensus that it carries information characterising many aspect of the speech signal. From the physical point of view, it matches well with our understanding of the speech production system. In addition, both fundamental and resonance frequencies can be easily detected and estimated using this part of the Fourier transform. Mathematically speaking, its minima, maxima and trend have a clear interpretation and can be modelled effectively. The magnitude spectrum's statistical behaviour is studied, too, and successfully utilised in techniques like MMSE for speech enhancement [8, 9]. It also well fits the short-term analysis paradigm for non-stationary signal processing.

1.2 Research Questions and Goals

Despite the aforementioned points, the phase spectrum has been usefully employed in some applications in speech processing which will be detailed in Chapter 2. Although the conducted researches report some improvement in performance after embedding the phase in the processing pipeline, the community is still doubtful about the usefulness of phase. In parallel with researches motivating phase application in speech processing, there is a need for a fundamental model which sheds light on how the phase encodes speech information and provides a framework for manipulating the signal through phase processing.

One of the basic models which facilitates the magnitude-based speech processing is the source-filter model. It allows for modelling the vocal tract and excitation components in the magnitude spectrum domain and offers an effective framework for separating them. If such a model is developed in the phase domain, the doubts about the usefulness of the phase spectrum will be highly alleviated and phase application will be highly facilitated.

To realise the goal of source-filter modelling and separation in the phase domain, the following issues/questions should be addressed:

- Do source and filter, as speech elemental pieces of information, exist in the phase spectrum? In general, how is the signal information distributed between the phase and magnitude spectra?

- If source/filter information exists in the phase spectrum, how are they combined together and how they can be separated through phase processing?
- If the modelling/separation is carried out in the phase domain, how successful, accurate and robust is it in comparison with its magnitude-based counterpart?
- Is it possible to extract the fundamental frequency from the source component of the phase? What process is needed to do that? How accurate and robust is the extracted pitch?
- How can one parametrise the filter component of the phase spectrum and extract features form it, e.g. for ASR? How much discriminability and robustness do these features afford?
- What are the possible rooms for further optimisation of the source-filter separation framework in the phase domain to better meet the downstream processing requirements?
- What are the statistical properties of the phase spectrum and its representations?
- Having studied the statistical properties of the phase, is it possible to extend the idea of statistical normalisation and the model-based noise compensation techniques to the phase-based features to achieve higher noise robustness, similar to the magnitude spectrum?

This PhD thesis aims to address the aforementioned questions.

1.3 Contributions

- Introducing the concept of signal information regions, clarifying which pieces of information are captured uniquely by either phase or magnitude spectra and what is shared between them. The relative importance of each information region in the short and long-term analysis is studied, too (Chapter 3)
- Proposing a novel source-filter modelling and separation framework in the phase domain and comparing it with its magnitude-based counterpart in the clean and noisy conditions (Chapter 4)
- Estimating the fundamental frequency from the source component of the phase spectrum and examining its accuracy and robustness in the clean and noisy conditions (Chapter 4)

- Feature extraction from the filter component of the phase spectrum for ASR and investigating its robustness in a connected-digit (Aurora-2 [10]) and medium to large vocabulary continuous speech recognition (Aurora-4 [11]) ASR tasks at the clean and multi-style training modes along with DNNs[2] (Chapter 4)
- Studying the statistical[3] behaviour of the phase spectrum and its representations along the feature extraction pipeline, showing that phase spectrum contrary to the general uniform assumption [12] has a bell-shaped distribution (Chapter 4)
- Studying the effect of applying the statistical normalisation techniques such as mean-variance normalisation, Gaussianisation [13], Laplacianisation and histogram equalisation [14] to improve the robustness of the proposed phase's filter component-based feature at five points along the parametrisation process (Chapter 4)
- Improving the robustness of the proposed source-filter separation framework through replacing the log function with the generalised log function [15] (or power transformation) in the Hilbert transform and also substituting the sample difference with the regression filter [16] in computing the group delay (Chapter 4)
- Examining the effect of the additive and channel noise in the phase and group delay domains (Chapter 5)
- Combining the idea of vector Taylor series (VTS) [17] model-based approach to robust ASR with the power transformation which leads to developing a novel VTS-based formulation in the periodogram domain and extending the formulation to the group delay-power product spectrum domain leading to substantial performance improvement in dealing with additive noise (Chapter 5 and Appendix B)
- Developing a novel iterative channel noise estimation technique in the periodogram and the group delay-power product spectrum domains to leverage the proposed VTS-based framework resulting in significant gain in coping with the channel distortion (Chapter 5 and Appendix C)

1.4 Organisation of the Thesis

1.4.1 Chapters

The thesis is structured as follows.

[2]Deep Neural Network

[3]To get statistically significant results 244 minutes of clean speech was used.

Chapter 2 deals with the basics of the phase-based signal processing and provides a review of the phase spectrum applications in speech processing.

Chapter 3 is dedicated to investigating the importance of the phase spectrum based on its information content. The signal information regions concept is introduced which shows that in general there are three mutually exclusive regions of information for a mixed-phase signal: all-pass, minimum-phase-scale-excluded (MinPh*) and scale information. The all-pass part information is uniquely captured by the phase spectrum, MinPh* is shared by both spectra and the scale information is uniquely encoded in the magnitude spectrum. The relative importance of each region and the kind of information it encodes is evaluated. The discussions are supported by the results of the magnitude-only, phase-only, minimum-phase-only and all-pass-only reconstructed signals in short-, mid- and long-term. This chapter provides a systematic description of signal spectro-temporal information distribution and what is encoded in the phase spectrum. Also it is shown that the speech's elemental information, namely source and filter, are present in the phase spectrum.

Chapter 4 develops a source-filter model in the phase domain along with a framework for separating the vocal tract and excitation components through phase processing. To this end, the phase of the minimum-phase component of the speech is represented using a Trend-plus-Fluctuation structure and through proper filtering these two components are segregated. The efficacy of the proposed approach is investigated and compared with its magnitude-based counterpart. In this regard, the source and filter components are employed in the pitch tracking and feature extraction, respectively. Moreover, the usefulness of applying the statistical normalisation techniques such as mean-variance normalisation, Laplacianisation, Gaussianisation and Histogram equalisation (HEQ) to the proposed phase-based features is investigated. In addition, to further improve the performance of the proposed source-filter modelling and separation framework two further modifications are proposed: substituting the log with generalised logarithmic function and also applying the regression filter rather than sample difference in computing the group delay.

Chapter 5 is dedicated to robust model-based ASR. The robustness gain achieved through using statistical normalisation and generalised logarithmic function encouraged the use of more advanced model-based statistical techniques such as vector Taylor series (VTS). VTS in its original formulation assumes usage of the log function for compression. In order to simultaneously take advantage of the VTS and generalised logarithmic function, a new formulation is first developed to merge both into a unified framework called generalised VTS (gVTS). Also in order to leverage the gVTS framework, a novel channel noise estimation method is developed. The extensions of the gVTS framework and the proposed channel estimation to the group delay domain are then explored. The problems it presents are

analysed and discussed, some solutions are proposed and finally the corresponding formulae are derived. Moreover, the effect of additive noise and channel distortion in the phase and group delay domain are scrutinised and the results are utilised in deriving the gVTS equations. Experiments on Aurora-4 showed that, despite training only on the clean speech, the proposed features provide average WER[4] reductions of 0.8% absolute and 4.1% relative compared to an MFCC-based system trained on the multi-style data. Combining the gVTS with bottleneck DNN-based system led to average absolute (relative) WER improvements of 6.0 (23.5) when training on clean data and 2.5% (13.8%) when using multi-style training with additive noise.

Finally, **Chapter 6** provides a review of the main deliveries of Chapters 3, 4 and 5 along with some scope for future work.

1.4.2 Appendices

The thesis includes six appendices. In Appendix A the Hilbert transform relations between the phase and magnitude spectra for the minimum-phase and maximum-phase signals are derived in detail. The Hilbert transform is used in Chapters 3 and 4. Appendix B reviews the VTS approach to ASR along with a detailed treatment of the generalised VTS approach developed during this PhD thesis. It is utilised in Chapter 5. Appendix C explains the proposed novel technique for channel noise estimation to be used along with the (g)VTS in Chapter 5. Appendix D briefly reviews the DNN-based bottleneck set up used in the ASR experiments. The corresponding experimental results are presented in Chapters 4 and 5. Appendix E explains the databases used in different experiments and Appendix F presents the pseudo codes of the feature extraction algorithms utilised in this thesis.

1.5 List of Publications

1. E. Loweimi, J. Barker, and T. Hain, "Exploring the use of group delay for generalised VTS-based noise compensation," in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), March 2018.

2. E. Loweimi, J. Barker, and T. Hain, "Channel compensation in the generalised vector taylor series approach to robust ASR," in Interspeech 2017, Stockholm, Sweden, 2017.

3. E. Loweimi, J. Barker, O. Saz Torralba, and T. Hain, "Robust source-filter separation of speech signal in the phase domain," in Proc. Interspeech, Stockholm, Sweden, 2017.

[4] Word Error Rate

4. E. Loweimi, J. Barker, and T. Hain, "Statistical normalisation of phase-based feature representation for robust speech recognition," in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), March 2017, pp. 5310–5314.

5. E. Loweimi, J. Barker, and T. Hain, "Use of generalised nonlinearity in vector taylor series noise compensation for robust speech recognition," in Proc. Interspeech, San Francisco, USA, 2016, pp. 3798–3802.

6. E. Loweimi, J. Barker, and T. Hain, "Source-filter separation of speech signal in the phase domain." in Proc. Interspeech, 2015.

7. E. Loweimi, M. Doulaty, J. Barker, and T. Hain, "Long-term statistical feature extraction from speech signal and its application in emotion recognition," in Statistical Language and Speech Processing, SLSP 2015, 2015.

8. E. Loweimi, M. Doulaty, J. Barker, and T. Hain, "Emotion recognition from speech signal by effective combination of the generative and discriminative models," The University of Sheffield Engineering Symposium Conference Proceedings Vol. 2, vol. 2, 2015.

9. E. Loweimi, J. Barker, and T. Hain, "Compression of model-based group delay function for robust speech recognition," The University of Sheffield Engineering Symposium Conference Proceedings Vol. 1, vol. 1, 2014.

Chapter 2

Background and Related Work

I may not agree with what you say, but
I'll defend to the death your right to say it.
– Evelyn Beatrice Hall, *The Friends of Voltaire*[1]

Study the past, if you would divine the future.
– Confucius

In this chapter the basics of the phase-based signal processing and the related work are reviewed. The chapter starts with the definition of the phase spectrum and covers the problems in the phase-based processing, the available solutions and the phase representations. The properties of the group delay as the major representation of the phase spectrum, its advantages and disadvantages as well as its different variants are presented. In the second part of the chapter the related work and the applications of the phase spectrum in different branches of speech processing are reviewed.

2.1 Signal Analysis using Fourier Transform

Fourier analysis is one of the most important mathematical tools which has been extensively employed in signal processing. It takes the signal from time (or space) domain to the frequency domain and provides a new representation using complex exponentials as the basis functions. This linear operation is uniquely reversible so no information would be lost after applying it. In many applications, this representation better highlights and distinguishes the information of interest hidden in the signal and provides an effective way for manipulating

[1] Slightly misquoted: I disapprove of what you say, but I will defend to the death your right to say it.

the signal. Discrete-time Fourier transform, $\mathscr{F}$, is defined as follows [18]

$$\mathscr{F}\{x[n]\} = X(\omega) = \sum_{n=0}^{N-1} x[n]\, e^{-j\omega n} \tag{2.1}$$

where n, N, and ω denote the time, number of samples and radial frequency variables, respectively, and $X(\omega)$ is the Fourier transform of the (discrete-time) signal $x[n]$. The complex exponentials ($e^{-j\omega n}$) form the basis functions of this transform and makes $X(\omega)$ a complex quantity (in general). The complex numbers may be expressed in the Cartesian coordinates

$$X(\omega) = X_{Re}(\omega) + jX_{Im}(\omega) \tag{2.2}$$

where $X_{Re}(\omega)$ and $X_{Im}(\omega)$ indicate the real and imaginary parts, respectively. The complex numbers are also representable in the polar form as follows

$$X(\omega) = |X(\omega)|\, e^{\phi_X(\omega)} \tag{2.3}$$

where $|X(\omega)|$ and $\phi_X(\omega)$ are the magnitude[2] spectrum and phase spectrum, respectively, and are defined in the following way

$$|X(\omega)| = \sqrt{X_{Re}^2(\omega) + X_{Im}^2(\omega)} \tag{2.4}$$

$$\phi_X(\omega) = \arctan\left(\frac{X_{Im}(\omega)}{X_{Re}(\omega)}\right) \tag{2.5}$$

To be more precise, for the inverse of the tangent function, arctan, there are two variants, namely *2-quadrant* and *4-quadrant*. The former is generally referred as $atan(.)$, takes one input and its output is bounded in $(-\frac{\pi}{2}, \frac{\pi}{2})$ whereas the later is denoted by $atan2(.,.)$, has two inputs (real and imaginary parts) and its output is wrapped within the $(-\pi, \pi]$ range. The output of atan2 is called the *principle* phase spectrum, $ARG\{X(\omega)\}$. As such, a more accurate form of 2.5 would be

$$ARG\{X(\omega)\} = atan2\big(X_{Im}(\omega), X_{Re}(\omega)\big) = atan2\left(\frac{X_{Im}(\omega)}{X_{Re}(\omega)}\right) \tag{2.6}$$

In computing the $atan(.)$, only the division of $\frac{X_{Im}(\omega)}{X_{Re}(\omega)}$ is taken into account and if this value becomes negative it is not clear whether the real or the imaginary part was negative. However,

[2] Also known as amplitude spectrum.

since $atan2(.,.)$ has two inputs, the corresponding quadrant can be determined without ambiguity.

2.1.1 Short-Time Fourier Transform

The only independent variable of the Fourier transform is frequency. As a result, it only conveys spectral information of the signal and does not depend on time. This is fine if the properties of the process which generates the signal do not vary by time in the statistical sense. However, for signals like speech the characteristics of the production system changes continuously with time. As a result, the frequency content becomes time-dependent and for capturing such *dynamics* the Fourier analysis should become time-dependent. Such signals are referred to as *non-stationary* [18] signals and for analysing them through Fourier transform, one has to decompose the signal into frames in which the stationarity assumption holds i.e., the characteristics of the production system could be assumed to be fixed. The Fourier transform of a short segment of signal is called *short-time* (or *short-term*) Fourier transform (STFT) [19], $|X(t,\omega)|$, and is defined in the following way

$$X(t,\omega) = \sum_{n=0}^{N-1} x[n]\, w[tM-n]\, e^{-j\omega n} \tag{2.7}$$

where t, M and N denote the frame index (time), decimation factor (frame shift in samples) and frame length (in samples), respectively, and $w[n]$ is the analysis window. The $|X(t,\omega)|$ and $\phi_X(t,\omega)$ indicate the short-time magnitude spectrum and short-time phase spectrum, respectively. From here onward, "short-time" modifier is implied when mentioning the magnitude or phase spectra unless otherwise stated.

STFT provides an effective tool for performing a spectro-temporal analysis of the speech signal. This time-frequency representation allows for localising the events in both time and frequency domains. However, the resolution or accuracy of such localisation in the time-frequency plane is limited and higher accuracy in one axis necessarily leads to more error on the other one. This is similar to Heisenberg uncertainty principles in physics [20]. In the context of time-frequency analysis, it is referred to as the *Heisenberg-Gabor limit* [20]. By increasing the length of each segment the frequency resolution increases and simultaneously the resolution in the time domain decreases. Resolution here means the accuracy in sharply localising the events occur in a specific time or frequency bin. In short, the Heisenberg-Gabor limit says that one cannot simultaneously improve both time and frequency resolutions.

2.1.2 Usefulness of the Magnitude Spectrum

Figure 2.1 illustrates the waveform and the spectrogram of a speech signal, along with the real and imaginary parts as well as the magnitude and phase spectra. As can be observed, among the Fourier transform parts, the magnitude spectrum better characterises the speech signal and matches our understanding of the human speech production system. This system could be considered as connection of a number of tubes with different cross-sections. Based on Physics law, such system has some resonance frequencies. These frequencies in speech processing context are called *formants* [21], denoted by F_1, F_2, F_3, etc. In addition, for the voiced sounds there is a quasi-periodic motion in the vocal folds leading to some quasi-periodicity in the time domain. This periodicity relates to the *fundamental frequency* often denoted by F_0. Fundamental and formant frequencies contain useful information about the excitation and vocal tract components of the speech signal. As shown in Figure 2.1, they are well highlighted and distinguished in the magnitude spectrum domain. Therefore, the magnitude spectrum appears to be a natural choice for processing the information decoded in the speech signal.

Compliance of a function with our understanding of the production process facilitates the modelling. As seen in Figure 2.1, fine structure of the log of the magnitude spectrum is closely related to the fundamental frequency (F_0) and its harmonics. Also, its coarse structure (envelope) is linked to the vocal tract configuration and the formants. This allows for considering the log of the magnitude spectrum as a Fluctuation component modulated by a Trend component. The former is connected to the excitation and the latter is related to the vocal tract. The Trend and Fluctuation components in a Trend-plus-Fluctuation structure [23], could be separated based on their change rate through the Fourier transform. Low-pass filtering of the sequence returns the Trend and having computed the Trend, a simple subtraction gives the Fluctuation part. This forms a basis of the source-filter separation using the (log of the) magnitude spectrum which is well-established and has wide applications in a multitude of speech processing algorithms.

Another advantage of the magnitude spectrum is that its local minimum and maximum points have an interpretation. If the z-transform of a signal is expressed in a rational form, the local minima are related to the zeros and the local maxima are connected to the poles. The poles and zeros are useful in characterising the corresponding signal/system and studying its properties. This, among others, facilitates the magnitude-based signal processing.

Furthermore, the square of the magnitude spectrum, namely the periodogram, can be used for estimating the power spectrum [24]. This is useful in dealing with noise and enhancing a signal. The clean signal and additive noise are additive in the periodogram domain (assuming they are uncorrelated) and this forms the basis for enhancing a signal using techniques like

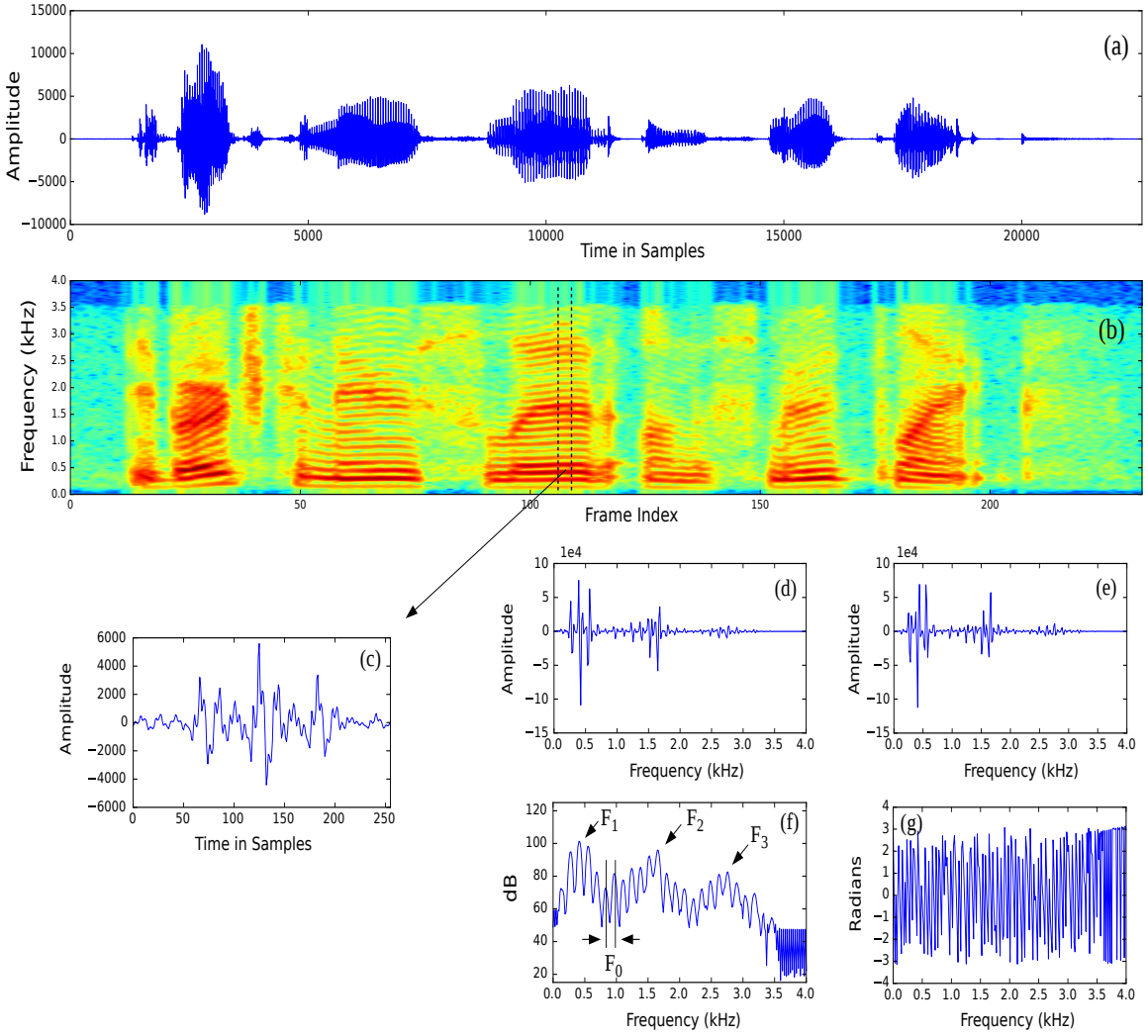

Fig. 2.1 Comparing signal representations using different parts of the Fourier transform. (a) speech signal (sp07 [22]: "We find joy in the simplest things", $f_s = 8000\ Hz$), (b) spectrogram, (c) waveform of a segment of 32 ms length, (d) real part of the STFT, (e) imaginary part of the STFT, (f) short-time magnitude spectrum, (g) short-time (principle) phase spectrum. Magnitude spectrum is the most expressive part of the Fourier transform. The cutoff frequencies at 300 Hz and 3400 Hz stem from the intermediate reference system (IRS) filter applied to NOIZEUS signals to simulate the receiving frequency characteristics of telephone handsets [22].

spectral subtraction [25]. Also, the statistical properties of the magnitude spectrum have been studied and successfully utilised in the speech enhancement based on MMSE[3] estimation criterion [8, 9].

[3]Minimum Mean Square Error

In summary, the understandable and expressive behaviour of the magnitude spectrum allows for using it in a wide range of applications in the speech and signal processing.

2.1.3 Usefulness of the Phase Spectrum

As depicted in Figure 2.1, phase spectrum does not behave as clear as the magnitude spectrum. Apparently it resembles a noise with no informative trend nor extremum points which may facilitate the physical interpretation and/or mathematical modelling. Such ambiguous structure even poses a fundamental question: is there any information in the phase at all? If there is no information in the phase spectrum, it is not worthwhile to work on it.

It should be noted that for unique recovery of the signal in the time domain both magnitude and phase spectra are required. This implies that the phase spectrum carries some information. However, the amount, type and significance of the information encoded in the phase spectrum remain unclear. The information content of the phase spectrum will be studied in detail in the next chapter.

The question which should be addressed at this point is that what is the cause of such chaotic behaviour?

2.1.4 Phase Wrapping

The main reason behind the chaotic shape of the phase spectrum is the *phase wrapping* phenomenon stemming from the four-quadrant arctangent function which is utilised in computing the principle phase. Wrapping allows for representing a quantity when it exceeds the admissible range. The other solution for keeping the values in a given range is to clip the values below or above the allowed range. The downside of the clipping is that it is irreversible and the original values can be no longer recovered. Wrapping is a mechanism which keeps the values within the given limits and the same time allows for recovering the original values through unwrapping.

Assuming that the valid range is (l_{min}, l_{max}) and $\Delta L = l_{max} - l_{min}$, wrapping process acts as follows: If the sample was outside the given range add the $m\,\Delta L$ such that $sample + m\Delta\, L$ is placed in (l_{min}, l_{max}) (m is an integer number). The principle phase spectrum is wrapped between $(-\pi, \pi]$ and this gives rise to its chaotic shape with many discontinuities. One ramification of such discontinuities, as well as rendering a chaotic shape, is that it hinders computing the phase derivative[4].

[4]In Chapter 4, it is demonstrated that the differentiation, *demodulates* the information lie in the phase spectrum, similar to FM demodulation through *slope detector* [26].

The reverse process is called *phase unwrapping* in which the target is to find a continuous representation for the phase spectrum. Unwrapped or *continuous* phase spectrum is denoted by $arg\{X(\omega)\}$. The fundamental difficulty in unwrapping the phase is that addition of any integer multiple of 2π to the principle phase does not change the complex number $X(\omega)$

$$X(\omega) = |X(\omega)|\, e^{j\phi_X(\omega)]} = |X(\omega)|\, e^{j(\phi_X(\omega)+2m\pi)}. \tag{2.8}$$

Thus, there are many possible options for m but only one correct m and continuous phase exist. Mathematically, $arg\{X(\omega)\} = ARG\{X(\omega)\} + 2\,m(\omega)\pi$ and the unwrapping process is about finding the proper value for $m(\omega)$.

Application-wise, phase unwrapping is particularly important in image processing [27] and is an integral part of methods such as Magnetic Resonance Imaging (MRI) [28], Synthetic aperture radar (SAR) [29] and fringe-pattern processing [30]. That is why most of the research carried out on the phase wrapping issue are in the image processing field. Computing the complex cepstrum is an important application of phase unwrapping which plays an important role in homomorphic signal processing and deconvolution. It also helps in determining the minimum- or maximum-phase property of the systems. For the former, the complex cepstrum becomes causal (zero at the negative quefrencies) and for the latter it takes anti-causal form (zero at the positive quefrencies) [18].

2.1.5 Unwrapping the Phase

For unwrapping the phase one can make use of the derivative of the phase spectrum, which is computable without unwrapping the phase spectrum, based on the following formulation:

$$\begin{aligned} arg'\{X(\Omega)\} &= \frac{d\; arg\{X(\omega)\}}{d\omega} = \frac{X_{Re}(\omega)X'_{Im}(\omega) - X'_{Re}(\omega)X_{Im}(\omega)}{|X(\omega)|^2} \\ &\Rightarrow \begin{cases} arg\{X(\omega)\} = \int_0^{\omega} arg'\{X(\Omega)\}d\Omega \\ arg\{X(\omega)\}\big|_{\omega=0} = 0 \end{cases} \end{aligned} \tag{2.9}$$

where $'$ denotes derivative with respect to ω. In practice, since the variables are digital, the differentiation and integration are approximately carried out using sample difference and sum, respectively. These approximations increase the error associated with this approach. Tribolet [31] proposed an adaptive numerical integration (using trapezoidal method) which first computes the unwrapped phase given the unwrapped phase at the previous bin and then checks the consistency given a permissible range for the values the unwrapped phase can take. In case of inconsistency, the integration step is halved and the process is repeated.

A more popular method for unwrapping is based on the difference between two successive samples [18]. The command *unwrap* in Matlab or Numpy Python library implement this algorithm. In this approach, the difference between the adjacent principle phase samples is computed. If the modulus of the difference becomes more than a threshold (usually π), 2π is added or subtracted such that the difference between the adjacent samples becomes less than π. In [32], this technique is referred to as discontinuity detection (DD).

Although in a controlled condition this approach works well, there are two factors which limit its functionality [33]: *undersampling*[5] and noise. They make the difference between the samples a less reliable measure and cause $false\ alarm$[6] or *miss*[7]. Note that this approach has a cumulative nature, that is, making an error in a bin gives rise to getting inaccurate results in all the higher frequency bins.

By increasing the size of the FFT, N_{FFT}, the integration step in the first approach and undersampling error in the second technique decrease and this elevates the accuracy of both methods at the cost of higher computational load. Drugman and Stylianou [32] proposed a method for computing the optimal N_{FFT}. In the proposed method, first number of zeros of the Z-transform which lie outside the unit circle is computed using the Schur-Cohn method [34]. Then, the unwrapped phase is computed using the DD technique for a FFT size of N_{FFT}. In the next step the number of zeros outside the unit circle is computed using the unwrapped phase based on Cauchy's residue theorem [35]. If it was not equal to the value returned by the Schur-Cohn method, the N_{FFT} is doubled and the unwrapping process is repeated. This techniques provides the minimal N_{FFT} for accurate unwrapping.

In general, the phase unwrapping problem is NP-hard [27], there is no global/ultimate solution for it and all the proposed methods are heuristic.

2.2 Group Delay

As shown in Figure 2.1, in contrast to the magnitude spectrum whose fine and coarse structures have a clear relation to speech perception, the phase spectrum is difficult to interpret and manipulate. Apparently, there is not a meaningful trend or extrema which may facilitate the model construction process. Dealing with this issue, researchers turned to work with other representations of the phase spectrum which carry the phase information but at the same time are more tractable. Group delay function is the major representation of the phase spectrum. It is defined as the negative of the spectral derivative of the continuous

[5] Undersampling here is not related to the Nyquist sampling rate and means the FFT size is not large enough.
[6] Unwrapping when no wrapping has taken place
[7] Wrapping takes place but the algorithm fails to detect it

(unwrapped) phase spectrum

$$\tau(t,\omega) = -\frac{\partial\ arg\{X(t,\omega)\}}{\partial\omega} \tag{2.10}$$

where $\tau(t,\omega)$ is the (short-time) group delay of the t^{th} frame. One can also compute the (short-time) group delay function in the following way

$$\tau(t,\omega) = -\frac{\partial\ arg\{X(t,\omega)\}}{\partial\omega} = -\frac{X_{Re}(t,\omega)X'_{Im}(t,\omega) - X'_{Re}(t,\omega)X_{Im}(t,\omega)}{|X(t,\omega)|^2} \tag{2.11}$$

The main advantage of this formulation is that there is no need to do phase unwrapping before computing the GD.

2.2.1 Physical Interpretation of the Group Delay

Before discussing the properties and applications of the group delay, it is helpful to discuss its physical interpretation. Consider a simple time-delay system in which the output is just the delayed version of the input. The impulse response, $h[n]$, and the transfer function, $H(\omega)$, of this system are as follows

$$\begin{cases} h[n] & = \delta[n-n_0] \\ H(\omega) & = \mathscr{F}[h(n)] = e^{-j\omega n_0} \end{cases} \tag{2.12}$$

where $\mathscr{F}$ denotes the Fourier transform and n_0 indicates the time delay.

The magnitude spectrum of this system is unity (constant) for all frequency bins whereas its phase spectrum is linear and equals $-n_0\,\omega$. It is straightforward to see that the group delay of this system equals the time delay value, i.e. n_0, for all the frequency components. Therefore, for such systems and from physical viewpoint, the group delay is related to the overall delay which the input signal is exposed to, during passing through the system.

In practice, the systems are much more complicated than a simple pure delay filter, but one may still think of the group delay as a measure of the delay of the system, although this relation is no longer as obvious as a pure time-delay system. Now, consider a narrowband filter $H(\omega)$ centred in the frequency domain around ω_0

$$H(\omega) = \begin{cases} H(\omega) & |\omega-\omega_0| < B \\ 0 & otherwise \end{cases} \tag{2.13}$$

where B is the bandwidth. If the bandwidth B is small enough, then in the vicinity of ω_0 the magnitude and phase spectra could be approximated by constant and a line, respectively,

$$\begin{aligned} |H(\omega)| &\approx |H(\omega_0)| \\ \phi_H(\omega) &\approx \phi_0 - \tau_0\,(\omega - \omega_0) \end{aligned} \tag{2.14}$$

where $\tau_0 = \tau_H(\omega_0)$. Note that these approximations are reasonable only in the neighbourhood of ω_0. In frequencies outside $|\omega - \omega_0| < B$ the magnitude spectrum is zero and consequently the phase value is not important as the $e^{j\phi}$ is multiplied by zero. As such the response, $y[n]$, of the narrowband system $h[n]$ to the input $x[n]$ will be

$$\begin{aligned} Y(\omega) = X(\omega)\,H(\omega) &\approx |X(\omega)|\,e^{j\phi_X(\omega)}\,|H(\omega_0)|\,e^{j(\phi_0 - \tau_0\,(\omega-\omega_0))} \\ \Rightarrow Y(\omega) &\approx c\,X(\omega)\,e^{-j\tau_0\omega} \\ y[n] &\approx c\,x[n-\tau_0] \end{aligned} \tag{2.15}$$

which is the shifted version of the input with a delay equal to τ_0. In this case, one can say the envelope of the carrier signal (with frequency ω_0) is delayed by the group delay, τ_0. As the bandwidth B increases, the constant magnitude and linear phase approximations are accompanied by higher error. Although it was assumed that the channel is narrowband, the same argument holds when the input signal is narrowband, too.

Having an output which is a (scaled) time-shifted version of the input, means that the system does not distort the input and this is one of the main advantages of linear phase systems. Such systems have a constant group delay which means all the frequency components are delayed by the same amount. However, in general the phase may be non-linear and consequently the group delay becomes frequency-dependent. This causes different frequency components to undergo different delay times when they pass through the system. As a result, the output signal will be a distorted version of the input. Figure 2.2 shows this effect which is called phase distortion.

It should be noted that the delay-based interpretation for the group delay is not always correct. As a matter of fact, group delay sometimes becomes negative which means that the corresponding frequency experience time-advance in passing the system. Practically this does not make sense as it violates the causality of the system [8].

[8] Physically realisable systems ought to be causal. It means that the output at each moment depends on the input on the current moment and the previous times [36].

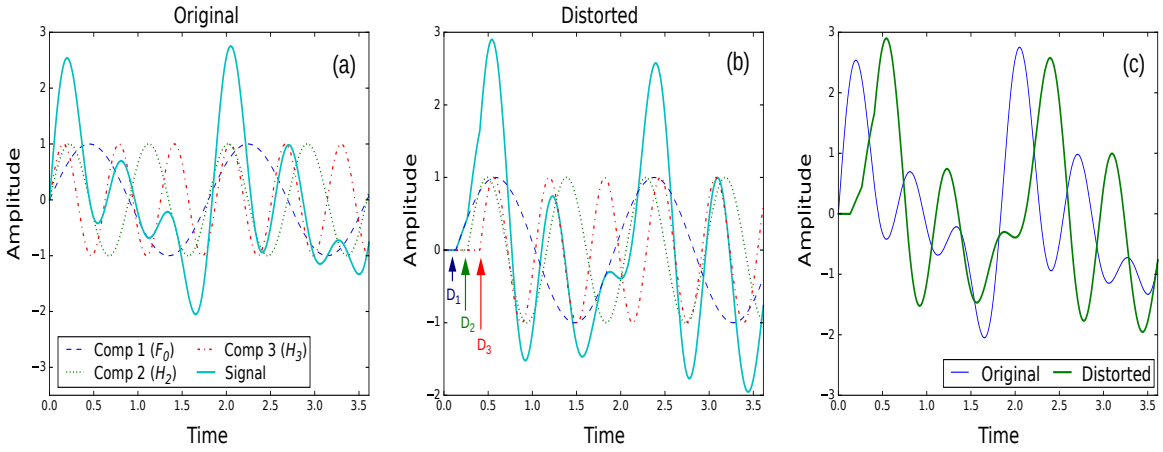

Fig. 2.2 Phase distortion after passing a signal through a filter with a flat magnitude spectrum and a non-linear phase characteristic which leads to different group delay values for different frequency components. Here, the original signal consists of three tones: fundamental frequency F_0, second harmonic, $H_2 = 2F_0$ and third harmonic, $H_3 = 3F_0$. The group delay of the filter at F_0, H_2 and H_3 equals D_1, D_2 and D_3, respectively. (a) the original signal and its components, (b) the distorted signal and its components, (c) the original signal vs the distorted signal.

2.2.2 Computing the Group Delay

For computing the group delay one needs to differentiate the $X_{Re}(\omega)$ and $X_{Im}(\omega)$. Calculation of the derivative of the discrete samples is not straightforward and is accompanied by some error. In order to evade this problem, one may take advantage of the following property of the Fourier transform

$$\mathscr{F}[nx(n)] = j\frac{dX(\omega)}{d\omega} = -X'_{Im}(\omega) + jX'_{Re}(\omega) \tag{2.16}$$

As such instead of using 2.11 for computing the group delay, the following formula can be utilised

$$\tau(t,\omega) = \frac{X_{Re}(p,\omega)Y_{Im}(t,\omega) + Y_{Re}(t,\omega)X_{Im}(t,\omega)}{|X(t,\omega)|^2} \tag{2.17}$$

where $Y(\omega)$ is the Fourier transform of $y[n] = nx(n)$. The proof is straightforward considering the following relations

$$\begin{cases} Y_{Re}(\omega) &= -X'_{Im}(\omega) \\ Y_{Im}(\omega) &= X'_{Re}(\omega) \end{cases} \tag{2.18}$$

Therefore, by this trick the group delay can be calculated without dealing with the phase wrapping and the derivative issues.

There is also another way of computing the group delay. Since $LnX(\omega) = Ln|X(\omega)| + jarg\{X(\omega)\}$ where Ln is the complex logarithm, one can write

$$\tau(\omega) = -\frac{d}{d\omega}\arg\{X(\omega)\} = -\frac{d}{d\omega}Im\{LnX(\omega)\} = -Im\{\frac{d}{d\omega}LnX(\omega)\} \tag{2.19}$$

because the $Im\{.\}$ and the derivative($\frac{d}{d\omega}$) are linear operations. Thus, their order may be swapped. Using Fourier transform and complex number properties the group delay can be computed as follows

$$\tau(\omega) = -Im\{\frac{X'(\omega)}{X(\omega)}\} = Re\{\frac{jX'(\omega)}{X(\omega)}\} = Re\{\frac{\mathscr{F}[nx[x]]}{\mathscr{F}[x[n]]}\} \tag{2.20}$$

where $Re\{.\}$ denotes the real part.

2.2.3 Group Delay and Complex Cepstrum Relationship

$LnX(\omega)$ and complex cepstrum, $\hat{x}[q]$, form a Fourier pair

$$\hat{x}[q] \underset{\mathscr{F}^{-1}}{\overset{\mathscr{F}}{\rightleftarrows}} LnX(\omega) \tag{2.21}$$

which means that

$$LnX(\omega) = \sum_{q=-\infty}^{+\infty} \hat{x}[q]e^{-j\omega q} \tag{2.22}$$

where q is the quefrency (independent variable in the cepstral domain). Rewriting both sides using Cartesian coordinates yields

$$Ln|X(\omega)| = \sum_{q=-\infty}^{+\infty} \hat{x}[q]\cos(\omega q) \tag{2.23}$$

$$arg\{X(\omega)\} = -\sum_{q=-\infty}^{+\infty} \hat{x}[q]\sin(\omega q) \tag{2.24}$$

As a result,

$$\tau(\omega) = -\frac{d}{d\omega}arg\{X(\omega)\} = \sum_{q=-\infty}^{+\infty} q\hat{x}[q]\cos(\omega q). \tag{2.25}$$

There is also another way of looking at the relationship between the complex cepstrum and the group delay. Complex cepstrum is a real quantity for real signals [9] and can be expressed as the addition of the even and odd parts as follows

$$\begin{cases} \hat{x}_{even}[q] = \frac{\hat{x}[q]+\hat{x}[-q]}{2} & \& \quad \hat{x}_{even}[q] = \hat{x}_{even}(-q) \\ \\ \hat{x}_{odd}[q] = \frac{\hat{x}[q]-\hat{x}[-q]}{2} & \& \quad \hat{x}_{odd}[q] = -\hat{x}_{odd}[-q] \end{cases} \tag{2.26}$$

where *even* and *odd* subscripts denote the even and odd parts, respectively. Based on the Fourier transform properties, the following Fourier pairs get formed

$$\hat{x}_{even}[q] \underset{\mathscr{F}^{-1}}{\overset{\mathscr{F}}{\rightleftarrows}} Ln|X(\omega)| \tag{2.27}$$

$$\hat{x}_{odd}[q] \underset{\mathscr{F}^{-1}}{\overset{\mathscr{F}}{\rightleftarrows}} j\arg\{X(\omega)\} \tag{2.28}$$

where $\hat{x}_{even}[q]$ equals the *real cepstrum*. Taking advantage of the properties of the even and odd sequences

$$Ln|X(\omega)| = \sum_{q=-\infty}^{+\infty} \hat{x}_{even}[q]e^{-j\omega q} = \hat{x}_{even}[0] + 2\sum_{q=1}^{+\infty} \hat{x}_{even}[q]\cos(q\,\omega) \tag{2.29}$$

$$arg\{X(\omega)\} = -\sum_{q=-\infty}^{+\infty} \hat{x}_{odd}[q]e^{-j\omega q} = -\,2\sum_{q=1}^{+\infty} \hat{x}_{odd}[q]\,\sin(q\,\omega) \tag{2.30}$$

Therefore,

$$\tau(\omega) = -\frac{d}{d\omega}arg\{X(\omega)\} = \,2\sum_{q=1}^{+\infty} q\,\hat{x}_{odd}[q]\,\cos(q\omega) = \sum_{q=-\infty}^{+\infty} q\hat{x}_{odd}[q]e^{-j\,\omega\,q} \tag{2.31}$$

$$\Rightarrow\; q\,\hat{x}_{odd}[q] \underset{\mathscr{F}^{-1}}{\overset{\mathscr{F}}{\rightleftarrows}} \tau(\omega) \tag{2.32}$$

which shows that $q\hat{x}_{odd}[q]$ and the group delay form a Fourier pair.

[9] It is called complex cepstrum because of using complex logarithm in its computation.

Special Case: Minimum-phase Signals

An important special case that is worth mentioning is the minimum-phase case for which the complex cepstrum becomes causal [18]

$$\hat{x}[q] = 0, \quad \forall\, q < 0. \tag{2.33}$$

Consequently,

$$\hat{x}_{even}[q] = \begin{cases} \frac{\hat{x}[-q]}{2}, & q < 0 \\ \hat{x}[0], & q = 0 \\ \frac{\hat{x}[q]}{2}, & q > 0 \end{cases} \tag{2.34}$$

and

$$\hat{x}_{odd}[q] = \begin{cases} \frac{-\hat{x}[-q]}{2}, & q < 0 \\ 0, & q = 0 \\ \frac{\hat{x}[q]}{2}, & q > 0 \end{cases} \tag{2.35}$$

Therefore, the (2.29) and (2.30) take the following forms

$$Ln|X(\omega)| = \hat{x}[0] + \sum_{q=1}^{+\infty} \hat{x}[q]\ \cos(q\omega) \tag{2.36}$$

$$arg\{X(\omega)\} = -\sum_{q=1}^{+\infty} \hat{x}[q]\ \sin(q\omega) \tag{2.37}$$

and finally the group delay will be

$$\tau(\omega) = \sum_{q=1}^{+\infty} q\,\hat{x}[q]\ \cos(q\omega). \tag{2.38}$$

2.2.4 Advantages of the Group Delay

The group delay function has four main advantages:

- high spectral resolution
- low spectral leakage
- additivity
- resemblance to the magnitude spectrum

Group delay represents the phase spectrum information but its overall shape is more understandable. For minimum-phase signals, similar to the magnitude spectrum, it peaks at the poles and becomes minimum at the zeros. However, for the maximum-phase signals, opposite to the magnitude spectrum it has a minimum at poles and maximum at zeros. Figure 2.3 illustrates this point.

Figure 2.4 shows the magnitude spectrum and the group delay of a simple signal/system characterised by six poles estimated using LPC[10] [37] analysis. As can be seen, the group delay has a noticeably better frequency resolution and lower frequency leakage. It also resembles the magnitude spectrum in terms of having peaks at the poles and valleys at the zeros [11]. In Chapter 4, the reason behind the resemblance of the group delay to the magnitude spectrum is explained.

Finally, if two signals get convolved in the time domain, e.g. a signal ($x[n]$) and an impulse response ($h[n]$) of a filter, their group delays will be added

$$\begin{aligned} \mathscr{F}\{x[n] * h[n]\} = X(\omega)\,H(\omega) = |X(\omega)|\,|H(\omega)|\,e^{j(arg\{X(\omega)\}+arg\{H(\omega)\})} \\ \Rightarrow \tau_{x*h}(\omega) = \tau_x(\omega) + \tau_h(\omega) \end{aligned} \tag{2.39}$$

Therefore, convolution in the time domain is equivalent to addition in the unwrapped phase and group delay domains.

2.2.5 Main Problem with the Group Delay

Group delay suffers from a significant issue which limits its usefulness in practice. Figure 2.5 shows the magnitude spectrum and the group delay of a typical speech signal. As can be seen, in this case the group delay is very spiky and neither the fundamental frequency nor the formants can be distinguished visually. The spikes are due to poles and/or zeros located close to the unit circle. For a general auto-regressive moving average (ARMA) model

$$\begin{aligned} X(z) = \frac{B(z)}{A(z)} = \frac{\prod_{q=1}^{Q}(z-z_q)}{\prod_{p=P}(z-z_p)} \\ \Rightarrow \tau_X(\omega) = \sum_{q=1}^{Q}\tau_q(\omega) - \sum_{p=1}^{P}\tau_p(\omega) \end{aligned} \tag{2.40}$$

[10] Linear Predictive Coding

[11] This is true for the minimum-phase signals. For the zeros/poles outside the unit circle, the group delay exhibits an opposite behaviour, namely peaks at zero and valley at poles. Models like LPC are guaranteed to give poles inside unit circles. So, the group delay of the all-pole model behaves similar to the magnitude spectrum.

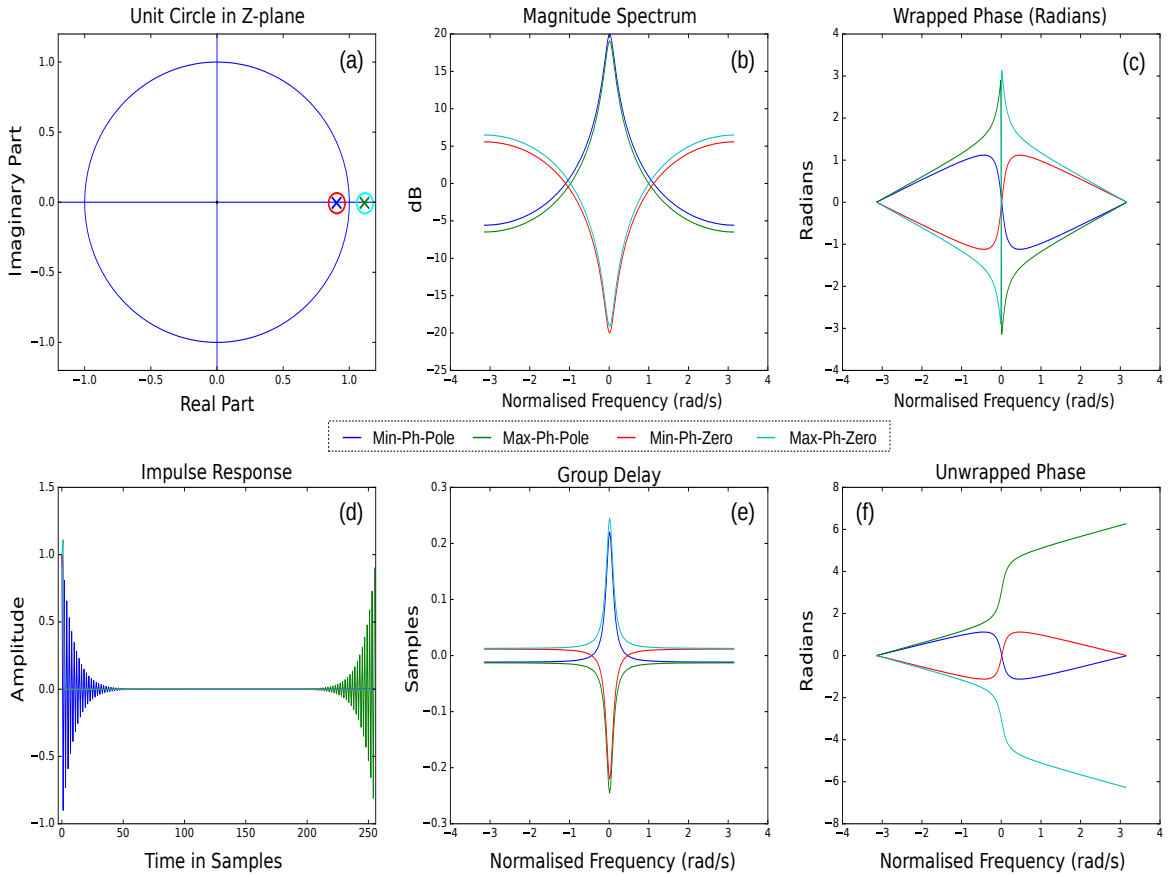

Fig. 2.3 Behaviour of the magnitude spectrum, principle phase spectrum, unwrapped phase spectrum and group delay of a single pole or a single zero inside or outside the unit circle. (a) zero/pole location, (a) magnitude spectrum, (c) principle (wrapped) phase spectrum, (d) impulse response in time domain, (e) group delay, (f) unwrapped phase spectrum. For minimum-phase poles or zeros the magnitude spectrum and group delay have similar behaviour in terms of having peak at poles and valley at zeros whereas for the maximum-phase poles or zeros the group delay and the magnitude spectrum have opposite behaviour.

where z_q, z_p, Q, P, τ_q and τ_p indicate the zeros, poles, number of zeros, number of poles, group delay of z_q and group delay of z_p, respectively. Using (2.19), the group delay of such system can be calculated as follows

$$\tau(\omega) = Im\frac{d\,Log\{X(\omega)\}}{d\omega} = -Im\{\frac{\frac{d}{d\omega}[\frac{B(\omega)}{A(\omega)}]}{\frac{B(\omega)}{A(\omega)}}\} = Im\{\frac{A'(\omega)B(\omega) - A(\omega)B'(\omega)}{A(\omega)B(\omega)}\}. \quad (2.41)$$

As can be observed in (2.41), when a pole or zero approaches the unit circle, the denominator tends to zero and the group delay at the corresponding frequency becomes spiky. Since the group delay of the poles/zeros are added together, if only one of the group delays

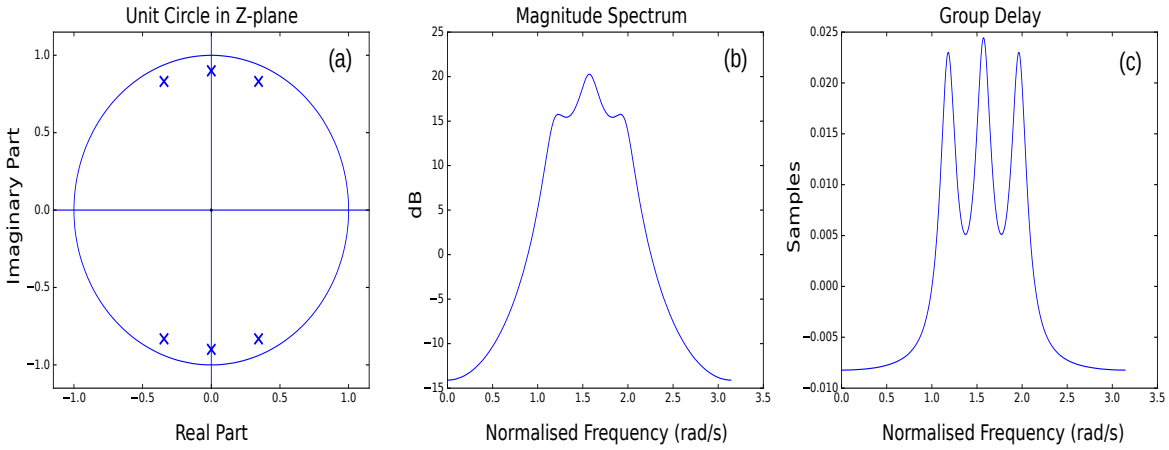

Fig. 2.4 LPC and GD of a signal characterised by having six poles. As seen group delay resembles the LPC-based (parametric) power spectrum estimate. (a) poles in the z-plane, (b) LPC-based (parametric) power spectrum estimation, (c) group delay function.

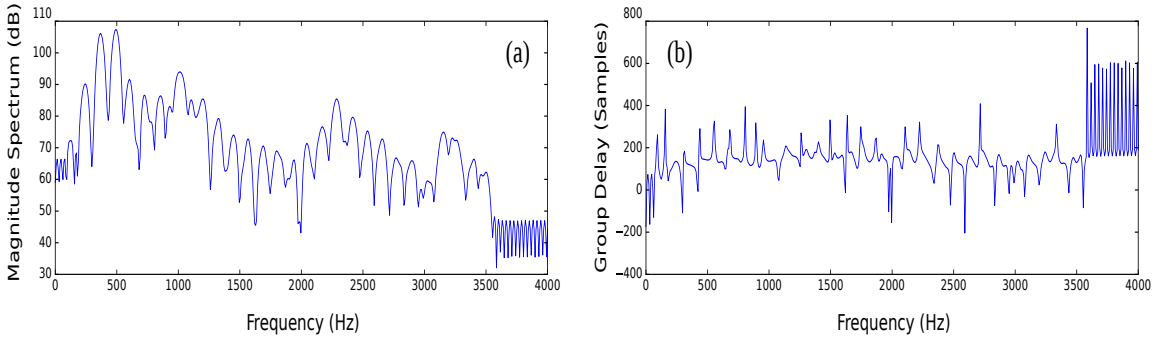

Fig. 2.5 The (a) log of the magnitude spectrum and (b) group delay for a voiced speech frame. The group delay could be very spiky, if is left uncontrolled.

becomes spiky, it will make the whole group delay spiky. These spikes mask the formant and other useful aspects of the spectrum and ultimately make the group delay a less useful representation. For speech signals, the poles which are mainly associated with the vocal tract, are located in an almost safe distant from the unit circle. That is why the group delay of an all-pole model of speech is smooth. However, the zeros of the speech which are mainly related to the excitation component are located close to the unit circle and lead to a spiky group delay.

2.2.6 Dealing with the Group Delay Spikiness

To exploit the advantages of the group delay, the spikiness problem should be solved. There are four main solutions to this issue

- Modified group delay (MODGD)
- Product Spectrum (PS)
- Chirp group delay function (CGDF)
- Model-based (parametric) group delay

Modified Group Delay

In 1992, Murthy and Yegnanarayana proposed the modified group delay (MODGD) which solves the spikiness issue of the group delay to a great extent [38]. They proposed to filter out the excitation component from the denominator of the group delay in (2.17), through cepstral smoothing [39] the squared magnitude spectrum

$$\tau(\omega) = \frac{X_{Re}(\omega)Y_{Re}(\omega) + X_{Im}(\omega)Y_{Im}(\omega)}{S(\omega)} \tag{2.42}$$

where $S(\omega)$ is the cepstrally smoothed power spectrum.

Cepstral smoothing is basically a low-time liftering of the cepstrum. In fact, the logarithm of the speech magnitude spectrum can be thought of as a superposition of two elements: a quickly oscillating component modulated by a slowly varying envelope. The former is related to the excitation component and the latter is linked to the vocal tract. In the cepstrum domain, the low-quefrency components are related to the slowly varying component, namely the vocal tract and high-quefrency components are connected to the excitation part. Now if one applies a low-time lifter the excitation component can be filtered out and a smoothed power spectrum is achieved. Note that smoothing is another name for low-pass filtering and cepstral smoothing is basically low-time liftering in the quefrency domain. The low-time lifter can be sharp (brick-wall) or may have some roll-off rate. In the work presented here, a brick-wall filter is utilised with length l. The higher the l, the lower the smoothness.

Modified group delay still has another problem which needs to be addressed. The dynamic range is still high and the bandwidth of the formants is too low due to the sharpness of the peaks. In this regard, Murthy and Gadde [40] proposed the following extra modification

$$\begin{aligned} \tau(\omega) &= \frac{X_{Re}(\omega)Y_{Re}(\omega) + X_{Im}(\omega)Y_{Im}(\omega)}{S^{\gamma}(\omega)} \\ \tau(\omega) &\leftarrow sign\{\tau(\omega)\}\,|\tau(\omega)|^{\alpha} \end{aligned} \tag{2.43}$$

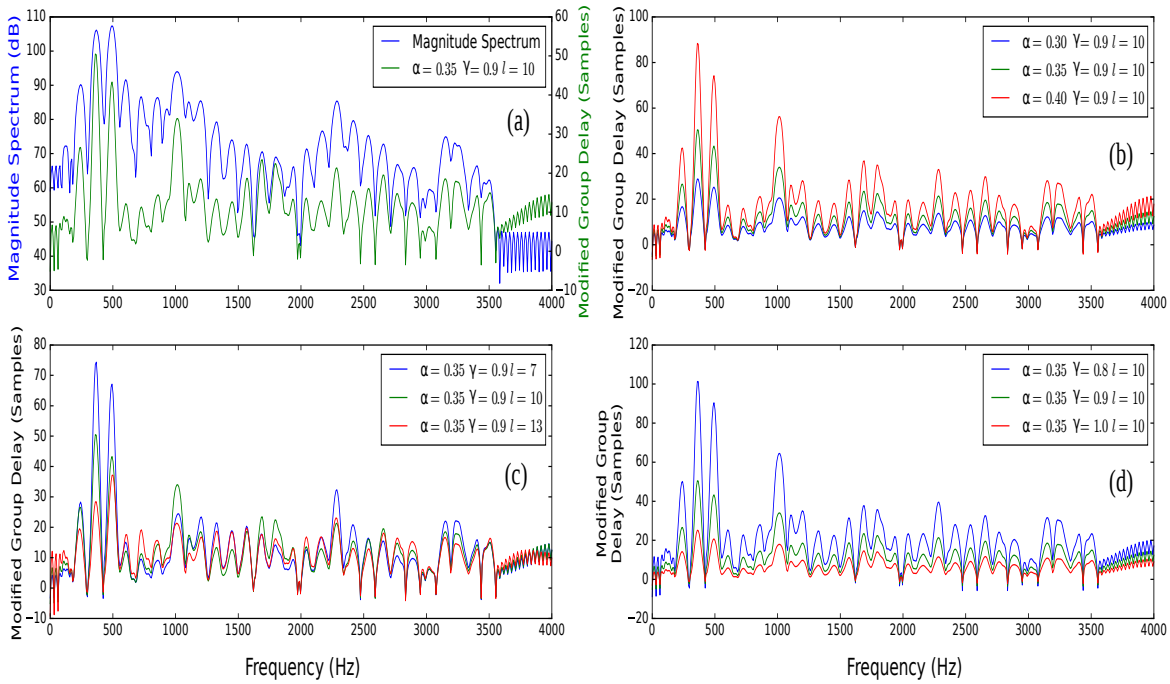

Fig. 2.6 Comparing the power spectrum with the modified group delay. (a) power spectrum plotted along with the modified group delay function (2.43), (b) effect of α on the modified group delay, (c) effect of l on the modified group delay, (d) effect of γ on the modified group delay.

where α and γ are new parameters introduced for dynamic range and formant bandwidth adjustment. The appropriate value for α is proposed to be in the range of 0.3 to 0.4 and optimum value for γ is suggested to be around 0.9 [40, 41]. Therefore, the MODGD has three parameters, namely α, γ and l. Figure 2.6 shows the influence of these parameters on this function. By taking the DCT[12] of the modified group delay a set of features can be extracted from it. The recognition results of this feature will be presented in Chapter 4.

Product Spectrum

This method was proposed in 2004 by Zhu and Paliwal [42] and as the name implies the group delay is multiplied by *something*. As mentioned the spikiness issue stems from the excitation component of the speech which pushes the denominator toward zero. Dealing with this problem, in this work, the denominator was removed through multiplying the group delay with the periodogram estimate of the power spectrum

$$Q(\omega) = \frac{X_{Re}(\omega)Y_{Re}(\omega) + X_{Im}(\omega)Y_{Im}(\omega)}{|X(\omega)|^2}|X(\omega)|^2 \tag{2.44}$$

[12]Discrete Cosine Transform. Mostly, type II is used.

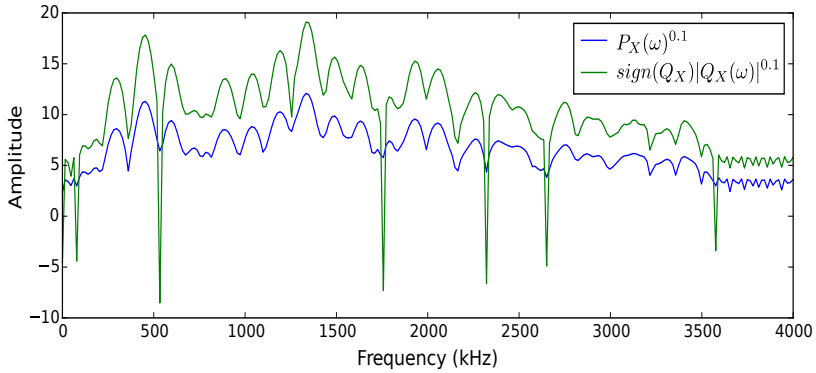

Fig. 2.7 Product spectrum ($Q_X(\omega)$) along with the periodogram power spectrum estimate ($P_X(\omega) = |X|^2$). Since the product spectrum could be negative, for a better visualisation, it was compressed through 2.43 ($\alpha = 0.1$). Deep valleys in the product spectrum stem from zeros placed next to the unit circle.

where $Q(\omega)$ is the group delay-power product spectrum or simply the product spectrum (PS). Figure 2.7 illustrates the magnitude spectrum along with the product spectrum. As can be seen, the product spectrum highly resembles the magnitude spectrum. The main difference between them lies in the sharp valleys in the product spectrum which stems from the sensitivity of the group delay to the zeros located in the vicinity of the unit circle.

Chirp Group Delay

Bozkurt and Dutoit [43] proposed the chirp group delay (CGD) for dealing with the spikiness issue which takes advantage of the chirp z-transform [44]. Evaluating the z-transform on the unit circle i.e. replacing the z with $e^{j\omega}$, yields the Fourier transform. In the chirp z-transform, however, the z-transform is evaluated off the unit circle either on a circle with a different radius or a spiral. The benefit is that by pushing the evaluation circle or spiral closer to the poles or zeros, the corresponding peaks and/or valleys get more pronounced. This potentially could result in having higher frequency resolution, without changing the frame length. The inverse is also true, that is, evaluating the z-transform on a circle which is farther from a pole/zero decreases the contribution of that pole/zero and leads to higher smoothness. Using a circle for evaluating the z-transform leads to having the same frequency resolution over all frequency bins whereas using spiral allows for having non-uniform frequency resolution over different frequency bands. Although the chirp processing appears to be a more flexible method, finding the optimal value for the circle radius or optimal shape for the spiral is not straightforward.

Backing to the group delay problem, based on the above argument, if the z-transform is evaluated on a circle with a radius larger than one, the effect of the zeros will be decreased

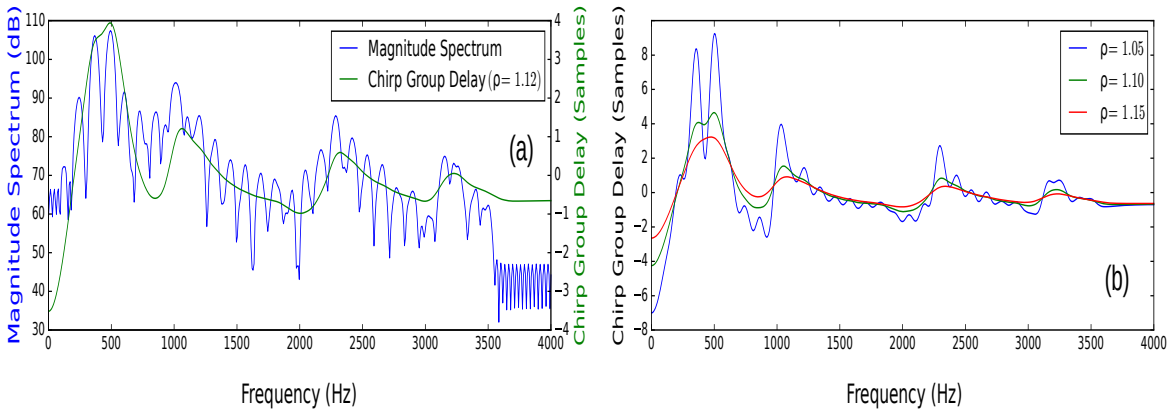

Fig. 2.8 Chirp group delay proposed in [43]. (a) Chirp group delay along with the magnitude spectrum, (b) effect of the radius (ρ) of the circle on which the z-transform is evaluated. The larger the radius, the higher the smoothness.

because they are mostly concentrated around the unit circle. It means that their negative effect, namely too sharp peaks, will be mitigated. However, it comes at the cost of losing some frequency resolution because the poles contribution also gets weaker since they are located inside the unit circle. Therefore, there is a trade-off: The higher the radius, the more the smoothness (as far as the zeros are concerned) and the less the resolution (as far as the poles are concerned).

There is one more point which should be considered. Although the zeros are mainly distributed around the unit circle (both inside and outside), there is no limit or restriction on their radius. As such increasing the radius of the circle where the z-transform is evaluated, does not necessarily solve the spikiness problem. This also poses a problem in finding the optimal radius for the chirp processing. To deal with this issue, Bozkurt and Dutoit [43] proposed to first compute the zero-phase signal, namely inverse Fourier transform of the signal after setting its phase spectrum to zero ($\mathscr{F}^{-1}\{|X(\omega)|\}$), and then conduct the chirp processing. The advantage is that the zero-phase signal includes only the magnitude spectrum and the magnitude spectrum is uniquely linked to the minimum-phase signal. This guarantees that all the zeros of the z-transform of the zero-phase signal remain inside the unit circle. As such using a radius larger than 1 necessarily leads to smoothing. The optimum value for radius is reported to be 1.12 in [43]. Figure 2.8 depicts the effect of the radius and the trade-off between the spectral resolution and smoothness after chirp processing.

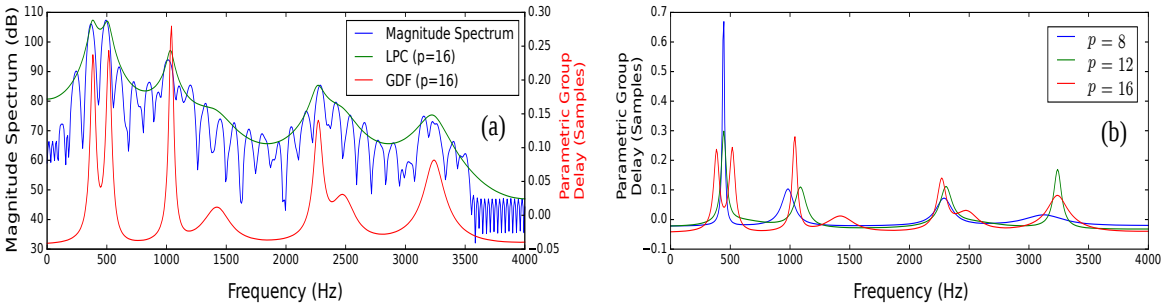

Fig. 2.9 Group delay of an all-pole model. (a) Parametric group delay along with the parametric magnitude spectrum and (non-parametric) magnitude spectrum, (b) group delay of the all-pole model for different model orders (sampling rate is 8 kHz).

Model-based Group Delay

The idea of the model-based or parametric group delay is to instead of directly computing the GD of the signal, fit a, in general, ARMA[13] model and then compute the group delay. Based on the negative effect of the zeros on the group delay, all-pole model is a better option and helps in removing the spikes caused by the zeros associated with the excitation component. Figure 2.9 shows the group delay computed after LPC analysis of the speech signal for different model orders. As can be observed, the group delay affords a remarkably higher spectral resolution and lower frequency leakage in comparison with the log magnitude spectrum and the parametric magnitude spectrum after all-pole modelling.

2.3 Applications of Phase spectrum in Speech Processing

In the first part of this chapter, the theoretical aspects of the phase-based signal processing was covered. The second part is dedicated to reviewing the main applications of the phase spectrum in speech processing.

2.3.1 Speech Analysis

To the best of our knowledge, Yegnanarayana [45] was the first who used the group delay of an all-pole model for extracting the formants. As shown in Figure 2.9, in such case, the group delay behaves similar to the magnitude spectrum, namely peaks at poles but with the advantage of having higher frequency resolution and less spectral leakage. This study initialised a wide range of group delay-based research in India.

[13] Autoregressive Moving Average

Note that the linear predictive modelling plays a crucial role and if the group delay is computed directly on the signal, it will be too spiky to be useful. In the early 90s modified group delay (MODGD) was proposed which directly calculates the group delay from the speech signal [46, 38] without the need for all-pole modelling. As explained earlier, the spikiness of the group delay stems from zeros of the excitation component and in the MODGD the cepstral smoothing was employed to deal with this issue. In [47], the idea of the cepstral smoothing was carried out using the root cepstral analysis (originally proposed by Lim in 1979 [48]) and shown to be more robust to noise. Application-wise, the modified group delay [38] was proposed for the power spectrum estimation and was compared with the periodogram method. It was shown that by using the modified group delay function, the high resolution property of the periodogram power spectrum estimation is preserved while the variance is reduced due to having less leakage. As a side note, the main advantage of the periodogram method is its high spectral resolution and the main disadvantage is having high variance (due to frequency leakage) [24].

Group delay also found application in pitch extraction. As shown in Figure 2.6, the modified group delay function also includes the speech source component information. In particular, the spectral distance between the peaks in the modified group delay domain corresponds to the fundamental frequency (F_0) [49], similar to the power spectrum domain. In [50, 51], it has been employed for melodic pitch extraction from music. They propose to first flatten the spectrum using high-time liftering in the root (or generalised) cepstrum domain and then compute the modified group delay. In [52], the idea was extended to the two-pitch extraction problem from speech and music signals.

Another application of the group delay in speech analysis is in determining the instants of the significant excitation [14] in speech [54, 55]. In [54], the average slope of the unwrapped phase of the short-time Fourier transform of the linear prediction residual was calculated as a function of time. Instants where the phase slope function (which is defined as a function of time or frame index) makes a positive zero-crossing (from negative to positive) where identified as significant excitations. In [55], first an approximate epoch locations were determined using the Hilbert envelope of the linear prediction residual and then an accurate locations of the significant excitation instants were computed using the group delay around the approximate epoch locations.

In addition to the spectral derivative of the short-time phase spectrum, its temporal derivative, namely instantaneous frequency (IF) has been utilised for speech analysis, too. In [56], it was observed that the instantaneous frequency in narrowband analysis includes informa-

[14] Significant excitation primarily refers the instants of glottal closure (epoch) in the case of voiced speech and onset of burst in the case of unvoiced speech [53].

tion related to speech excitation component. In [57], Charpentier proposed a technique for extracting the pitch frequency from the narrowband instantaneous frequency. In [58, 59], it was demonstrated that if the signal is decomposed into normal frame lengths (20 to 40 ms) and under *pitch-synchronous* analysis, the IF can be employed in computing the pitch frequency. It was called *narrowband* IF. On the hand, if the signal decomposed into very short frames, as long as half of the pitch period and again after pitch synchronous analysis, IF captures the formant frequencies. It was called *wideband* IF [60].

Stark et al [61] proposed a representation based on the *instantaneous frequency deviation* (IFD) as an alternative to the magnitude spectrum. The IFD is defined as follows

$$IFD(t,\omega) = ARG\{X(t+1,\omega)\, X^*(t,\omega)\, e^{-j\omega}\} \tag{2.45}$$

and they observed that the inverse of the IFD, namely $|IFD(\omega)|^{-1}$, is proportional to the magnitude spectrum and resembles it. Contrary to the narrowband IF which only carries pitch information, the proposed representation contains both pitch and formant frequencies.

Also note that in (2.45), first the FFTs are multiplied and then the *ARG* is computed instead of directly subtracting the corresponding phase spectra. It is a clever trick proposed by Steven Kay in [62] to avoid phase (un)wrapping issue because for adding/subtracting the phase spectra first they should be unwrapped. However, using this simple yet effective point, one only needs to multiply the FFTs and then compute the phase spectrum which can be easily carried out without any ambiguity.

Stark and Paliwal also proposed a similar function to IFD called *inverse group delay deviation* (IGDD) [63] which is an extension of the IFD idea to the group delay domain and leads to a representation that resembles the magnitude spectrum, too. Inverse group delay deviation, $\eta(\omega)$, is defined as

$$\eta(\omega) = |\tau_{win}(\omega) - \tau_x(\omega)|^{-1} \tag{2.46}$$

where $\tau_x(\omega)$ and τ_{win} indicate the group delay of the frame x and the group delay of the analysis window, respectively. For a causal symmetric windows, $\tau_{win} = \frac{N-1}{2}$ where N is the frame length in samples. It was also proposed to post-process the $\eta(\omega)$ with a smoothing filter.

Relative Phase Shift (RPS) is another phase-based representation for speech analysis used in harmonic modelling framework. It converts the phase to a measure of a relative phase shift between harmonic components and the fundamental frequency [64]. It was employed for polarity detection in [65] and was demonstrated to be a relatively accurate and robust measure. RPS was converted to a feature by passing it through Mel filter bank followed

by the DCT. It was used for ASR [66] and also in a GMM-based speaker recognition [67] system. Performance-wise, it was shown to be useful as an augmentation to the MFCC feature.

McCowan et al [68] proposed another representation for the short-time phase spectrum based on its temporal derivative, called *delta-phase* (DP). It bears resemblance with the instantaneous frequency deviation in the sense that both are related to the temporal evolution of the short-time phase spectrum. However, contrary to the IFD in which the temporal derivative of the phase is computed, delta phase captures a coarser phase change over a longer temporal context, as long as the frame shift. Delta phase was turned into a feature vector in a MFCC-like framework by replacing the power spectrum with the absolute value of the delta phase. The extracted feature was named MFDP[15] and evaluated in speaker recognition and voice activity detection (VAD). In both tasks MFCC outperformed the MFDP but it was shown that the combination of MFCC with MFDP could result in some performance improvement.

2.3.2 Quality/Intelligibility of the Phase-only Reconstructed Speech

Signal reconstruction from partial Fourier transform was a hot topic in the early 80s. Oppenheim and Lim [69, 6] studied the importance of the phase spectrum for some signals like speech and image. It was illustrated that the phase-only reconstructed speech is intelligible when the signal is decomposed into long frames. For both image and speech signals, it was shown that the signal reconstructed through 1 $e^{j\phi(\omega)}$, where $\phi(\omega)$ is the global phase spectrum (computed over the whole signal) highly resembles the original signal and was intelligible.

Hayes et al [70] demonstrated that a finite length signal is recoverable from its phase spectrum up to a scale error. The corresponding theorem is as follows

Theorem 1 *A sequence which is known to be zero outside the interval* $0 \leq n \leq N-1$ *is uniquely specified to within a scale factor by* $N-1$ *distinct samples of its phase spectrum in the interval* $0 \leq \omega \leq \pi$ *if it has a z-transform with no zeros on the unit circle or in conjugate reciprocal pairs* [16] *[70].*

For obtaining $N-1$ distinct phase samples at the positive frequencies the length of the FFT should be (greater than or) equal to $2N$. They also proposed an iterative algorithm for reconstructing a signal (a single frame) from its phase or magnitude spectrum which is explained in the next chapter.

[15] Mel-Frequency Delta-Phase

[16] z_0 and $\frac{1}{z_0^*}$ form a conjugate reciprocal pair.

Note that for mixed-phase signals like speech [71], a signal (or the frame) is not recoverable from its magnitude spectrum. Van Hove et al [72] showed that the signal is recoverable from its *signed-magnitude spectrum* which is defined as follows

$$X(\omega;\alpha) = |X(\omega)| \begin{cases} +1 & \alpha - \pi \leq \phi_X(\omega) \leq \alpha \\ -1 & otherwise \end{cases} \tag{2.47}$$

Theorem 2 *Let $x[n]$ and $y[n]$ be two real, causal and finite extent sequences with z-transform which have no zeros on the unit circle. If $X(\omega;\alpha) = Y(\omega;\alpha)$ for all ω then $x[n] = y[n]$.*

It should be noted that for realising the potentials of the phase and magnitude spectra in reconstructing the original signal, in the most cases, iterative algorithms like [70] are required. However, for a certain class of signals, there is an (almost) one-to-one relationship between the phase and magnitude spectra. As such, by having one of them, the other part and then the whole signal is recoverable entirely. Minimum-phase and maximum-phase signals fall in this category. This unique relationship is imposed by the causality and anti-causality of the complex cepstrum for the minimum-phase and maximum phase signals, respectively [18]. In this case, the unwrapped phase spectrum and consequently the signal can be uniquely computed from the magnitude spectrum. On the other hand, the magnitude spectrum and the signal can be recovered from the (unwrapped) phase spectrum to within a scale error [17]. For these types of signals there is no need for turning to iterative techniques, and using the Hilbert transform [18] in a non-iterative framework suffices to recover the signal from its partial Fourier transform. For more details about the Hilbert transform please refer to Appendix A.

Yegnanarayana et al [73] studied the significance of the group delay in signal reconstruction and attempted to unify the problems of signal reconstruction only from the magnitude and phase spectra through the use of the group delay. They demonstrated that phase information is more important for recovering an image signal, whereas for speech signals the short-term magnitude spectrum information is more important.

The aforementioned researches were carried out from the signal processing perspective. Liu He, and Palm [7] studied the relative importance of the speech phase spectrum relative to the magnitude spectrum using a perceptually motivated approach. They measured the information content of the phase and magnitude spectra of the speech signal in terms of the intelligibility of the phase- and magnitude-only reconstructed signals. The subjects were asked to listen to a carrier sentence "Hear aCa now" and recognise the consonant *C*.

[17] The scale error equals $exp(\hat{x}[0])$ where $\hat{x}[0]$ is the value of the real (or complex) cepstrum at zero.

The *C* is called *intervocalic consonant* since it occurs between two vowels. The C was one of the six *stop* (also known as *plosive*) consonants, namely /b/, /d/, /g/, /p/, /t/ and /k/. In recording process 10 speakers were participated. The signals with the aforementioned lingual content were decomposed into frame lengths of 16, 32, 64, 128, 256 and 512 ms with 50% overlap along with Hamming windowing. Then the Fourier transform of each frame was computed. For constructing the magnitude-only stimuli the phase spectrum was replaced with a uniformly distributed random number in range of $(-\pi, \pi)$. Similarly, for generating the phase-only stimuli the magnitude spectrum was set to unity. Finally, the speech signals were synthesised using overlap-add (OLA) method (non-iteratively). They played the reconstructed speeches for the subjects and asked them to recognise the intervocalic consonant C.

Experimental results showed that by frame length extension the intelligibility of the magnitude-only reconstructed signal decreases. On the other hand, the phase-only stimuli behave differently and by frame length extension they become more intelligible. This trend is in agreement with the observations of Oppenheim and Lim [69, 6] for speech signal. Liu et al also showed that the intelligibility of the phase-only stimuli exceeds that of its magnitude-only counterpart at frames longer than 128 ms.

Paliwal and Wojcicki [74] showed that the optimal frame length to achieve maximum intelligibility for the magnitude-only reconstructed speech is in range of 15 to 35 ms. The intelligibility was measured using speech transmission index (STI) [75, 76]. Similar result was observed in [71], with the difference that PESQ [77] measure was employed for evaluating the quality of the magnitude-only stimuli. PESQ quality measure is among the most reliable objective measures and in [78] it is shown to be highly correlated with the subjective mean opinion score (MOS).

Alsteris and Paliwal [79–83] did a similar set of experiments to [7] with the following differences

- Instead of six consonants they used 16 commonly occurring consonants in Australian English in the same carrier sentence, "Hear aCa now"
- Increased the overlap to 87.5% (frame shift equals $\frac{\text{frame length}}{8}$)
- As well as Hamming they investigated the effect of the rectangular window
- Influence of the FFT length and zero padding were examined
- Two frame lengths were studied in the subjective tests: 32 ms and 1024 ms.

The frame overlap was increased because for recovering the signal after short-term analysis using Hamming window, one needs at least 75% overlap [84]. For staying in a safer

side, they increased the overlap to 87.5%. Higher overlap imposes a heavier constrain on the frames during synthesis through OLA and leads to a signal with a higher quality. This comes at the cost of increasing the computational load as the number of frames is doubled in comparison with 75% overlap.

Regarding the window shape, they observed that for the magnitude-only stimuli, using the Hamming window leads to a higher intelligibility and for the phase-only signal reconstruction rectangular window results in a better intelligibility.

Effect of the window shape was further studied in [85]. This time as well as the Rectangular and Hamming windows, the Chebyshev (a.k.a. Dolph-Chebyshev) window was studied. This window, similar to Kaiser window [86] has two parameters where the second parameter equals the attenuation at the side-lobes relative to the main-lobe in dB and is called dynamic range. One special property of the Chebyshev window is that all the side-lobes have the same amplitude. It was shown that applying this window can further improve the intelligibility of the phase-only reconstructed speech. The optimal range of the dynamic range was reported to be 15 to 25 dB [85] and maximum intelligibility was obtained using Chebyshev window with dynamic range of 15 dB. The reason behind optimality of the Chebyshev window was explained as follows: in case of having too small magnitude spectrum both real and imaginary parts become almost zero and as such phase spectrum $arctan(\frac{X_{Im}}{X_{Re}}) = arctan(\frac{0}{0})$ become unreliable as $\frac{0}{0}$ leads to numerical instability.

Another issue is the effect of the zero-padding. In [81, 80], it was observed that using FFT length of $2N$ instead of N where N is the number of samples of each frame, improves the intelligibility of the phase-only reconstructed speech by 3.1% (absolute). Although in these set of experiments they have not reconstructed the signal iteratively, in iterative signal reconstruction, zero padding plays a significant role, as mentioned earlier in this section, in Theorem 1 [70].

Shi, Modirhanechi and Aarabi [87] investigated the importance of the phase spectrum in human speech recognition. They used a database of signals of 78 preselected words sampled at 44.1 kHz and asked the subjects to recognise the word in the played signal. The procedure is as follows. First, they decomposed the speech signals into frame lengths of 11.6 ms (512 samples per frame) with 50% overlap and multiplied it in Hanning window. Then, the signals were contaminated by additive Gaussian noise at -10, -5, 0 and 20 dB levels. In the next step, they modified the phase spectrum of the noisy signal according to this formula

$$\hat{X}_{noisy}(\omega) = (1-\alpha)\phi_{clean}(\omega) + \alpha\phi_{rand} \tag{2.48}$$

where α is called *phase noise factor* and ϕ is a sequence of uniformly distributed random numbers in $(-\pi, \pi)$. The α was varied between 0 (perfect phase) and 1 (completely random phase). Finally, they reconstructed the signal via LSEE [18] [88], using 100 iterations.

It was observed that at high SNRs changing the phase noise factor does not alter the recognition rates. However, by SNR reduction the effect of the α on the recognition rates become more noticeable. They conclude that in high SNRs the phase information is not important, however, as the SNR decreases, using the clean phase spectrum becomes more useful and influential on the intelligibility of the words. They also separated the results for male and female speakers. It was observed that randomising the phase is slightly more problematic for the male speakers and concluded that phase is a bit more important for male (low-pitch) speakers.

In [89], for a wide range of frame lengths ($\frac{1}{16}$ to 2048 ms), two hybrid signals were obtained by a cross-wise combination[19] of the magnitude and phase spectra of the speech and white noise. Speech intelligibility experiments showed that the significance of the phase spectrum for the frames longer than 256 ms and shorter than 4 ms is higher than the magnitude spectrum.

Loweimi and Ahadi [90] studied the quality of the phase and magnitude-only reconstructed speech versus frame length and for different window shapes. For the phase and magnitude-only reconstruction they used iterative approach with 100 iterations and measured the quality of the reconstructed signals using PESQ [77], weighted spectral slope (WSS) [91] and log-likelihood ratio (LLR) [92]. For synthesis, both OLA and LSEE were utilised.

Similar to [7], they observed that the quality of the magnitude-only stimuli decreases by frame length extension, contrary to the phase-only reconstructed signals. The cross-over point (where the magnitude-only and phase-only stimuli show the same quality) was illustrated to be in 256 ms using Hamming window and 128 ms after employing the rectangular window. It was also shown that the rectangular and Hamming window result in a higher quality for the phase-only and magnitude-only speech reconstruction, respectively. Regarding the synthesis method, the LSEE and OLA resulted in higher quality in case of the magnitude-only and phase-only speech reconstruction, respectively. In [93], they showed that the phase-only signal reconstruction requires more iterations than the magnitude-only one for full realisation of the phase spectrum potentials. The effective number of iterations were determined by comparing the PESQ scores in two consecutive iterations such that when it become less than a preset threshold the iteration would stop. It was shown that the crossover point in case of using Rectangular window and after enough iterations would happen in 64 ms.

[18] Least square error estimation which also known as weighted OLA (WOLA).

[19] Cross-wise combination in this paper means swapping a part of the Fourier transform of two signals.

In many researches cited above, it was observed that by frame length extension the quality/intelligibility of the phase-only reconstructed speech increases. However, the reason behind this trend was not justified. In [71], Loweimi, Ahadi and Sheikhzadeh showed that this is due to the scale incompatibility in the synthesis stage. As mentioned earlier, potentially a finite length signal can be recovered from its phase spectrum up to a scale error. Now, if a frame is taken into account on its own, the scale information just serves as the intensity with no effect on the quality/intelligibility of the signal. However, when different frames are overlapped and added for recovering the speech signal, the aforementioned scale issue becomes problematic because each frame has a specific scale error. In [71], this error was called *scale incompatibility error* (SIE) and was quantitatively shown that it decreases by frame length extension.

In addition, they illustrated that if in the phase-only signal reconstruction the magnitude spectrum of frame t is initialised by a constant $exp(\hat{x}[t,0])$, near perfect phase-only signal reconstruction could be achieved in all the frame lengths (both short and long). In this paper also the effect of a wide range of windows, including Chebyshev with a many dynamic ranges was studied. It was shown that maximum quality in PESQ scale is achieved in case of using the Chebyshev window with dynamic range of 20-30 dB[20]. It was also shown that for achieving a near-perfect phase-only speech reconstruction both scale information and appropriate window shape are required.

2.3.3 Speech Coding

One of the first practical application of the phase spectrum in speech coding is the phase vocoder proposed by Flanagan [21] and Golden [56]. Before the phase vocoder, channel vocoder was the dominant coding scheme proposed by Dudley [95] in 1939. The idea of the channel vocoder is to encode each frame using a bank of contiguous bandpass filters (typically 16-20 [21]) and append the code with the voicing information and the pitch frequency (for the voiced frames). In the phase vocoder, similar to the channel vocoder, the magnitude spectrum is encoded by the filter bank (typically 25-30 [21]) but instead of the pitch and voicing information, the temporal derivative of the phase spectrum is utilised. In the decoder (synthesis stage), the channel vocoder is designed based on the idea of source-filter model for speech [96] whereas the phase vocoder synthesises the speech from the (envelope of) the magnitude spectrum and the phase temporal derivative.

[20] In [85] maximum consonant intelligibility (or identification) was achieved in an almost similar range (15-25 dB).

[21] James L. Flanagan: A Scholar and a True Gentleman [94]

The resulting bit-rate was in the range of 7200 bps [22], higher than the channel vocoder which had a bit rate in range of 2400 to 3000 bps. *Time-scale modification*[23] and *frequency shifting*[24] are two applications of the phase vocoder [56]. In [97] an efficient implementation for phase vocoder using the FFT was proposed by Portnoff. The phase vocoder also found some application in the sub-band coding [98] and computer music [99]. Two main artifacts of the phase vocoder, especially in time-scale modification, are the *transient smearing* and the so-called *phasiness* (also known as reverberation or loss of presence) which are studied in [100] and some technical improvements have been suggested.

In sinusoidal coding, instead of using the phase evolution over time like phase vocoder, phase is estimated at the decoder based on the minimum-phase assumption [101]. It turns out that this assumption is more problematic for low-pitch sounds. In other words, sinusoidal coders, which dedicate more bits to encode the magnitude information, generally produce higher quality speech for female speech than for the male speech, whereas the quality of the male speech is better than female speech in Code Excited Linear Prediction (CELP) coders [102]. In CELP scheme, the excitation component has two parts which are a fixed code book for unvoiced speech and an adaptive code book for the voiced sounds. The input of the adaptive part is the delayed version of the excitation component and this feedback provides some temporal information, normally captured by the phase spectrum. As a result, the lack of the phase information is compensated to a certain extent.

Polloth and Keign [103] studied the phase importance in coding through investigating the perceptual phase capacity, C, which is defined as follows

$$C = log_2(k_{opt}) \tag{2.49}$$

where k_{opt} is the optimal (minimal) size of the codebook which can represent all the phase spectra in a perceptually accurate way. Number of bits needed for encoding the indices of the optimal codebook forms the perceptual phase entropy. If all the entries are assumed to be equiprobable the entropy will equal the C. Note that assigning equal probability to all the possible events leads to maximum entropy, so perceptual phase capacity is the upper bound for the required number of bits. It was shown that perceptual phase capacity closely relates to the pitch such that the lower the pitch, the higher the C. This is compatible with the well-known fact that speech coding schemes which reserve the phase information more accurately, work better for male voices while coders which assign more bits for coding the magnitude spectrum result in a better quality for female (high-pitched) speech.

[22] bit per second

[23] Temporal expansion and/or compression without changing the pitch

[24] Change of the pitch without changing the articulation speed

In [104], the characteristics of the human phase perception were analysed in terms of just-noticeable difference (JND)[25] of phase. Experiments were carried out for five vowels and seven subjects participated in the subjective tests. Results indicated that human perception of phase varies with frequency, especially for low-pitched speakers. This dependence was reported to be particularly strong in the mid-frequency range (1-3 kHz) [104].

Agiomyrgiannakis and Stylianou [106] studied the importance of the phase quantisation for speech coding in the context of harmonic representation of the speech signal. They used the wrapped Gaussian mixture model (WGMM) to construct a phase quantiser. WGMM statistically fits the random variables that belong to circular spaces like phase. For training the WGMM, an EM-based formulation was derived. The proposed quantiser was employed in a prototype variable rate narrow-band VoIP sinusoidal codec that is equivalent to iLBC [107] in terms of PESQ-MOS [77], at about 13 kbps. For more details please refer to [108].

2.3.4 Speech Synthesis

Two major paradigms in speech synthesis are the unit-selection [109] and statistical parametric (HMM-based) [110] approach. One issue in the unit-selection speech synthesis is that the concatenation of the acoustic units should be done coherently such that the continuity of the synthesised speech at the concatenation point is preserved. To achieve such coherency, the concatenated units should be time-synchronous which requires removing the linear phase mismatch between the adjacent segments. In [111], Stylianou has studied this problem and two solutions based on *centre of gravity* and *differentiated phase data* were proposed. It was demonstrated that both techniques improve the quality of the synthesised speech without imposing extra computational burden on the synthesiser because the required extra processing could be done off-line.

In the HMM-based statistical parametric speech synthesis, the phase and consequently the signal is synthesised using the minimum-phase assumption [110], namely negative quefrency components which are related to the all-pass component of the signal are set to zero. In [112], the importance of the phase spectrum in HMM-based speech synthesis was studied. In this regard, the all-pass component was retrieved using the non-causal part (negative quefrencies) of the complex cepstrum and the first 13 coefficients of the all-pass component in the cepstral domain were directly used as parameters in the observation vector assuming to represent the phase overlooked information. Experimental results show that inclusion of the phase information in the aforementioned way improves the quality of the synthesised speech at the cost of computation overhead on unwrapping the phase for computing the complex cepstrum.

[25] JND is the amount something must be changed in order for a difference to be noticeable, detectable at least half the time [105].

2.3.5 Feature Extraction for ASR

The power spectrum is a key ingredient in the feature extraction process from the speech signal. Since the group delay resembles it, speech parametrisation based on the group delay could be carried out using the well-established power spectrum-based techniques such as MFCC, the Swiss Army knife of speech processing. Of course, some appropriate modifications and optimisations should be conducted.

Paper published by Bayya and Yegnanarayana [113] in 1999 is among the first attempts to extract feature from the group delay for automatic speech recognition. Assuming the $r[l] << 1$ (and $r[0] = 1$) where $r[l]$ is the autocorrelation at the l^{th} lag, they establish a relation between the group delay and the autocorrelation. Then by truncating the autocorrelation, they derived a smoothed estimated of the group delay. Finally, by taking DCT a set of feature was extracted from the smoothed group delay. Recognition results on an isolated-digit speech recognition task showed comparable results to the LPCC[26] feature.

The modified group delay (MODGD) was used for feature extraction from speech in [40] and evaluated in a phoneme recognition task. In this paper, the MODGD was directly turned into feature through DCT and lead to comparable results to MFCC. In [114**?**], the modified group delay feature was evaluated in language identification, speaker recognition and syllabus recognition tasks. It was shown that combining the MFCC and MODGD in a single feature vector leads to slightly higher recognition accuracy.

The chirp group delay function is another phase-based representation that provides an estimate for the power spectrum. As such it can be parametrised using MFCC framework. In this regard, Bozkurt et al [43] extracted a feature from the chirp group delay for ASR using a MFCC-like framework with two major modifications: the periodogram estimate was replaced by the chirp group delay and the filter bank energies were directly passed to the DCT block without compression through logarithm.

Loweimi and Ahadi used the group delay of the all-pole model for feature extraction [115]. They extracted an all-pole model from each frame through LPC and Burg methods[27] [24] and compressed the group delay of the model using two-stage DCT to decrease the compression loss imposed by one stage DCT. In [117], instead of two stage DCT, first the group delay spectrum of the all-pole model was passed through the Mel filter bank, similar to MFCC, and then DCT was applied. In addition, the phase-based features were warped (Gaussianised [13]) and instead of log of energy, the scale information were derived using

[26] Linear Prediction Cepstral Coefficient

[27] In the LPC method the all-pole model coefficients are estimated through minimising the forward prediction error in the least squares sense whereas Burg's method is based on minimisation of the sum of the forward and backward squared prediction errors [116].

the Hilbert transform which is similar to c_0 in MFCC framework. These modifications lead to substantial robustness improvement in Aurora-2 [10] ASR task. In both aforementioned work, the output of the Mel filter bank was directly passed to the DCT block without applying any non-linearity. In [118], it was shown that applying power transformation non-linearity in between the filter bank and the DCT stages can slightly improve the performance.

As well as the spectral derivative of the phase, temporal derivative of the phase also has been used in speech recognition. In [119], standard MFCC feature vector was appended by phase temporal evolution and a new feature vector was built with and without using additional linear discriminant analysis (LDA). Experimental results on a German digit recogniser showed upto 25% relative WER reduction.

Wang et al [120] extracted two features from the instantaneous frequency called average instantaneous frequency (AIF) and average log-envelope (ALE) and employed them in a connected digit speech recognition (Aurora-2) task. In [121], instantaneous frequency was used along with other short-term features such as Teager energy, spectral moments and cepstral features for ASR. Paliwal and Atal [122] used the instantaneous frequency in vowel recognition and showed that (for this task) it works as well as LPCC and MFCC. They filtered the *analytic signal*[28] with a number of bandpass filters (10), uniformly spaced on the Mel frequency scale. The instantaneous amplitude and instantaneous frequency were computed for each filter. In the last step the histogram of the IF was computed and turned into features for speech recognition through taking DCT.

2.3.6 Speaker Recognition

Hegde et al [123] applied the modified group delay function in speaker identification tasks. In [124] the robustness of the group delay to the additive noise was studied in three cases, namely $SNR >> 1$, $SNR \approx 1$ and $SNR << 1$. Then, the MODGD features were used in the NIST 2003 speaker verification task and reportedly outperformed the MFCCs.

In [125], the MODGDs were used in an i-Vector-based speaker recognition system [126] to model target speakers in the total variability space. Vijayana and Murty [127] investigated the importance of the phase spectrum through all-pass modelling. They fitted an $M^{th} order$

[28] Analytic signal is defined as follows

$$x_a(t) = x(t) + j \underbrace{\mathscr{H}\{x(t)\}}_{H_x} = A(t)\, e^{j\phi(t)} \tag{2.50}$$

where $\mathscr{H}$, $A(t)$ and $\phi(t)$ denote the Hilbert transform, instantaneous amplitude and instantaneous phase, respectively, and here t means time, not the frame index.

all-pass model on what they called it *phase signal*[29]. The coefficients of a parametric all-pass model [30] are estimated iteratively through maximising the fourth-order cumulants using the method proposed by Chi et al [128]. The order of the all-pass model was set in the range of 15-25 and the extracted coefficients were used as features in a GMM-based speaker recognition system leading to 6% equal error rate (EER[31]) on a population of 50 speakers.

In [129], the significance of the instantaneous phase (phase of the analytic signal which the authors have called the analytic phase) was studied in a speaker verification task. It was demonstrated that when the analytic phase of the speech signal is distorted, the resulting signal sounds like whispered speech and this made it difficult for the subjects to verify the identity of the speaker. Also an algorithm was proposed for extracting feature from the instantaneous frequency, named IFCC[32] and led to comparable results to FDLP [33] [130] and MFCC in NIST 2010 speaker verification task. They also investigated the feature fusion by concatenating the i-vectors from all the three features (MFCC+FDLP+IFCC) and showed that the resulting hybrid i-vector system outperformed the individual systems.

Another application of the phase spectrum in speaker verification and identification was in [131] where the MFCC features got augmented by the phase information and in both tasks some improvements were achieved. In this paper, the phase spectrum was turned into feature vector in a special and different way. It was argued that the phase spectrum depends on the point from which the frame starts. To minimise such dependency, authors suggested to fix the phase of a certain reference frequency like ω_0 in ϕ_0 ($\phi_X(\omega_0) = \phi_0$) and recompute the phase of other frequencies relatively as follows

$$\begin{aligned} \hat{X}(\omega) &= |X(\omega)|\, e^{j\phi(\omega)}\, e^{j\frac{\omega}{\omega_0}(\phi_0 - \phi(\omega))} \\ \hat{\phi}(\omega) &= \angle\{\hat{X}(\omega)\} \xrightarrow{\text{Feature}} \{cos(\hat{\phi}(\omega)), sin(\hat{\phi}(\omega))\}. \end{aligned} \tag{2.52}$$

[29] phase signal = $IFFT\{e^{j\phi(\omega)}\}$ where $\phi(\omega)$ is the phase spectrum. Note that this representation is all-pass in terms of having unit magnitude spectrum, but differs from the all-pass component achieved through all-pass/minimum-phase decomposition. In fact, $\phi(\omega) = \phi_{MinPhase}(\omega) + \phi_{AllPass}(\omega)$ which obviously includes the minimum-phase information.

[30] The z-transform of a M^{th} order all-pass filter is as follows

$$\begin{aligned} H_{AllPass} &= z^{-M}\, \frac{\prod_{m=1}^{M}(1 - z_m z)}{\prod_{m=1}^{M}(1 - z_m^* z^{-1})} \\ &= \frac{a_M + a_{M-1}z^{-1} + a_{M-2}z^{-2} + ... + z^{-M}}{1 + a_1 z^{-1} + a_2 z^{-2} + ... + a_M z^{-M}} \end{aligned} \tag{2.51}$$

where $|z_m| < 1$ and * denotes the conjugate operator.

[31] EER is a point (threshold) in which two types of errors, namely the *false reject rate* and the *false accept rate* become equal. In speech processing, it is mainly employed in the speaker verification task.

[32] Instantaneous Frequency Cepstral Coefficient

[33] Frequency Domain Leaner Prediction

As a feature vector, only the $\hat{\phi}(\omega)$ at the frequency range of 60 to 700 Hz was utilised. Using $cos(\hat{\phi}), sin(\hat{\phi})$ as feature led to better results than directly using $\hat{\phi}$, although MFCCs returned a noticeably better results than both. Combining MFCCs with these phase-based features provided some improvement in both speaker identification and verification tasks.

2.3.7 Emotion Recognition

Once the phase or group delay were turned into a set of features, they potentially can be employed in a wide range of classification problems. For example, in [132] the group delay function was applied for emotion recognition as follows: LPC coefficients were extracted for each frame, the corresponding group delay was computed and finally turned into features by taking DCT. The feature vector was augmented by prosodic features such as pitch, energy and ZCR. In the last stage, just before passing the features to the classifier, the extracted parameters were post-processed through feature warping which Gaussianises [13] them. In [133], both modified group delay and group delay of the all-pole model were employed in emotion recognition from whispered speech. It was observed that merging the magnitude and phase-based features can improve the performance.

2.3.8 Synthetic Speech and Spoofing Detection

Wu et al [134], used the modified group delay function for synthetic speech and spoofing detection. As well as the modified group delay feature which captures the signal characteristics in the short-term, a set of features were extracted from the modulation spectrum of the group delay aiming at capturing the long-term properties of the speech signal. In [135] and [136] phase information in the form of the relative phase shift (RPS) along with MFCC was employed for synthetic speech and spoofing detection. It was shown that the phase-based parameters are useful in improving the security and decreasing the vulnerability of the speaker verification systems in dealing with the unknown spoofing and synthetic speech attacks.

2.3.9 Speech Enhancement

In most of the speech enhancement methods, after computing the Fourier transform, only the noisy magnitude spectrum enters into the enhancement process and the phase spectrum is directly transferred to the output without any modification [12]. It has been shown that under certain conditions, the noisy phase is the optimal phase in the MMSE sense [8] assuming

- independence between the Fourier transform coefficients

- the phase and magnitude spectra are independent
- the phase spectrum is uniformly distributed in $(-\pi, \pi)$ and the magnitude spectrum has a Rayleigh pdf
- the real and imaginary parts' distribution is Gaussian.

As such MMSE estimate for the clean phase, given noisy observation, could be computed as follows

$$\begin{aligned} \min_{e^{j\hat{\phi}_X(\omega)}} \quad & \mathbb{E}\{|e^{j\phi_X(\omega)} - e^{j\hat{\phi}_X}(\omega)|\} \\ & \text{subject to: } |e^{j\hat{\phi}_X(\omega)}| = 1 \end{aligned} \tag{2.53}$$

where $\phi_X(\omega)$ and $\hat{\phi}_X(\omega)$ are the clean phase spectrum and its estimate, respectively. The optimal solution is shown to be

$$e^{j\hat{\phi}_X(\omega)} = e^{j\phi_Y(\omega)} \tag{2.54}$$

where $\phi_Y(\omega)$ is the noisy phase spectrum. Note that removing the constraint $|e^{j\hat{\phi}_X(\omega)}| = 1$, results in the following estimate [12]

$$\begin{aligned} exp(j\hat{\phi}_X(\omega)) &= \frac{\sqrt{\pi}}{2}\sqrt{\nu_\omega}\, exp(\frac{-\nu_\omega}{2})\big[I_0(\frac{\nu_\omega}{2}) + I_1(\frac{\nu_\omega}{2})\big]\, e^{j\phi_Y(\omega)} \\ &= B(\omega)\, e^{j\phi_Y(\omega)} \qquad (2.55) \\ \Rightarrow \hat{X}(\omega) &= |X_{Enh}(\omega)|\, B(\omega)\, e^{j\phi_Y(\omega)} \qquad (2.56) \end{aligned}$$

where ν_ω is a function of SNR [34], I_0 and I_1 indicate the modified Bessel functions of zero and first order, and the $\hat{X}(\omega)$ and $|X_{Enh}(\omega)|$ are the Fourier transform of the enhanced signal and the enhanced magnitude spectrum, respectively. As seen, the overall magnitude of the enhanced signal is $|X_{Enh}(\omega)|\, B(\omega)$. It should be noted that the optimal magnitude spectrum is $|X_{Enh}(\omega)|$ and after multiplication with $B(\omega)$, it is no longer optimal. This negatively affects the magnitude spectrum of the enhanced Fourier transform. That is why the constraint $|e^{j\hat{\phi}_X(\omega)}| = 1$ [35] was imposed which results in $\phi_{\hat{X}}(\omega) = \phi_Y(\omega)$.

Note that in case of the Wiener filter and spectral subtraction [25] techniques the gain function is real and therefore noisy phase similar to MMSE remains unchanged. Vary [137]

[34] $\nu_\omega = \frac{\xi_\omega}{1+\xi_\omega}\gamma_\omega$ where ξ and γ denote the a priori and a posteriori SNRs, respectively.

[35] It seems obvious that $|e^{j\hat{\phi}_X(\omega)}| = 1$, so why is it mentioned explicitly? Because in the estimation process $e^{j\hat{\phi}_X(\omega)}$ is treated as a single variable, if first the phase was estimated and then $exp(j\phi)$, the modulus would be one and there was no need to express this constraint explicitly.

demonstrated that for local SNRs larger than 6 dB, the noisy phase is a reasonable estimate of the clean phase. However, for the voiced speech, phase distortions are only perceivable if the local SNR in a time-frequency bin is lower than 6 dB.

From perceptual side, Wang and Lim [138] were the first who studied the (un)importance of the phase spectrum in speech enhancement context. In their experiments, the 10 kHz sampled signals were decomposed into frame lengths of 6.4 ms, 51.2 ms and 409.6 ms with 50% overlap along with Hanning widow. They first contaminated the clean signal through white Gaussian noise at two different SNR levels, say $y_1[n]$ and $y_2[n]$, and then built synthetic stimuli by combining them as follows $y[n] = \mathscr{F}^{-1}\{|Y_1(\omega)|\, e^{\phi_{Y_2}(\omega)}\}$. They played a noisy signal with a known SNR for the subjects and asked them to compare it with the synthetic stimuli and let them know when both have the same SNR level. So, by this clever trick, they indirectly estimated the SNR of the synthetic stimuli.

Their study showed that better estimation of the noisy phase spectrum does not lead to noticeable improvement in the speech quality. Better estimate means phase spectrum is taken from a signal with a higher SNR. It was shown that the SNR gain obtained by mixing the noisy magnitude spectrum with an almost clean phase (SNR = 25 dB) results in SNR improvements of upto 1 dB at the SNR levels of -5, 5 and 15 dB. As such Wang and Lim concluded that phase is *unimportant* in speech enhancement and that is why it is stated in the title of their paper.

In 2006 Shannon and Paliwal [139] revisited the significance or possible role of the phase spectrum in speech enhancement. In their experiments the enhanced speech was synthesised using the magnitude spectrum taken from the noisy signal in conjunction with the phase spectrum of the corresponding clean signal. For measuring the quality of the enhanced signal PESQ and Enhanced Modified Bark Spectral Distortion (EMBSD) [140] objective measures [36] were employed. In the analysis stage, the magnitude spectrum of the noisy signal was computed using the Hamming window and in computing the phase spectrum six window types, namely Rectangular, Hamming and Chebyshev with dynamic range of 10, 20, 30 and 40 dB were utilised. Rectangular window was used in synthesis. Frame length was set to 32 ms and the frame shift was 4 ms (one eighth of the frame length). Experimental results showed that in all the tested SNRs (0, 5, 10 and 15 dB), using the clean phase spectrum consistently leads to quality improvement. Chebyshev window with dynamic range of 30 dB turned out to be the best option and led to about 0.6-0.7 PESQ score improvement in all the tested SNRs. Applying the clean phase spectrum extracted using Hamming window in analysis stage resulted in about 0.2-0.3 PESQ score elevation, almost in all the SNR levels.

[36] Note that the higher the PESQ, the higher the quality whereas the lower the EMBSD the better the quality.

In 2008, Wojcicky et al [141] proposed a novel phase-based speech enhancement method which is called *phase spectrum compensation (PSC)*. The idea is that the weaker spectral components of the noisy signal which are assumed to be dominantly noise, relative to the stronger components which are presumably clean speech, get more affected by offsetting the complex spectrum ($X(t,\omega)$) with a constant. The algorithm runs as follows

$$X_\Lambda(t,\omega) = X(t,\omega) + \Lambda(\omega) = |X_\Lambda|\, e^{\phi_\Lambda(\omega)} \tag{2.57}$$

where $X_\Lambda(t,\omega)$ is the modified complex spectrum and $\Lambda(\omega)$ is a real-valued frequency dependent function defined as follows

$$\Lambda(\omega) \begin{cases} +\lambda & 0 < \omega < \pi \\ -\lambda & -\pi < \omega < 0 \end{cases} \tag{2.58}$$

Note that $\Lambda(\omega)$ should be anti-symmetric to achieve noise cancellation (Fig. 2 in [141]). The λ should be proportional with the noise level and the higher the λ, the higher the noise cancellation. Clearly, the optimal value of λ is SNR dependent and the lower the SNR, the higher the optimal λ.

$$X_{Enh}(t,\omega) = |X(t,\omega)|\, e^{j\phi_\Lambda(t,\omega)} \;\rightarrow\; x_{Enh}(t,n) = Real\{\mathscr{F}^{-1}\{X_{Enh}(t,\omega)\}\} \;\rightarrow\; OLA \tag{2.59}$$

Now the issue is to optimise the λ which depends on the SNR level and noise-type. They studied three noise types, namely white Gaussian, Train and Babble. Maximum improvement was observed for Gaussian noise: 0.55 PESQ scores for $\lambda \approx 1$ in 15 dB SNR and 0.36 PESQ scores using $\lambda \approx 3.7$ in 0 dB. Minimum improvement was reported for Babble noise: 0.15 PESQ scores for $\lambda \approx 1$ in 15 dB SNR and 0.29 PESQ scores using $\lambda \approx 9.5$ in 0 dB. As seen, for the Babble noise the algorithm works better in the low SNRs, contrary to Gaussian and Train noise types, which is counter intuitive and backs to the non-stationarity of Babble noise. Note that white Gaussian noise is stationary and predictable; hence, naturally the enhancement algorithms better cope with it. In general, it was shown that the performance of this approach is comparable to the spectral subtraction and MMSE (given that the optimal value for λ is applied).

One issue with this approach is estimation of the λ. As explained it should be proportional with the noise level. In this regard, Stark et al [142] made the Λ SNR-dependent through the following modification

$$\hat{\Lambda}(\omega) = \Lambda(\omega)\, |W(t,\omega)| \tag{2.60}$$

where $W(t,\omega)$ is the estimate of the additive noise at the frame t. It was shown that after such noise-driven modification the optimal value for λ stabilised at 3.74 across different SNRs. Quality improvement in terms of the PESQ score was maximum in SNR range of 10-15 dB for white Gaussian noise (0.6 PESQ), 5-15 dB for $F16$ noise (0.55 PESQ) and about 20 dB for Babble noise (0.3 PESQ).

In [143] Loweimi et al similar to [139], studied the importance of the phase and magnitude spectra in speech enhancement through replacing the noisy phase and magnitude with their clean counterparts. They decomposed the signal into six frame lengths, from 32 ms to 1024 ms with 87.5% overlap. For computing the magnitude spectrum Hamming window was used in analysis while the phase spectrum was computed after using the Chebyshev window with dynamic range of 35 dB for analysis. It was shown that the quality improvement (in terms of PESQ) after replacing the phase and magnitude spectra with their clean counterparts depends on the frame length and the synthesis method as well as the SNR level. In case of replacing the magnitude spectrum with its clean version, the maximum improvement was achieved for 32 ms frame length (1.8, 2.0 and 2.1 PESQ scores for SNRs 15, 5 and 0 dB, respectively). With frame length extension, usefulness of the magnitude spectrum cleaning decreases and in the frame length of 1024 ms, replacing the noisy magnitude spectrum with the clean one results in quality improvement of upto 0.5 in PESQ scale. Also it was illustrated that using OLA or WOLA for synthesis almost has no effect on the quality of the enhanced signals.

On the other hand, after replacing the noisy phase spectrum with its clean version, maximum quality improvement was achieved in the frame length of 64 ms when the OLA was used for synthesis and in the frame length of 128 ms when synthesis was done by LSEE (WOLA). Maximum quality improvement for the former was 0.7 whereas for the later it was 1.1 in PESQ scale. At 32 ms frame length and in 5 dB SNR, replacing the noisy phase with its clean counterpart lead to 0.8 and 0.6 quality improvement in PESQ scale in case of using LSEE and OLA, respectively. They also studied the effect of gender. It was demonstrated that cleaning the magnitude spectrum was slightly more useful for the male speakers whereas replacing the noisy phase with its clean version resulted in a higher quality improvement for female speakers.

In MMSE speech enhancement technique, the magnitude estimator is phase blind and is merely a function of a priori and a posteriori SNRs. A recent approach in speech enhancement aims to employ the clean phase spectrum during estimating the magnitude spectrum and is called *phase-sensitive* or *phase-aware* speech enhancement [144, 145] in which the magnitude estimator becomes also a function of $cos(\underbrace{\phi_Y(\omega) - \phi_X(\omega)}_{\Delta\phi})$ where Y and X denote the noisy and clean signals, respectively. The gain function ($\frac{|X_{Enh}|}{|Y|}$) is proportional with

$cos(\Delta\phi)$ and also it can be shown that the $\Delta\phi$ is reciprocally proportional with SNR: the higher the SNR the closer the $\Delta\phi$ to zero.

The advantage of the phase-aware approach is as follows: assume that at a time-frequency bin a posteriori SNR ($\gamma = \frac{|Y|^2}{|W|^2}$) is high. This could be due to two reasons: either the signal is very strong or the signal is weak but the noise is too small. In the first case the gain function should not attenuate the noisy observation whereas in the second case more attenuation could be better. Note that even a priori SNR could not disambiguate this because in both cases it would be high, too. Extra phase information, however, can help in finding the true reason behind that: if the signal is strong the $\Delta\phi \to 0$ and in the second case $\Delta\phi > 0$.

It was shown that using clean phase spectrum (oracle experiments) improves the speech quality by 0.35 in PESQ scale in the SNR range of 0 to 15 dB. In the aforementioned work the enhanced signal was synthesised using the noisy phase spectrum and the clean phase was only used in estimating the magnitude spectrum. It was also shown that using the clean phase in both phase-aware magnitude estimation and speech synthesis improves the quality in PESQ scale by 0.5 in SNR range of 0 to 15 dB. In both experiments the frame length and overlap was set to 32 ms and 50%, respectively. In the case which the clean phase should be estimated blindly, for example using [146], it was shown that the PESQ score could be improved by 0.25 points for the voiced speech at 0 dB. For more details about the phase-aware approach to single channel speech enhancement and its variants please refer to [147].

2.4 Summary

In this chapter the basic aspects of phase-based signal processing was reviewed. The chapter began with the definition of the phase spectrum and the problems associated with using the phase spectrum in speech signal processing. The group delay as the major representation of the phase spectrum, was scrutinised in detail. The definition, physical interpretation, its properties, its relation to the magnitude spectrum, the spikiness issue and the proposed solutions in the literature were covered. The rest of the chapter was dedicated to reviewing the applications of the phase spectrum in speech processing. The goal was to shed light on where we stand in the phase-based speech processing. In particular the applications of the phase spectrum in speech analysis, speech coding, speech synthesis, speech recognition, speaker recognition, emotion recognition, spoofing and synthetic speech detection and speech enhancement were reviewed. In the next chapter, the phase information content is investigated.

Chapter 3

Phase Information

An investment in information[1] *pays the best interest.*
– Benjamin Franklin

The problems are solved, not by giving new information,
but by arranging what we have known since long.
– Ludwig Wittgenstein, *Philosophical Investigations*

3.1 Introduction

A signal is defined as a physical manifestation of information that changes with some independent variable [148] or simply a sequence of numbers which carries information. As such signal processing is about processing the information that resides within the signal. The phase spectrum is a sequence of numbers, too, and before taking the challenge of processing it, one needs incentive and the best one is to be sure that some information of interest is encoded in the phase. The goal of this chapter is to provide some motivation for the phase-based speech signal processing through shedding light on the the information content of the phase and magnitude spectra in a systematic way.

3.1.1 Information from Information Theory Perspective

Claude Shannon [149], the father of Information Theory, expressed the concept of information as a measurable commodity and defined it as the *expected surprise*. If an event is highly probable to happen, it is considered least informative as it comes with minimum surprise [150]. On the other hand, occurrence of an event which is least probable is highly

[1] Slightly misquoted: An investment in *knowledge* pays the best interest.

informative because it is accompanied with maximum surprise. Shannon defined the surprise as $-log(P_Z(z))$ where $P_Z(z)$ denotes the probability density function (pdf)[2] of event z. As such the expected surprise, which Shannon called it *entropy*, is defined as follows

$$H = -\sum_{z} P_Z(z)\, log_b(P_Z(z)) \tag{3.1}$$

where H denotes the entropy, b indicates the base of the logarithm and z is an event or a letter in an alphabet[3]. If b is set to two, H determines the minimum number of bits needed to encode the sequence or alphabet set z, without losing information in decoding stage [150]. The higher the entropy, the more information is carried by z and consequently the more bits required for lossless encoding.

So, if one wishes to evaluate the phase information from this standpoint, should estimate the pdf and compute the entropy. But is the expected surprise what we are looking for here? Based on this approach, pure noise has high entropy and consequently high information whereas in speech processing, that is not what is meant by information.

3.1.2 Information from Speech Processing Perspective

The entropy-based definition of information is helpful in data coding and transmission. However, when one talks about speech signal information, it does not mean the number of bits required for speech transmission. Speech signal information, generally speaking, means linguistic content, speaker-dependent attributes (speaker ID, gender, emotional state, accent/dialect, age and health) and the environment related information in the background.

If the speech signal is considered as a vector in the information space, such information categories are the unit vectors spanning this hypothetical space. However, such space is abstract and relates to the subjective impression the speech signal gives our brain. It is really difficult, if not impossible, to decompose this signal in such way, objectively. Our brain performs a miracle in information processing, can decompose a speech signal in the information space and extract all the aforementioned information from it with effortlessness. Such information decomposition allows for processing different streams of information independently and with high robustness. For example, separating the background noise and focusing on the lingual content leads to a high level of noise robustness. Our brain, finds the projection of the speech as a vector into each subjective subspace (speech information categories) and process/filter the information streams based on the intended goal. However,

[2] If z is a discrete random variable, instead of pdf, the term probability mass function (pmf) should be used.

[3] Alphabet is a set of all the possible events and the letter is either members.

designing algorithms and building machines with such capability is extremely challenging due to the involved abstraction.

Speech information content can be analysed from another angle with a lower abstraction level. This signal can be studied using the properties of the system which produces it, namely human speech production system. Based on the well-studied physiological and physical properties of such system, a speech signal is the result of a filtering process on the air flow produced by the lungs. This view leads to the source-filter model [96] for speech signals. Using this approach, speech is the convolution of two components: the source (excitation) and filter (vocal tract), both varying with time. Such low-level[4] objective presentation has ties with the previous high-level presentation of the speech information. For example, the lingual content is closely connected to the filter component and the speaker characteristic is related to both filter and the source parts.

Also it should be noted that the information content in the time and frequency domains are equal because the Fourier transform is reversible and does not lose information. On the other hand, the Fourier transform is completely characterised by its magnitude and phase in the polar coordinates or the real and imaginary parts in the Cartesian coordinates. As such the signal information equals the union (or loosely speaking sum) of the information encoded in the phase and magnitude spectra or the union of information of the real and imaginary parts. This poses the question that how the information is divided between them. For example, is the filter component captured by the magnitude and the source component encoded by the phase or vice versa? Do phase and magnitude spectra share any pieces of information? In general, how the information is allocated to different parts of the Fourier transform?

The goal of this chapter is to investigate how much and what kind of information resides in each part of the Fourier transform of the signal. To this end, the signal is reconstructed using *only* a partial Fourier transform information, $\mathscr{X}$, which can be phase, magnitude, etc. Then, the $\mathscr{X}$*-only* reconstructed speech signal is compared with the original one. The higher the similarity, the more informative the $\mathscr{X}$ is.

This Chapter is organised as follows. In Section 2, the signal information components are introduced and discussed. Section 3 is dedicated to explaining the iterative $\mathscr{X}$-only signal reconstruction process, the underlying theory and the influential factors. Section 4 includes the experimental results illustrating the importance of $\mathscr{X}$ through comparing the $\mathscr{X}$-only reconstructed signal with the original signal and discussion. Section 5 summarises this chapter.

[4]It is called *low-level* because it does not involve any abstraction.

3.2 Information Regions of the Mixed-phase Signals

Fourier Transform of signal $x[n]$, namely $X(\omega)$, is a complex quantity which can be decomposed into the real and imaginary parts in the Cartesian coordinates or can represented by the magnitude and phase spectra in the polar coordinates

$$X(\omega) = X_R(\omega) + jX_I(\omega) = |X(\omega)|\, exp(j\phi_X(\omega)) \tag{3.2}$$

where ω is radial frequency, subscripts R and I denote the real and imaginary parts, $|X|$ and ϕ_X indicate the magnitude and phase[5] spectra.

In addition to the Cartesian and polar representation, Fourier transform of the *mixed-phase* signals [36] can be decomposed as multiplication of the *minimum-phase* (MinPh) and *all-pass* (AllP) parts, too,

$$X(\omega) = X_{MinPh}(\omega)\, X_{AllP}(\omega) \tag{3.3}$$

where X_{MinPh} and X_{AllP} denote the Fourier transform of the minimum-phase and all-pass components, respectively.

There is an important difference between the signal decomposition using (3.2) and (3.3). In (3.2), there is an overlap between the information resides in each part. For a finite causal signal, there is an almost one-to-one relationship between the real and imaginary parts[6] [18]. Therefore, it is possible to recover one from another and then reconstruct the signal in the time domain. On the other hand, for the minimum-phase[7] signals, there is a one-to-one relationship between the magnitude and phase spectra and one can be recovered from the other one[8] [18].

So, for a large class of causal signals and a smaller class of the minimum-phase signals, there is a remarkable overlap between the information carried by each part of the Fourier transform because the missed part can be recovered from the available part. This means that the real and imaginary parts in the Cartesian decomposition and the magnitude and phase spectra in the polar coordinates are merely two different mathematical realisation of the same pieces of information. However, when a signal is decomposed into minimum-phase/all-pass

[5] As far as phase appears in the $j\, exp(\phi_X(\omega))$ context, it does not make any difference whether ϕ is the principle phase or the unwrapped phase.

[6] From the imaginary part, X_{Re} and $x[n]$ can be computed to within an additive constant error equals $x[0]$.

[7] This property holds for maximum-phase signals too, but due to practical issues such as stability the maximum-phase signals are not of much interest.

[8] For the minimum/maximum-phase signals, from the phase spectrum, the magnitude spectrum can be recovered upto a scale error. This error equals $\tilde{x}[0]$ where $\tilde{x}$ is the complex cepstrum.

components, there is no relationship between its minimum-phase and all-pass parts and they are independent.

From the information content point of view, these points can be expressed as follows

- Total Signal Information $= I_{MinPh} \cup I_{AllP} = I_{Magnitude} \cup I_{Phase} = I_{Real} \cup I_{Imginary}$
- Causal Signal Information $= I_{Real} = I_{Imaginary} \cup$ Shift Information
- MinPh Signal Information $= I_{Magnitude} = I_{Phase} \cup$ Scale Information
- $I_{MinPh} \cap I_{AllP} = \varnothing$
- $I_{Real} \cap I_{Imaginary} \neq \varnothing$
- $I_{Magnitude} \cap I_{Phase} \neq \varnothing$
- $I_{Real} \cap I_{Imginary} > I_{Magnitude} \cap I_{Phase}$

where the symbols $\cup$, $\cap$ and $\varnothing$ denote the union, intersection and empty set, respectively, and I_X indicates the information encoded in the $\mathscr{X}$ part of the Fourier transform. The shift information equals the $x[0]$ and the scale information equals $exp(\tilde{x}[0])$ where $\tilde{x}$ denotes the complex cepstrum of the signal $x[n]$ [18].

Figure 3.1 shows the information regions for a mixed-phase signal where the total information content of the signal is illustrated as the area of a rectangle. As can be seen, in the minimum-phase/all-pass decomposition, there is no overlap or intersection in the information regions covered by each part. However, in the Cartesian and polar decomposition there is some overlap (intersection) between the corresponding elements. Note that the overlap between the real and imaginary parts in terms of information they contain is larger than the overlap between the magnitude and phase spectra. This is due to the fact that the characteristic which ties the real and imaginary spectra together, namely causality, is a milder condition and easier to meet than the minimum-phase (or the maximum-phase) constraint needed for connecting the magnitude and phase together.

This concept can be also portrayed using a vector notation. If the signal is considered as a vector in the *information space*, either real and imaginary parts or the magnitude and phase spectra or the minimum-phase and all-pass components can be the basis vectors spanning the information space because the signal is uniquely expressible using its projection along each pair. In case of the real/imaginary parts or the magnitude/phase spectra, since there is some overlap between the information each part carries, the aforementioned basis vectors are not perpendicular. However, in case of the minimum-phase/all-pass elements because they are

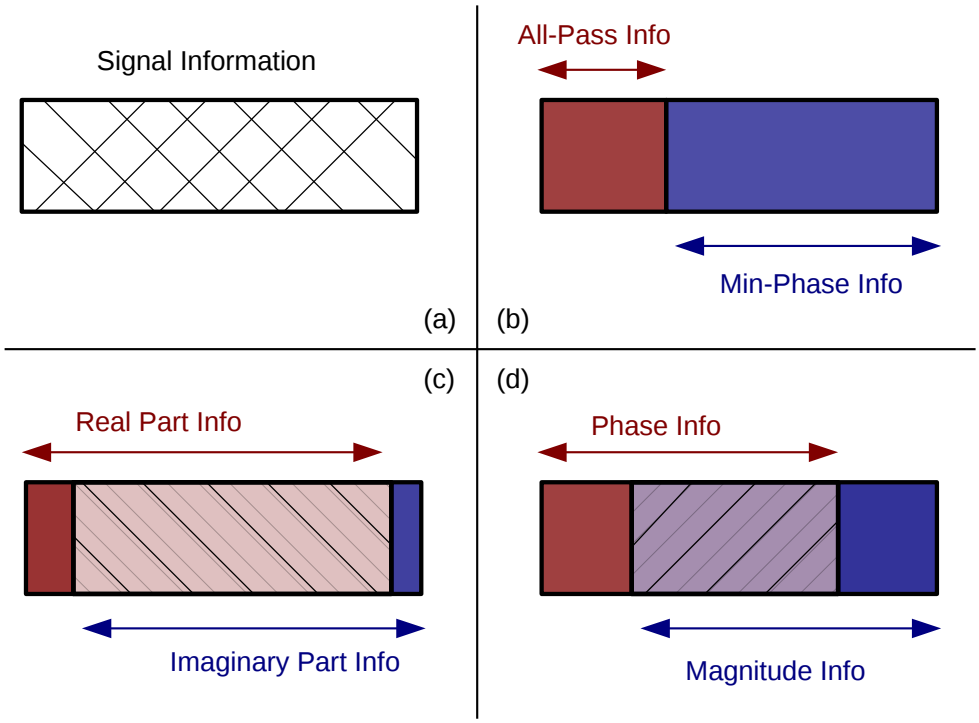

Fig. 3.1 Signal information regions for different signal decompositions, area of the rectangle indicates the total amount of information encoded in the signal. (a) total information, (b) information split after Minimum-phase/All-pass decomposition, (c) information split in the Cartesian coordinates between the real and imaginary parts, (d) information split after polar decomposition between the phase and magnitude spectra. Hatched area is proportional to the information shared by the two underlying components.

independent, the corresponding vectors are orthogonal[9]. As mentioned earlier, the overlap between the real and imaginary parts is larger than the magnitude and phase spectra. This results in higher correlation and consequently a smaller angle between the vectors represent them. Figure 3.2 illustrates these points.

Note that since there is intersection (overlap) between the information encoded in the real and imaginary parts or the phase and magnitude spectra, to be more precise, the total signal information should be rewritten as follows

$$\begin{aligned}\text{Total Signal Information} &= I_{MinPh} \cup I_{AllP} \\ &= I_{Real} \cup I_{Imginary} - (I_{Real} \cap I_{Imaginary}) \\ &= I_{Magnitude} \cup I_{Phase} - (I_{Magnitude} \cap I_{Phase})\end{aligned}$$

[9]Mathematically speaking, the complex logarithms of the minimum-phase and all-pass parts are orthogonal in the complex cepstrum domain. That is, the former shows up in the positive quefrencies and the latter appears only in the negative quefrencies; hence the inner product would be zero.

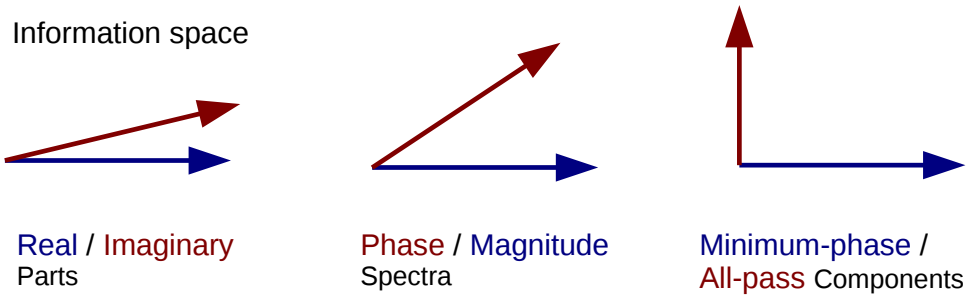

Fig. 3.2 Hypothetical information space spanned by the real/imaginary parts, phase/magnitude spectra and the minimum-phase/all-pass components as basis vectors. The closer the angle between the vectors to orthogonality, the lower the shared information.

3.2.1 Relation between the Elements of a Mixed-phase Signal

The Hilbert transform, under certain conditions, can mathematically underpin the relation between the real and imaginary parts and also the magnitude and phase spectra. The required condition is causality (or anti-causality). So, based on the Hilbert transform, causality in the time domain links the real and imaginary parts together. By the same token, the causality in the complex cepstrum domain connects the phase and (log of) magnitude spectra together.

The real and imaginary parts for a causal signal are recovered from each other as follows

$$X_{Re}(\omega) = x[0] + \frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} X_{Im}(\theta)\, cot(\frac{\omega-\theta}{2})\, d\theta = x[0] + \frac{1}{2\pi}X_{Im}(\omega) * cot(\frac{\omega}{2}) \quad (3.4)$$

$$X_{Im}(\omega) = -\frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} X_{Re}(\theta)\, cot(\frac{\omega-\theta}{2})\, d\theta = -\frac{1}{2\pi}X_{Re}(\omega) * cot(\frac{\omega}{2}) \quad (3.5)$$

where $\mathscr{P}$ denotes Cauchy principle value of the integral. Similarly, assuming the causality of the complex cepstrum, magnitude and phase spectra relate to each other as follows

$$arg\{X(\omega)\} = -\frac{1}{2\pi}\, log|X(\omega)| \,*\, cot(\frac{\omega}{2}) = -\frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} log|X(\theta)|\, cot(\frac{\omega-\theta}{2})\, d\theta \quad (3.6)$$

$$log|X(\omega)| = \tilde{x}[0] + \frac{1}{2\pi}arg\{X(\omega)\} * cot(\frac{\omega}{2}) = \tilde{x}[0] + \frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} arg\{X(\theta)\}\, cot(\frac{\omega-\theta}{2})\, d\theta \quad (3.7)$$

where $\tilde{x}[0]$ denotes the complex cepstrum at zero and equals

$$\tilde{x}[0] = \frac{1}{2\pi}\int_{-\pi}^{\pi} log|X(\omega)|\, d\omega. \quad (3.8)$$

For a detailed derivation of the above equations, please refer to Appendix A. The interpretation of such (almost) one-to-one relationship from information content point of view is

that both phase and magnitude spectra carry the same amount of information and are merely two mathematical realisation of the same information. If there is no one-to-one relationship, then at least a fraction of signal information is uniquely captured by each part. The question which arises at this point is that how the information is divided between the magnitude and phase spectra of a mixed-phase signal because in such case there is no one-to-one relation between these two spectra.

3.2.2 Minimum-phase/All-pass Components versus Magnitude and Phase Spectra

The magnitude and phase spectra can be expressed using the the minimum-phase and all-pass components as follows

$$X(\omega) = |X(\omega)|\, e^{j arg\{X(\omega)\}} = X_{MinPh}(\omega)\, X_{AllP}(\omega), \tag{3.9}$$

and given $|X_{AllP}(\omega)| = 1$,

$$\begin{aligned} |X(\omega)| &= |X_{MinPh}(\omega)| \\ arg\{X(\omega)\} &= arg\{X_{MinPh}(\omega)\} \; + \; arg\{X_{AllP}(\omega)\}. \end{aligned} \tag{3.10}$$

Note that unwrapped phase must be used in (3.10) because only the unwrapped phase values can be added together, not the principle wrapped phase values. Therefore, *arg* was used instead of ϕ.

From the magnitude spectrum point of view, the all-pass component equals unity. As such the magnitude spectrum does not see the information encoded in the all-pass part while the phase spectrum captures the information of the all-pass part entirely. On the other hand, the information of the minimum-phase part is shared by the phase and magnitude spectra

$$log|X_{MinPh}(\omega)| \quad \underset{\longleftarrow}{\text{Hilbert Transform}} \quad arg\{X_{MinPh}(\omega)\} \tag{3.11}$$

which means that phase and magnitude spectra of a minimum-phase signal carry the same amount of information. To be more precise, for a minimum-phase signal the (unwrapped) phase spectrum is uniquely recoverable from the magnitude spectrum whereas using the unwrapped phase spectrum, the magnitude spectrum is obtainable upto a scale error. So, the phase spectrum includes the all-pass information as well as the minimum-phase information excluding the scale. Here, it is referred to as minimum-phase-scale-excluded part, denoted by $MinPh^*$. The magnitude spectrum contains all the information lies in the minimum-phase

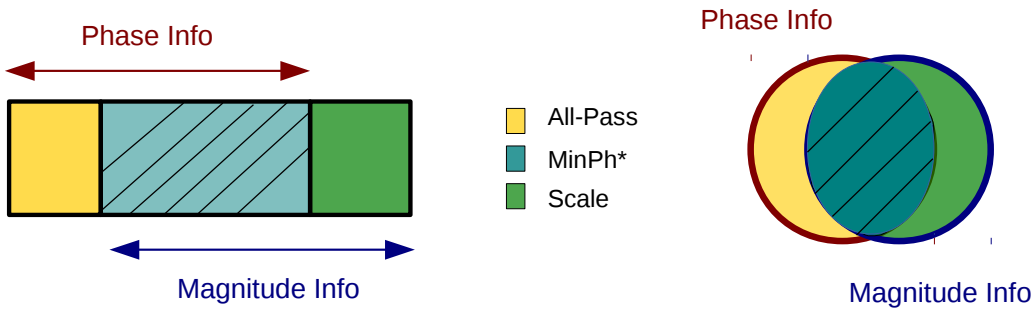

Fig. 3.3 Phase and magnitude information content based on the minimum-phase/all-pass decomposition using Venn diagram. MinPh* is shared between the magnitude and phase spectra, scale information is uniquely captured by the magnitude spectrum and the all-pass part resides in the phase spectrum.

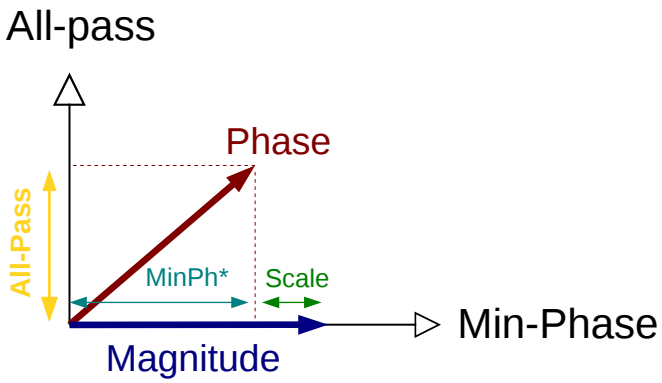

Fig. 3.4 Vector presentation of the phase and magnitude spectra in a hypothetical information space spanned by the minimum-phase and all-pass components. Phase spectrum has a projection onto both AllP and MinPh* axes whereas the magnitude spectrum only has a projection onto the MinPh part.

part. Information-wise,

$$\begin{aligned} I_{Magnitude} &= I_{MinPh^*} \cup I_{Scale} \\ I_{Phase} &= I_{MinPh^*} \cup I_{AllP}. \end{aligned} \tag{3.12}$$

Figure 3.3 depicts these points using a Venn diagram and Figure 3.4 illustrates them using vector notation.

At this point, the concept of signal's information regions/components, what is shared and what is uniquely captured by each spectrum is hopefully clear. The next step is to measure the area of each part which is proportional to the amount of information it carries and reflects its

relative importance. Actually, the information regions depiction in Figure 3.3 is qualitatively true, but quantitatively is rather sloppy. Now, the relative area (or importance) of each part and its relation to the whole signal information should be investigated.

3.3 Quantitative Evaluation of the Significance of the Signal Information Regions

Quantitative evaluation of the information and the corresponding area in Figure 3.3 is not straightforward. In this regard, one may resort to signal reconstruction from the part of interest, say $\mathscr{X}$, where $\mathscr{X}$ is partial information for example the magnitude spectrum, phase spectrum, minimum-phase component or the all-pass element. In the next step the $\mathscr{X}$-only reconstructed signal is compared with the original one. The higher the similarity (or the less the distance), the higher the information content of $\mathscr{X}$ and the larger the corresponding area in the information regions. To implement this approach, two issues should be addressed

1. How to reconstruct the signal only from partial information, $\mathscr{X}$?
2. How to measure the similarity of two speech signals reliably?

Signal reconstruction from the partial information resembles solving a set of m equations with n unknown variables where $m < n$, namely knowns are less than unknowns. As such it is not a well-posed[10] problem and does not have a unique answer, either many solutions or no solution. On the other hand, objective measurement of the the similarity/distance between two patterns in a reliable way, such that the results correlate acceptably with the subjective tests is not straightforward, too.

3.3.1 Iterative Signal Reconstruction

Since the problem of $\mathscr{X}$-only signal reconstruction is ill-posed, there are many possible solutions. To deal with such problems, the lack of enough knowns should be somehow compensated. If the number of knowns becomes enough, there is a unique answer and the unknown variables could be determined exactly. However, if the unknowns become more than knowns, the unknown can be expressed using intervals rather than a point in the space. So, one rather intuitive solution for dealing with ill-posed problems is to impose constraints on the possible solutions and the values which unknowns can take. Now the question would

[10] A mathematical problem is well-posed in the sense of Hadamard if (i) a solution exists, (ii) it is unique, and (iii) the solution depends continuously on the input data [151].

be what the constraints should be in order to achieve an acceptable answer. Prior knowledge such as the general properties of the data or the system which has generated it can be helpful.

3.3.2 Imposing Spectro-temporal Constraints

Human speech production system has some inertia and for normal speech it varies relatively smoothly over time. Also, due to non-stationarity, the speech signal is usually decomposed into overlapping frames which leads to increasing the correlation between the adjacent frames. As such when, after some spectral modifications, the frames are overlapped and added together for synthesis, they should not be that different at the overlap region. This constraint is imposed by adding the values in the overlap interval. So, in this case, the temporal contextual information put a constraint on the values which the samples can take and to some extend compensate for (known) unknowns[11].

There is also a constraint to be imposed in the spectral domain. The goal of the iterative $\mathscr{X}$-only signal reconstruction is to estimate the missed part of the Fourier transform subject to the constraint that the $\mathscr{X}$ part of the reconstructed signal over different iterations should remain equal to the given $\mathscr{X}$. For example, in the magnitude-only signal reconstruction, the magnitude of the reconstructed signal should be kept fixed over different iterations. However, after each iteration, the $\mathscr{X}$ part of the FT of the reconstructed signal changes. So, the spectral constraint would be substituting the $\mathscr{X}$ part at each iteration with the original one. In instance, for the magnitude-only signal reconstruction at the i^{th} iteration $|X^i(\omega)| \leftarrow |X(\omega)|$ or for phase-only reconstruction $\phi_{X^i}(\omega) \leftarrow \phi_X(\omega)$.

The length of the Fourier transform is important, too. If the number of samples of the signal (or frame) equals N, and the X is computed using FFT of length $N_{FFT} \geq 2N$ (zero-padding), the reconstruction will be more accurate [70]. Note that when the FFT length is $2N$, taking the inverse FFT provides $2N$ samples in the time domain. It is a priori known that the samples of the frame are N and the rest should be set to zero. This is another temporal constraint to be imposed. Increasing the FFT size imposes a heavier constraint as it involves setting more samples to zero. Intuitively, this should speed up the convergence at the cost of higher computation load in taking longer FFT. Figure 3.5 depicts this process.

Pseudo-code for $\mathscr{X}$-only signal reconstruction is as follows

FOR $i = 0$ to *number-of-iterations*

Frame blocking and windowing

[11] There are known knowns; there are things we know we know. We also know there are *known unknowns*; that is to say we know there are some things we do not know. But there are also unknown unknowns – the ones we don't know we don't know. And if one looks throughout the history of our country and other free countries, it is the latter category that tend to be the difficult ones. – Donald Rumsfeld

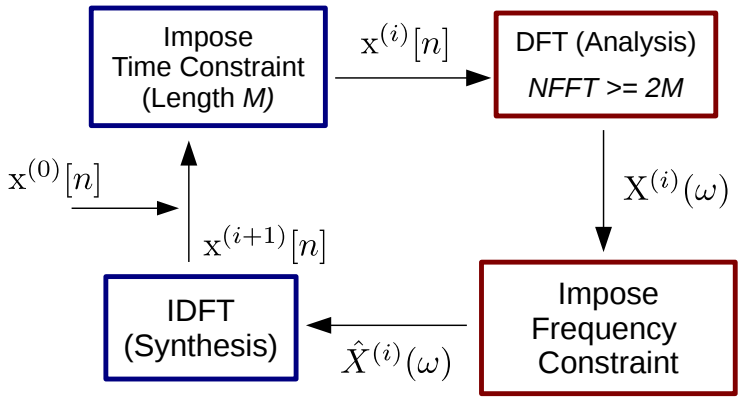

Fig. 3.5 Workflow of the iterative $\mathscr{X}$-only signal reconstruction [70]. $\mathscr{X}$ could be magnitude spectrum, phase spectrum, MinPh part or AllP part. The algorithm involves switching back and forth between the time and frequency domains.

FOR $n = 1$ to *number-of-frames*

Compute the (short-time) Fourier transform

IF $i == 0$

Preserve the $\mathscr{X}$ part and initialise the missing part

ELSE

Replace the $\mathscr{X}$ part with the original $\mathscr{X}$

END IF

Compute the Inverse Fourier transform

END FOR

Signal synthesis through OLA[12] or WOLA[13]

END FOR

This approach falls within the *Analysis-Modification-Synthesis* framework. Analysis and Synthesis stages clearly are related to the Fourier transform and the inverse Fourier transform in conjunction with the (weighted) overlap-add process and the Modification part relates to the step six to eight which are described in the pseudo-code. Figure 3.6 shows the

[12]Overlap-add [18]
[13]Weighted overlap-add [19]

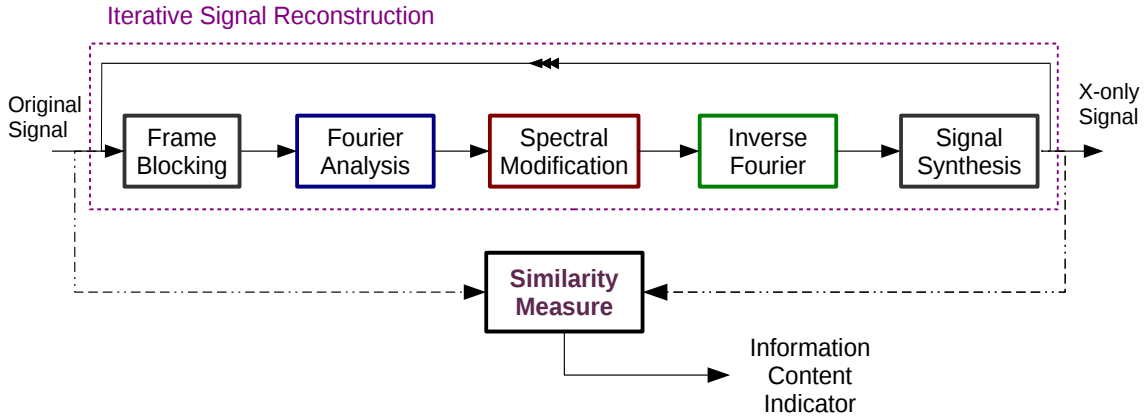

Fig. 3.6 Analysis-modification-synthesis (AMS) framework for measuring the information content of $\mathscr{X}$ part of the Fourier transform through iterative $\mathscr{X}$-only signal reconstruction. The similarity between the $\mathscr{X}$-only reconstructed signal and the original signal serves as a proxy for the information content of the $\mathscr{X}$ part. The $\mathscr{X}$ could be magnitude spectrum, phase spectrum, etc.

workflow of the $\mathscr{X}$-only signal reconstruction and the information content measurement of the reconstructed signal through comparing it with the original one.

3.3.3 Modification Step

The modification step is where the spectral constraint is imposed to ensure the $\mathscr{X}$ part of the Fourier transform remains fixed. In this section the modification step for the magnitude-only, phase-only, minimum-phase-only and all-pass-only signal reconstruction is quickly reviewed.

Phase-only Signal Reconstruction

For the phase-only signal reconstruction, the modification step takes the following form

$$\hat{X}^{(i)}(t,\omega) = |\hat{X}^{(i)}(t,\omega)|\, exp(j\, \hat{\phi}_X^{(i)}(t,\omega)\,) \leftarrow \begin{cases} |X^{(0)}(t,\omega)|\, exp(j\, \phi_X(t,\omega)\,) & i=0 \\ |X^{(i)}(t,\omega)|\, exp(j\, \phi_X(t,\omega)) & i>0 \end{cases} \tag{3.13}$$

where $X^{(i)}(t,\omega)$ and $\hat{X}^{(i)}(t,\omega)$ are the short-time Fourier transforms at the i^{th} iteration before and after performing the Modification, respectively, and $|X^0(t,\omega)|$ denotes the initial magnitude spectrum. In particular, at the i^{th} iteration of the phase-only signal reconstruction

the following constraint is imposed

$$\phi_X^{(i)}(t,\omega) \quad \leftarrow \quad \phi_X(t,\omega). \tag{3.14}$$

The algorithm aims to iteratively recover the missed complementary part which is the magnitude spectrum. The initial magnitude spectrum, $|X^{(0)}(t,\omega)|$, could be set to unity or a random sequence.

Magnitude-only Signal Reconstruction

The modification step in case of the magnitude-only signal reconstruction is as follows

$$\hat{X}^{(i)}(t,\omega) = |\hat{X}^{(i)}(t,\omega)|\, exp(j\, \hat{\phi}_X^{(i)}(t,\omega)\,) \leftarrow \begin{cases} |X(t,\omega)|\, exp(j\, \phi_X^{(0)}(t,\omega)\,) & i=0 \\ |X(t,\omega)|\, exp(j\, \phi_X^{(i)}(t,\omega)) & i>0 \end{cases} \tag{3.15}$$

with the spectral constraint

$$|X^{(i)}(t,\omega)| \quad \leftarrow \quad |X(t,\omega)|. \tag{3.16}$$

Phase spectrum could be initialised through zero or random values with uniform distribution in $(-\pi,\pi)$ or $(0,2\pi)$.

Minimum-phase-only Signal Reconstruction

For minimum-phase-only signal reconstruction the modification step runs as follows

$$\hat{X}^{(i)}(t,\omega) = \hat{X}_{MinPh}^{(i)}(t,\omega)\, \hat{X}_{AllP}^{(i)}(t,\omega) \leftarrow \begin{cases} \hat{X}_{MinPh}(t,\omega)\, \hat{X}_{AllP}^{(0)}(t,\omega) & i=0 \\ \hat{X}_{MinPh}(t,\omega)\, \hat{X}_{AllP}^{(i)}(t,\omega) & i>0 \end{cases} \tag{3.17}$$

with the spectral constraint

$$X_{MinPh}^{(i)}(t,\omega) \quad \leftarrow \quad X_{MinPh}(t,\omega). \tag{3.18}$$

The all-pass part is initialised with unity.

All-pass-only Signal Reconstruction

In case of the all-pass-only signal reconstruction the modification step takes the following form

$$\hat{X}^{(i)}(t,\omega) = \hat{X}^{(i)}_{MinPh}(t,\omega)\,\hat{X}^{(i)}_{AllP}(t,\omega) \leftarrow \begin{cases} \hat{X}^{(0)}_{AllPh}(t,\omega)\,\hat{X}_{AllP}(t,\omega) & i=0 \\ \hat{X}^{(i)}_{MinPh}(t,\omega)\,\hat{X}_{AllP}(t,\omega) & i>0 \end{cases} \tag{3.19}$$

with the spectral constraint

$$X^{(i)}_{AllP}(t,\omega) \quad \leftarrow \quad X_{AllP}(t,\omega). \tag{3.20}$$

The minimum-phase part is initialised with unity.

3.4 Experimental Results

The basics of iterative $\mathscr{X}$-only signal reconstruction was briefly reviewed and explained. In this section, the aforementioned four $\mathscr{X}$-only reconstruction scenarios are implemented and studied. The phase and magnitude-only signal reconstructions are already studied in the literature as mentioned in Section 2.3.2, however, the all-pass-only and minimum-phase-only signal reconstruction have not been investigated. In this section, the signal is reconstructed from either of the aforementioned parts and the relative importance of each part in short-, mid- and long-term analysis is investigated. In addition, the role and the extent to which iteration numbers and window shape affects the results is examined and discussed. First a single-frame $\mathscr{X}$-only signal reconstruction problem is studied. After that the problem of $\mathscr{X}$-only speech reconstruction is addressed.

3.4.1 Single-frame Signal Reconstruction

Figure 3.7 shows the $\mathscr{X}$-only reconstructed signals for different Fourier transform parts when the original signal consists of a single frame. Hence, the synthesis process just consists of taking an inverse Fourier transform and does not involve overlap-add step. The number of iterations is set to 1000 and the original signal is a frame of speech signal of length 25.6 ms belongs to *sp*07 from NOIZEUS database [22].

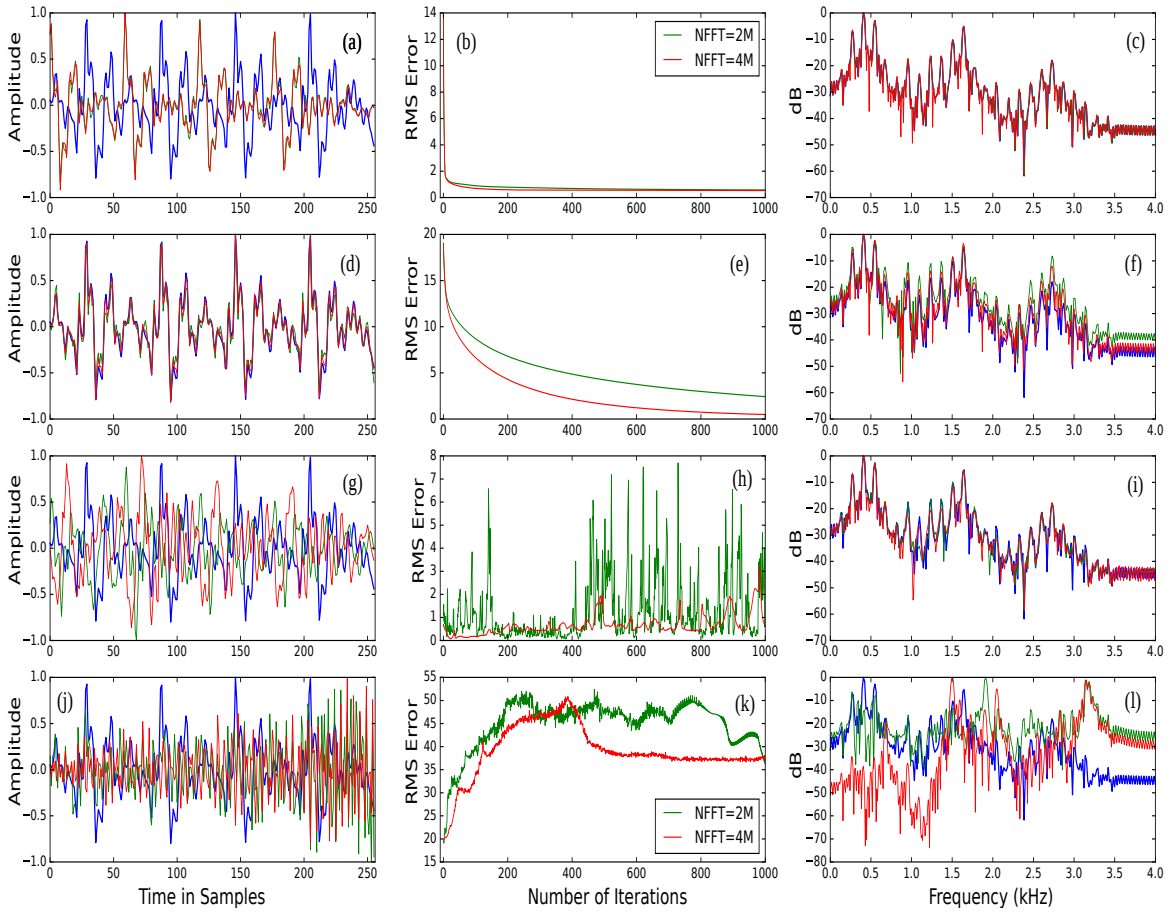

Fig. 3.7 $\mathscr{X}$-only (single-frame) signal reconstruction after 1000 iterations. (a) magnitude-only reconstructed signal, (b) RMS error of the magnitude-only reconstructed signal versus iteration number for two FFT sizes: $2M$ and $4M$ where M denotes number of frame samples, (c) original magnitude spectrum versus the magnitude spectrum of the magnitude-only reconstructed signal, (d) phase-only reconstructed signal, (e) RMS error of the phase-only reconstructed signal versus iteration number for two FFT sizes: $2M$ and $4M$, (f) original magnitude spectrum versus the magnitude spectrum of the phase-only reconstructed signal, (g) minimum-phase-only reconstructed signal, (h) RMS error of the minimum-phase-only reconstructed signal versus iteration number for two FFT sizes: $2M$ and $4M$, (i) original magnitude spectrum versus magnitude spectrum of the minimum-phase-only reconstructed signal, (j) all-pass-only reconstructed signal, (k) RMS error of the all-pass-only reconstructed signal versus iteration number for two FFT sizes: $2M$ and $4M$, (l) original magnitude spectrum versus magnitude spectrum of the all-pass-only reconstructed signal. In case of the magnitude and phase-only signal reconstruction the algorithm converges whereas for the minimum-phase-only and all-pass-only reconstruction it does not converge.

Magnitude-only

In case of magnitude-only signal reconstruction, convergence happens early and as it appears, the iteration is not that much useful. It can be justified based on what the magnitude spectrum

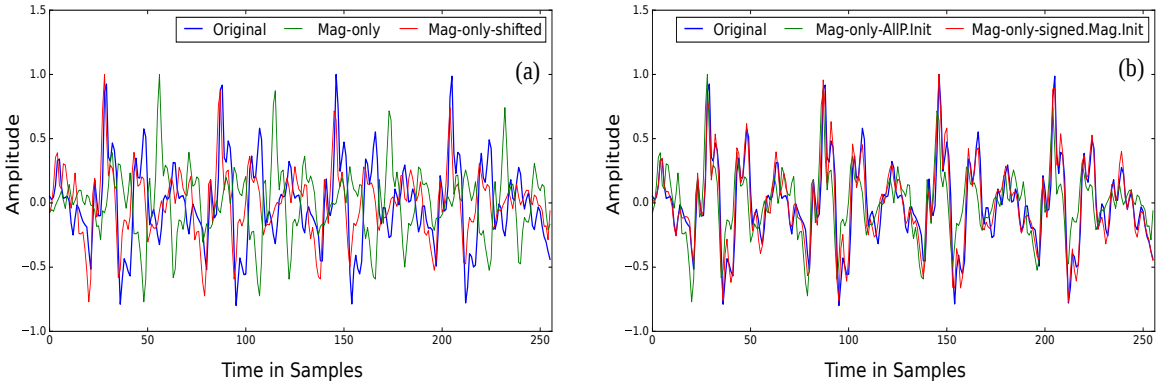

Fig. 3.8 Effect of initialisation with the all-pass component and the signed magnitude spectrum on the magnitude-only reconstructed signal. (a) phase spectrum is initialised by zero, (b) all-pass component (green curve) and the signed magnitude spectrum (red curve) are used for initialisation. Using the signed magnitude spectrum or the all-pass component in initialisation step lead to near-perfect magnitude-only signal reconstruction.

misses. The magnitude spectrum includes the minimum-phase component but lacks the all-pass part. As discussed earlier, the minimum-phase and all-pass parts are orthogonal in the information space and has no link together. As such the missed information in case of the magnitude-only reconstruction cannot be recovered by iterating. Note that since only a single frame is given, no contextual information is available.

Another observation is that, in the example provided here, the magnitude-only signal, to some extent, resembles the time-shifted version of the signal. Therefore, after optimal shift in time, it approximately matches the original signal, as shown in Figure 3.8(a). In Figure 3.8(b) the effect of initialising the phase spectrum with the phase of the all-pass component and also with the signed magnitude spectrum is depicted. As can be seen, both all-pass component and the signed version of the magnitude spectrum can lead to a higher quality magnitude-only signal reconstruction.

Phase-only

As shown in Figure 3.7(d), (e) and (f), through phase-only signal reconstruction using enough iterations, the magnitude spectrum and consequently the original signal can be recovered within a scale error. In addition, employing a larger FFT length leads to a faster convergence rate. Recall that based on the Theorem 1 in Section 2.3.2, a finite length signal is recoverable from its phase spectrum up to a scale error. Here we have normalised the scale of original and the phase-only reconstructed signal to 1. However, as seen although the phase-only reconstructed signal highly resembles the original one, it does not match it perfectly.

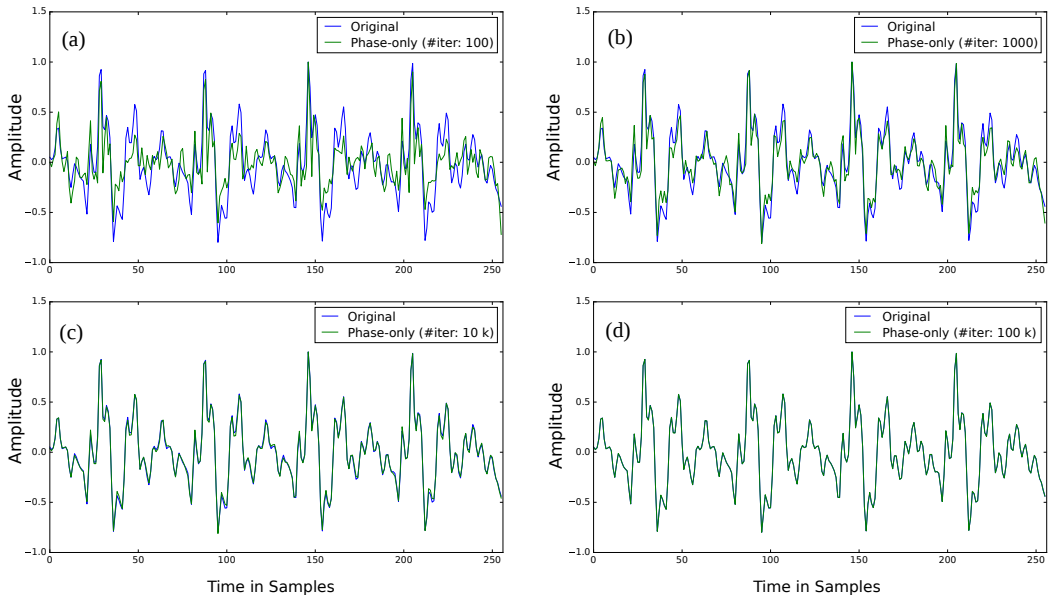

Fig. 3.9 Phase-only reconstructed signal for different iteration numbers (#iter) versus the original signal. As mentioned in Theorem 1 in Section 2.3.2, the signal is recoverable from its phase spectrum up to a scale error, however, the number of required iterations is unknown and should be determined empirically. Number of iterations: (a) 100, (b) 1000, (c) 10000, (d) 100000.

Note that the theorem does not mention the required number of iterations. In fact, in limit or after performing enough iterations, the signal is recoverable from its phase spectrum upto a scale error. The enough number of iteration, among others factors, depends to the complexity of the signal. Figure 3.9 shows the phase-only reconstructed signal after 100, 1000, 10000 and 100000 iterations. As seen, after enough iterations the phase-only reconstructed signal matches the original one (within a scale error).

Minimum-phase-only

Minimum-phase-only signal reconstruction is innately problematic because the missing part, namely all-pass component, is orthogonal to it. As shown in Figure 3.2 they have no projection on each other. Therefore, any knowledge about either of them does not put any constraint on the possible options for the other one. This argument holds for the all-pass-only signal reconstruction, too. As such the iteration can not help in recovering the missing part and consequently the signal. That is why the error change pattern versus iteration is chaotic and does not converge. In addition, the length of the FFT also does not help in reaching convergence or getting better results. As Figure 3.7(g)-(i) show, although the minimum-

phase-only reconstructed signals has the same magnitude spectra, their behaviour in the time domain is different.

Note that the magnitude spectrum was only linked to the minimum-phase information, too, but the magnitude-only signal reconstruction was more successful and converged. The reason backs to the missed complementary part. In case of the magnitude-only signal reconstruction, the algorithm searches for the missing part, namely the phase spectrum which is not independent of the magnitude (Figure 3.2). In terms of information space, the vectors corresponds to the phase and magnitude spectra are not orthogonal and share a remarkable fraction of signal information. Determining the magnitude spectrum, although in general does not lead to a unique phase spectrum, limits the possible options for the phase. However, putting constraint on the minimum-phase part does not restrain the corresponding all-pass part and vice versa. This leaves no room for recovering the missed part.

All-pass-only

Similar to the minimum-phase-only signal reconstruction, the known knowledge does not constrain the unknown. So, from the perspective of solving an ill-posed problem, the strategy of imposing constraints does not work and the iterative algorithm does not converge. Figure 3.7(j)-(l) demonstrate the all-pass-only reconstructed signal. Notice that (after synthesis) the magnitude of the all-pass-only reconstructed signal is not flat, although does not match the original magnitude spectrum.

For the magnitude-only signal reconstruction, adding the all-pass information to the algorithm lead to some temporal re-organisation of the algorithm. This hints at a link between the all-pass part and time-related information. In the next section where the signal consists of many frames, the information content of the all-pass part will be better clarified.

3.4.2 Speech Signal Reconstruction

In this part, the signal to be $\mathscr{X}$-only reconstructed, consists of overlapping frames. So, the synthesis stage in the AMS framework consists of the inverse Fourier transform and the overlap-add. Overlapping and adding the frames imposes contextual information and constraints on the frames in the synthesis stage. In order to simultaneously study the spectro-temporal properties of information encoded in each part of the Fourier transform, the signal was analysed in short- (32 ms), mid- (128 ms) and long-term (512 ms). The frame overlap and number of iterations were set to 75% and 100, respectively.

Table 3.1 *Average PESQ score of the magnitude-only reconstructed speech of the NOIZEUS database [22] for different window shapes and frame lengths. Phase spectrum was initialised with zero, frames overlap was set to 75%, and 100 iterations were applied. The z in Cheb-z denotes the dynamic range of the Chebyshev window in dB.*

Frame length (ms)	32	64	128	256	512	1024
Rectangular	4.00	3.91	3.61	3.05	2.36	1.90
Triangular	4.18	3.99	3.54	2.73	1.95	1.18
Hanning	3.34	3.37	3.19	3.18	0.81	0.50
Hamming	4.25	4.09	3.83	3.28	2.66	1.43
Cheb-20	2.42	2.44	2.29	2.18	1.91	1.95
Cheb-25	2.54	2.54	2.40	2.29	2.10	1.88
Cheb-30	2.86	2.65	2.53	2.43	2.27	1.77
Cheb-35	3.40	2.90	2.61	2.57	2.45	1.87
Cheb-40	3.78	3.36	2.88	2.72	2.37	1.53
Cheb-45	4.08	3.60	3.30	2.93	2.58	1.73
Cheb-50	4.18	3.86	3.50	3.09	2.63	1.48
Cheb-80	4.19	4.10	3.95	3.53	1.64	0.77
Cheb-110	3.61	3.54	3.44	3.13	0.75	0.53

Magnitude-only

Figure 3.9 portrays the spectrogram of the magnitude-only reconstructed signal in case of short-, mid- and long-term analysis. In short-term analysis, the magnitude-only reconstructed speech highly resembles the original speech signal. This also has been verified perceptually for example using PESQ measure as shown in Table 3.1, taken from [71]. However, by frame length extension, the similarity between the magnitude-only reconstructed and the original signal decreases. In particular, in long-term analysis, the spectrogram of the magnitude-only reconstructed signal seems to have lost all the timing information and the events no longer can be localised on the time axis. If the underlying signal generation process does not change by time, the events do not vary in time, so temporal localisation becomes irrelevant. Variability of the signal generation process leads to the non-stationarity and that is why the speech signal is analysed in short-term. So, the shortcoming of the magnitude spectrum in mid and long-term processing backs to the speech non-stationarity. Regarding the window shape, Table 3.1 shows that Hamming window is an optimal choice for working with the magnitude spectrum and it leads to the highest quality in the PESQ scale. In the next section, the effect of window is investigated in more details.

It should be noted that the magnitude spectrum lacks the all-pass part. Divergence between the original signal and the magnitude-only reconstructed signal means the fraction of information which it misses, namely the all-pass part, becomes more important. So, frame

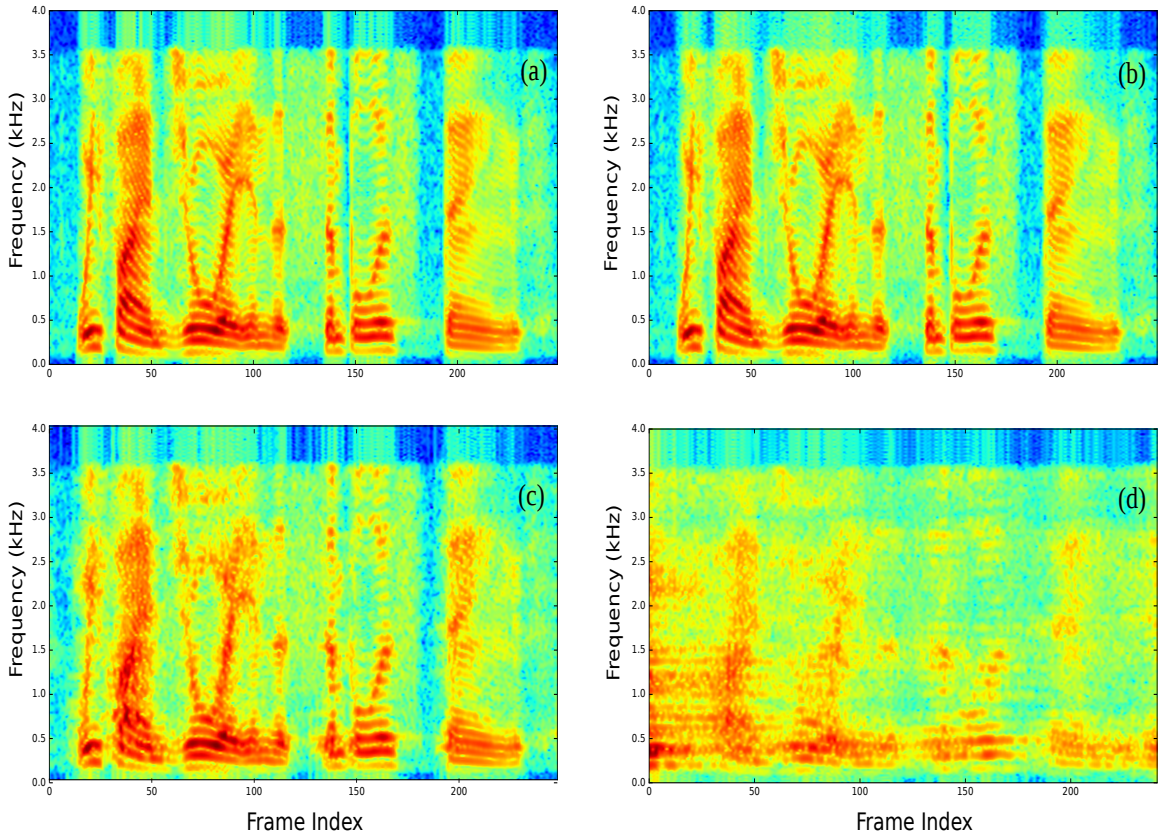

Fig. 3.10 Magnitude-only signal reconstruction after short-term, mid-term and long-term Fourier analysis. The phase spectrum was initialised with zero, a Hamming window was applied for analysis/synthesis and 100 iterations were used. Frames overlap was set to 75%. (a) Original signal, (b) short-term analysis, frame length: 32 ms, (c) mid-term analysis, frame length: 128 ms, (d) long-term analysis, frame length: 512 ms.

length extension apparently leads to more importance for the all-pass part. Increasing the frame length, although improves the frequency resolution, reduces the temporal resolution. This gives more importance to the part of signal information which is related to the timing of the events. This indicates that the all-pass part includes the timing information. In short-term analysis, the timing information is less important as the temporal resolution is already high. However, by frame length extension, the timing information lying in the all-pass part becomes more influential.

To test this hypothesis, one can initialise the phase spectrum in the magnitude-only signal reconstruction with the phase of the all-pass part, instead of random or zero-phase. As depicted in Figure 3.11, by employing the all-pass information, no matter how long the frame is, the signal gets reconstructed near-perfectly. This corroborates the point that the all-pass component includes timing information. All-pass-only signal reconstruction could

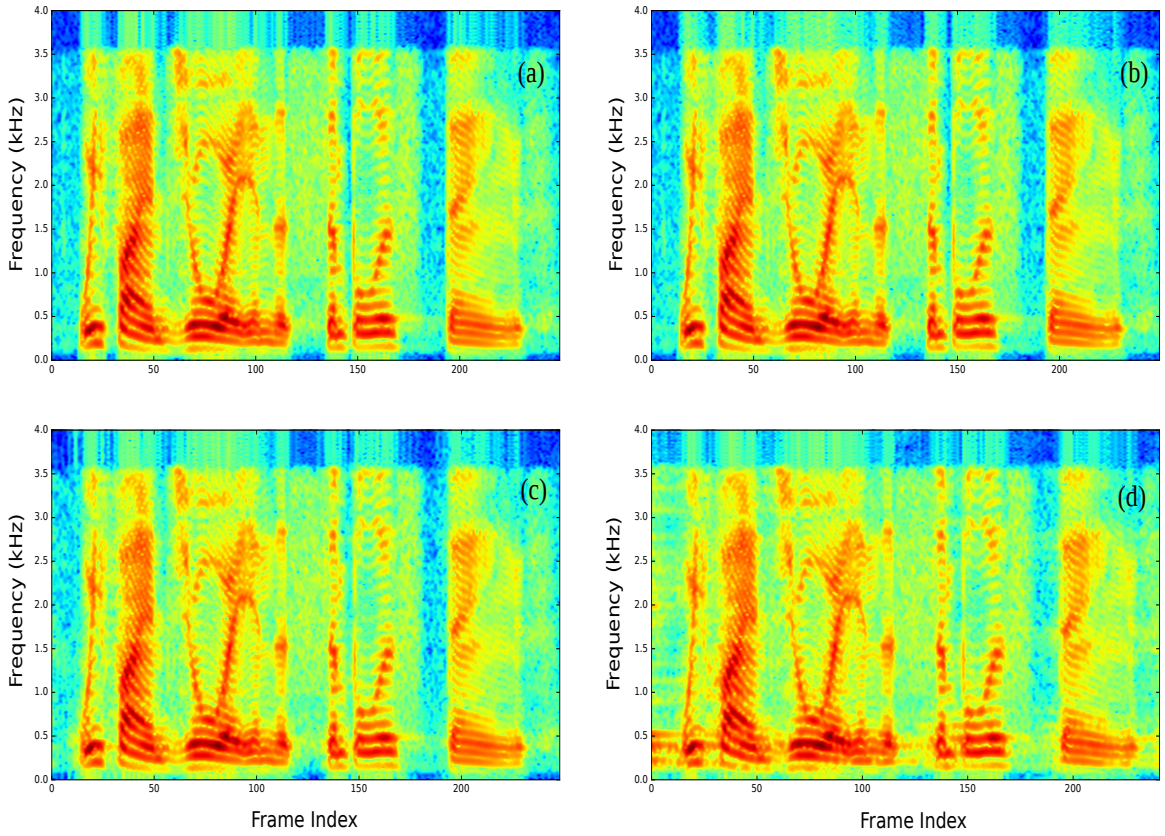

Fig. 3.11 Magnitude-only signal reconstruction after short-term, mid-term and long-term Fourier analysis. The phase spectrum was initialised with the phase spectrum of the all-pass component, a Hamming window was applied for analysis/synthesis and 100 iterations were used. Frames overlap was set to 75%. (a) Original signal, (b) short-term analysis, frame length: 32 ms, (c) mid-term analysis, frame length: 128 ms, (d) long-term analysis, frame length: 512 ms.

shed further light on this point. Lastly, initialisation with signed magnitude spectrum can also provide the algorithm with the all-pass information and as Figure 3.12 demonstrates, it leads to a near-perfect magnitude-only signal reconstruction, too.

Phase-only

Table 3.2 shows the PESQ quality values for the phase-only reconstructed signal versus frame length and for different window shapes [71]. The table shows rectangular and Chebyshev (25-35 dB) windows are optimal options for reconstructing the signal from the phase spectrum, leading to high quality. Figure 3.13 illustrates the spectrogram of the phase-only reconstructed signal using 100 iterations after short-, mid- and long-term analysis using rectangular window and Figure 3.14 portrays the effect of applying the Chebyshev (25 dB) window in the same

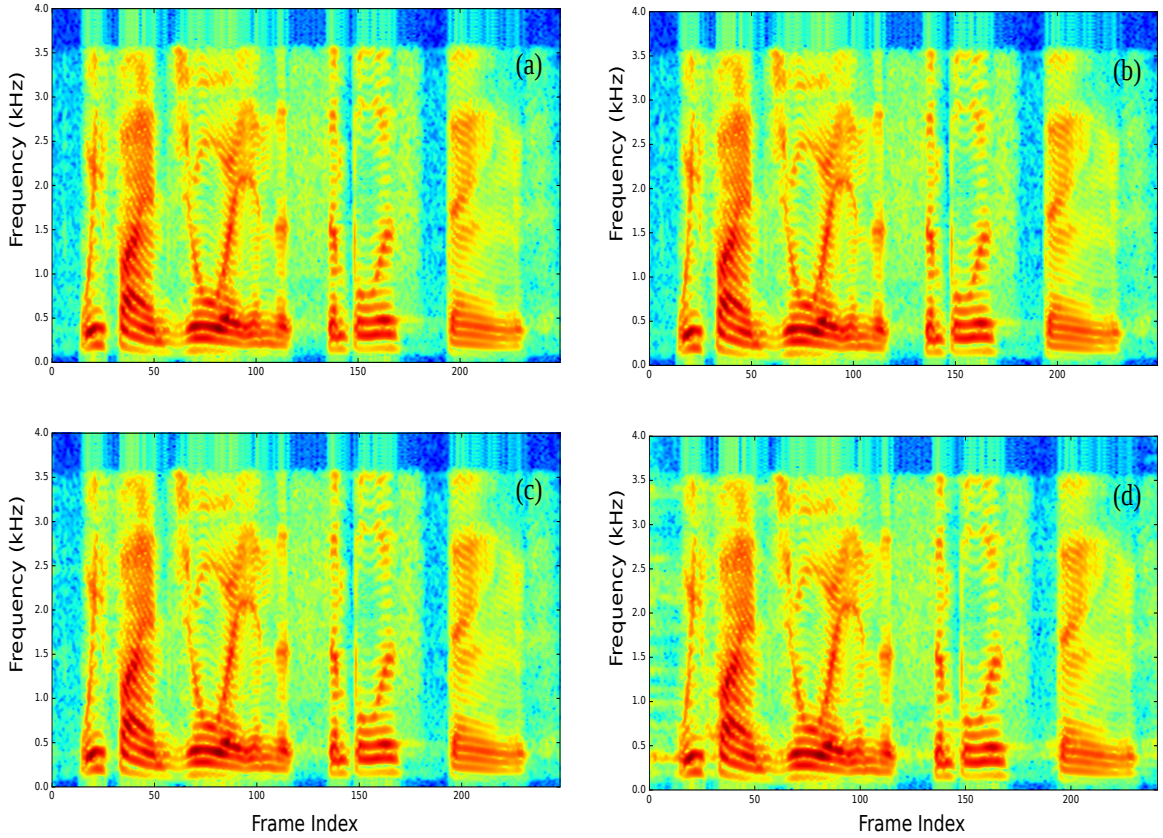

Fig. 3.12 Magnitude-only signal reconstruction after short-term, mid-term and long-term Fourier analysis. The initialisation was done with the signed magnitude spectrum, a Hamming window was applied for analysis/synthesis and 100 iterations were used. Frames overlap was set to 75%. (a) Original signal, (b) short-term analysis, frame length: 32 ms, (c) mid-term analysis, frame length: 128 ms, (d) long-term analysis, frame length: 512 ms.

conditions. The spectrograms are in agreement with the high PESQ scores. Windows such as Hanning and Chebyshev with high dynamic range are inappropriate options for working with phase. Effect of the window will be discussed in the next section.

The PESQ score and the spectrograms show that by frame length expansion, the quality of the phase-only reconstructed speech improves. As mentioned in Section 2.3.2, this trend was observed in many researches such as [69, 6, 7, 79]. The reason for it was explained in [71].

The reason can be explained using the concept of the information regions. The phase spectrum includes all the signal information except for the scale. In the extreme case where the whole signal is analysed without framing, the scale information has the minimum perceptual role to play and turns into merely intensity. However, on the other extreme, where the signal is decomposed into frames as long as one sample, the scale information

Table 3.2 *Average PESQ score of the phase-only reconstructed speech of the NOIZEUS database [22] for different window shapes and frame lengths. Magnitude spectrum was initialised with unity, frames overlap was set to 75%, and 100 iterations were applied. The z in Cheb-z denotes the dynamic range of the Chebyshev window in dB.*

Frame length (ms)	32	64	128	256	512	1024
Rectangular	3.51	3.55	3.66	3.78	3.83	3.59
Triangular	1.85	1.96	2.03	2.33	2.58	2.53
Hanning	1.37	2.25	1.92	2.48	0.89	0.77
Hamming	2.41	2.47	2.58	2.80	2.99	2.88
Cheb-20	3.78	3.98	4.04	4.20	4.20	4.04
Cheb-25	3.78	3.98	4.09	4.24	4.17	4.21
Cheb-30	3.73	3.96	4.05	4.20	4.16	4.24
Cheb-35	3.63	3.88	3.97	4.12	4.05	4.08
Cheb-40	3.36	3.65	3.81	3.92	3.86	3.80
Cheb-45	2.92	3.26	3.54	3.64	3.60	3.44
Cheb-50	2.57	2.83	3.14	3.32	3.33	3.08
Cheb-80	1.88	1.85	2.02	2.36	2.72	2.63
Cheb-110	1.88	2.10	2.29	2.53	2.31	1.54

become very important. So, intuitively by frame length extension the importance of the scale information decreases. As a result, what the phase spectrum lacks, becomes less influential by frame length extension.

As far as a single frame is concerned, the signal can be recovered from the phase within a scale error. Therefore, intra-frame-wise the phase can capture all the signal variabilities (Figure 3.9). Assume all the frames are phase-only reconstructed and match the original frame perfectly, within merely a scale error. For synthesis, the frames should be brought together, overlapped and added. However, each one suffers from a different scale error in comparison with the exact signal frame. When the overlapping frames are added together, since in general the scales do not match with each other, the stronger frames overshadow the weaker ones in the neighbourhood. For perfect signal recovery, as well as capturing the intra-frame variability of the signal (which phase can afford), the scale should be in harmony with the neighbouring frames, too. The error due to the inconsistency of the scales of the adjacent phase-only reconstructed frames is called *scale incompatibility error* (SIE) in [71]. It is computed as follows

$$SIE = \frac{1}{T}\sum_{t=1}^{T}\left(1 - exp(\tilde{x}[t,0])\right) \tag{3.21}$$

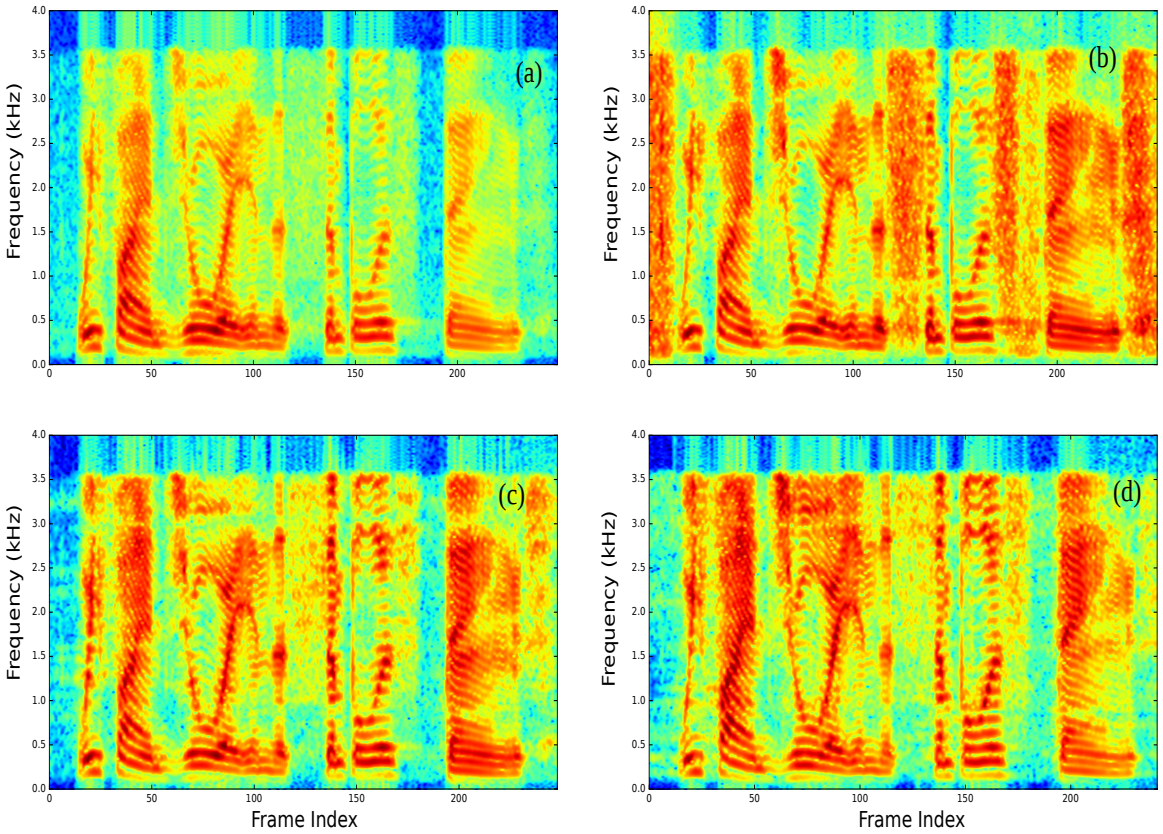

Fig. 3.13 Phase-only signal reconstruction after short-term, mid-term and long-term Fourier analysis. The magnitude spectrum was initialised with unity, frames overlap was set to 75%, a Rectangular window was used for analysis/synthesis and 100 iterations were applied. (a) Original signal, (b) short-term analysis, frame length: 32 ms, (c) mid-term analysis, frame length: 128 ms, (d) long-term analysis, frame length: 512 ms.

where 1 relates to initialising the magnitude spectrum with unity and the $exp(\tilde{x}[t,0]$ is the correct value for the scale of the frame t computed using the Hilbert transform relations, (3.7). As Figure 3.15 shows, this error decreases by frame length extension which agrees with the aforementioned discussion regarding the scale information importance. By initialising the magnitude spectrum with a constant equals the frame scale information, namely $exp(\tilde{x}[t,0])$, near-perfect phase-only signal reconstruction can be achieved in all the frame lengths (short-, mid- and long-term), as demonstrated in Figure 3.16.

Note that in case of the phase-only reconstructed signal, regardless of the frame length and availability of the scale information, the timing of the events is exact. This is due the connection of the phase spectrum to the all-pass component. As mentioned earlier the timing information of the signal closely linked to the all-pass part. In addition, as the spectrograms show, irrespective of the availability of the scale information, the source and filter components

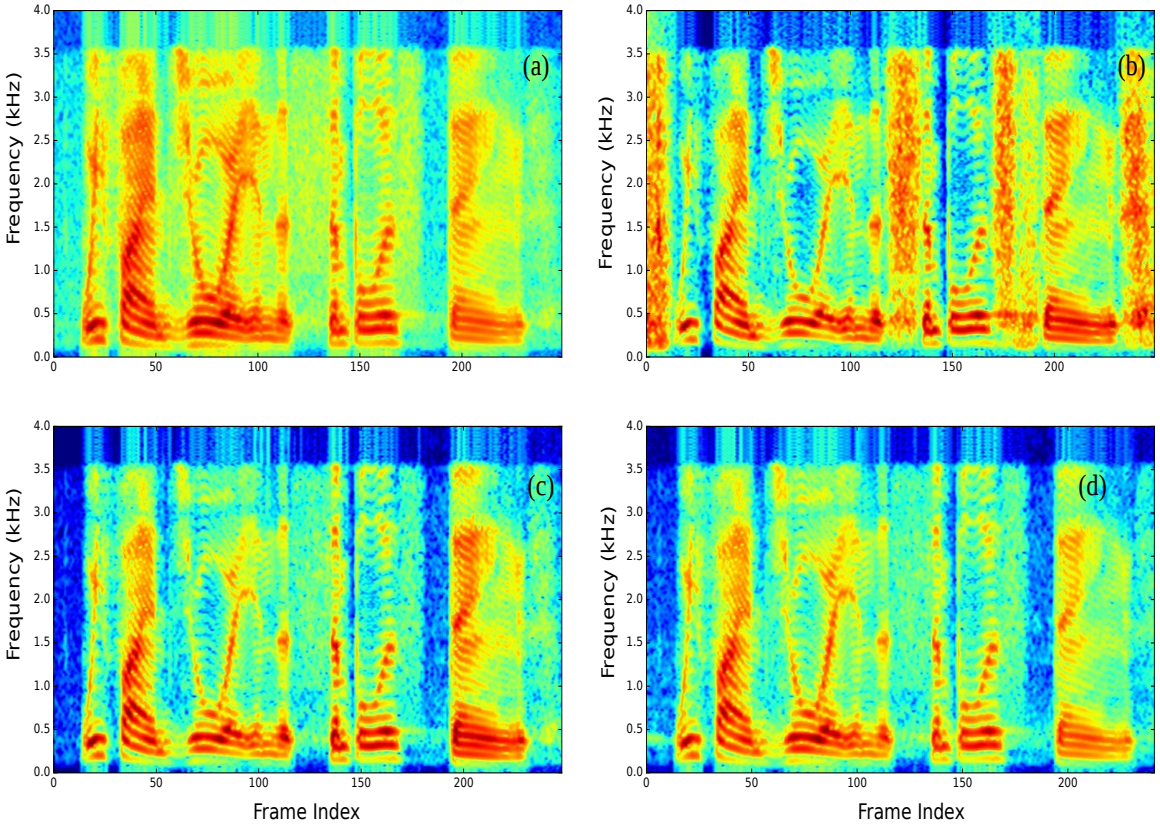

Fig. 3.14 Phase-only signal reconstruction after short-term, mid-term and long-term Fourier analysis. The magnitude spectrum was initialised with unity, frames overlap was set to 75%, a Chebyshev (25 dB) window was used for analysis/synthesis and 100 iterations were applied. (a) Original signal, (b) short-term analysis, frame length: 32 ms, (c) mid-term analysis, frame length: 128 ms, (d) long-term analysis, frame length: 512 ms.

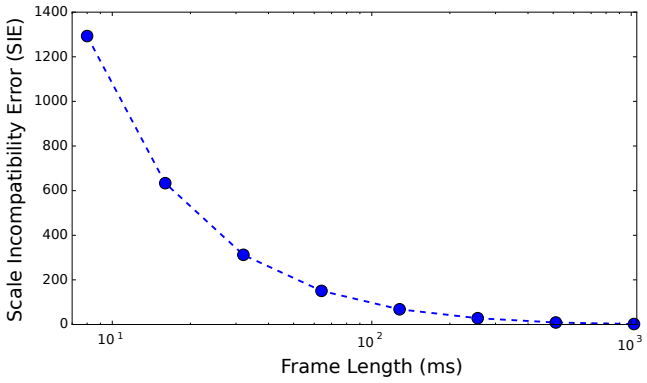

Fig. 3.15 Scale incompatibility error (SIE) in the phase-only signal reconstruction. SIE decreases by frame length extension.

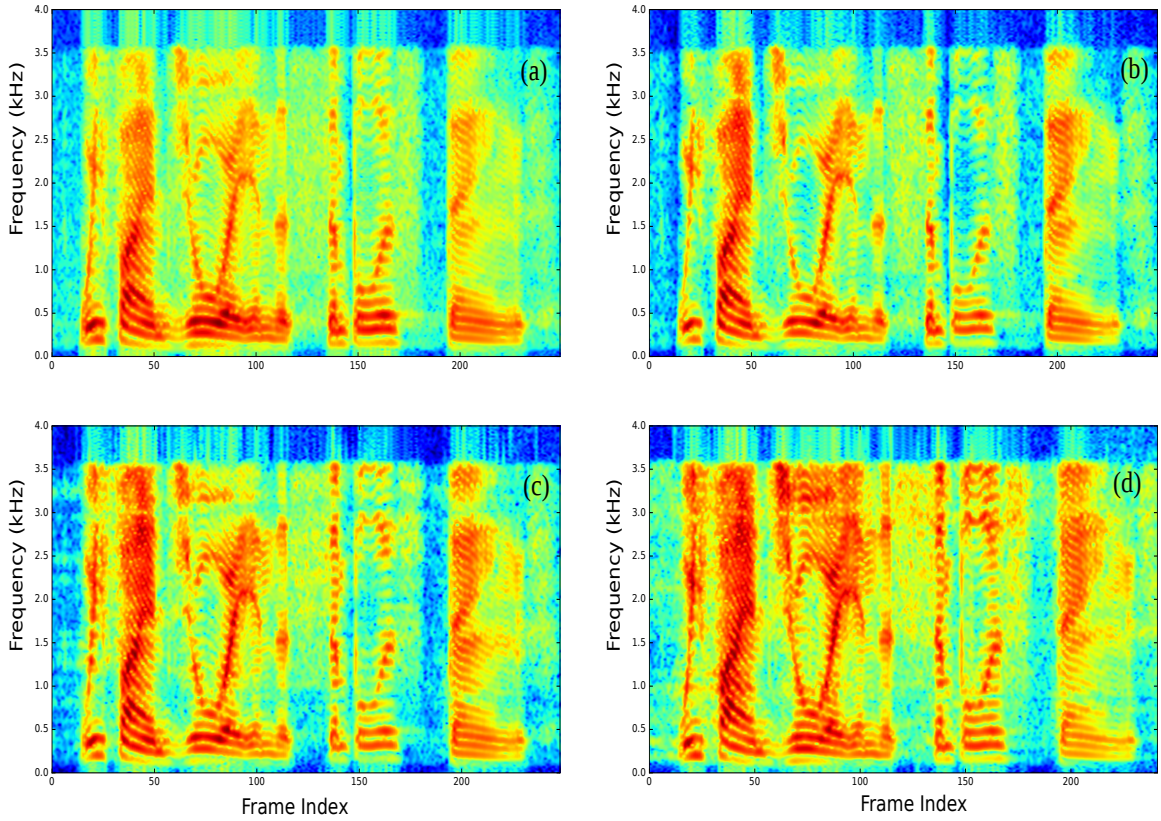

Fig. 3.16 Phase-only signal reconstruction after short-term, mid-term and long-term Fourier analysis. The magnitude spectrum at frame t was initialised with $exp(\tilde{x}[t,0])$, frames overlap was set to 75%, a Chebyshev (25 dB) window was used for analysis/synthesis and 100 iterations were applied. (a) short-term analysis, frame length: 32 ms, (b) mid-term analysis, frame length: 128 ms, (c) long-term analysis, frame length: 512 ms. Distance in mean square error for frame length (d) 32 ms, (e) 128 ms, (f) 512 ms.

of the speech signal exist in the phase-only reconstructed signal which means that phase spectrum includes such elemental information.

Minimum-phase-only

Figure 3.16 shows the spectrograms of the minimum-phase-only reconstructed signal in short-, mid- and long-term analysis. As explained in case of the single-frame signal reconstruction, recovering the all-pass component given the minimum-phase part is problematic and iteration is not beneficial. That is why the error pattern in case of the minimum-phase-only and all-pass only signal reconstruction was random and did not converge. However, compared with the single-frame signal reconstruction, there is an advantage related to the contextual information due to having overlapping frames. In case of the minimum-phase-only signal

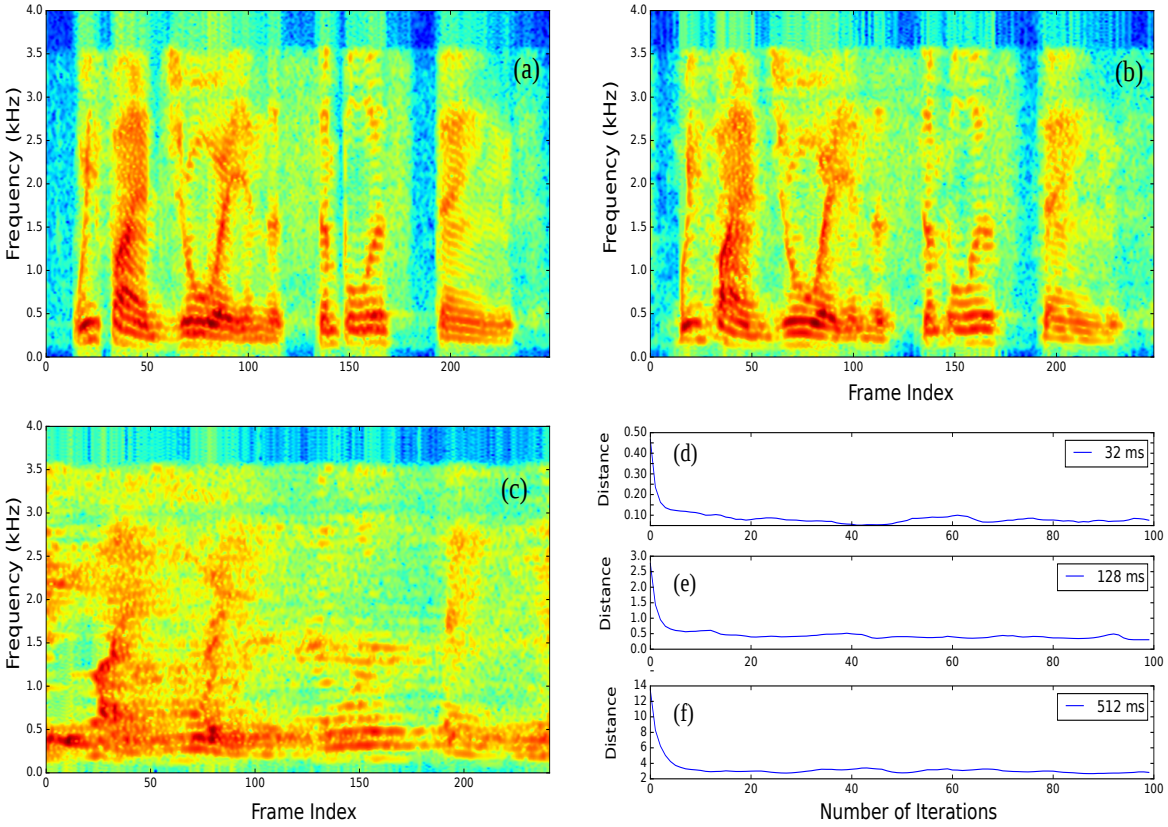

Fig. 3.17 Minimum-phase-only signal reconstruction after short-term, mid-term and long-term Fourier analysis. The all-pass part was initialised with unity, frames overlap was set to 75%, a Hamming window was used for analysis/synthesis and 100 iterations were applied. (a) short-term analysis, frame length: 32 ms, (b) mid-term analysis, frame length: 128 ms, (c) long-term analysis, frame length: 512 ms. Distance in mean square error for frame length (d) 32 ms, (e) 128 ms, (f) 512 ms.

reconstruction, the timing information which is encoded in the all-pass part is missed. When the frames are overlapped and added together, adding or basically tying the samples of the neighbouring frames together can supply some limited timing information. That is why the iteration in this case may have some limited advantages.

Anyhow, the minimum-phase-only reconstructed signal after short-term analysis, highly resembles the original signal. By frame length expansion, the distance between the minimum-phase-only reconstructed signal and the original signal increases which means less information resides in this part of the Fourier transform. Note that the minimum-phase and the all-pass components together constitute the whole signal information and also do not share any information. Since the information content of the signal is constant regardless of the frame length, the higher importance of one of them necessarily translates into lower significance for the other one. As such from Figure 3.17 one can infer that the minimum-

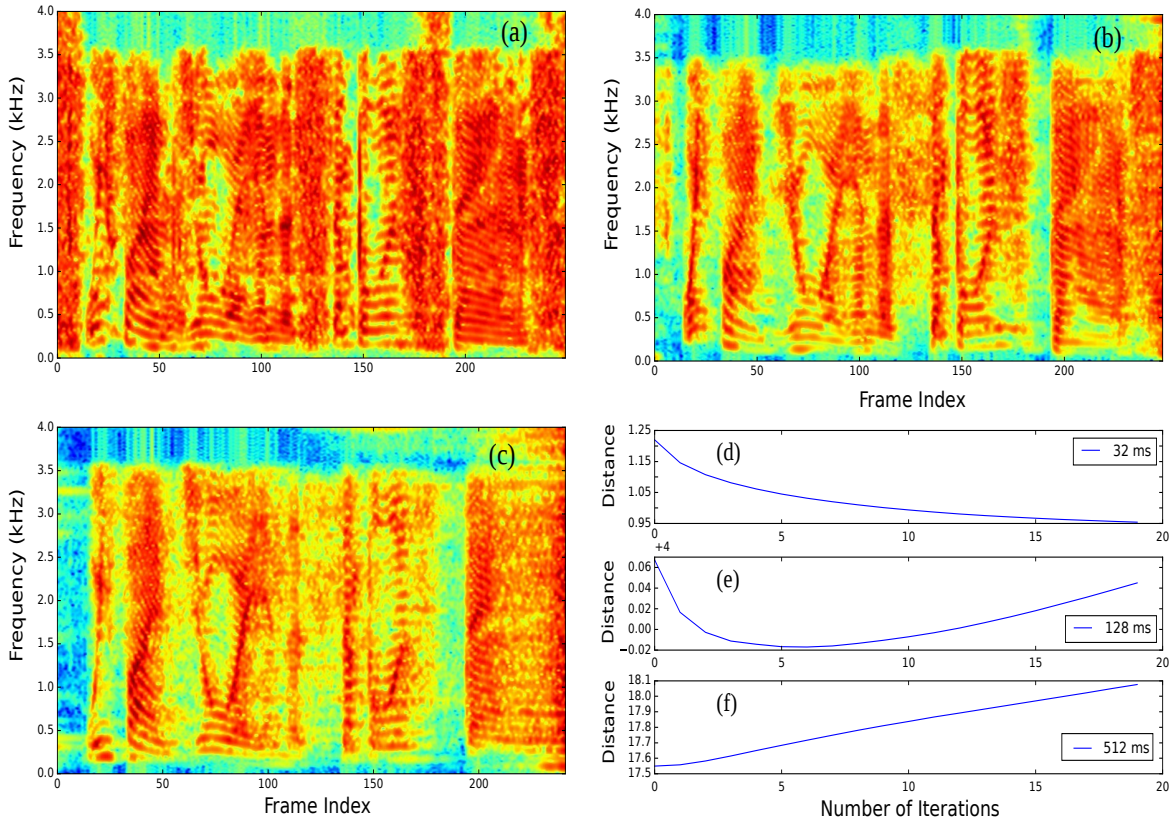

Fig. 3.18 All-pass-only signal reconstruction after short-term, mid-term and long-term Fourier analysis. The minimum-phase part was initialised with unity, frames overlap was set to 75%, a Hamming window was used for analysis/synthesis and 100 iterations were applied. (a) Original signal, (b) short-term analysis, frame length: 32 ms, (c) mid-term analysis, frame length: 128 ms, (d) long-term analysis, frame length: 512 ms.

phase part is the dominant component of the Fourier transform in short-term analysis. Also, the minimum-phase component after short-term analysis includes both source and filter components of the speech signal. This means that both magnitude and phase spectra would share these pieces of information.

All-pass-only

Finally, Figure 3.18 portrays the spectrograms of the all-pass-only reconstructed signals after short-, mid- and long-term analysis. As expected, by frame length extension, the similarity of the all-pass-only reconstructed signal with the original signal increases. In addition, although iteration is not that much useful, still can lead to some limited quality improvement due to the contextual constraints imposed by the overlap-add synthesis. The timing of the events in the all-pass-only signal is preserved, regardless of the short or long

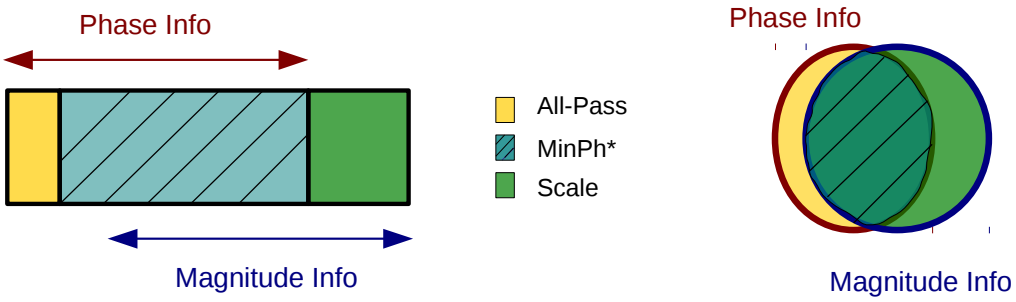

Fig. 3.19 Speech signal information distribution between the all-pass and minimum-phase components after short-term analysis. The minimum-phase part is the dominant component and the all-pass part plays a marginal role.

term signal decomposition which verifies the hypothesis that the all-pass part contains the timing information of the signal. Another noteworthy point is that in short-term analysis, the source (excitation) component (specially for voiced parts) simultaneously exists in both minimum-phase and all-pass parts.

3.4.3 Redrawing the Information Regions

It was demonstrated that the minimum-phase component is the dominant part in the short-term processing whereas the all-pass part prevails by frame-length extension. Figure 3.19 illustrates the signal information distribution between different regions after short-term analysis and Figure 3.20 shows the distribution of information in the long-term analysis.

The dominance of the minimum-phase component in the short-term and the fact that for the minimum-phase signals the magnitude and phase spectra are (almost) equi-informative, overrides the belief that the phase spectrum is perceptually useless or non-informative. Actually, as far as the minimum-phase is the dominant component, phase is as informative as the magnitude spectrum. However, it should be noted that in this condition employing the phase spectrum in an algorithm which already benefits from the magnitude spectrum does not necessarily lead to significant performance improvement because it does not have extra information to offer on top of what already provided by the magnitude spectrum. However, in the mid- and long-term analysis/processing where the all-pass component obtains more importance, utilising the phase spectrum along with the magnitude spectrum could be more influential and useful.

Figure 3.21 shows the spectro-temporal information distribution of the signal, inter- and intra-framely. Intra-frame, (local) temporal information of the signal is captured by the all-pass part and the inter-frame timing information (relative order of the frames) resides

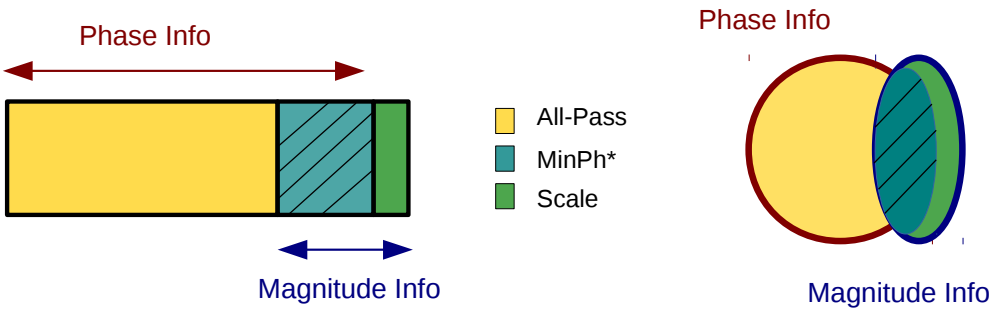

Fig. 3.20 Signal information distribution between the all-pass and minimum-phase components after long-term analysis. The minimum-phase part plays a marginal role and the all-pass part is the dominant component.

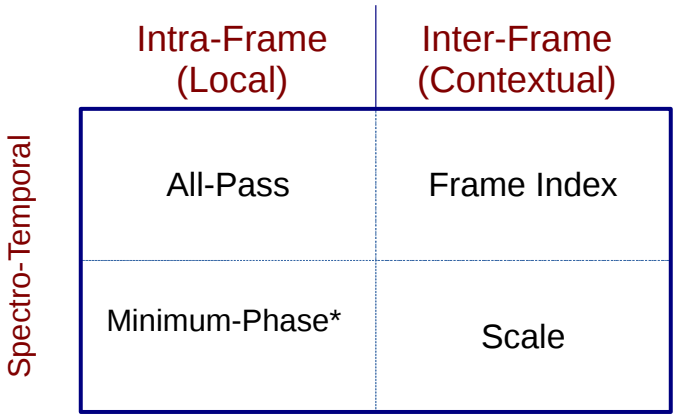

Fig. 3.21 Spectro-temporal information distribution for a non-stationary mixed-phase signal.

in the frame index. On the other hand, the intra-frame spectral information is captured by the minimum-phase-scale-excluded ($MinPh^*$) part whereas the the required information for joining the frame is captured by the scale information ($exp(\tilde{x}[0])$).

3.4.4 Effect of the Window Shape

In addition to the frame length, window shape is another factor which plays a notable role in $\mathscr{X}$-only signal reconstruction. In general, windowing is done to alleviate the spurious high frequency components caused by cutting the signal into smaller segments. Tapered windows provide a better spectral resolution-leakage trade-off, compared with the rectangular (non-tapered) window. They have a wider mainlobe and smaller sidelobes which leads to less frequency resolution but better (less) frequency leakage.

In magnitude-only signal reconstruction as shown in Table 3.1, Hamming window works better than others whereas in case of the phase-only signal reconstruction Chebyshev window

with dynamic range of 25-35 dB leads to higher quality. In general, one can argue that the optimal resolution-leakage trade-off for each case is provided by the aforementioned windows. This can be also justified using the information regions concept. In case of the magnitude-only signal reconstruction, the missed information is the all-pass part which carries temporal information. Hence, as far as the window shape is concerned, a window with a tapered shape is a better option than the rectangular one because a tapered window imposes a temporal constraint on the frame and emphasises signal content located in its centre.

For the phase-only reconstructed signal the timing is not important because the timing information already exists in the phase spectrum. What is missed is the scale information, namely $exp(\tilde{x}[0])$. As Table 3.2 shows, the phase-only signal reconstruction using the rectangular and Chebyshev (25-35 dB) windows results in higher quality in PESQ scale than using the tapered windows. Based on the information region concept, there should be a link between the characteristics of these windows and the scale information which hopefully helps in justifying/explaining the appropriateness of each window.

Let us revisit the scale info at the frame t along with considering the window effect

$$\begin{aligned}\tilde{x}[0] &= \frac{1}{2\pi}\int_{-\pi}^{\pi} log|\frac{1}{2\pi}X(\omega) * W(\omega)|^2 d\omega \\ &= \frac{1}{(2\pi)^2}\int_{-\pi}^{\pi} log\Big|\int_{-\pi}^{\pi} X(\theta)W(\omega-\theta)d\theta\Big|^2 d\omega \end{aligned} \tag{3.22}$$

Now consider a frequency in which a zero of $X(\omega)$ occurs. In this case, if the spectral leakage is very small the argument of log would be close to zero and consequently the $\tilde{x}[0]$ tends to minus infinity. In theory, if it becomes minus infinity or practically a very big negative number, the values of the other bins will be masked and destroyed. Having some spectral leakage could be useful to avoid having too small values for scale information, although it comes at the cost of spectral smoothing and resolution loss. Rectangular and Chebyshev windows provide enough spectral leakage to prevent this problem and that is why they better fit working with the phase spectrum.

On the other hand, large dynamic range of windows like Hanning aggravates the aforementioned problem. Note that if instead of the Hannig window, its square-root, namely square-root Hanning [152] is employed, such problem alleviates because taking square roots leads to stronger side-lobes. That is why the square-root Hanning window has been successfully utilised in some phase-related work like [153] despite the fact that the Hanning window itself is not an optimal option for working with phase (Table 3.2). Figure 3.22 shows the mainlobe and sidelobes of different windows and further clarifies the aforementioned point.

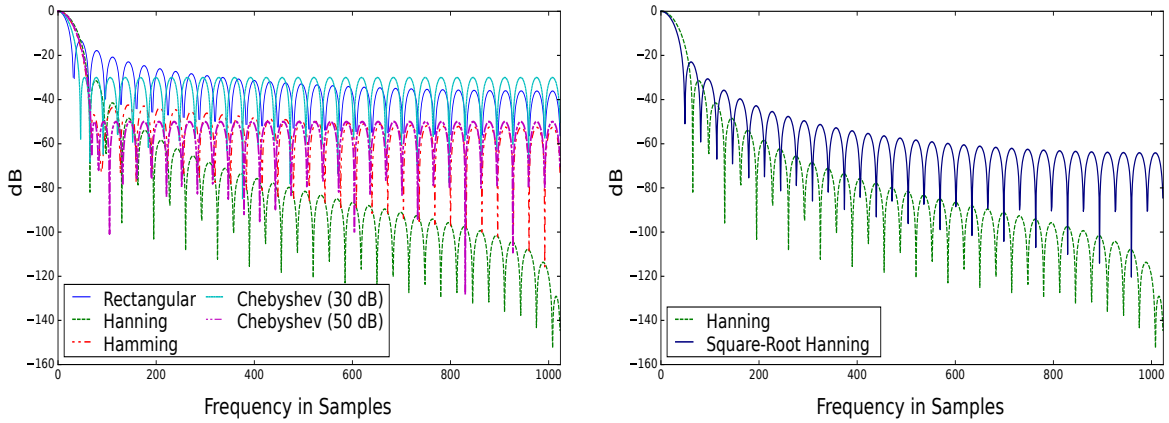

Fig. 3.22 Magnitude spectrum of different windows. Based on (3.22), windows with very small sidelobes are not optimal choices for working with phase spectrum. Square-root Hanning window is a better option than the Hanning window in working with the phase spectrum because of having less attenuation in the sidelobes.

In [85], the authors have related the appropriateness of the windows like Rectangular and Chebyshev to the sidelobes issue. However, they explain it based on how the phase spectrum is computed and argue that since $\phi(\omega) = arctan(\frac{X_{Im}(\omega)}{X_{Re}(\omega)})$, in case of spectral zeros (small side-lobes) both X_{Re} and X_{Im} simultaneously tend to zero. As such there would be $\frac{0}{0}$ which results in numerical instability and consequently unreliable value for the phase spectrum.

It should be noted that numerical instability means the results at each computation would be (at least slightly) different. The phase spectrum does not behave like a random variable, even if zeros are located next to the unit circle. Also, it is unlikely to have the $\frac{0}{0}$ or $\frac{\varepsilon}{\varepsilon}$[14] fractions in practice and for zeros placed even very close to the unit circle, still the accuracy of the computers is enough to get stable correct results.

3.4.5 Importance of Information Regions in Speech Enhancement

Investigating the noise-sensitivity of the information regions gives an idea about the relative importance of each part in speech enhancement. To this end, one can replace the information regions of the noisy part with the corresponding clean version as shown in Figure 3.23. The clean signal is contaminated with noise and then both clean and noisy signals are analysed and decomposed in the information space. In the modification stage, the information components of the noisy signal are substituted with the corresponding clean part. This serves as the modification step in the aforementioned AMS framework. The signal analysis was carried out using 32 ms and 512 ms frame lengths to study the issue in both short- and long-term

[14]The ε is the smallest positive number the computer can handle, just before the underflow happens.

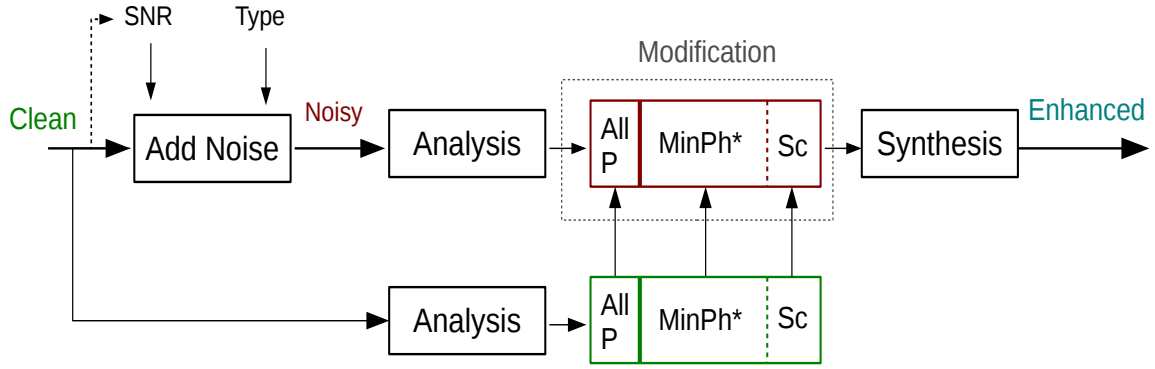

Fig. 3.23 Workflow for studying the noise-sensitivity and the relative importance of each information component in speech enhancement through replacing it with its clean counterpart. MinPh*: minimum-phase-scale-excluded, AllP: all-pass part and Sc: scale information.

analysis. In modification stage, the effect of substituting the minimum-phase (MinPh), minimum-phase-scale excluded (MinPh*), scale Information (Sc) and the all-pass (AllP) parts with their clean counterpart was studied. Figure 3.23 shows the process of building the stimuli. Figure 3.24 and Figure 3.25 depicts the results for short- and long-term analysis, respectively.

As Figure 3.24 demonstrates, based on the spectrograms, in the short-term analysis, cleaning the minimum-phase part of the noisy signal and to a less extend the MinPh* lead to the highest possible enhancement. Replacing the all-pass part with its clean version does not bring much change to the signal. This should not be surprising as the role of the all-pass part in the short-term analysis is limited. Regarding the scale information cleaning, it seems that it mainly affects the non-speech regions. This is due to the fact that the scale information mainly gets affected at the frames in which the speech signal is weak (relative to the noise level) or is absent as shown in Figure 3.24(g). Thus, cleaning the scale information could be helpful in finding the borders between speech and non-speech parts. The distribution of the scale information was studied, too. In order to get statistically significant results, the histograms were computed using all the 30 signals of NOIZEUS database (7470 frames). As seen in Figure 3.24(h), by adding noise, depending on the SNR and noise type, the overall shape and the support of the distribution changes. In particular, by SNR reduction the mean increases and the variance decreases.

On the other hand, cleaning the different information regions of the noisy signal, when it is decomposed into long frames has the opposite effect. As Figure 3.25 shows, the all-pass part plays the major role and its cleaning will lead to a higher level of enhancement. Replacing the scale and minimum-phase part with their clean version, however, are not effective due to the fact that they have a minor role to play in the long-term analysis. Evaluating the quality

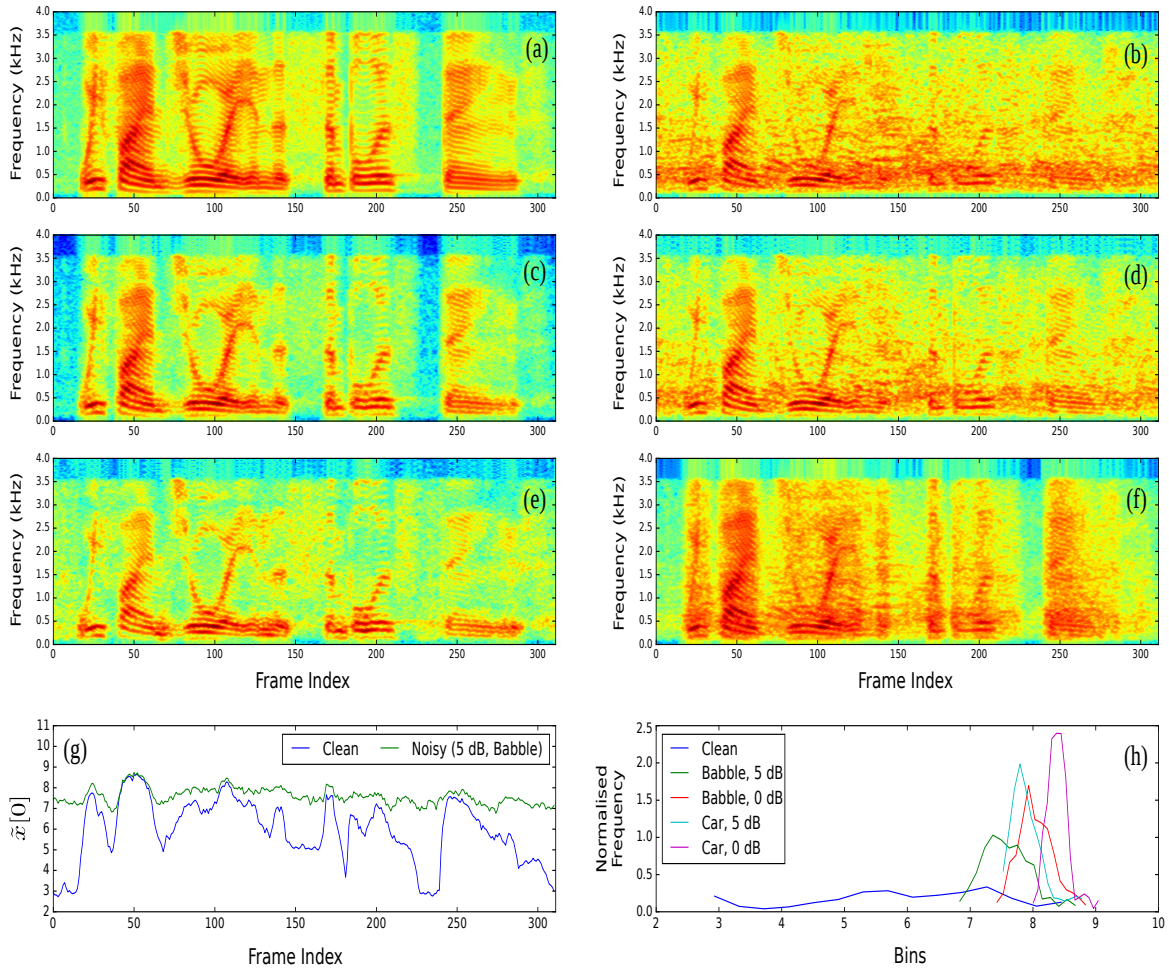

Fig. 3.24 Effect of replacing the minimum-phase and all-pass parts with the corresponding clean version in short-term analysis (frame length:32 ms, overlap: 75%). (a) original clean signal, (b) noisy signal (5 dB, Babble noise), (c) replacing the noisy minimum-phase part with its clean version, (d) replacing the noisy all-pass part with its clean version, (e) replacing the noisy min-phase-scale-excluded ($MinPh^*$) part with its clean version, (f) replacing the noisy scale information with its clean version, (g) effect of the noise on the scale information, (h) Effect of the noise on the histogram of the scale information, $\tilde{x}[0]$. Histograms computed using all the signals of the NOIZEUS database ($\approx$ 7470 frames).

improvement using PESQ score or WER change in an ASR task, can quantify the influence of cleaning each information region with a higher accuracy and reliability.

The extra information of the phase spectrum is related to the all-pass part. Figure 3.24 illustrates that in short-term analysis although the effect of the all-pass cleaning is limited, still it is not zero. As a result, some limited gain can be achieved through entering the phase into the magnitude-based enhancement algorithms.

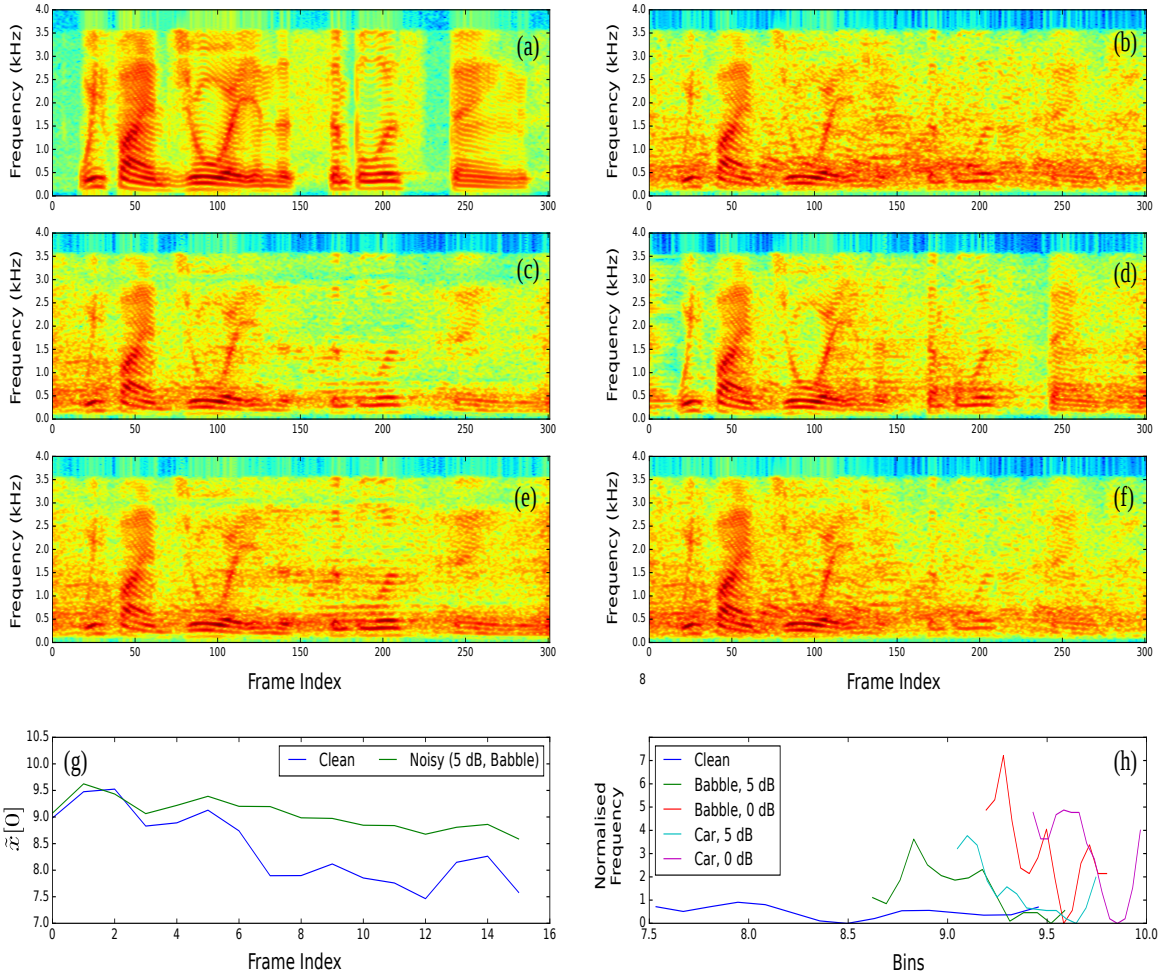

Fig. 3.25 Effect of replacing the minimum-phase and all-pass parts with the corresponding clean version in long-term analysis (frame length:32 ms, overlap: 75%). (a) original clean signal, (b) noisy signal (5 dB, Babble noise), (c) replacing the noisy minimum-phase part with its clean version, (d) replacing the noisy all-pass part with its clean version, (e) replacing the noisy min-phase-scale-excluded ($MinPh^*$) part with its clean version, (f) replacing the noisy scale information with its clean version, (g) effect of the noise on the scale information, (h) Effect of the noise on the histogram of the scale information, $\tilde{x}[0]$. Histograms computed using all the signals of the NOIZEUS database ($\approx$ 7470 frames).

3.5 Summary

In this chapter the information content of the phase and magnitude spectra along with the minimum-phase and all-pass parts of the Fourier transform was evaluated. In this regard, the signal was reconstructed only from the aforementioned parts and the similarity of the reconstructed signal to the original one was taken as a proxy for the information content. It was shown the the total information encoded in the signal can be divided into three regions,

namely all-pass, minimum-phase-scale-excluded ($MinPh^*$) and scale information. The phase spectrum uniquely captures the all-pass information, magnitude spectrum uniquely includes the scale information and the $MinPh^*$ information is shared by both phase and magnitude spectra. This concept was depicted using vector notation and also using the Venn diagram. The signal was reconstructed in the AllP-only, MinPh-only, magnitude-only and phase-only modes in short-, mid- and long-term. It was demonstrated that the all-pass part includes the timing information of the signal and its importance increases by frame length extension. The minimum-phase component is the dominant element in short-term analysis and by frame length expansion its importance decreases. It was illustrated that the scale information is important for joining the frames together in synthesis and by frame length expansion its importance decreases. The usefulness of the non-tapered windows for working with the phase spectrum was discussed, too. The spectrogram of the phase-only reconstructed signal showed that the phase spectrum includes the excitation and vocal tract information. The goal of the next chapter is to perform source-filter modelling in the phase domain and separate these two components through phase processing.

Chapter 4

Source-Filter Separation in the Phase Domain

Simin: He [Nader's Father] doesn't even know who you are.
Nader: He does not know me but I know that he is my father.
– Asghar Farhadi, *A Separation*

You can't separate peace from freedom because no
one can be at peace unless he has his freedom.
– Malcolm X

4.1 Introduction

In the last chapter the potential for phase-based speech processing in terms of the information phase carries in short, mid, and long-term was established. Also it was shown that the phase-only reconstructed speech contains signal's elemental components, namely source and filter. This removes doubts about the potential and the usefulness of this part of the Fourier transform. Now, the next problem to be tackled is to propose a mathematical framework to model the phase and harness the information that resides in it.

One of the fundamental models in speech processing is the source-filter model [96]. The current approaches are based on the magnitude spectrum and they highly facilitate applying the magnitude spectrum to process the speech signal. This chapter aims at extending the idea of the source-filter modelling to the phase domain and tries to separate the source and filter components through phase processing. Such a foundational model as well as shedding further light on the phase behaviour and the way it encodes the speech information, paves the way for applying the phase spectrum in a wide range of applications in speech processing.

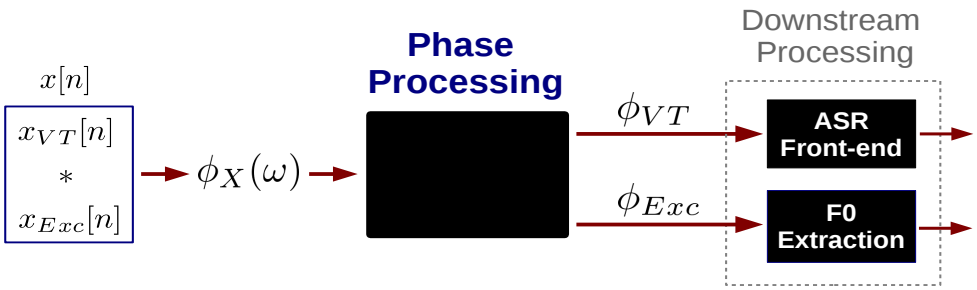

Fig. 4.1 Chapter goals are shown as three black boxes to be filled: source-filter separation in the phase domain, feature extraction from the filter component for ASR and fundamental frequency extraction from the source component.

In order to construct such a model, first the relationship between the source and filter components in the phase domain is derived. This helps in devising a mathematical framework for separating or deconvolving the excitation and vocal tract parts of the speech signal through phase manipulation. The efficacy of the proposed approach is initially evaluated using the spectrograms of the separated source and filter components. In the next stage of the evaluation, the filter component is turned into some features and the performance is examined on connected-digit and continuous speech recognition tasks in the clean and multi-style training modes. Finally, the phase spectrum of the excitation part is utilised for pitch estimation. The accuracy and robustness are tested and compared with other well-known magnitude-based pitch estimation techniques. Experimental results demonstrate the success of the phase-based speech processing. In short, the goal of this chapter is to fill the black boxes shown in Figure 4.1.

The organisation of this chapter is as follows. In Section 2 the basics of source-filter separation using cepstral (log-magnitude) processing and LPC analysis is briefly reviewed. Section 3 deals with the proposed method for non-parametric phase-based source-filter separation and examines its theoretical and practical aspects. In Section 4, the phase filter component is turned into features and they are tested in ASR. In Section 5 the phase excitation part is employed in fundamental frequency extraction and the accuracy and robustness are evaluated. Finally, Section 6 summarises the chapter.

4.2 Source-Filter Modelling and Separation in Magnitude Spectrum Domain

The idea of the source-filter modelling of the speech signal originally dates back to the work of Johannes Müller in 1848 [154] and Chiba and Kajiyama in 1941 [155] which was later

used in the more modern model proposed by Gunnar Fant [96] and Kenneth Stevens [156]. It provides a simple model of how the speech signal is generated and has been employed as a basis for a wide range of applications for processing the speech signal. In this model, speech is considered as a local-stationary signal, which is assumed to be stationary in short segments or frames. A stationary segment of the speech signal, $x[n]$, equals the convolution of the excitation (Exc) component and the vocal tract (VT) in the time domain

$$x[n] = x_{Exc}[n] * x_{VT}[n] \tag{4.1}$$

where n, x_{Exc} and x_{VT} denote the time (in samples), excitation signal and the impulse response of the vocal tract, respectively. As well as stationarity, this formula also assumes that

- the speech production system is linear
- the excitation (source) and vocal tract (filter) components are independent and do not interact with each other.

The speech production system does not meet these two demands perfectly but such approximations are not far from the real characteristics of this system and bring about much mathematical convenience.

Applying the Fourier transform yields

$$X(\omega) = X_{Exc}(\omega)\, X_{VT}(\omega) \tag{4.2}$$

where ω, $X_{Exc}(\omega)$ and $X_{VT}(\omega)$ are the angular frequency, short-time Fourier transforms of the vocal tract and excitation components, respectively. As such the magnitude spectra of these two elements are multiplicative and the phase spectra would be additive

$$\begin{aligned} |X(\omega)| &= |X_{Exc}(\omega)|\, |X_{VT}(\omega)| \\ arg\{X(\omega)\} &= arg\{X_{Exc}(\omega)\} + arg\{X_{VT}(\omega)\} \end{aligned} \tag{4.3}$$

where $|.|$ and $arg\{\ \}$ denote the short-time magnitude and unwrapped (or continuous) phase spectrum, respectively. Note that using $ARG\{\ .\}$ instead of $arg\{\ .\}$ is mathematically incorrect because principle phases cannot be added together.

A goal of the source-filter modelling is to separate the vocal tract and excitation parts (Figure 4.2). Linear predictive coding (LPC) and low-time cepstral liftering are two well-established techniques in this regard [21]. LPC could be considered as a parametric [1]

[1] Due to all-pole model assumption

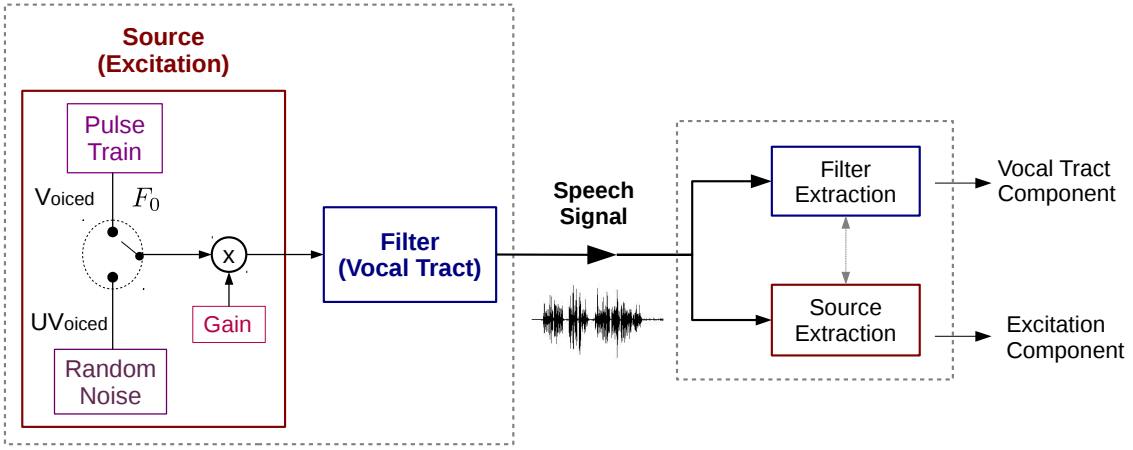

Fig. 4.2 Source-Filter modelling of the speech signal. The goal is to separate the vocal tract and excitation components.

approach to this problem while cepstral liftering is a non-parametric method. In the next subsection both of these two magnitude-based techniques are reviewed.

4.2.1 Parametric Source-Filter Separation using LPC Analysis

The envelope of the power spectrum of the speech signal can be estimated using all-pole modelling [37] and the envelope is closely associated with the vocal tract component of the speech signal. So, the filter element of the speech signal can be estimated using linear prediction techniques

$$\hat{X}_{VT}(\omega) \approx \frac{\sigma^2}{1+\sum_{k=1}^{P} a_k z^{-k}}\Bigg|_{z=exp(j\omega)} \tag{4.4}$$

where $\hat{X}_{VT}(\omega)$, P and σ^2 indicate an estimate of the Fourier transform of the vocal tract, the order of the autoregressive model and the variance of the prediction error, respectively. The order of the all-pole model should be large enough to capture the envelope and the coarse structure of the magnitude spectrum, but small enough to avoid capturing the fine structure, which is related to the excitation component. Having estimated the filter component, the source part equals

$$\hat{X}_{Exc}(\omega) = \frac{X(\omega)}{\hat{X}_{VT}(\omega)}. \tag{4.5}$$

where $\hat{X}_{Exc}$ is an estimate of the excitation component. The linear predictive coding approach shows good robustness to noise, can be used in low-bit-rate coding and the roots of the

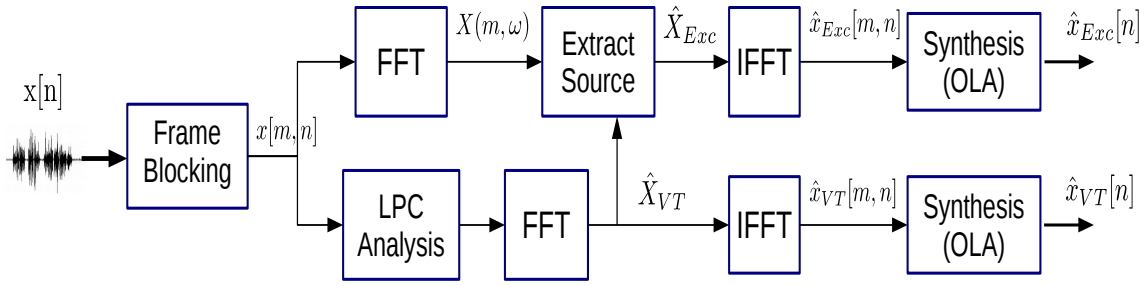

Fig. 4.3 Workflow for extracting the excitation and vocal tract components evolution over time. LPC analysis provides an estimate for the vocal tract part based on (4.4) and using (4.5) the excitation component can be extracted. The excitation-only, $\hat{x}_{Exc}[n]$, and the vocal tract-only, $\hat{x}_{VT}[n]$, signals are synthesised using the overlap-add (OLA) method. Variable m indicates the frame index.

corresponding polynomial (the denominator in (4.4)) can be used for computing the formants and their bandwidth.

Figure 4.3 depicts the process of building the source-only and filter-only speech signals using the LPC method. Figure 4.4 shows the waveforms juxtaposed with the corresponding spectrograms of the original, source-only and filter-only signals. As can be seen, the excitation component includes the voicing information along with the fundamental frequency and the vocal tract part captures the formants track.

4.2.2 Non-parametric Source-Filter Separation using Cepstral Liftering

Another popular approach to source-filter separation is to use the log of the magnitude spectrum along with cepstral processing. This approach does not make a parametric assumption like LPC, is straightforward and easy to implement. In the log-magnitude spectrum and cepstral domains, source and filter components are additive,

$$\begin{aligned} log|X(\omega)| &= log|X_{Exc}(\omega)| \; + \; log|X_{VT}(\omega)| \\ \tilde{x}[q] &= \tilde{x}_{Exc}[q] \; + \; \tilde{x}_{VT}[q] \end{aligned} \tag{4.6}$$

where q and $\tilde{x}$ denote the quefrency and real cepstrum, respectively. Additivity allows the components to be separated by applying appropriate linear liftering. A complementary assumption to the additivity is that the components to be segregated has a different rate of change with respect to the independent variable (frequency). This assumption plays a pivotal

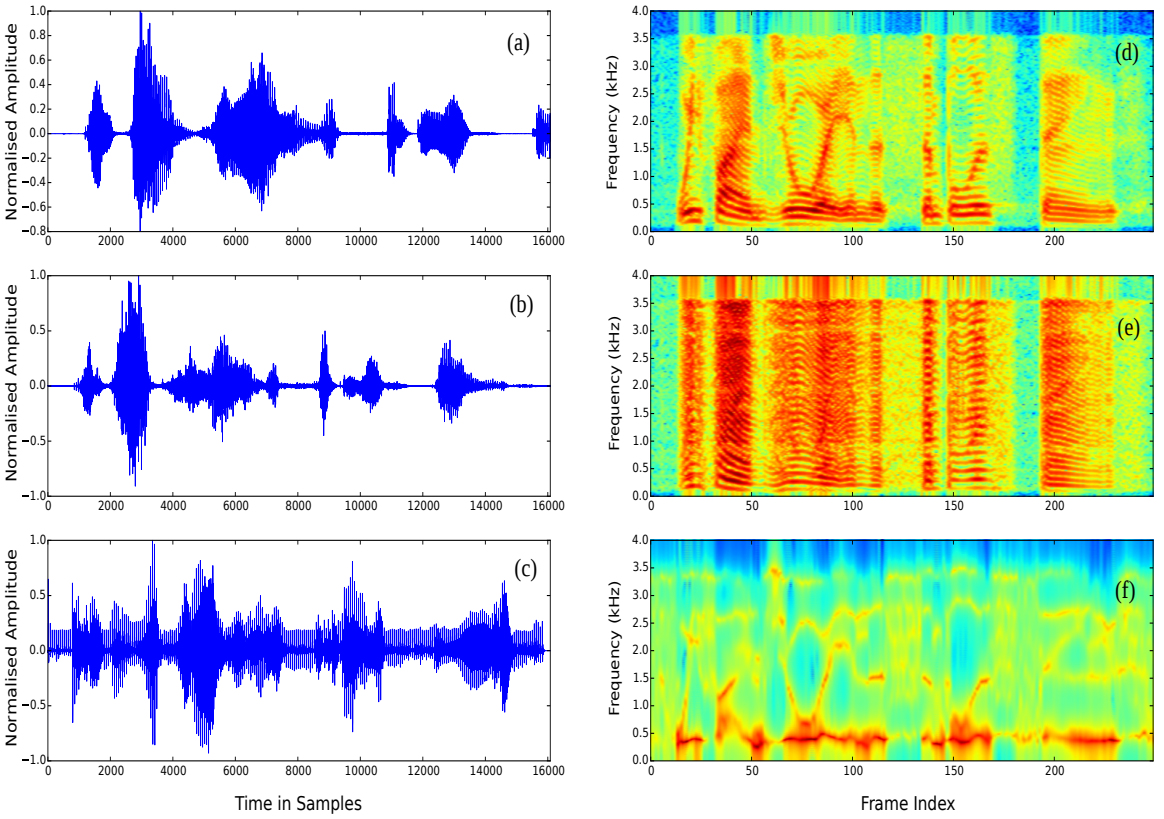

Fig. 4.4 LPC-based Source-Filter separation (LPC order: $12 \approx 1.5\frac{f_s}{1000}$, frame size: 25 ms, frame shift: 10 ms). (a) speech signal (sp07 [22]: "We find joy in the simplest things", $f_s = 8000\ Hz$), (b) Source component in the time domain, (c) Filter component over time, (d) spectrogram of the original signal, (e) spectrogram of the estimated excitation component, (f) spectrogram of the estimated vocal tract component. The cutoff frequencies at 300 Hz and 3400 Hz stem from the intermediate reference system (IRS) filter applied to NOIZEUS signals to simulate the receiving frequency characteristics of telephone handsets [22].

role and permits (4.6) to be rewritten as follows

$$log|X(\omega)| = Trend + Fluctutation \tag{4.7}$$

where $Trend$ is the slowly varying modulating component and $Fluctuation$ is the rapidly oscillating part. Note that the Trend-plus-Fluctuation structure holds only in the log-magnitude spectrum domain and is not extendible to the cepstrum domain, although the additivity holds in both frequency (log-magnitude) and quefrency domains.

Since the Trend varies slowly, assuming the independent variable is time, its high-frequency components after taking the Fourier transform are weak and most of its energy in the frequency domain is concentrated in the low-frequency bins. On the other hand, the

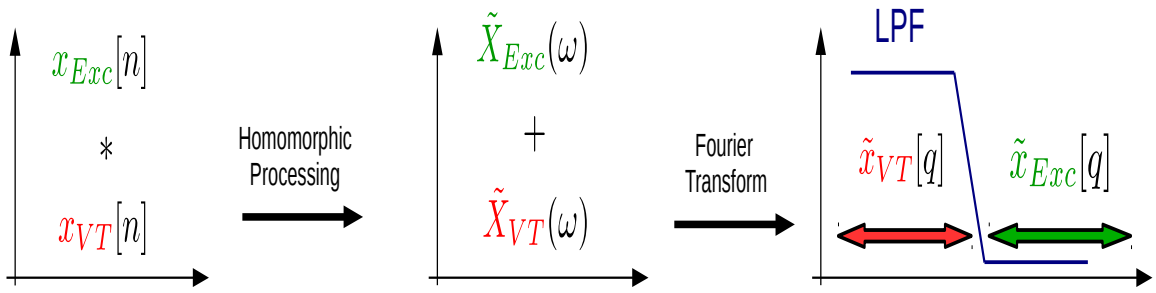

Fig. 4.5 Separating the excitation and vocal tract components based on a Trend-plus-Fluctuation paradigm using low-pass filtering (LPF). Homomorphic processing is used to turn the convolution to sum, paving the way for the linear filter to separate the source and filter parts.

source component changes faster and therefore, most of its spectral energy is in the high-frequency part of the spectrum. As such applying an appropriate low-time lifter (or low-pass filter[2]) to the $log|X(\omega)|$ provides an estimate of the Trend which, in this context, corresponds to the vocal tract and excitation parts. Having estimated the vocal tract component and using (4.6), a simple subtraction gives the Fluctuation part which corresponds to the excitation component. Figure 4.5 illustrates the low-pass filtering (low-time liftering) process for source-filter separation using Homomorphic processing [157] which turns the convolution to a sum using the Fourier transform along with the log function.

4.2.3 Low-pass Filtering for Trend Extraction

For implementing the low-pass filter (LPF), a wide range of options are available. Using a linear filter with an impulse response that resembles a window function can serve as a low-pass filter. In such case, employing a rectangular window acts like a moving average (MA) filter. By the same token, applying a tapered window results in a weighted moving average (WMA) filter. Since the low-pass filter eliminates the rapidly changing components that reside in the high frequencies, e.g. fluctuations, its output would be a smoother version of the input[3].

Another option for low-pass filtering is the brick-wall or ideal filter. In this case, the impulse response of the filter is a sinc function. The practical problem with the ideal filter

[2]Low-time liftering is mathematically equivalent to the low-pass filtering with the difference that in case of the low-pass filtering the independent variables before and after taking the Fourier transform are time and frequency, respectively, whereas for low-time liftering the independent variable before and after taking the Fourier transform are frequency and quefrency, respectively.

[3]Analogously, an adjusted high-pass filter can serve as a de-trender, because it removes the low-frequency components mainly associated with the Trend element.

is its non-causality. Given that all the samples of $log|X(\omega)|$ are available simultaneously, because the independent variable is frequency not time, the ideal low-pass filter can be used in this context and the non-causality is not an issue. Although cepstral smoothing is a general term for smoothing the log-magnitude spectrum through low-time liftering, from now onwards when cepstral smoothing is mentioned we mean using a brick-wall filter in the cepstral domain.

Median smoothing, Hodrick-Prescott [158] and Savitzky-Golay [159] are other options for the low-pass filtering and trend extraction. Median smoothing is a non-linear filter which takes a window of M samples and returns the median. It offers some robustness due to using median but has a higher computational cost and introduces some distortion. The Hodrick-Prescott approach is widely used in macroeconomics [158] to obtain a smooth estimate of the (long-term) trend component of a series subject to a penalty term (with weighting coefficient λ) that constrains the second derivative of the trend component. The higher the λ, the closer the trend to a linear estimation. Savitzky-Golay is another smoothing method in which for each point, first a polynomial of degree P is fitted using the M (contextual) points around it and then the point value is replaced with the value returned by the fitted polynomial. Figure 4.6 shows the results of using different types of low-pass filters in the log-magnitude-based source-filter separation process for a single voiced speech frame.

What could be more informative is to depict how the filter and source components evolve over time. In this regard, the excitation-only and the vocal-tract-only signals were reconstructed and the corresponding spectrograms were plotted. Figure 4.7 illustrates the workflow of the filter-only and source-only signal reconstruction. Since in this approach only the magnitude spectrum of the filter and source are available, the corresponding phase spectrum of each component was computed using the Hilbert transform which gives the equivalent minimum-phase phase spectrum. Figure 4.8 and Figure 4.9 illustrate the corresponding waveforms along with the spectrograms for the case of using cepstral smoothing and the Hamming window for low-pass filtering, respectively. In comparison with the LPC-based approach, the resolution of the formants track is lower in this case.

4.2.4 Main Shortcoming of the Low-Pass Filtering Approach

Separation of the Trend (Filter) and Fluctuation (Source) based on the low-pass filtering is straightforward and relatively easy, however, it is not perfect. The main problem stems from the key underlying premise of such a model illustrated in Figure 4.5. In fact, in order to separate the Trend and Fluctuation, as well as the rate of change w.r.t. the independent variable, there is another important implicit assumption. If the supports of the Trend and Fluctuation after taking the Fourier transform (in the quefrency domain), do not overlap,

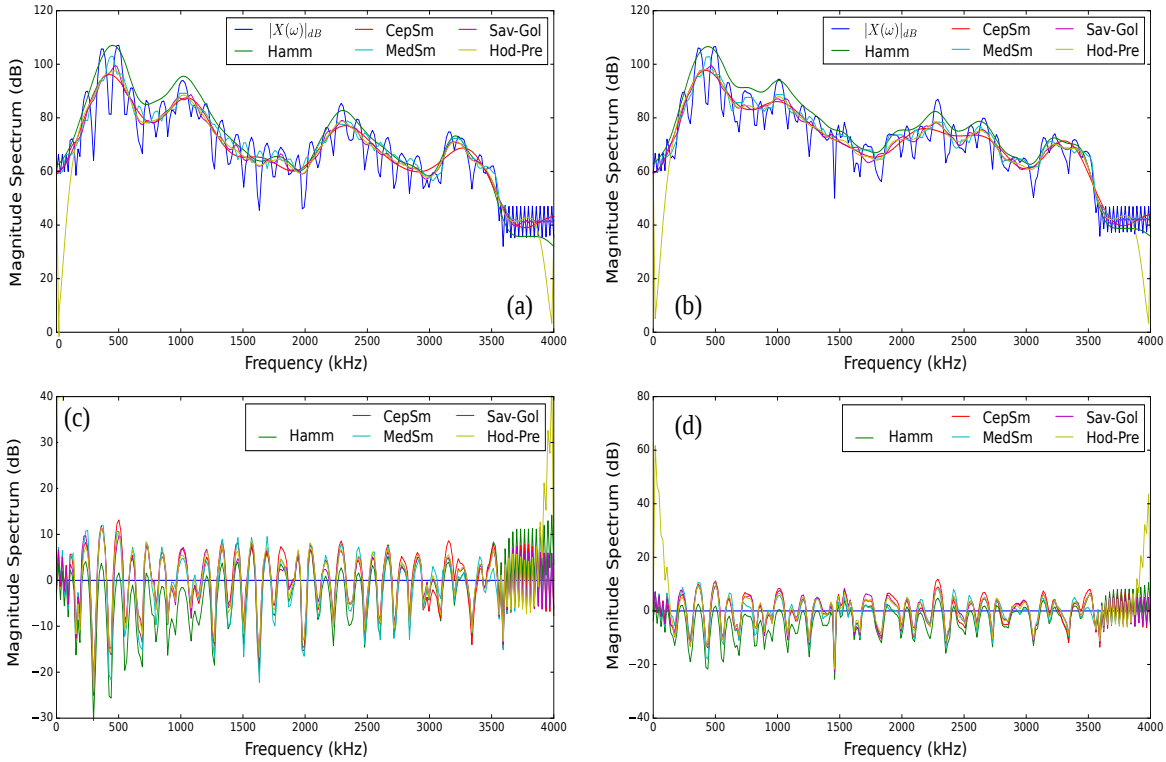

Fig. 4.6 Log-magnitude-based source-filter separation using different smoothing methods, (a) vocal tract component in clean condition, (b) vocal tract component in noisy condition (Babble, 5 dB), (c) excitation component in the clean condition, (d) excitation component in noisy condition (Babble, 5 dB). Hamm: Hamming window (as a low-pass filter), CepSm: Cepstral Smoothing, MedSm: Median smoothing, Sav-Gol: Savitzky-Golay (M=12, P=3), Hod-Pre: Hodrick-Prescot ($\lambda = 50$).

then separation through low-pass filtering works almost ideally (assuming the high cut-off frequency is adjusted properly). If two signals have no overlap in the time or frequency domains, they could be considered *orthogonal*[4]. So, the orthogonality assumption in the domain after taking the Fourier transform is the underlying premise for separating the Trend and Fluctuation components successfully. The higher the overlap, the higher the error associated with the separation. However, as shown in Figure 4.10, in practice this demand is not necessarily met, e.g. when the distance between two adjacent formants become less than the fundamental frequency. Having said that, this approach is widely used in practice and leads to reasonable accuracy.

[4] i.e., if the signals are considered to be vectors, in the domain that they do not overlap, the inner product will be zero.

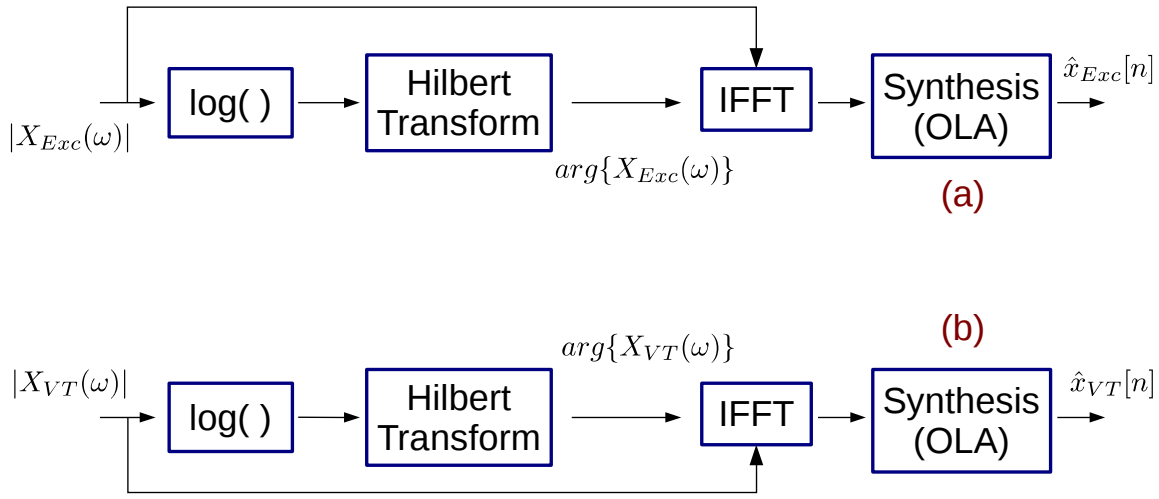

Fig. 4.7 Reconstructing the excitation-only and vocal tract-only waveforms using the minimum-phase assumption and Hilbert transform. The estimated (a) excitation-only signal ($\hat{x}_{Exc}[n]$), (b) vocal tract-only ($\hat{x}_{VT}[n]$) signal.

4.3 Source-Filter Separation in the Phase Domain

In the previous section the classic approach to the source-filter separation using the magnitude spectrum was reviewed. This section presents a novel method for source-filter separation using the speech phase spectrum. As shown in Section 3.2.2, the phase spectrum includes all the signal information except for the scale information and the spectrogram of the phase-only reconstructed speech signal contains both vocal tract and excitation components. It follows that such information exists in this spectrum. This section first examines in which way source and filter get mixed in the phase domain and then a method for separating them is presented.

4.3.1 Source and Filter Information in the Phase Domain

Speech is a mixed-phase signal because its complex cepstrum is neither causal nor anti-causal [71]. As such its (short-term) Fourier transform, $X(\omega)$, can be decomposed as follows

$$X(\omega) = |X(\omega)|\, e^{j\,arg\{X(\omega)\}} = X_{MinPh}(\omega)\, X_{AllP}(\omega) \tag{4.8}$$

where X_{MinPh} and X_{AllP} are the minimum-phase (MinPh) and all-pass (AllP) components of the (short-time Fourier transform) X. Given that $|X_{AllP}(\omega)| = 1$,

$$|X(\omega)| = |X_{MinPh}(\omega)|. \tag{4.9}$$

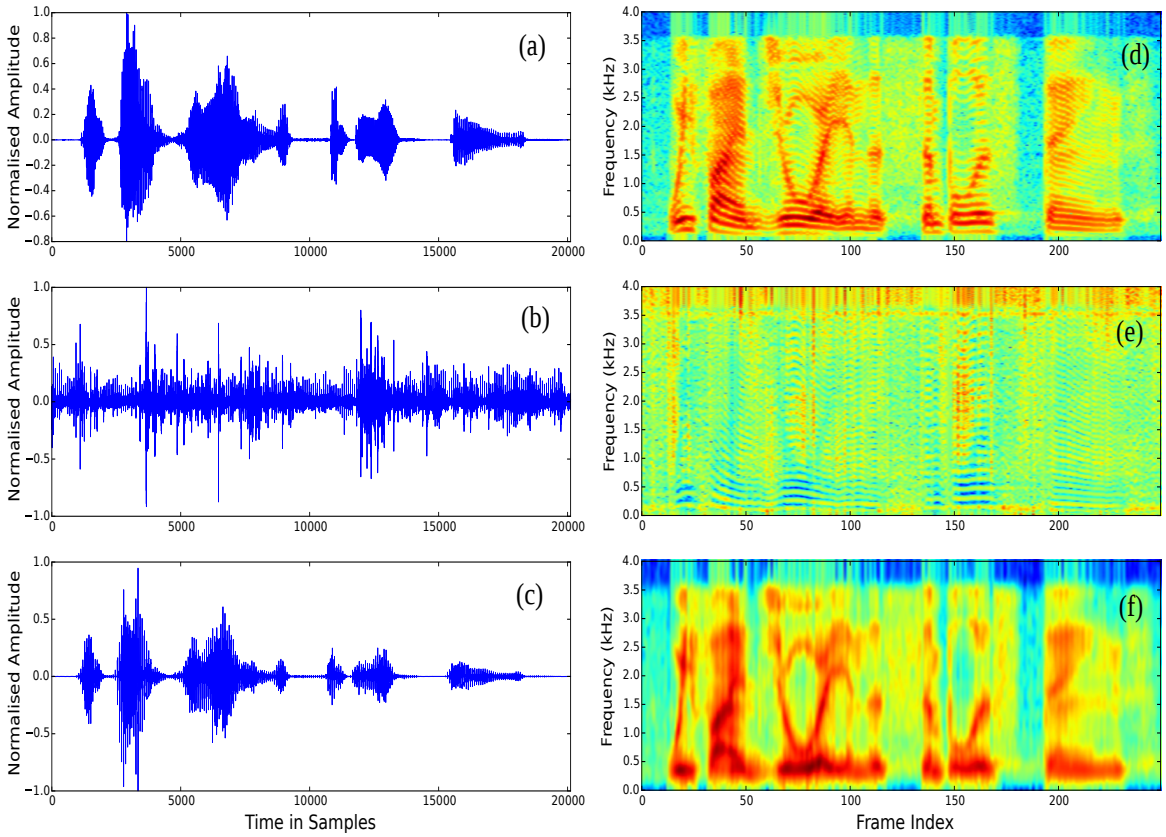

Fig. 4.8 LPF-based Source-Filter separation using Hamming window (as a low-pass filter) with length of 25 samples ($f_s = 8kHz$). (a) speech signal (sp01 [22]), (b) Source component in the time domain, (c) Filter component in the time domain, (d) spectrogram of the clean signal, (e) spectrogram of the estimated excitation component, (f) spectrogram of the estimated vocal tract component.

On the other hand,

$$arg\{X(\omega)\} = arg\{X_{MinPh}(\omega)\} + arg\{X_{AllP}(\omega)\} \tag{4.10}$$

As discussed in Section 3.3.1, in general, the problem of estimating one spectrum from the other one is ill-posed and to have a unique solution, extra constraints like minimum-phase (MinPh) or maximum-phase (MaxPh) should be imposed. For the MinPh signals, due to the causality of the complex cepstrum, the Hilbert transform relates the phase and magnitude spectra together. Also, it was shown that in the short-term analysis, the MinPh component is the dominant element and includes both source and filter components. Now let us use the Hilbert transform to relate the vocal tract and excitation components of the magnitude spectrum to their counterparts in the phase domain.

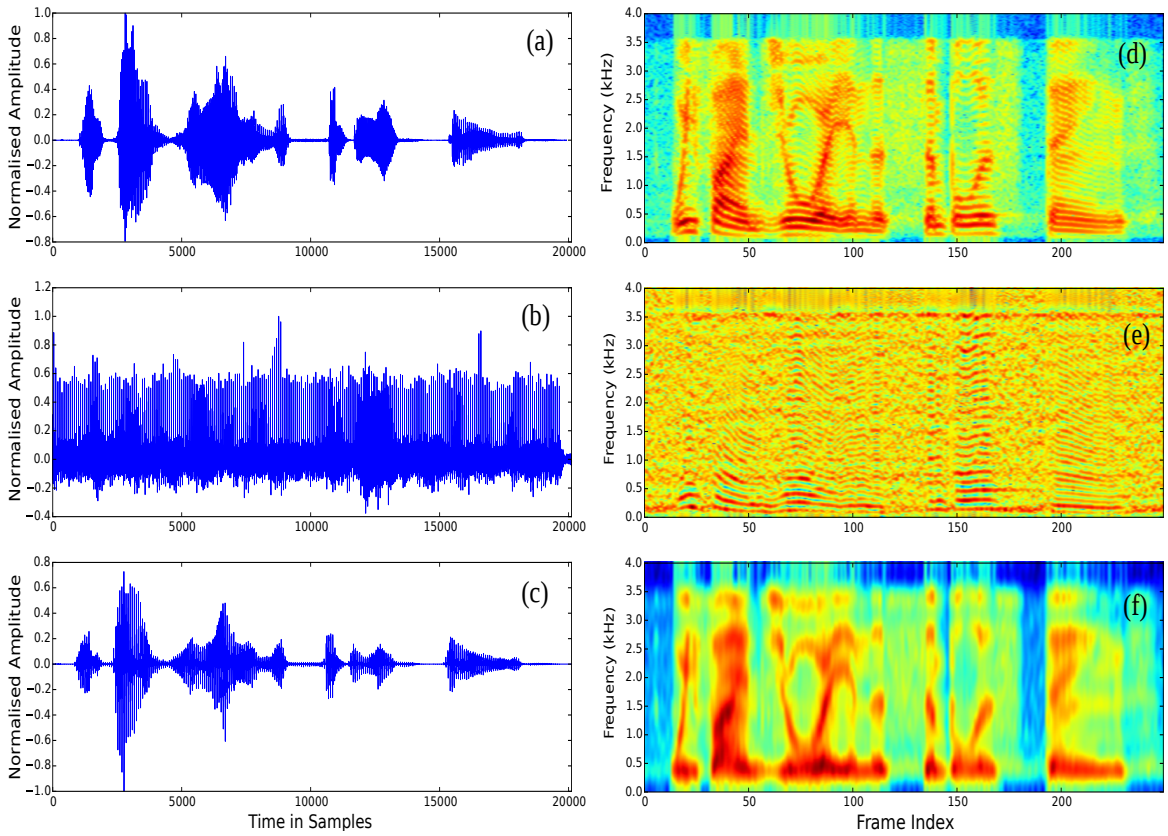

Fig. 4.9 Cepstrum-based Source-Filter separation with length of 15 samples. (a) speech signal (sp01 [22]), (b) Source component in the time domain, (c) Filter component in the time domain, (d) spectrogram of the clean signal, (e) spectrogram of the estimated excitation component, (f) spectrogram of the estimated vocal tract component.

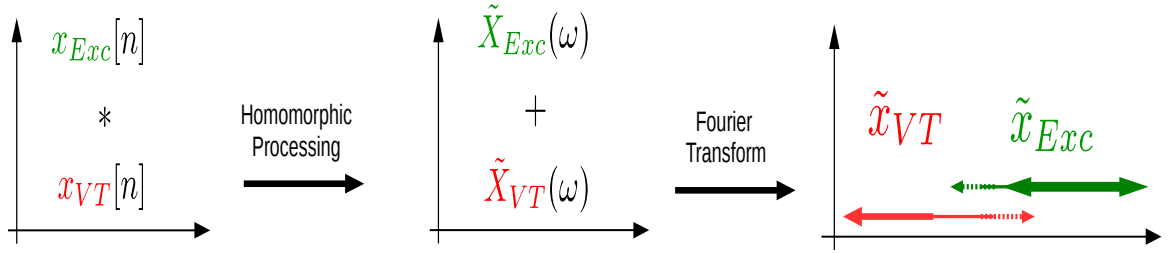

Fig. 4.10 Overlap between the supports of the excitation and vocal tract components in the quefrency domain causes error in the low-pass-filtering approach to source-filter separation.

For the minimum-phase signals, the Hilbert Transform provides the following

$$\begin{aligned} log|X_{MinPh}(\omega)| &= \tilde{x}[0] + \frac{1}{2\pi} arg\{X_{MinPh}(\omega)\} * cot(\frac{\omega}{2}) \\ &= \tilde{x}[0] + \frac{1}{2\pi} \mathscr{P} \int_{-\pi}^{\pi} arg\{X_{MinPh}(\theta)\} \, cot(\frac{\omega-\theta}{2}) \, d\theta \end{aligned} \tag{4.11}$$

where $\mathscr{P}$ denotes the Cauchy principle value of the integral and

$$\tilde{x}[0] = \frac{1}{2\pi}\int_{-\pi}^{\pi} log|X_{MinPh}(\theta)|\, d\theta \tag{4.12}$$

By the same token,

$$\begin{aligned} arg\{X_{MinPh}(\omega)\} &= -\frac{1}{2\pi}\, log\,|X_{MinPh}(\omega)| * cot(\frac{\omega}{2}) \\ &= -\frac{1}{2\pi}\,\mathscr{P}\int_{-\pi}^{\pi} log|X_{MinPh}(\theta)|\, cot(\frac{\omega-\theta}{2})\, d\,\theta \end{aligned} \tag{4.13}$$

A detailed derivation of the above equations is presented in Appendix A. Now let us rewrite (4.13) using (4.9) and (4.3)

$$\begin{aligned} arg\{X_{MinPh}(\omega)\} &= -\frac{1}{2\pi}\, log\big(\,|X_{VT}(\omega)|\,|X_{Esc}(\omega)|\big) * cot(\frac{\omega}{2}) \\ &= -\frac{1}{2\pi}\,\big(log\,|X_{VT}(\omega)| + log\,|X_{Esc}(\omega)|\big) * cot(\frac{\omega}{2}) \\ &= -\frac{1}{2\pi}\, log\,|X_{VT}(\omega)| * cot(\frac{\omega}{2}) \;-\; \frac{1}{2\pi}\, log\,|X_{Esc}(\omega)| * cot(\frac{\omega}{2}) \\ &= arg\{X_{VT}(\omega)\} \;+\; arg\{X_{Exc}(\omega)\}. \end{aligned} \tag{4.14}$$

This shows that the vocal tract and the excitation components are additive in the minimum-phase's phase domain. The role of the AllP part was discussed in Section 3.4.2. It was shown that it mainly includes the timing information of the signal, it plays a marginal role in the short-term analysis and also it encodes a fraction of the excitation component information (Section 3.4.2). Since the signal is analysed in the short-term (frame length $\approx$ 25 ms) and in such circumstance the MinPh part is the dominant component, we concentrate on the phase of the MinPh part in which the source and filter components are additive. Now the goal is to dissociate these two elements through phase processing.

4.3.2 Source-filter Separation in the Phase Domain

In order to separate the filter and source components of the phase spectrum through phase manipulation, as well as knowing they are additive, more information is required. In this regard, let us first visualise the phase spectrum of the MinPh part to see whether it can supply some extra information. Figure 4.11 illustrates the magnitude spectrum, the principle phase, the phase of the MinPh component and the phase of the all-pass part. As seen, relatively speaking, the phase of the MinPh element behaves smoothly whereas the phase spectrum of the all-pass part (similarly to the principle phase) is noise-like and chaotic. This also

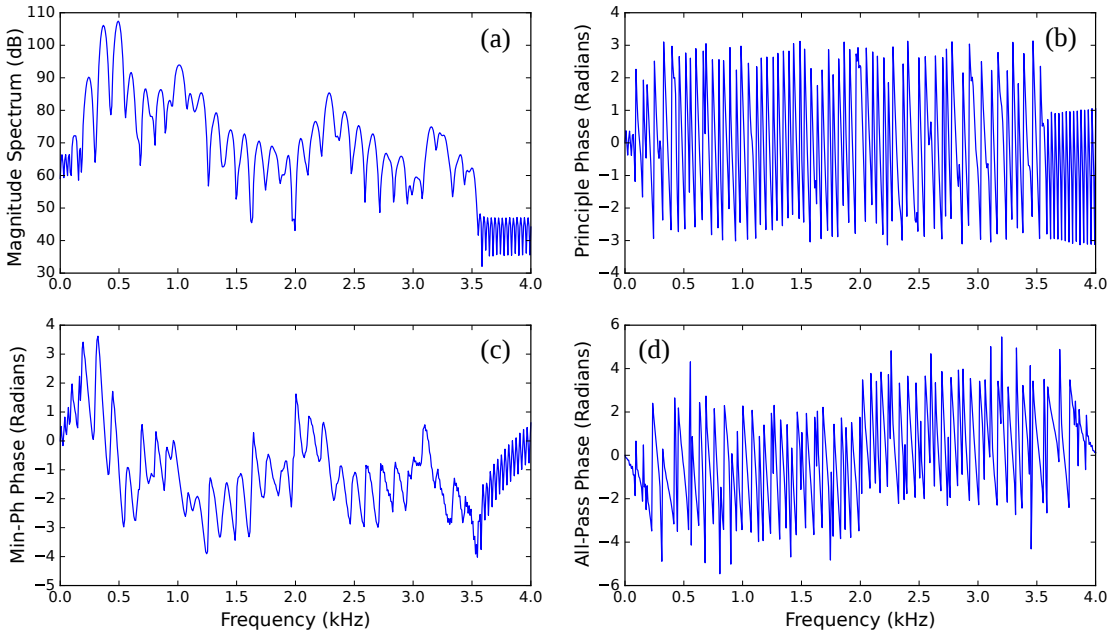

Fig. 4.11 Phase spectrum representations along with the magnitude spectrum. (a) Magnitude spectrum in dB, (b) principle phase ($ARG\{X(\omega)\}$), (c) phase of the MinPh component ($arg\{X_{MinPh}(\omega)\}$), (d) phase of the all-pass component ($arg\{X_{AllP}(\omega)\}$). Phase spectrum of the minimum-phase component behaves relatively smoothly whereas the phase spectrum of the all-pass part is chaotic.

implicitly shows that the all-pass component is the problematic part of the phase spectrum when one wants to work on the group delay function or unwrap the principle phase spectrum. Also note that in computing the phase of the Minimum-phase component there is no need to unwrap the phase because the Hilbert transform, which is used to compute it, directly outputs the unwrapped phase of the MinPh part.

Figure 4.11 (a) and (c) pictorially demonstrate that the source and filter are additive in the $log|X(\omega)|$ and $arg_{MinPh}\{X\}$ domains. Recall that the log-magnitude spectrum was morphed into Trend-plus-Fluctuation structure and the vocal tract and excitation were separated through applying proper filtering. Likewise these two components are additive in the phase domain, too. Furthermore, by inspecting Figure 4.11 (c), it turns out that the phase of the MinPh component, $arg_{MinPh}\{X(\omega)\}$, can be viewed as a sum of two components: a fast oscillating component, modulated by a slowly varying element. As such the phase of the minimum-phase part can be expressed using a $Trend-plus-Fluctuation$ structure, too,

$$arg_{MinPh}\{X(\omega)\} = Trend \; + \; Fluctutation \tag{4.15}$$

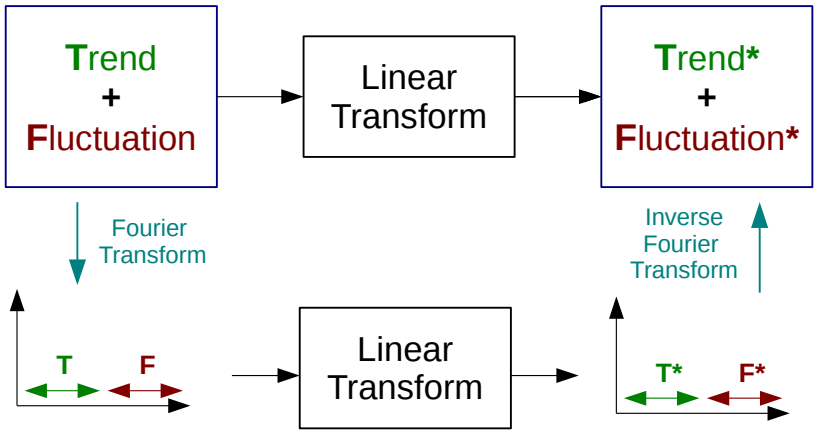

Fig. 4.12 Applying a linear transformation does not change the Trend-plus-Fluctuation structure. In other words, the Trend component remains Trend (occupying the low-pass frequency region after linear transformation) and the Fluctuation part remains the oscillating component occupying the high frequencies.

Mathematically speaking, this should not be surprising. Based on the Hilbert transform relations, the $arg_{MinPh}\{X(\omega)\}$ is a linearly-transformed version of the log-magnitude spectrum. Under a linear transform the spectral support[5] of a signal does not expand. In other words, if two signals are orthogonal (do not overlap) in the frequency domain, after applying a linear transform they remain orthogonal. As such the Trend and Fluctuation components after linear transformation, namely $Trend*$ and $Fluctuation*$, still have the same support. Therefore, the Trend* remains in the low-frequencies and the Fluctuation* occupies the high frequencies. Since the Trend component of the log-magnitude is related to the vocal tract part, the Trend component of its linearly transformed version is connected to the filter component. The same holds for the excitation part. Figure 4.12 illustrates this point. As such, in theory, through low-pass filtering of the phase of the MinPh component, one should be able to separate the excitation and vocal tract elements.

4.3.3 Extracting the Trend and Fluctuation from the Phase

Since the distinguishing characteristic of the source and filter components relates to the pace of change with respect to the independent variable, one can separate them through filtering the phase spectrum of the MinPh component. As such a properly tuned low-pass filter gives the vocal tract component of the phase spectrum. We refer to this filter as the *Trend Extractor*. On the other hand, since the source and filter are additive in the (unwrapped) MinPh phase

[5]The frequency band where outside it the signal has no spectral component.

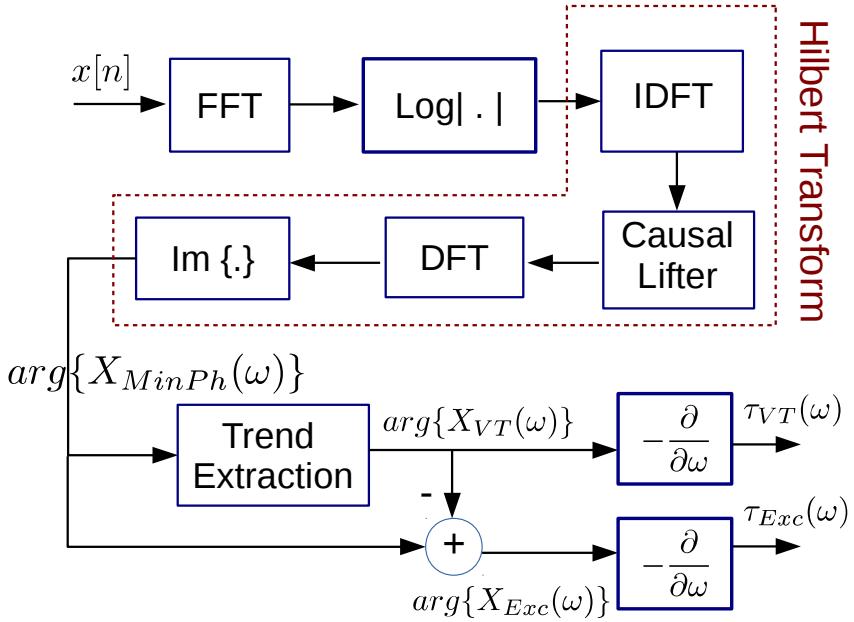

Fig. 4.13 Block diagram of the proposed phase-based speech source-filter separation.

domain, subtracting the Trend from the phase returns the excitation part

$$\begin{cases} arg\{X_{VT}(\omega)\} & \approx h(\omega) * arg\{X_{MinPh}(\omega)\} \\ arg\{X_{Exc}(\omega)\} & \approx arg\{X_{MinPh}(\omega)\} - h(\omega) * arg\{X_{MinPh}(\omega)\} \end{cases} \tag{4.16}$$

where $*$ denotes the convolution operator, $h(\omega)$ indicates the impulse response of the Trend Extractor (low-pass) filter and $arg\{X_{VT}(\omega)\}$ and $arg\{X_{Exc}(\omega)\}$ are the phase spectra of the vocal tract and excitation components, respectively. Note that the frequency response of the filter is $H[q]$, where q denotes quefrency. Figure 4.13 depicts the workflow of the proposed source filter separation process. In the last step, the group delay of the vocal tract ($\tau_{VT}(\omega)$) and excitation ($\tau_{Exc}(\omega)$) components are extracted from the corresponding phase spectra. Working with group delay is favourable as it resembles the magnitude spectrum and is easier to interpret and process [6].

Note that the $arg\{X_{MinPh}(\omega)\}$ can be computed through (4.13), namely in the frequency domain via convolution with $cot(\frac{\omega}{2})$ [7] or in the quefrency domain via applying a causal lifter (zero at the negative quefrencies). In theory and for a continuous variable, both should return the same results. However, given that variables are discrete and that the *cot* function becomes

[6] Later in this chapter we explain what is special about the derivative of the phase which leads to such useful similarity between the group delay and the magnitude spectrum.

[7] For more details about how *cot* enters into the equation, please refer to Appendix A, Equation (A.12).

singular at the roots of its argument, the results are slightly different. It was empirically observed that the cepstral liftering approach is more numerically stable and leads to a little better results, so we use that from now onwards.

4.3.4 Low-pass Filtering for Phase-based Source-filter Separation

The low-time liftering (or low-pass filtering) can be implemented using a wide range of techniques, including window function, cepstral smoothing, median smoothing, Hodrick-Prescott and Savitzky-Golay. For example, in the case of using a Hamming window for low-pass filtering

$$arg\{X_{VT}(\omega)\} \;=\; (0.54 - 0.46\,cos(\omega)) \;*\; arg\{X_{MinPh}(\omega)\} \tag{4.17}$$

and using the brick-wall filter (cepstral smoothing) leads to

$$arg\{X_{VT}(\omega)\} \;=\; 2L\,sinc(2L\,\omega) \;*\; arg\{X_{MinPh}(\omega)\} \tag{4.18}$$

where L is the bandwidth of the filter and *sinc* is the normalised *sinc* function, $\frac{sin(\pi x)}{\pi x}$.

Figure 4.14 shows the source and filter components extracted from the $arg_{MinPh}(\omega)$ for a voiced speech frame by utilising the aforementioned filters as Trend Extractor. As seen, the proposed approach effectively separates the vocal tract and source component through phase-based signal manipulation. Also, the shape of the low-pass filter does not appear to be a critical factor if the filter parameter(s) are adjusted properly. From now onwards we use the ideal low-pass filter, namely the brick-wall filter. The advantage of this filter is that it has only one parameter to adjust (L), the optimal range of its parameter is easy to find and it is computationally efficient. Setting L too large runs the risk of capturing Fluctuation as well as Trend whereas setting it too low leads to over-smoothing. To have a better evaluation of the proposed source-filter separation, it is more helpful to carry out this process for the whole signal in the clean and noisy conditions. Also it should be compared with the excitation and vocal tract components estimated using the magnitude-based approach. Figure 4.15 juxtaposes the phase and magnitude-based source-filter separation results in the clean and noisy (Babble, 5 dB) conditions. For the low-pass filtering, in both magnitude and phase-based approaches, a brick-wall filter was employed. The filter parameter, L, was set to 13, equal to length of the static coefficients of the feature vector because the filter component will be later used in feature extraction for ASR. Note that in Figure 4.15, instead of the phase spectrum, the group delay was used because of its resemblance to the magnitude spectrum which facilitates the comparison.

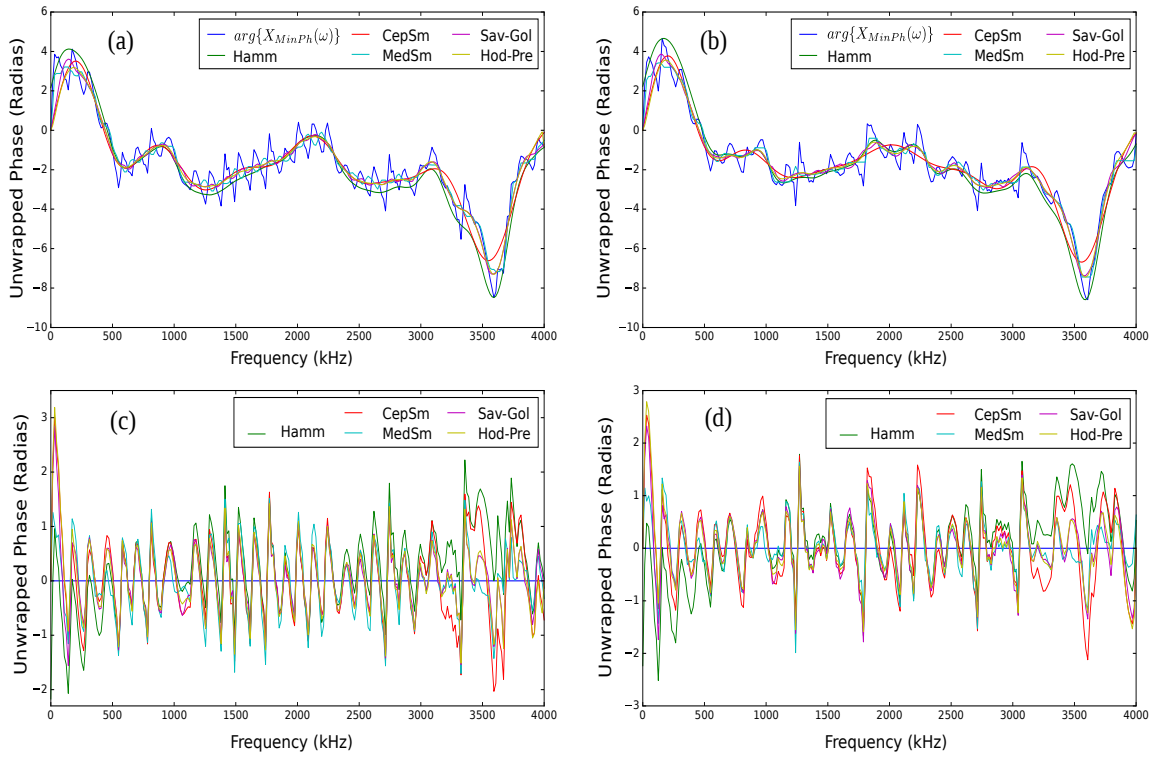

Fig. 4.14 Phase-based source-filter separation using different smoothing methods, (a) vocal tract component in clean condition, (b) vocal tract component in noisy condition (Babble, 5 dB), (c) excitation component in clean condition, (d) excitation component in noisy condition (Babble, 5 dB). Hamm: Hamming window, CepSm: Cepstral Smoothing, MedSm: Median smoothing, Sav-Gol: Savitzky-Golay (M=12, P=3), Hod-Pre: Hodrick-Prescot ($\lambda = 50$).

4.3.5 Phase vs Magnitude Spectrum for Source-filter Separation

As Figure 4.16 shows, the proposed phase-based approach effectively separates the source and filter components, in both clean and noisy conditions. The filter component tracks the formants contours and the excitation components captures the periodicity in the voiced segments of the signal. Based on the spectrograms and in comparison with the magnitude-based approach, the phase-based approach offers three main advantages:

- higher frequency resolution,
- lower spectral leakage,
- better noise robustness.

As the spectrograms of the phase and magnitude-based filter components show, although an identical value for L (parameter of the Trend extractor filter) has been used for both the magnitude and phase-based approaches, the formant track in the clean condition is more

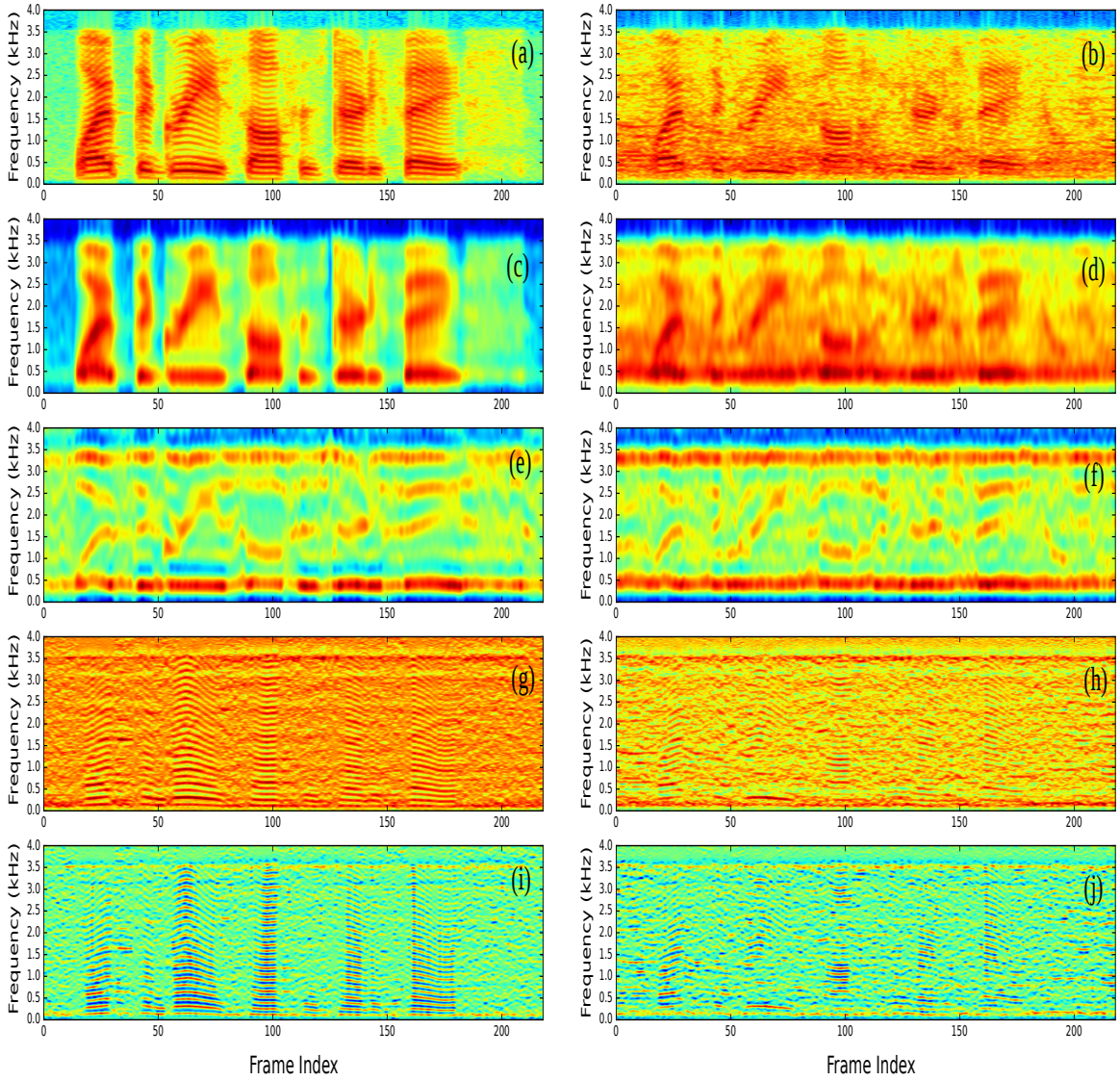

Fig. 4.15 Phase-based and magnitude-based source-filter separation in the clean and noisy (Babble, 5 dB) conditions for *sp*04 from NOIZEUS database. (a) clean signal, (b) noisy signal, (c) magnitude-based filter component in the clean condition, (d) magnitude-based filter component in the noisy condition, (e) phase-based filter component in the clean condition, (f) phase-based filter component in the noisy condition, (g) magnitude-based source component in the clean condition, (h) magnitude-based source component in the noisy condition, (i) phase-based source component in the clean condition, (j) phase-based source component in the noisy condition. The red strips in (e), (f), (i) and (j) at 300 Hz and 3400 Hz stem from the intermediate reference system (IRS) filter applied to NOIZEUS signals to simulate the receiving frequency characteristics of telephone handsets [22].

distinct and clear due to higher spectral resolution and less frequency leakage. Potentially, this can help in extracting features with higher phoneme-discriminability because one of

the phonemes distinguishing attributes is the formants and they are more distinct here. This can also lead to a higher level of robustness as by adding noise, the distance between the features belonging to different classes becomes less which, in turn, gives rise to higher misclassification error. When the classes in the feature space are more separated, the robustness to the noise would be higher, too.

The second column in Figure 4.15 illustrates the effect of the noise on the source-filter modelling and shed further light on the robustness issue. In case of the filter components, the phase spectrum appears to be more robust and the formant track is better preserved while in the magnitude spectrum both leakage and resolution issues get noticeably worse. The formant track alone, however, is not enough for drawing a firm conclusion about the robustness of the phase-based/magnitude-based representations. In the next section, a set of features are extracted from the phase filter component and the robustness is studied in a more rigorous way.

Regarding the robustness of the excitation component, spectrograms apparently do not provide sufficient evidence to make a judgement. Pitch frequency can be extracted in the cepstral domain (after taking DCT/FFT from the $log|X(\omega)|$) as shown in Figure 4.16(a). Based on the same logic[8], taking DCT or FFT from the $arg\{X_{MinPh}(\omega)\}$ could help in extracting periodicity, too. We refer to this domain as *cepstrum**

$$\begin{aligned} DCT\{log|X(\omega)|\} &\Rightarrow cesptrum \\ DCT\{arg\{X_{MinPh}(\omega)\} &\Rightarrow cesptrum* \end{aligned}$$

In the cepstral-based technique for extracting the fundamental frequency, the quefrency of the second peak[9], equals the fundamental periodicity. As Figure 4.16 demonstrates, in the clean conditions the second peak in the cepstrum* domain is more pronounced than in the cepstrum domain. Furthermore, in the noisy condition (Car, 10 dB), in the cesptrum* domain the second peak is still detectable whereas in the cepstrum domain it is buried under the noise. This is further evidence for the robustness of the phase-based approach.

4.3.6 Source-filter Separation in the Group Delay Domain

Group delay is the major representation of the phase spectrum and equals the negative derivative of the unwrapped phase. The derivative could be thought of as a special linear high-pass filter. On the other hand, the Trend Extraction filter acts like a (low-pass) linear

[8]By taking DCT or FFT, periodicity turns into impulses (or spikes) in the fundamental frequency (periodicity) and its harmonics.

[9]First peak occurs at zero.

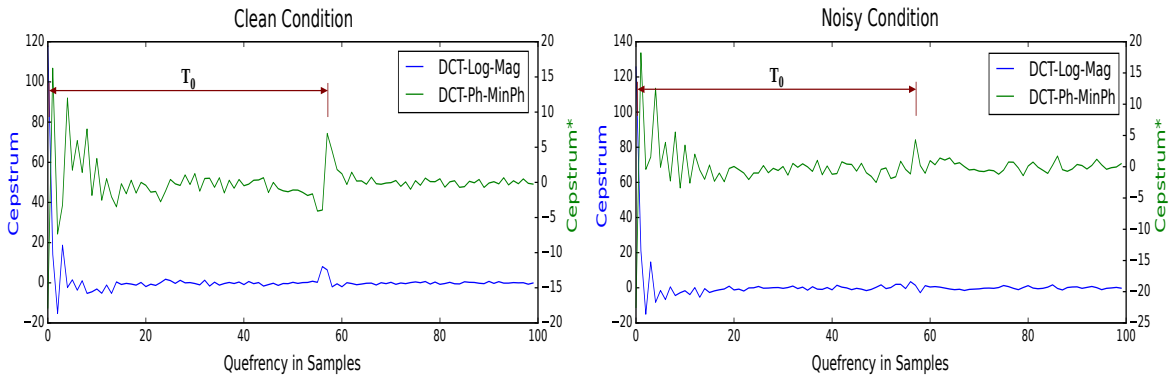

Fig. 4.16 Pitch extraction in the cepstrum and cepstrum* domains in the clean and noisy (Car, 10 dB) conditions.

filter, too. Also, it goes without saying that if the order of linear operators changes, the results will remain the same regardless of the order. So, the Trend-plus-Fluctuation structure of the phase remains intact in the group delay domain, too. In addition, swapping the order of the Trend Extraction filter and the derivative, based on the aforementioned argument, should return identical results. Therefore, the proposed technique should work well in the group delay domain, too. Figure 4.17 shows the separation processes in the phase and group delay domains.

However, remember that the underlying premise which paves the way for dissociation of the Trend and Fluctuation through filtering was orthogonality. This means that the two components should *ideally* have no support overlap (Figure 4.5). The greater the overlap, the less the efficacy of the low-pass filtering approach. The derivative can increase the error associated with the overlap (Figure 4.10). In practice, the overlap is not necessarily zero and after computing the derivative, the vocal tract component in the overlap region will have a stronger presence which implies higher violation of the orthogonality assumption. Mathematically, it runs as follows

$$\begin{cases} \tau_X(\omega) = -\frac{d}{d\omega} arg\{X_{MinPh}(\omega)\} = -\frac{d}{d\omega} Trend - \frac{d}{d\omega} Fluctuation \\ \\ \mathscr{F}\{\tau_X(\omega)\} = j\,q\,\mathscr{F}\{Trend(\omega)\} + j\,q\,\mathscr{F}\{Fluctuation(\omega)\} \end{cases} \tag{4.19}$$

where $\mathscr{F}$ indicates the Fourier transform and q denotes quefrency. Although the high-quefrency components of the vocal tract are weak, they are not zero. Taking the derivative magnifies them linearly with frequency, i.e. the higher the frequency the larger the amplification. This increases the error caused by the overlap between the supports of the source and filter components in the quefrency domain. As such by source-filter separation in the

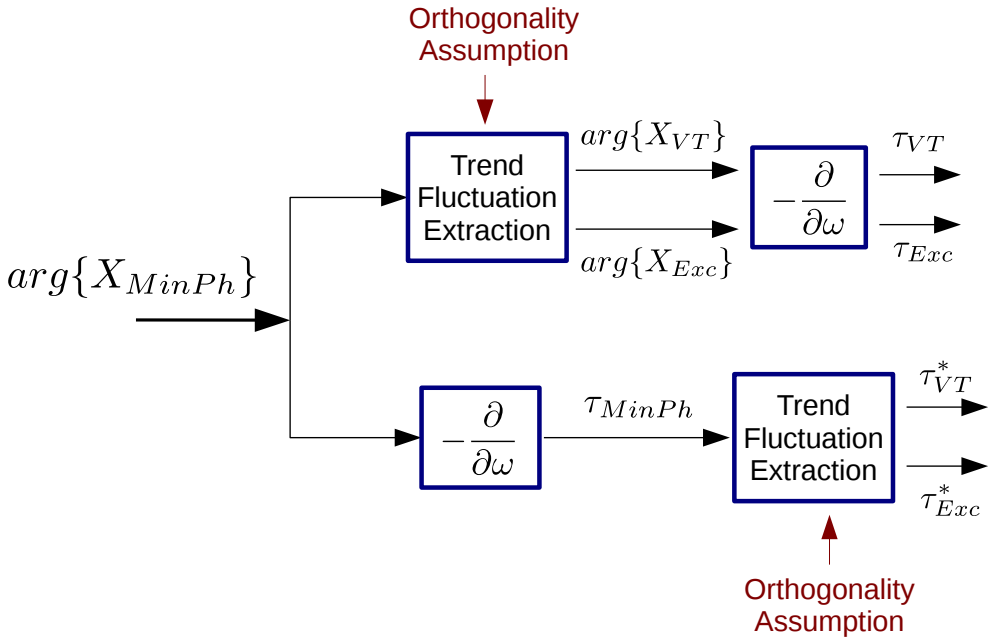

Fig. 4.17 Source-filter separation in the group delay domain. Orthogonality (zero overlap) assumption holds better in the phase domain.

phase domain the probability of violating the underlying assumption would be lower and the segregation could be more effective.

4.3.7 Post-Processing for the Source and Filter Components

To have a better representation of the information residing in the filter and source components of the phase spectrum and to make them better match the downstream processing some post-processing blocks could be added to each stream. In case of the excitation part, the information of interest is the periodicity, so the post-processing block should boost this attribute, paving the way for capturing the pitch frequency. Simple yet effective approaches are the autocorrelation function (ACF) and average magnitude difference function (AMDF) [160] which can better highlight the periodicity in the $\tau_{Exc}(\omega)$ for the voiced sections

$$\begin{aligned}\hat{\tau}_{Exc}(\omega) &= ACF\{\tau_{Exc}(\omega)\}\\ \hat{\tau}_{Exc}(\omega) &= AMDF\{\tau_{Exc}(\omega)\}\end{aligned} \tag{4.20}$$

where $\hat{\tau}_{Exc}$ denotes the post-processed group delay of the excitation component.

In the case of the vocal tract, the formant tracks are the important piece of the information. In order to further distinguish the formants track from the rest of the spectrum, one may use

the following approach originally proposed in [161] and was used in the modified group delay [40]

$$\begin{cases} \hat{\tau}_{VT}(\omega) = sign(\tau_{VT}(\omega)) \, |\tau_{VT}(\omega)|^{\alpha} \\ \\ sign(\tau_{VT}(\omega)) = \frac{\tau_{VT}(\omega)}{|\tau_{VT}(\omega)|} \end{cases} \tag{4.21}$$

where $\hat{\tau}_{VT}$, α and *sign* denote the post-processed group delay of the vocal tract component, the parameter of the transform and the sign function, respectively. This transform is an extension of the power transformation to the variables which may take negative values. Power transformation elevates the flexibility of the framework and allows for boosting the formants as well as adjusting the statistical properties of the representation for optimal performance. The statistical properties of the power transformation is investigated in this chapter and in Appendix B.

Figure 4.18 and Figure 4.19 show the effect of post-processing the source and filter components of the phase spectrum using the aforementioned techniques. As can be seen, the autocorrelation highlights the periodicity and the power transformation further boosts and distinguishes the formant tracks.

4.4 Evaluation of Usefulness of Phase Filter Component

So far the effectiveness of the suggested technique has been evaluated through spectrograms of the source and filter components in the clean and noisy conditions. It was observed that, the proposed method in comparison with its magnitude-based alternative affords higher spectral resolution, lower spectral leakage and higher noise robustness. For a more systematic investigation some features from the filter component of the phase spectrum are extracted and tested in ASR experiments. The performance is compared with the well-known magnitude and phase-based features. Experiments start with Aurora-2, a connected-digit task which makes it easier to study the effect of different parameters and techniques. After examining the effect of these factors, the optimal parameter set is used in the Aurora-4 continuous speech recognition task in clean and multi-style (a.k.a. multi-condition) training modes. The usefulness of the phase source component will be studied in the fundamental frequency estimation in the last section of this chapter.

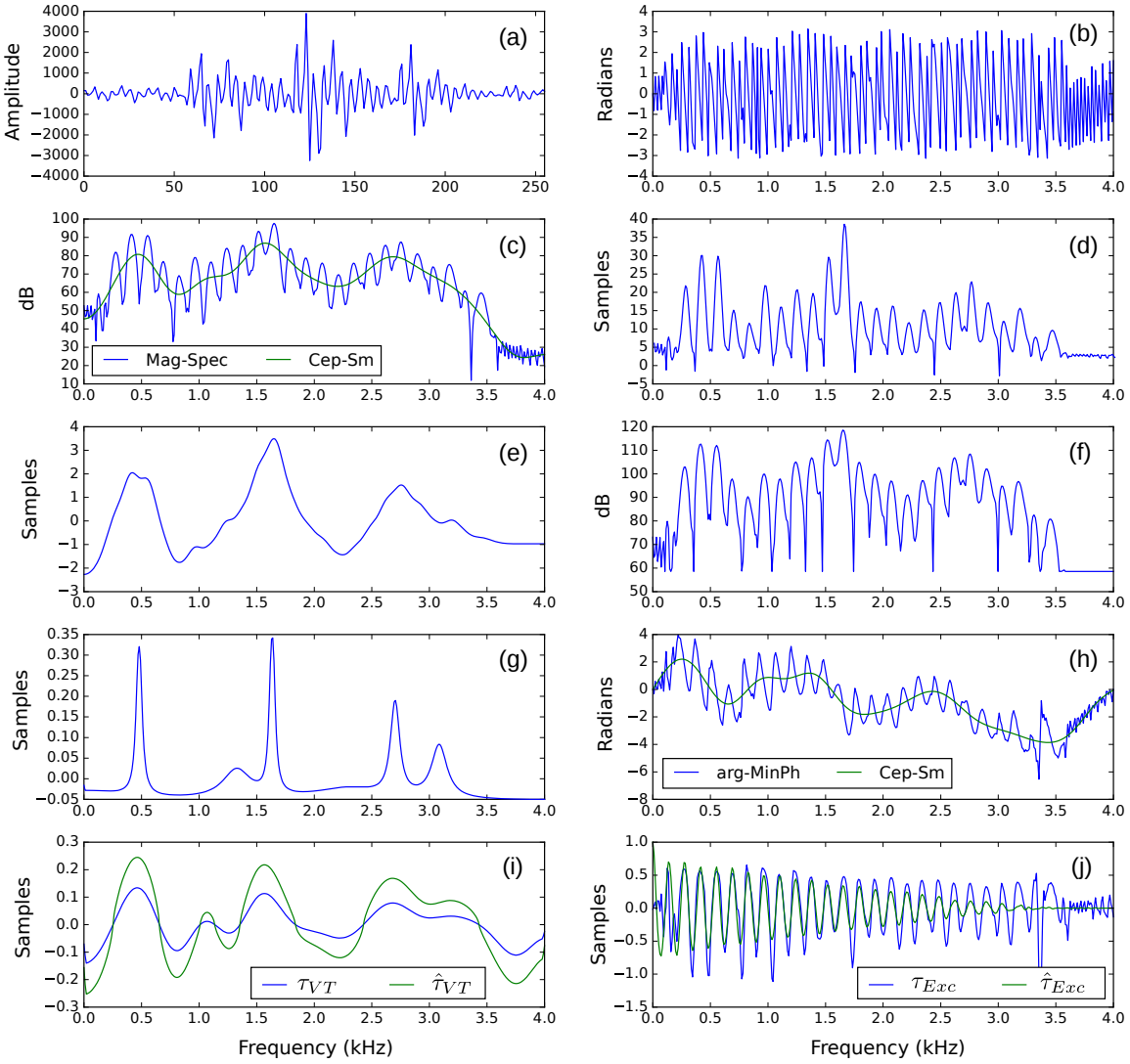

Fig. 4.18 Different representations of a speech signal. (a) waveform (length: 32 ms, sampling frequency: 8 kHz, x-axis label: Time in Samples), (b) principle (wrapped) phase spectrum ($ARG\{X(\omega)\}$), (c) magnitude spectrum (Mag-Spec) and its cepstrally smoothed (Cep-Sm) version, (d) modified group delay function [40] ($\alpha = 0.3$, γ=0.9), (e) chirp group delay function [43] ($\rho = 1.12$), (f) product spectrum [42], (g) group delay of the all-pole model (order 13), (h) phase spectrum of the minimum-phase component computed using causal liftering (Figure 4.13) and its cepstrally smoothed (Cep-Sm) version, (i) $\tau_{VT}(\omega)$: group delay of the Filter component, $\hat{\tau}_{VT}(\omega)$: $\tau_{VT}(\omega)$ after post-processing through (4.21), (j) $\tau_{Exc}(\omega)$: group delay of the Source component, $\hat{\tau}_{Exc}(\omega)$: $\tau_{Exc}(\omega)$ after post-processing through ACF.

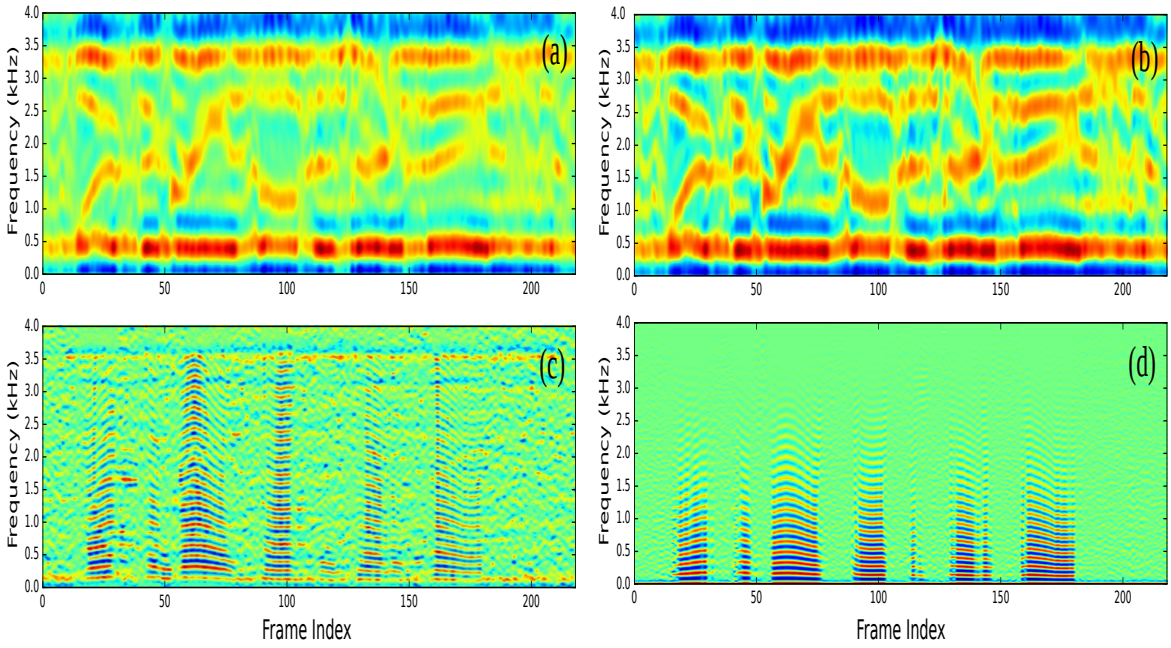

Fig. 4.19 Effect of the proposed post-processing on the phase-based source and filter components. (a) filter component before post-processing, (b) filter component after post-processing, (c) source component before post-processing, (d) source component after post-processing using ACF.

4.4.1 Feature Extraction from Phase Filter Component for ASR

For evaluating the proposed filter component of the phase spectrum one can turn this component into features and test it in ASR. Generally speaking, a good feature extraction algorithm should

- capture/enhance aspects of the signal/data relevant to the task,
- filter out the irrelevant information encoded in the signal/data,
- be as compact as possible (in terms of the length of the feature vector),
- conform to the assumptions made by the back-end.

The lingual content is generally assumed to be correlated with the envelope of the magnitude spectrum which is related to the filter component. Also the envelope is usually warped in a way that low frequencies get a higher spectral resolution and high-frequency components get a lower resolution. The warped envelope is usually passed through a non-linearity for adjusting the dynamic range of the features which statistically pushes the distribution closer to the Gaussian density. The next step is decorrelation which makes the data better match the

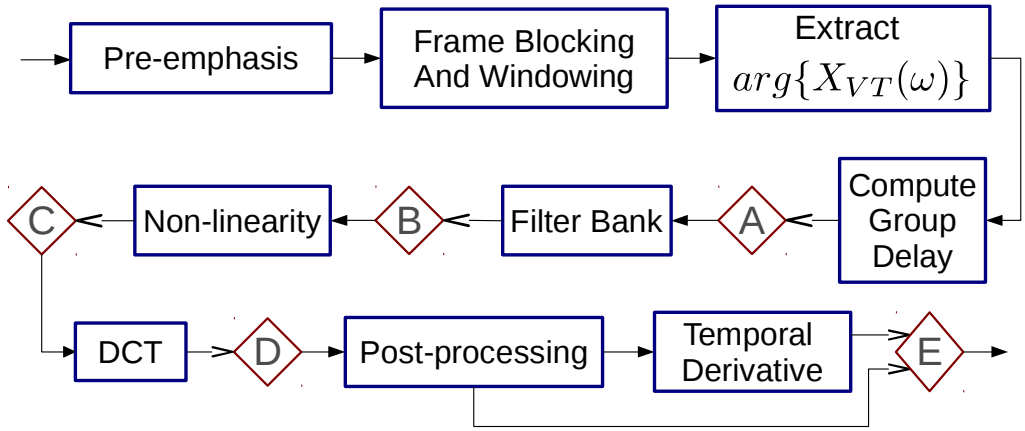

Fig. 4.20 Block diagram of the proposed feature extraction from the phase spectrum. A-E denote points in which later some statistical normalisation may be applied.

diagonal covariance matrices utilised in the GMM/HMM systems. Finally, the feature vector is post-processed, for example using cepstral mean normalisation (CMN), and appended by delta (dynamic) features.

Since the the group delay resembles the magnitude spectrum, it can be turned into features for ASR following the step described above. In this regard, the following approaches were examined for turning the phase filter component into a feature for ASR

i) $arg\{X_{VT}\} \rightarrow DCT \Rightarrow PHVT$

ii) $\hat{\tau}_{VT} \rightarrow DCT \Rightarrow GDVT$

iii) $\hat{\tau}_{VT} \rightarrow Mel\ Filter\ bank \rightarrow DCT \Rightarrow MFGDVT$

iv) $\hat{\tau}_{VT} \rightarrow Mel\ Filter\ bank \rightarrow Boost\ (post\ processing) \rightarrow DCT \Rightarrow BMFGDVT$

where the rightmost name is the name assigned to each feature derived from the filter component of the phase spectrum and *boost* in BMFGDVT means post processing the group delay using (4.21). Figure 4.20 shows the overall workflow of the proposed feature extraction schemes from the filter component.

4.4.2 Parametrisation and ASR Setup

Now, we run the first set of the ASR experiments. For other tested features, the default parameters reported in the respective publications were used. Appendix F explains the details of the parametrisation process for the features utilised here. Frame length, frame shift and number of filters were set to 25 ms and 10 ms and 23, respectively. Feature vectors have been

augmented by log-energy as well as by delta and delta-delta coefficients. CMN is performed for all the features on the utterance level. In the proposed method, the α coefficient (4.21) was set to 0.7 and a brick-wall low-pass filter with 20 taps was used for extracting the Trend. Aurora-2 [10] has been used as the database and the HMMs were trained by employing only the clean data using HTK [162] based on the Aurora-2 standard simple recipe. For a detailed description of the Aurora-2 database please refer to Appendix E. Recognition results (average 0-20 dB) achieved when using the various magnitude-based and phase-based features are reported in Table 4.1.

4.4.3 Experimental Results and Discussion

As seen in Table 4.1, the proposed methods for parametrising the phase filter component returns comparable results to the magnitude and other phase-based features. The relative robustness of the proposed method can be explained in part by considering the better distinction of the formants due to higher frequency resolution and less spectral leakage which, in turn, decrease the disorder occurring after SNR reduction in the spectrum (Figure 4.15).

Taking the DCT of the phase spectrum of the filter component ($PHVT$), despite being too simple and coarse for parametrisation, performs relatively well in comparison with features like modified group delay function (MODGDF) and chirp group delay function (CGDF). Decorrelating the group delay using DCT ($GDVT$) improves the performance to some extent. This could be justified assuming that the speech part varies with a faster pace (with respect to frequency) than the noise, so the derivative (as a high-pass filter) amplifies the contribution of the clean speech relative to the noise. In the case of the channel distortion (test set C), since it is additive in the phase domain and assuming that the channel frequency response is relatively constant, the derivative can attenuate the channel effect and the mismatch it brings about. Applying the Mel filter bank ($MFGDVT$) clearly improves the performance for all the test sets which is in agreement with its wide application in speech parametrisation. Finally, boosting the formants using the power transformation ($BMFGDVT$) with $\alpha = 0.7$ leads to higher robustness in all the test sets.

One possible room for improving the performance of a feature is applying some statistical normalisation which is addressed in the next section.

4.5 Statistical Normalisation of Phase Filter-based Features

Variability in the data representation due to nuisance factors is a significant issue in pattern recognition posing considerably more difficulties for the back-end in mapping the features

Table 4.1 *Average (0-20 dB) recognition rates (accuracy=100-WER, in %) for Aurora-2 [10]. For more detail about each feature please refer to Appendix F.*

Feature	Test Set A	Test Set B	Test Set C
MFCC	66.5	71.7	65.3
PLP	67.3	70.6	66.2
MODGDF	64.3	66.4	59.5
CGDF	67.0	73.0	59.4
PS	66.0	71.2	64.6
ARGDMF	75.4	79.0	76.0
PHVT	69.0	74.8	67.1
GDVT	70.5	75.9	69.1
MFGDVT	72.8	77.3	72.8
BMFGDVT	**73.2**	**77.4**	**73.4**

onto the correct class. This problem could be alleviated by developing either a more robust front-end which is less affected by the nuisance factors or a back-end which can better handle such variability. In the front-end, one sensible approach could be applying some knowledge about the properties of the clean data to mitigate the effect of the unwanted disturbances. This entails evaluating the behaviour of the extracted patterns in a noise-free condition and embedding such knowledge into the parametrisation pipeline, in a principled way, to attenuate the deviations induced by noise.

The prior knowledge about the clean data could have a deterministic or statistical basis. For example, flooring the filter bank energies below a pre-set threshold (say 60 dB) is a deterministic approach and histogram equalisation of the features exemplifies the statistical normalisation techniques. The deterministic approach is easy to implement but can not effectively handle the variability problem. On the other hand, the statistical approach is more effective in dealing with the variability issue, although involves higher complexity. It is added to the parametrisation process as a normalisation block aiming at giving the features a *desired* statistical property. Note that the term desired does not necessarily mean clean (free of noise). It essentially means the matched condition with the training data. Since at this stage the ASR models are built using only clean data, the term desired means the (statistical) characteristic of the clean features. Therefore, a good feature transformation should give the noisy feature the properties of its clean counterpart, suppress the noise effect and/or enhance the contribution of the clean part.

A general pattern recognition system, in addition to a front-end, also includes a back-end[10]. In an optimal setup, the feature transformation should consider the properties of the

[10]Excluding the end-to-end systems.

back-end, too. From the back-end perspective, desired features are those which, among others, are in harmony with the assumptions classifier makes about its input. Although any mismatch could be costly performance-wise, transforming the data by only considering the back-end could be problematic, too. In fact, there is the possibility of distorting the features through imposing some properties on them which do not comply with their original structure. As such an optimal normalisation scheme should take both ends into account.

Let us first investigate the behaviour of the phase and its representations from a statistical standpoint in the clean conditions before applying the statistical normalisation techniques. Estimation of the statistical structure of the phase in the clean condition provides a fresh perspective on the behaviour of this spectrum. It also helps in explaining the reason behind the success or failure of each normalisation scheme and paves the way towards finding the best one.

For evaluating the distribution of the speech phase spectrum and its representations, the histograms at various points along the proposed workflow (Figure 4.20) was computed. In order to get statistically significant results, all the clean training data of the Aurora 2 [10] database has been employed which includes 8440 waves yielding about 1.47 million frames ($\approx$ 244 minutes). For comparison, the same process has been carried out for the MFCC features along the pipeline. Number of bins for each histogram was estimated using the Freedman–Diaconis rule [163] which aims to minimise the difference between the area under the empirical probability distribution and the area under the theoretical probability distribution.

4.5.1 Distribution of the Magnitude-based Representations

As shown in Figure 4.21(a), in the clean condition, the distribution of the magnitude spectrum is heavily right-skewed and bears the hallmarks of the Rayleigh density, the assumption which is made in speech enhancement in techniques like MMSE [8]. Taking the logarithm of the filter bank energies (FBE) results in a bimodal distribution. The left mode relates to the low-energy speech and/or silence and the right one is connected to the speech parts with a normal energy level. Applying the logarithm has a significant statistical impact on the FBEs and pushes the distribution toward the Gaussian density by decreasing both skewness and kurtosis. This allows the GMM-based back-end to obtain a much better fit. Another point is that as the index of the filter goes up, the centre of the corresponding distribution moves to the right (increases) and the left mode become more pronounced. Also, the support/spread of the histogram becomes shorter[11].

[11] The less technical terms *centre* and *support/spread* were used instead of the mean and variance because the mean and variance are meaningful for unimodal distributions whereas the distribution here is bimodal.

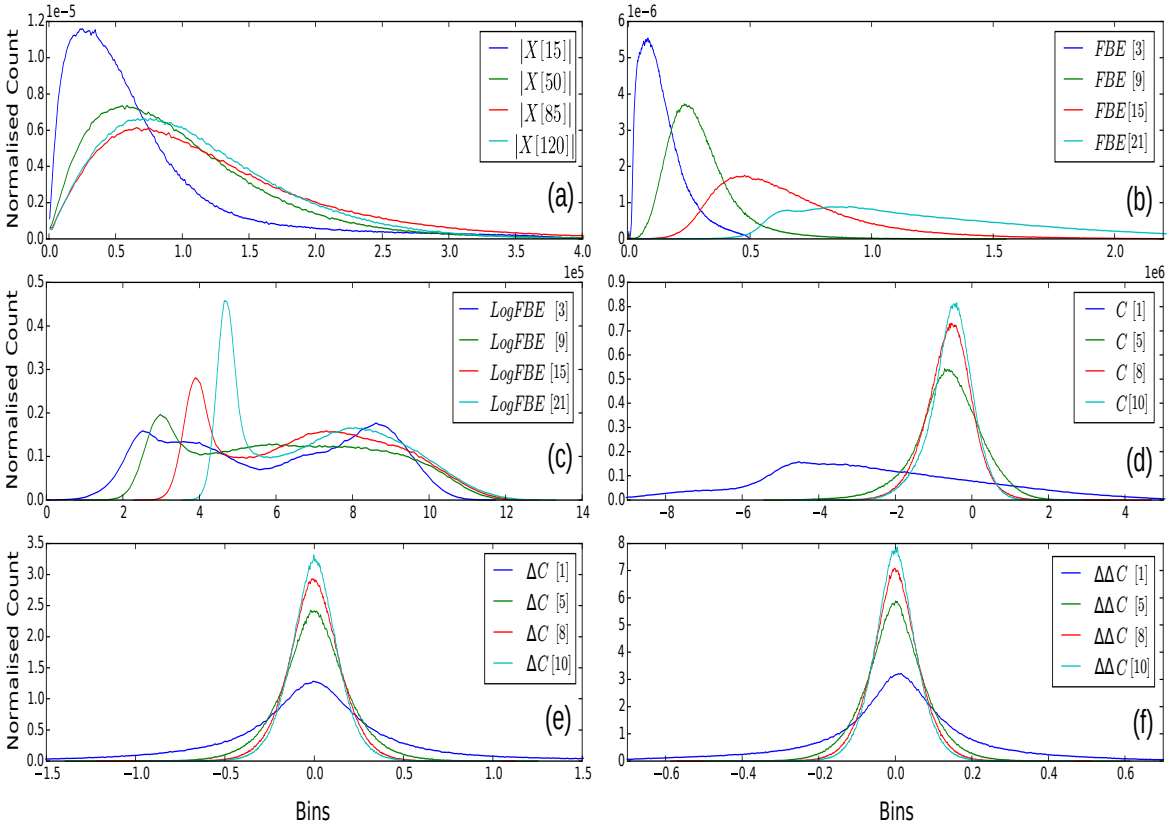

Fig. 4.21 Histograms at different stages of the MFCC pipeline. Histograms of the (a) magnitude spectrum, (b) filter bank energies (FBE), (c) log of the FBEs, (d) DCT of the log of the FBEs, (e) delta coefficients, (f) delta-delta (acceleration) coefficients. Number in the square brackets denotes the index.

Taking the DCT, makes the density uni-modal and pushes it closer to the Gaussian distribution. This could be explained using the central limit theorem (CLT) as the DCT basically acts as a weighted sum. Of course the conditions of the CLT, namely sum of the i.i.d variables is not met entirely and the distribution is not perfectly Gaussian.

4.5.2 Distribution of the Phase-based Representations

Figure 4.22 demonstrates the histograms of the continuous phase and its representations at different points along the parametrisation workflow. In sharp contrast to the uniform assumption usually made about the phase spectrum, especially in the speech enhancement literature [12], $arg\{X_{MinPh}(\omega)\}$ has a bell-shaped distribution (Figure 4.22(a)). Note that the histogram here is computed for the unwrapped phase. On the other hand, Figure 4.23 illustrates the distribution of the principle phase spectrum, $ARG\{X_{MinPh}(\omega)\}$. As can be seen, the uniform density, $U(-\pi,\pi)$, appears to be a reasonable approximation for the distribution

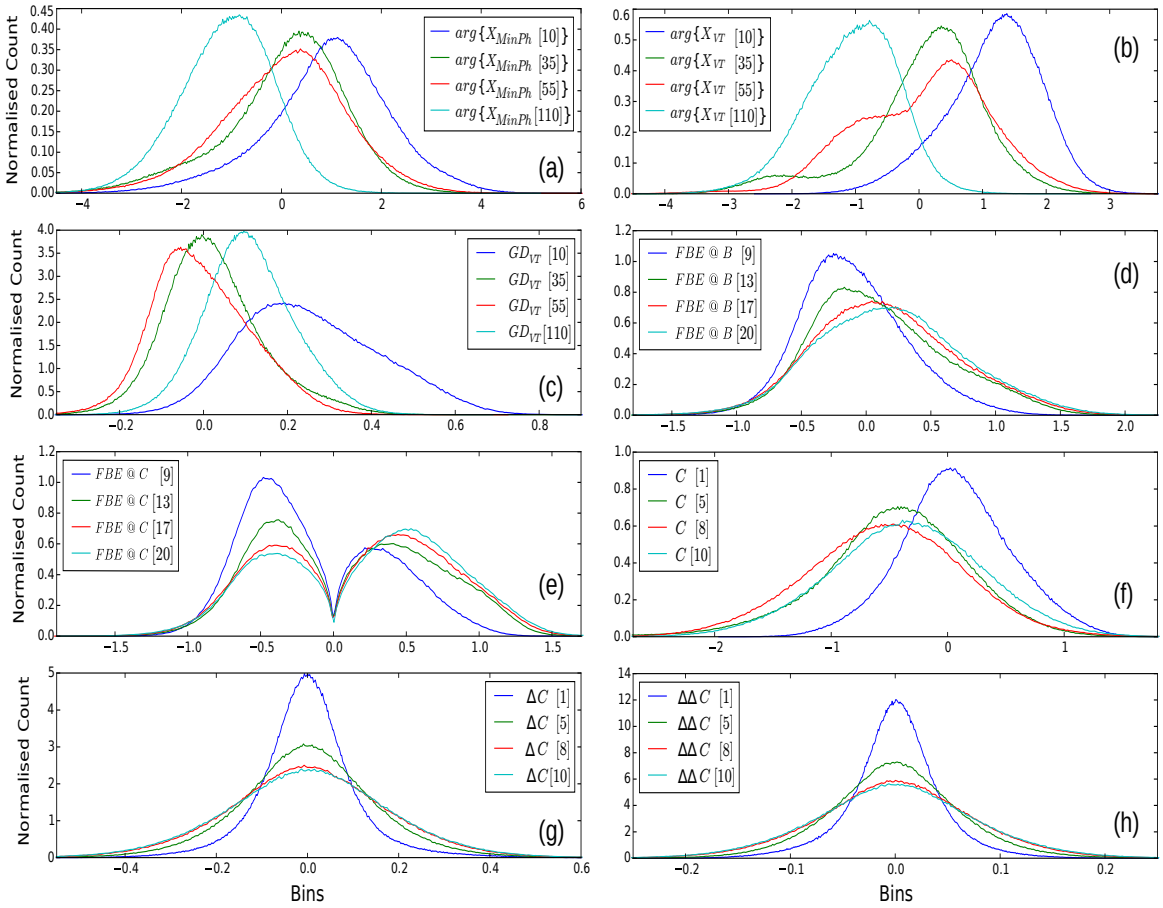

Fig. 4.22 Histograms of the phase spectrum and its representations along the workflow shown in Figure 4.20. Histograms of the (a) unwrapped phase spectrum of the minimum-phase component, (b) unwrapped phase spectrum of the vocal tract component, (c) group delay of the vocal tract component, (d) FBEs of the GD of the vocal tract component (point B in Figure 4.20), (e) FBEs after applying non-linearity (point C in Figure 4.20), (f) DCT of the FBEs after applying non-linearity (point D in Figure 4.20). (g) delta coefficients, (h) delta-delta (acceleration) coefficients. Number in the square brackets denotes the index.

of the principle phase. Comparing these two distributions illustrates that the uniform density is the artefact of the wrapping and is not an inherent property of the phase spectrum.

In order to better clarify this point, first the magnitude spectrum is wrapped similar to the phase spectrum as follows

$$wrapped\{|X(\omega)|\} = (|X(\omega)| + \pi) \; mod \; 2\pi - \pi, \tag{4.22}$$

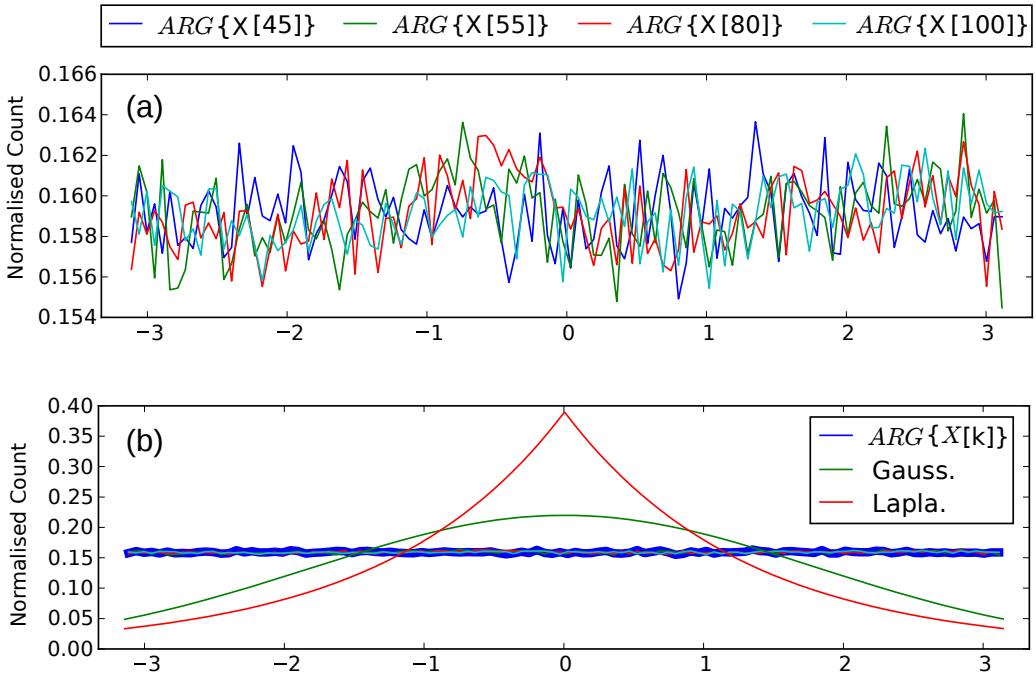

Fig. 4.23 (a) Histogram of the principle (wrapped) phase spectrum ($ARG\{X(\omega)\}$), (b) Histogram of the $ARG\{X(\omega)\}$ versus the Gaussian and Laplacian distributions. Number in the square brackets denotes the frequency bin.

where *mod* denotes modulo operation. Figure 4.24 shows that the distribution of the *wrapped magnitude spectrum* which is $U(-\pi,\pi)$, too. This lends support to the claim that the uniform density is only due to the wrapping.

Second, a data sequence with the uniform distribution over its support has the maximum-level of disorderliness (entropy) [164]. In our empirical study, it means that after making more than 1.4 million observations, still all the possible values which the phase spectrum can take are equiprobable. This implies that there is no structure and the corresponding data is a random informationless sequence. As a matter of fact, making a uniform assumption for the phase density creates two paradoxes: First, as mentioned in Section 2.3.2, under some mild conditions, a signal is recoverable (within a scale error) from its phase spectrum [6]. A non-informative uniformly-distributed sequence should not have such capability. Second, there is an almost one-to-one relationship between the magnitude and phase spectra of a minimum-phase signal. Given that the minimum-phase component is dominant in the short-term analysis coupled with this point implies that the short-term phase and magnitude spectra should carry the same amount of information and are just two mathematical realisations of the same information. The apparent uniform distribution of the phase along with the one-to-one phase/magnitude relation imply that the magnitude spectrum is also devoid of

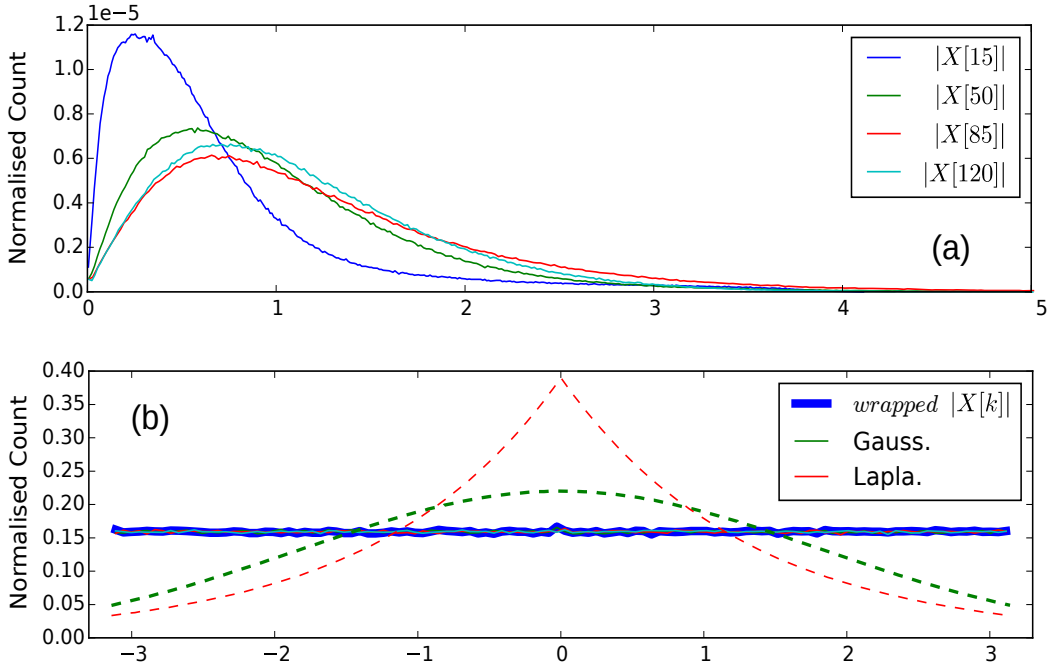

Fig. 4.24 Effect of wrapping on density of magnitude spectrum. (a) histogram of magnitude spectrum, (b) histogram of the wrapped magnitude spectrum. Wrapped magnitude spectrum has a uniform distribution. Number in the square brackets denotes the frequency bin.

information which is obviously incorrect. Figure 4.22(a), however, shows the true distribution and resolves these contradictions. It indicates that the uniform distribution is not a structural property of the phase spectrum but merely a repercussion of the phase wrapping.

After applying the non-linearity on the FBEs at the point C in Figure 4.20 ($FBE\ @\ C$) the distribution becomes bimodal (Figure 4.21(e)) which is similar to the log effect on the density of the FBEs in MFCC pipeline. However, this time the underlying reason is different. In case of the magnitude spectrum, as explained the modes are related to the low and normal energy levels of the speech. Contrary to the magnitude spectrum, phase and its representations are scale-blinded; hence, the energy level has no role to play. The reason here originates from the fact that zero is a *fixed-point*[12] of the power transformation used as a non-linearity ($sign(x)|x|^a$, where x is the $FBE\ @\ B$ in Figure 4.20). Therefore, values very close to zero remain almost identical whereas others move away and this gives rise to bimodality around zero.

Finally, there is a need to investigate where the bell-shaped density of the phase-based features (Figure 4.22(e)) stands in comparison with the Gaussian distribution. Figure 4.25 shows the histograms of the c_2 and c_3, second and third cepstral coefficients. Although the

[12]Point x is a fixed point for $F(x)$ if $F(x) = x$.

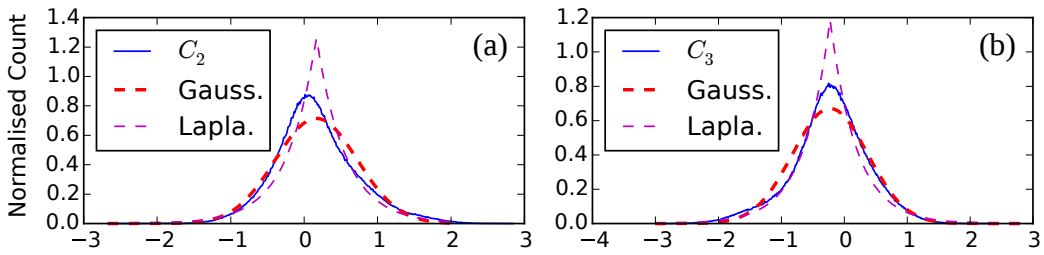

Fig. 4.25 Comparison of histograms of the proposed phase-based feature in the cepstrum domain (C) with Gaussian and Laplacian densities. (a) second cepstral coefficient, C_2, (b) third cepstral coefficient, C_3.

Gaussian assumption could be a fairly reasonable approximation, the distribution of these features has a higher kurtosis[13]. Since the features has a kurtosis higher than the Gaussian, their distribution is called *Leptokurtic* or *super-Gaussian*. The Laplace distribution is an example of such densities [14]. As juxtaposed in Figure 4.25, it forms the upper bound from this perspective and could be regarded as another approximation for the feature's distribution in the clean conditions.

Returning to the statistical normalisation issue, such upper and lower bounds for the features imply that the true distribution is located in between. Although suboptimally, both Gaussianisation and Laplacianisation might be helpful in pushing the noisy phase-based features toward their clean counterpart and improving the robustness. This hypothesis is validated in the end of this section after a brief review of how such statistical normalisations are implemented.

4.5.3 Implementing the Statistical Transformation

The overarching goal of statistical normalisation is to normalise the data such that it statistically follows a target/desired distribution. In this subsection the theory underlying this process is reviewed and the schemes which are used in the experiments, namely Gaussianisation [13], Laplacianisation and histogram equalisation (HEQ) [14] are explained. These techniques are computationally low-cost and neither need noise estimation nor stereo data. Also, contrary to techniques like vector Taylor Series [17], there is no need for an explicit expression showing how the features get contaminated by noise (environment model). The

[13] Kurtosis is the forth order statistics which reflects the peakedness and tailedness of a distribution. For the Gaussian density it is equal to 3. Excess kurtosis equals kurtosis minus 3, so for the normal distribution it is zero. Positive excess kurtosis indicates heavy tails and peakedness relative to the normal distribution, whereas negative kurtosis indicates light tails and flatness [165].

[14] With the excess kurtosis equal to 3

equation which mathematically underpins the statistical normalisation methods is as follows

$$CDF_Y(y) = CDF_X(x) \Rightarrow x = CDF_X^{-1}\left(CDF_Y(y)\right), \tag{4.23}$$

where CDF indicates cumulative distribution function, X and Y are the random variables (rv) associated with the clean and noisy observations, respectively, and x and y are the corresponding realisations. Implementing (4.23) involves finding the quantile function of X, i.e., $CDF_X^{-1}(x)$ as well as the CDF of Y.

If the rv Z is defined as $Z = CDF_Y(Y)$, the Probability Integral Transform (PIT) [166] shows that it follows $U(0,1)$. However, estimating the quantile function of the clean (reference) features, is not straightforward and a closed-form solution is only available for a certain classes of density functions. In practice mostly the numerical techniques are employed. Table-based HEQ [167] is an example of such methods in which this function is estimated from the training data. HEQ does not make any assumption about the target distribution and learns it from the training data, contrary to the Gaussianisation and Laplacianisation. This turns it into a more flexible approach, although robust learning of the histogram and estimating the quantile function are not straightforward. For the Gaussian and Laplace distributions, the closed-form expression for the quantile function exists

$$\begin{cases} Gaussianisation \;\rightarrow\; x_i = \sqrt{2}\, erf^{-1}(2z_i - 1) \\ Laplacianisation \;\rightarrow\; x_i = \begin{cases} ln(2z_i), & z_i < 0.5 \\ -ln(2-2z_i), & z_i \geq 0.5, \end{cases} \\ z_i = \frac{r_i - \beta}{N} \;\; , \; i = 1,\, 2,\, \ldots,\, N \end{cases} \tag{4.24}$$

where erf^{-1}, ln, z, N and r_i denote the inverse error function, natural logarithm, realisation of the rv Z, number of observations and the rank of y_i after ascending sort, respectively. β is used to avoid extreme values and usually set to 0.5 [13]. Note that for mathematical convenience, normalisation is (suboptimally) carried out for each dimension independently. Thus, decorrelating the features before statistical normalisation reduces the error associated with this assumption.

The difference between these techniques and the mean-variance-normalisation (MVN) should be noted: The former affects all the moments while the latter only touches the first- (mean) and second-order (variance) statistics. Changing only the first and second order statistics using a linear transform such as MVN does not change the family to which an rv belongs whereas Gaussianisation, Laplacianisation and HEQ are non-linear transforms which affect the higher-order statistics and reshape the distribution. As such they have a

deeper statistical impact on the features. Note that if the distribution is perfectly Gaussian, the result of MVN and Gaussianisation would be identical (given sufficient amount of data). By the same token, Laplacianisation and MVN lead to the same results if the distribution is Laplacian.

4.5.4 Experimental Results and Discussion

Comparing Table 4.1 with Tables 4.2-4.5 shows that normalisation, in most of the cases, enhances the recognition performance in noisy conditions. The amount of improvement depends on the type of the normalisation and the stage at which it is performed. As seen, point *E* (Figure 4.20), namely just before the back-end, appears to be the best place for applying normalisation. Gaussianisation at this point leads to upto 18.6% relative word error rate reduction (RER) which is a significant gain, considering the low computational overhead involved. MVN seems to be the least helpful approach, however. The remarkable difference between the performance of MVN and Gaussianication/Laplacianisation implies that normalising the higher order statistics is important and also the distance between the true distribution of the features and the Normal or Laplace distributions is significant.

Comparison of Tables 4.3 and 4.4 shows that Gaussianisation is a more effective normalisation scheme than Laplacianisation. Two arguments can be put forward to explain this: First, from the front-end perspective, as depicted in Figure 4.25, although the true distribution of the phase-based feature is super-Gaussian, it is not as Leptokurtic as Laplacian and seems to be closer to Gaussian. As a result, Gaussianisation leads to less distortion than Laplacianisation because it is more consistent with the original statistical structure of the features. Second, from the back-end standpoint, and in comparison with the Laplace density, the GMM-based speech recogniser better fits data with a Gaussian distribution.

Comparing Tables 4.3 and 4.4 with Table 4.5 demonstrates that both Gaussianisation and Laplacianisation return better results than HEQ. Two points should be noted: First, HEQ assumes that the noise-corruption process is a monotonic transform and does not cause any information loss. This demand is not met here due to the random effect of the noise [14]. Second, as shown in Figure 4.25, the true distribution of the phase-based feature is relatively close to both Gaussian and Laplacian distributions (unimodal and almost symmetric). Thus both can approximate it to a reasonable extent and the flexibility of the HEQ is unnecessary.

Figure 4.26 depicts the performance of these techniques (after applying each one at the corresponding optimal point) versus SNR (averaged over all test sets). As can be seen, these methods are especially useful in SNRs below 10 dB and can return absolute accuracy improvement of about 7% and 10% in SNRs of 5 and 0 dB, respectively.

Table 4.2 *Average (0-20 dB) accuracy (in %) after MVN at points $A-E$ in Figure 4.20.*

Feature	A	B	C	Ave. All	RER(%)
MV-A	73.9	77.9	74.4	**75.4**	2.8
MV-B	73.1	76.0	74.2	74.4	-1.2
MV-C	72.1	74.8	73.2	73.4	-5.1
MV-D	64.4	67.4	62.1	64.6	-39.9
MV-E	75.1	77.3	73.3	75.2	2.0

Table 4.3 *Average (0-20 dB) accuracy (in %) after Gaussianisation at points $A-E$ in Figure 4.20.*

Feature	A	B	C	Ave. All	RER(%)
Gaus-A	74.1	78.3	74.4	75.6	3.6
Gaus-B	73.0	76.0	74.1	74.4	-1.9
Gaus-C	74.0	76.7	74.9	75.2	2.0
Gaus-D	78.6	80.2	77.0	78.6	15.4
Gaus-E	79.3	81.0	77.8	**79.4**	18.6

Table 4.4 *Average (0-20 dB) accuracy (in %) after Laplacianisation at points $A-E$ in Figure 4.20.*

Feature	A	B	C	Ave. All	RER(%)
Lap-A	74.4	78.5	74.8	75.9	4.7
Lap-B	73.9	76.7	74.8	75.1	1.6
Lap-C	74.0	76.7	75.2	75.3	2.4
Lap-D	75.5	77.5	74.0	75.7	4.0
Lap-E	77.5	79.3	75.9	**77.6**	11.5

Table 4.5 *Average (0-20 dB) accuracy (in %) after HEQ at points $A-E$ in Figure 4.20.*

Feature	A	B	C	Ave. All	RER(%)
HEQ-A	74.0	78.0	74.9	75.6	3.5
HEQ-B	74.2	78.0	75.2	75.8	4.3
HEQ-C	74.5	78.4	75.4	76.1	5.5
HEQ-D	76.5	78.2	73.5	76.1	5.5
HEQ-E	77.0	78.7	74.9	**76.9**	8.7

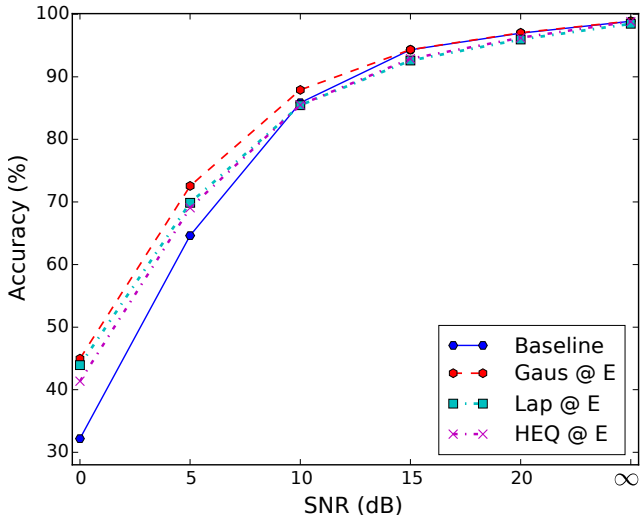

Fig. 4.26 Accuracy (100-WER) versus SNR for different normalisation schemes (Baseline: BMFGDVT, averaged over all test sets).

4.6 Improving the Source-filter Model in the Phase Domain

In this section two modifications are suggested for improving the performance of the proposed source-filter separation model in the phase domain. In the first part, the logarithm in the Hilbert transform is replaced with the generalised logarithmic function (GenLog). In the second subsection, the regression filter is employed for computing the group delay function instead of the sample difference of the phase spectrum.

4.6.1 Replacing Log with Generalised-Log in the Hilbert Transform

Based on the classic definition of the Hilbert transform, $arg\{X_{MinPh}|\}$ is computed through (4.13). Here, we modify this and instead of log, utilise the generalised logarithmic function ($GenLog$) [168]

$$\begin{cases} GenLog(x;\alpha) = \frac{1}{\alpha}(x^{\alpha}-1), \quad x>0 \quad \alpha \neq 0 \\ \lim_{\alpha \to 0} GenLog(x;\alpha) = log(x), \end{cases} \tag{4.25}$$

where α is the parameter of the transform. In the Statistics literature, this function is known as Box-Cox transformation [15]. It unifies the power and log transforms and under optimal adjustment of α it is shown to be helpful in improving the following

- Linearity: achieving a better fit in regression
- Homoscedasticity: stabilising the variance and reducing the heteroscedasticity

Table 4.6 *Effect of swapping the order of the filter bank and the logarithm in the MFCC features (accuracy in % for Aurora-2).*

Feature	Clean	TestSet A	TestSet B	TestSet C
MFCC	99.09	66.5	71.7	65.3
MFCC*	98.22	54.0	64.8	47.7

MFCC*: Order of the filter bank and non-linearity is swapped.

- Gaussianity[15]: pushing the distribution closer to the Gaussian density (an example for this is the effect of the log on the FBE in Figure 4.21(c)).

In fact, $GenLog(x;\alpha)$ provides one degree of freedom that allows two main properties of the representation to be adjusted, namely its dynamic range (DR) and statistical distribution. Figure 4.27 shows that by increasing α, the dynamic range of the representation gets larger. The point which should be underlined here is that for the magnitude-based features, the power spectrum which has a high dynamic range is fed into the filter bank and then the compression is carried out through power transformation (*log* is its special case). As Table 4.6 shows, if in the MFCC pipeline the order of the compression and filter bank is swapped (denoted by $MFCC*$ in Table 4.6), the performance will degrade sharply. However, in the case of the proposed phase-based feature, $\tau_{VT}(\omega)$ which has a limited dynamic range (comparable to $log|X(\omega)|$), enters the filter bank. Similar to the magnitude-based features, this could be costly performance-wise. So, the DR of the group delay should be increased before passing it through the filter bank.

On the contrary to the magnitude spectrum, the dynamic range of the GD is not related to the signal energy level at different bins[16]. It depends on the relative location of the poles/zeros with respect to the unit circle. As explained in Section 2.2.5, zeros or poles located in the vicinity of the unit circle increase the dynamic range of group delay and make it too spiky if left uncontrolled. In case of the speech signal, the poles which are mainly linked to the vocal tract component are far enough from the unit circle, so they are not the cause of the spikiness of the group delay. However, zeros primarily associated with the excitation component, are placed in the vicinity of the unit circle and give rise to this issue, if left uncontrolled. By removing the source part, the spikiness problem is greatly alleviated but the dynamic range of the group delay is significantly reduced, too. Tuning α allows the dynamic range of the $\tau_{VT}(\omega)$ to be adjusted without affecting the spikiness, as can be seen in Figure 4.27.

Another advantage of using the *GenLog* relates to the noisy condition where contamination with noise results in a flattening of the spectrum and dynamic range reduction. Increasing

[15]It should not be confused with Gaussianisation [13].

[16]Group delay is a phase-based representation and is scale-blind, too.

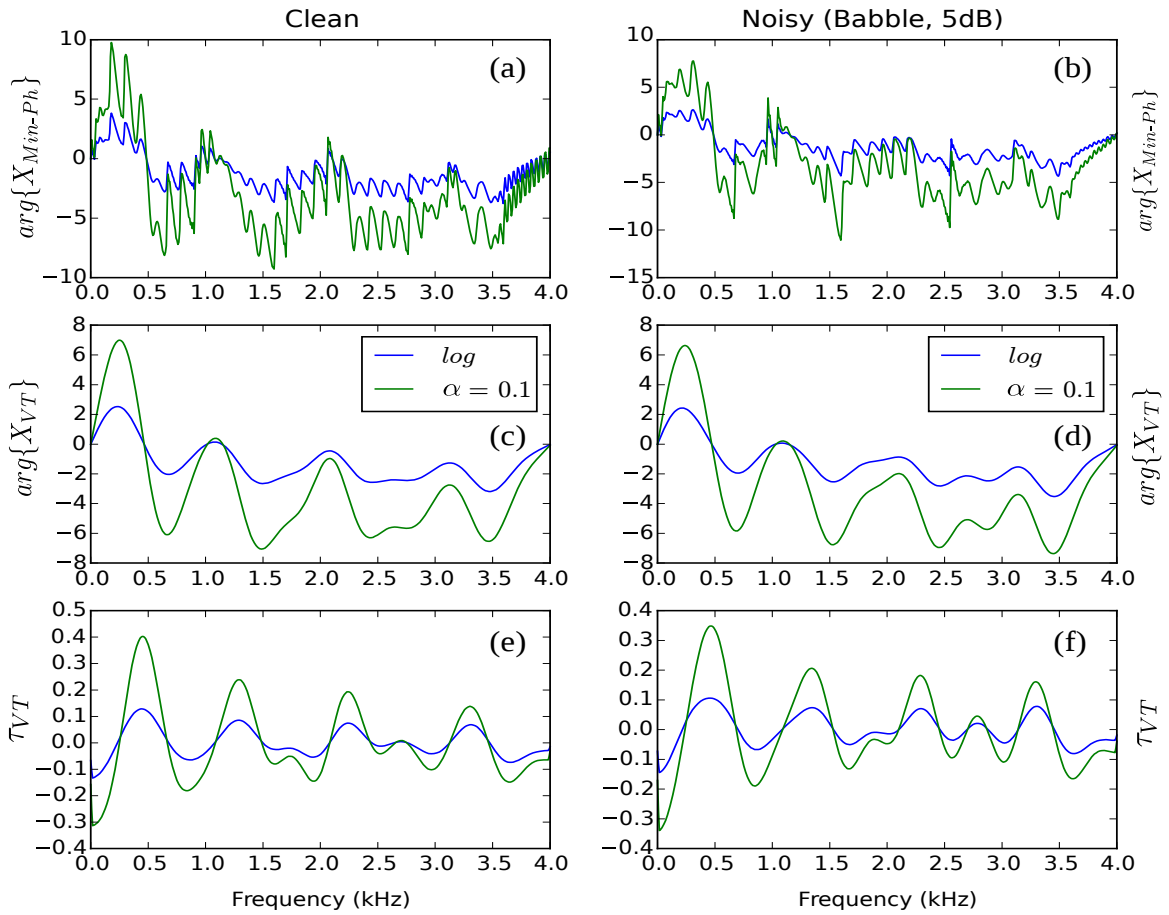

Fig. 4.27 Effect of substituting $log(|X(\omega)|)$ with the $GenLog(|X(\omega)|)$ in the Hilbert Transform at the clean and noisy (Babble, 5 dB) conditions. (a) $arg\{X_{MinPh}\}$-clean, (b) $arg\{X_{MinPh}\}$-noisy, (c) $arg\{X_{VT}\}$-clean, (d) $arg\{X_{VT}\}$-noisy (e), τ_{VT}-clean, (f) τ_{VT}-noisy.

α counters this effect of the noise and consequently improves the robustness. The effect of applying the GenLog function in the noisy condition can be explained in other way, too. Let us define the SNR after applying $GenLog(\alpha)$ as $\frac{X^\alpha}{W^\alpha}$ where X and W denotes the periodograms of the clean signal and the additive noise, respectively. Now, assuming that the clean signal is stronger than noise, namely $\frac{X}{W} > 1$, it follows that increasing α leads to an increase in SNR. Recall that using log is a special case where $\alpha \rightarrow 0$. Thus, using a power transformation instead of log in the Hilbert transform, also enhances the relative contribution

of the clean signal with respect to the additive noise

$$\begin{cases} SNR_1 = \frac{X^{\alpha_1}}{W^{\alpha_1}} \\ SNR_2 = \frac{X^{\alpha_2}}{W^{\alpha_2}} \\ \frac{X}{W} > 1 \\ \alpha_2 > \alpha_1 \end{cases} \quad \Rightarrow \quad SNR_2 > SNR_1 \tag{4.26}$$

The larger the α, the greater the gain in the SNR which leads to a more robust phase spectrum estimate. A natural question at this point is how much gain could be achieved through increasing α? Is there an upper bound on this parameter?

Let us rewrite (4.14) using the *GenLog* function (4.25)

$$\begin{aligned} arg\{X_{MinPh}(\omega);\alpha\} &= -\frac{1}{2\pi} GenLog\{|X(\omega)|\} * cot(\frac{\omega}{2}) \\ &= -\frac{1}{2\pi} GenLog\{|X_{VT}(\omega)|\,|X_{Exc}(\omega)|\} * cot(\frac{\omega}{2}) \\ &= -\frac{1}{2\pi\alpha} |X_{VT}(\omega)|^{\alpha}\,|X_{Exc}(\omega)|^{\alpha} * cot(\frac{\omega}{2}) - \frac{1}{2\pi\alpha} \underbrace{1 * cot(\frac{\omega}{2})}_{0} \\ &= -\frac{1}{2\pi\alpha} |X_{VT}(\omega)|^{\alpha}\,|X_{Exc}(\omega)|^{\alpha} * cot(\frac{\omega}{2}). \end{aligned} \tag{4.27}$$

Although the *GenLog* function brings flexibility and robustness to the framework, based on (4.27), it poses a substantial problem: the useful additive relationship between the source and filter resulting from the log function in (4.14) is replaced with some kind of multiplication (4.27). This could hinder source and filter separation because the Trend-*plus*-Fluctuation premise (4.15) is undermined.

However, as can be seen in Figure 4.27(a), as long as α is set to a sufficiently small value, e.g. 0.1, the Trend and Fluctuation remain *quasi-additive*. While given (4.27) this may seem counter-intuitive, *Maclaurin series* expansion of the function $f(\alpha) = z^{\alpha}$, where $z = |X_{VT}(\omega)|\,|X_{Exc}(\omega)|$, shows the reason

$$\begin{aligned} f(\alpha) &= 1 + \alpha\, logz + \alpha^2 (log\, z)^2 + \alpha^3 (log\, z)^3 + ... \\ &\approx 1 + \alpha\, logz = 1 + \alpha\,(log|X_{VT}(\omega)| + log|X_{Exc}(\omega)|). \end{aligned} \tag{4.28}$$

As long as $\alpha \ll 1$, non-linear terms in (4.28) remain negligible and the Trend-*plus*-Fluctuation assumption stays reasonable. Note that (4.28) can be also used for proving (4.25). Therefore, α should be set large enough to supply sufficient dynamic range and SNR gain but at the

same time small enough to avoid the violation of the quasi-additive combination of the source and filter components.

4.6.2 Computing the Group Delay through Regression Filter

The group delay, $\tau_X(\omega)$, is defined as the negative spectral derivative of $arg\{X(\omega)\}$. Advantages of the group delay, namely high spectral resolution and low leakage were reviewed and it was shown that its spikiness is a major problem when working with this function. Also the main solutions proposed for alleviating the spikiness problem, namely Cepstral smoothing [38], chirp processing [43] and signal modelling [132, 117] were explained in Section 2.2.6.

Another advantage which makes the group delay a special representation for the phase spectrum is that, if its spikiness issue is resolved for the minimum-phase signals, it will resemble the magnitude spectrum. That is, it has peaks at poles and valleys at zeros [17]. As a result, a wide range of the magnitude-based methods can be employed to process this phase-based representation. While the bulk of GD-related research is concerned with circumventing the spikiness (which is important from application point of view), an important theoretical question is overlooked: why does group delay bear a resemblance to the magnitude spectrum? In other words, among all the possible mathematical representations for the (unwrapped) phase spectrum, what is special about its derivative which renders the foregoing useful similarity?

Usefulness of the Group Delay

The similarity between the group delay and the magnitude spectrum stems from the way in which information is encoded in the phase spectrum. Contrary to the magnitude spectrum where information is distributed in the amplitude values, in the phase domain it resides in the level-crossing structure. For the sake of argument let us consider a simple single-pole ($z_p = r_p e^{j\theta_p}$) function where information means r_p and θ_p. For the magnitude spectrum, the frequency bin in which the maximum takes place gives the θ_p and the corresponding amplitude value determines r_p. In the phase domain, however, the bin at which downward zero-crossing occurs yields θ_p and the slope at that point gives r_p.

Loosely speaking, in the magnitude spectrum the information appears in an amplitude modulation (AM) format while for the phase spectrum the format looks like frequency modulation (FM) where information gets encoded in the slopes rather than the amplitude values. By computing the derivative, similar to FM demodulation through *discriminator*

[17] As we showed in Chapter 2, Figure 2.3, for the maximum-phase signals group delay would behave opposite, namely peaks at zeros, and has valley at poles.

(aka *slope detector*) [169], the information would be demodulated and moved into the amplitude domain. This pushes the overall structure of the group delay toward an AM signal, similar to the magnitude spectrum, and consequently facilitates the interpretation as well as processing. This argument can also explain the relatively higher noise robustness of the phase in comparison with the magnitude spectrum (e.g. in Figure 4.15) as the frequency modulation is less sensitive to the noise than the amplitude modulation.

To further clarify the aforementioned idea regarding the similarity of the group delay and the magnitude spectrum, let us compute the derivative of the phase of the minim-phase part of the signal to see how the group delay and the log of the magnitude spectrum relates to each other. Mathematically, it runs as follows

$$\begin{aligned}
\arg\{X_{MinPh}(\omega)\} &= -\frac{1}{2\pi}\, log|X(\omega)| * cot(\frac{\omega}{2}) \\
\tau_X(\omega) &= -\frac{d}{d\omega}\arg\{X_{MinPh}(\omega)\} = \frac{1}{2\pi}\,\frac{d}{d\omega}\{log|X(\omega)| * cot(\frac{\omega}{2})\} \\
&= -\frac{1}{4\pi} log|X(\omega)| * (1 + cot^2(\frac{\omega}{2})) \\
&= -\frac{1}{4\pi} log|X(\omega)| * 1 - \frac{1}{4\pi} log|X(\omega)| * cot^2(\frac{\omega}{2}) \\
&= -\frac{1}{2}\,\tilde{x}[0] - \frac{1}{4\pi}\, log|X(\omega)| * cot^2(\frac{\omega}{2})
\end{aligned} \tag{4.29}$$

As seen, a linear transform of the $log|X(\omega)|$ yields the phase spectrum, but for the group delay an affine transform is required to compute the group delay from the log of the magnitude spectrum. What is more important is the interpretation of the convolution with $cot^2(\frac{\omega}{2})$. Impulse or Dirac function (δ) is the neutral element under (linear) convolution, namely $x * \delta = x$. As seen in Figure 4.28(a), $cot^2(\frac{\omega}{2})$ behaves approximately like the Dirac function and the higher the FFT size [18], the closer the behaviour to the Delta function. This corroborates the argument we put forward earlier about the role of the derivative in demodulation of the phase spectrum and the link it creates between the group delay and magnitude spectrum.

Computing the Derivative Using Regression Filter

Since the phase of the DFT is a discrete sequence, numerical differentiation is typically approximated by a finite difference (diff). The 1st-order diff, as is typically used, is intrinsically noisy. We propose to fit a line to a short spectral interval around each bin and take the negative of the slope as the group delay. This is called *regression filter* [16] and

[18] FFT size = $C_{N_{FFT}}\ 2^{nextpow2(N)}$ where $C_{N_{FFT}}$ is a natural number and $nextpow2(N)$ returns the smallest power of two that is greater than or equal to N and N is the number of samples of the $x[n]$.

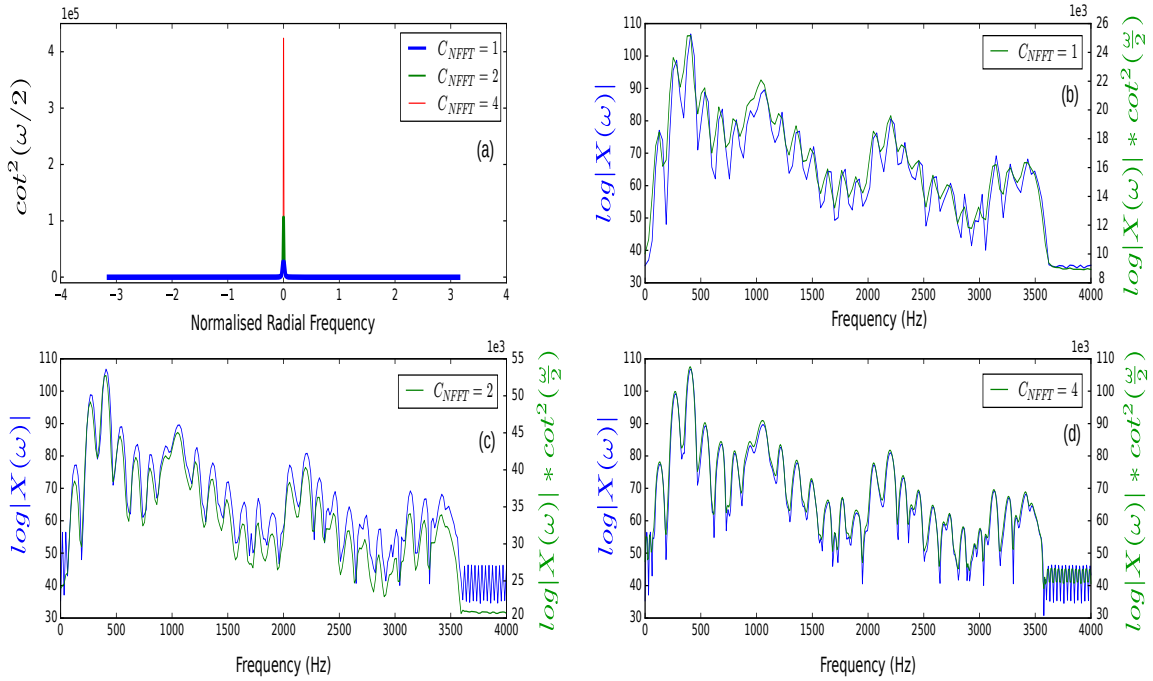

Fig. 4.28 $cot^2(\frac{\omega}{2})$ behaves approximately similar to the Dirac function in the sense of being infinite at zero and almost zero at other places. (a) $cot^2(\frac{\omega}{2})$ for different FFT sizes ($C_{N_{FFT}}\ 2^{nextpow2(N)}$), (b) $log|X(\omega)| * cot^2(\frac{\omega}{2})$ for $C_{N_{FFT}} = 1$, (c) $log|X(\omega)| * cot^2(\frac{\omega}{2})$ for $C_{N_{FFT}} = 2$, (d) $log|X(\omega)| * cot^2(\frac{\omega}{2})$ for $C_{N_{FFT}} = 4$.

mathematically runs as follows

$$\tau_X[k] = -\frac{d\ arg\{X(\omega)\}}{d\ \omega} \approx -\frac{\sum_{m=-k_0}^{k_0} m\ arg\{X[k+m]\}}{\sum_{i=-k_0}^{k_0} i^2}, \tag{4.30}$$

where k denote the discrete frequency and the group delay, respectively, and $2k_0+1$ is the length of the context in frames. The regression filter has a bandpass frequency response, contrary to the sample difference which acts like a high-pass filter. Figure 4.29 shows the frequency response of the regression filter for different values of k_0 and also versus using the sample difference. Increasing k_0 lowers the high cut-off frequency of the bandpass filter and smooths the τ_X. Too large k_0 leads to over-smoothing and the error accompanied with the linearisation increases. On the other hand, if k_0 becomes too small, the slope of the fitted line would be a less reliable estimate for the derivative.

Figure 4.30 illustrates the effect of the regression filter on the group delay of the filter and source components. As can be seen, the influence of the regression filter on the group delay of vocal tract is limited. This should come as no surprise because the vocal tract component is the output of a low-pass filter (Trend Extraction), and its high-frequency

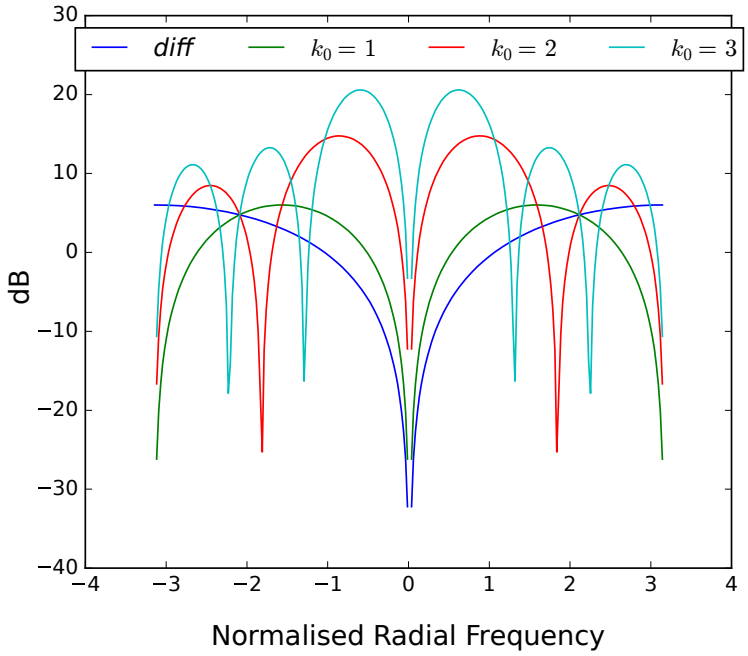

Fig. 4.29 Frequency response of the regression filters for different k_0 values versus sample difference (diff). Diff acts as a high-pass filter whereas regression filter behaves like a bandpass filter, and the larger the k_0 the higher the smoothing.

content is already weak and suppressed. However, using the regression filter is especially effective in processing the excitation component. It makes the source element smoother without inducing extra distortion on its periodic structure which, in turn, facilitates extracting the fundamental frequency from the phase spectrum. The application of the phase spectrum in pitch estimation is further studied at the end of this chapter.

4.6.3 Applying the Non-linearity Function

Applying the GenLog and a regression filter are general modifications and could be integrated into the source-filter separation process regardless of the downstream processing. This part would be specific to the feature extraction from the filter component.

In general, the non-linear compressive function applied to the filter bank energies (FBE) mimics the human auditory system's conversion of the sound pressure into the loudness and is usually implemented through the power transformation (log is its special case). From a machine standpoint, it is important for reshaping the distribution of the features so that their statistical behaviour, better fits the model utilised in the back-end. For the phase-based features, a power transformation cannot be applied directly since the admissible range is restricted to the positive values and the FBEs may become negative when the filter bank is

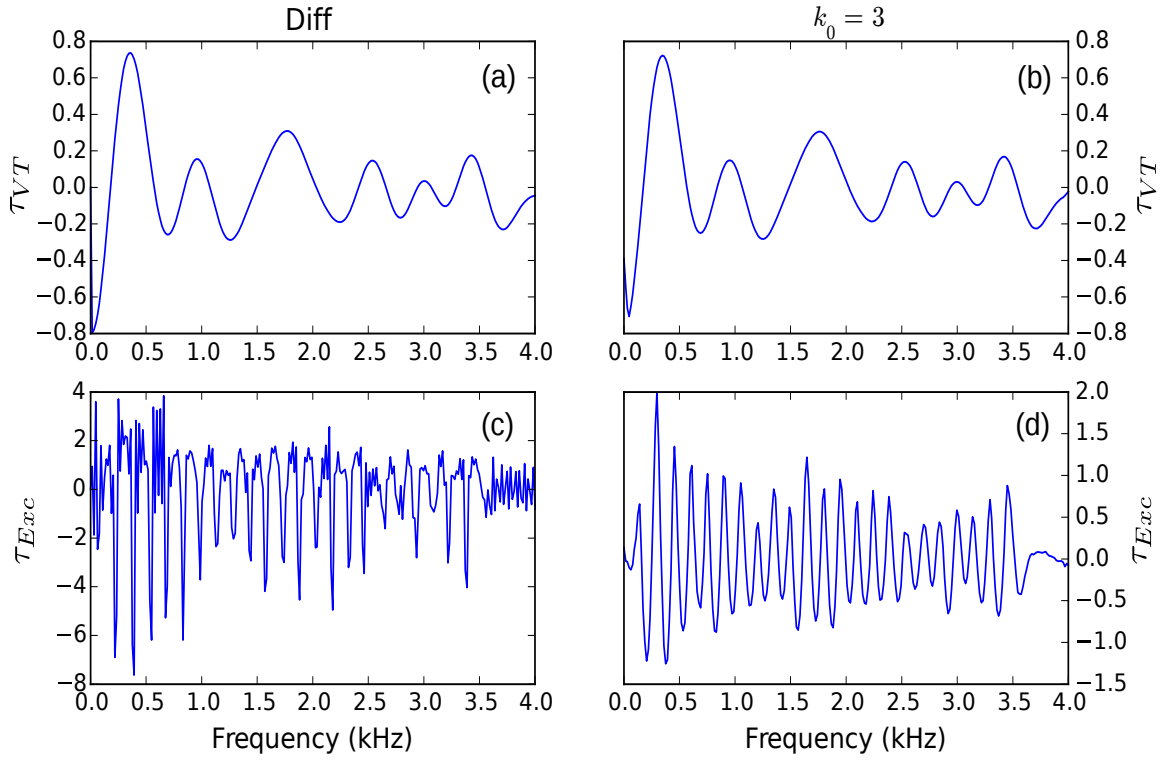

Fig. 4.30 Sample difference (Diff) vs regression filter for computing the group delay of the filter and source components ($\alpha = 0.1$). (a) Group delay of the vocal tract computed using the diff function, $\tau_{VT}[Diff]$, (b) group delay of the vocal tract computed using the regression filter, $\tau_{VT}[k_0 = 3]$, (c) group delay of the excitation component computed using the diff function, $\tau_{Exc}[Diff]$, (d) group delay of the excitation component computed using the regression filter, $\tau_{Exc}[k_0 = 3]$.

fed with the group delay. This, of course, has nothing to do with the concept of the negative energies in Physics [170] and merely relates to the way in which the FBE is computed

$$\begin{aligned} FBE_{|X|^2}[t,l] &= \sum_{k=1}^{N_{FFT}/2} |X[t,k]|^2 \, H^{(l)}[k] \\ FBE_{GD}[t,l] &= \sum_{k=1}^{N_{FFT}/2} \tau_X[t,k] \quad H^{(l)}[k] \end{aligned} \tag{4.31}$$

where t, i, k, N_{FFT}, $H^{(l)}$, $FBE_{|X|^2}$ and FBE_{GD} indicate the frame index, filter index, discrete frequency, FFT size, frequency response of the l^{th} filter of the filter bank, the FBE after applying the power spectrum and the FBE after applying the group delay, respectively. Since the group delay may become negative at some bins, FBE_{GD} can be negative, too.

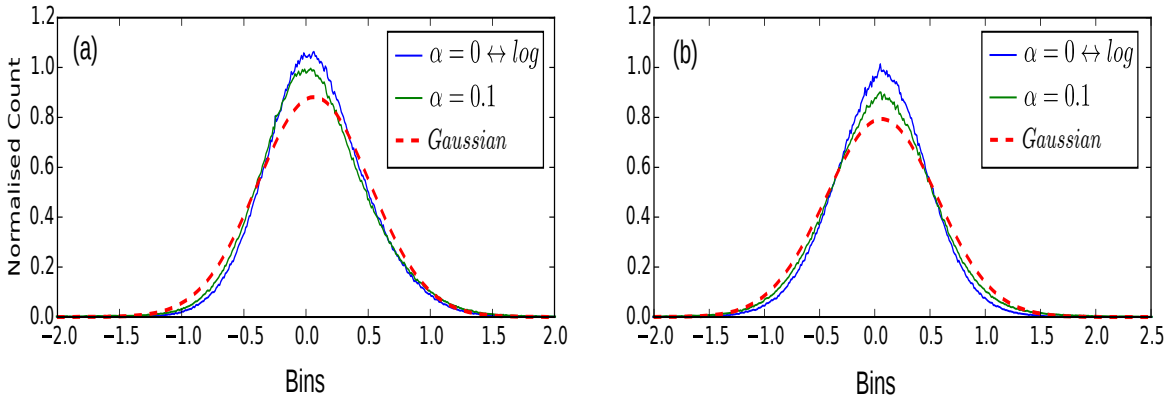

Fig. 4.31 Statistical effect of α on the histogram of the FBE_{GD}. (a) $l = 7$, (b) $l = 15$ (l indicates the filter index of the filter bank).

Bickel and Doksum [161] modified the Box-Cox transform such that it could also operate on the negative values using

$$y = \frac{sign(x)|x|^{\gamma} - 1}{\gamma}, \tag{4.32}$$

where $sign(\,.\,)$, $|\,.\,|$ and γ are the signum function, absolute value and the parameter of the transform, respectively. The -1 from the numerator and γ from the denominator may be removed without loss of generality as they do not affect the class discriminability of the features. That is why in [40, 41] and (4.21) only $sign(x)|x|^{\gamma}$ has been used.

Comparing (4.32) and (4.25) shows both α and γ have similar statistical functionality. As such keeping both is redundant and one of them may be eliminated. Based on the argument propounded regarding the requirement to increase the dynamic range of the group delay before passing it to the filter bank, we omit the non-linearity block placed after the filter bank (or integrate it into the *GenLog*). Figure 4.31 illustrates the effect of the α on the distribution of the FBE_{GD}. As seen, using GenLog with $\alpha = 0.1$ instead of the log in the Hilbert transforms, as well as providing a larger dynamic range and better noise robustness, helps in enhancing the Gaussianity of the features, too. Another point which is worth mentioning is that applying the non-linearity after the filter bank leads to bimodality (Figure 4.22(e)) whereas carrying non-linearity out before the filter bank not only does not result in bimodality but also boosts the Gaussianity.

4.6.4 Experimental Results and Discussion

Connected-digit Task: Aurora-2

For conducting a fair comparison, the effect of replacing *log* with *GenLog* in the MFCC pipeline which leads to generalised-MFCC [171][19], was also evaluated. It is denoted by γ-MFCC in Table 4.7 and results in a significant performance improvement in the noisy conditions, although in the clean condition *log* is a better option. Choosing an appropriate value for γ plays a key role and on average, 0.075 turned out to be an optimal choice for working with the magnitude spectrum in the noisy condition (for Aurora-2). Another advantage of the power transformation is that contrary to the log function it does not need any flooring. In case of the *log*, small FBEs could turn into very big negative values which is troublesome. Finding a proper threshold for flooring is problematic in practice as its optimum value depends on SNR level. However, in the case of using a power transformation the minimum value would be zero; hence, there is no need to set any parameter for this purpose.

In the proposed feature, α and k_0 should be adjusted for optimal performance. Table 4.7 shows that the optimum value for α is around 0.1 (α-BMFGDVT-0.1) and it provides a significant WER reduction and robustness improvement compared with the previous version which was applying the log function (BMFGDVT). This gain is achieved with no computational overhead except for tuning the α a priori. It can be explained by considering the robustness of the phase-based representation (Figure 4.15), tuning the dynamic range through α in the optimal stage along the pipeline and the statistical influence of the α in terms of boosting the normality (Figure 4.31).

The effect of using the regression filter for computing the group delay is shown in the last part of Table 4.7. Here, α is fixed to 0.1. Setting k_0 to 2 or 3 provides optimal results although, as illustrated earlier (Figure 4.30), it has a limited influence on the filter component. In comparison with employing the sample difference, it still affords some small advantage. In general, compared with other phase-based features, the proposed filter-based representation shows a superior performance. Figure 4.32 illustrates the WER versus SNR and demonstrates that the advantage of the proposed modifications is greatest in SNRs below 15 dB.

Aurora-4 along with DNN Setup

In order to further investigate the capabilities of the proposed parametrisation scheme, the Aurora-4 database was also employed. First, HMMs were trained using only the clean data.

[19]The generalised-MFCC [171] feature as well as replacing the log with *GenLog*, includes another modification which is warping the Fourier transform through using first-order all-pass filter ($\frac{e^{-j\omega}-\beta^*}{1-\beta e^{-j\omega}}$) instead of $e^{-j\omega}$ in the Fourier transform definition. In our implementation only the GenLog modification is imposed.

Table 4.7 *WER (average 0-20 dB in %) for Aurora-2 [10].*

Feature	α	k_0	Test Set A	Test Set B	Test Set C
MFCC	log	-	32.7	27.5	34.0
γ-MFCC	0.05	-	25.5	23.3	24.0
γ-MFCC	0.075	-	24.6	23.8	23.1
γ-MFCC	0.1	-	26.7	25.7	25.5
PLP	0.333	-	32.7	29.4	33.8
MODGDF	0.3	-	35.7	33.6	40.5
CGDF	-	-	33.0	27.0	40.6
PS	log	-	34.0	28.8	35.4
ARGDMF	-	-	24.6	21.0	24.0
BMFGDVT	log	diff	**26.8**	**22.6**	**26.6**
α-BMFGDVT	0.12	diff	22.8	20.8	20.7
α-BMFGDVT	0.1	diff	**22.1**	**19.5**	**20.5**
α-BMFGDVT	0.08	diff	22.3	19.3	21.0
α-BMFGDVT	0.1	1	21.7	19.2	20.3
α-BMFGDVT	0.1	2	**21.5**	**18.9**	**20.3**
α-BMFGDVT	0.1	3	21.5	18.8	20.5

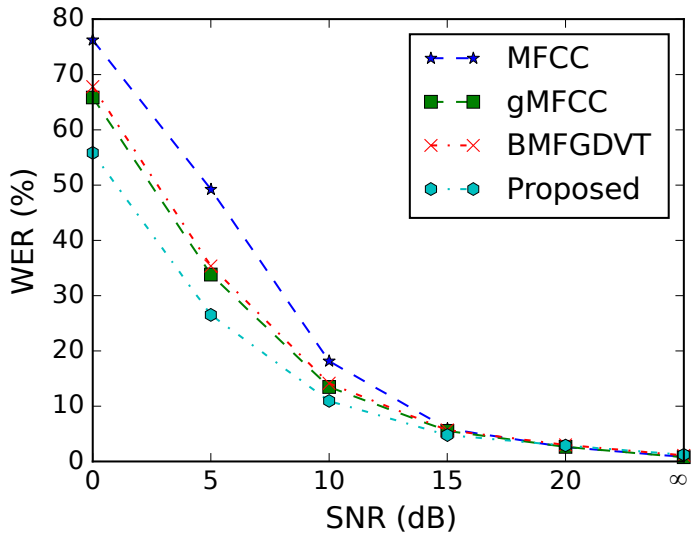

Fig. 4.32 WER of different features vs SNR for the Aurora-2 task (averaged over A, B and C test sets).

As seen in Table 4.8, as well as returning notably better results in the noisy condition (5.8% absolute WER reduction, on average), the proposed feature (α-BMFGDVT-0.1-2) in the clean condition performs as well as MFCC. Most of the other robust features, although

Table 4.8 *WER (in %) for Aurora-4 in clean (HMM/GMM) and multi-style training models (Bottleneck+HMM/GMM).*

Feature	A	B	C	D	*Ave*
MFCC [Clean]	7.4	34.5	29.4	50.3	39.0
Proposed [Clean]	7.1	26.5	28.1	45.1	33.2
BN-MFCC [Multi]	7.2	12.9	14.3	27.0	18.7
BN-Proposed [Multi]	7.0	12.7	14.3	26.6	18.4

outperform the MFCCs on average, perform more poorly in the matched scenarios (test set A).

Finally, bottleneck (BN) [172] features were extracted from the proposed phase-based representation and MFCCs in the multi-style training mode. Although in such circumstances MFCC work quite well, the proposed feature outperforms it. For a detailed explanation of the utilised DNN architecture and training procedure please refer to Appendix D.

4.7 Evaluation of Usefulness of Phase Source Component in Pitch Extraction

The speech signals can be aperiodic or quasi-periodic. Detecting the presence of the periodicity and quantifying the period value, are two common problems in speech source analysis. The excitation component could be a quasi-periodic train of laryngeal pulses in which vocal cords have a quasi-periodic vibration or could take the form of a turbulent air flow resulting in voiceless sounds. The problem of determining the existence of periodicity is referred as voicing determination. For speech sounds in which the voiced excitation is present, the rate of the vocal cord vibration is called the pitch or fundamental frequency (F_0). The problem of finding the value of the F_0 is referred to as pitch detection or pitch determination. To be more precise, fundamental frequency and pitch are not exactly the same quantities, although closely linked together. Fundamental frequency is an objective physical attribute whereas pitch is a subjective quality related to the human perceptual appreciation of the lowness and highness of a sound.

Fundamental frequency or fundamental period ($T_0 = \frac{1}{F_0}$) can be measured in the time domain, the frequency domain or a combination of both. Most of the frequency domain techniques utilise the magnitude spectrum. The goal of this section is to take a different approach and extract the pitch frequency from the source component of the phase spectrum. Recall that due to similarity between the group delay and the magnitude spectrum, magnitude-

based algorithms (with some modifications) can be employed for processing the group delay, too. So, let us first briefly review some of the well-known magnitude-based pitch extraction techniques.

4.7.1 Pitch Extraction using Magnitude Spectrum

Frequency domain algorithms, as the name suggests, start by computing the Fourier transform of the signal. If the signal is periodic with the fundamental periodicity of $T_0 = \frac{1}{F_0}$, its magnitude spectrum peaks in F_0 and the corresponding harmonics. In such case, the local maxima are (ideally) equally spaced and the distance between the adjacent peaks equals F_0. Such periodicity and peaks cannot be computed directly through an *argmax* process. So, there is a need for a function which converts this quasi-periodicity into a peak at the fundamental frequency, paving the way for estimating the pitch frequency through *argmax* (or *argmin*).

Harmonic Product Spectrum

Harmonic Product Spectrum (HPS) [173] is one of the frequency domain methods which multiplies the power spectrum, $|X[k]|^2$, by its downsampled versions, $|X[m\,k]|^2$ where m is the decimation factor. This process measures the maximum coincidence for the harmonics. The highest peaks (most likely) occur at the F_0 and its harmonics. It is computed as follows

$$HPS\,[k] = \prod_{h=1}^{N_{harm}} |\,X[h\,k]\,|^2 \tag{4.33}$$

where N_{harm} is the number of harmonics being considered. Having computed HPS, its *argmax* gives the fundamental frequency

$$F_0 = \frac{f_s}{N_{FFT}} \underset{k_{min} \le k \le k_{max}}{argmax} \; HPS\,[k] \tag{4.34}$$

where N_{FFT}, f_s, k_{min} and k_{max} are the FFT size and the sampling frequency in Hz, the minimum possible value for F_0 and the maximum possible value for F_0, respectively. Limiting the search space makes the argmax less computationally demanding and also decreases the probability of getting spurious results.

HPS is simple to implement, runs in real-time and performs relatively well under noise presence. Zero-padding is useful for improving the accuracy of these method. Although it will not improve the true frequency resolution, it provides more bins and a smoother spectrum which helps in resolving the peaks and consequently better estimating the pitch frequency.

Cepstral Analysis

Using cepstrum (DCT or FFT of the log of the magnitude spectrum) provides another way for pitch detection [174] as follows

$$T_0 = \frac{1}{F_0} = \frac{1}{f_s} \underset{q_{min} \leq q \leq q_{max}}{argmax} \; \tilde{x}[q] \tag{4.35}$$

where $\hat{x}$ is the (real) cepstrum of the signal x and q denotes the quefrency. One advantage of the cesptrum method over HPS is that often times the peaks in the harmonic frequencies are inharmonic because the F_0 falls between the discrete bins and does not exactly land on them. This could be problematic for HPS, although can be alleviated by zero-padding. The main disadvantage of the cepstrum method over HPS is that it is less robust to noise.

Summation Residual Harmonic

Summation residual harmonic (SRH) function [175] is another magnitude-based technique for estimating the fundamental frequency and is defined as follows

$$\begin{aligned} SRH\ [k] &= E_{res}[k] + \sum_{m=2}^{N_{harm}} \left(E_{res}[m\,k] - E_{res}[(m-0.5)\,k] \right) \\ F_0 &= \frac{f_s}{N_{FFT}} \underset{k_{min} \leq k \leq k_{max}}{argmax} \; SRH[k] \end{aligned} \tag{4.36}$$

where E_{res} denotes the magnitude spectrum of the residual signal after LPC analysis or generally, the magnitude spectrum of the excitation component. Similar to the HPS, it peaks at the fundamental frequency and its harmonics. Robustness is the main advantage of this technique.

4.7.2 Extension of the Magnitude-based Methods to the Phase Domain

Due to the similarity of the behaviour of the group delay and the magnitude spectrum, magnitude-based pitch extraction techniques can be employed with minimal modification for extracting the fundamental frequency from the group delay of the source component, namely $\tau_{Exc}(\omega)$. This section aims at extending the ideas of the HPS, cepstrum and SRH techniques to the phase domain and exploiting them to extract the pitch frequency from the group delay of the source component.

Earlier in this chapter, the FFT/DCT of the phase or group delay was referred to as the cepstrum* domain and it was shown that similar to the cepstrum domain, the pitch shows up

as a peak in the fundamental periodicity in this domain (Figure 4.16). Therefore, with the same logic the fundamental frequency can be calculated as follows

$$\begin{aligned} \tilde{x}^*[q] &= DCT\{\tau_{Exc}(\omega)\} \\ T_0 = \frac{1}{F_0} &= \frac{1}{f_s} \underset{q_{min} \le q \le q_{max}}{argmax} \quad \tilde{x}^*[q] \end{aligned} \tag{4.37}$$

where $\tilde{x}^*$ indicates the cepstrum*. In (4.37), $\tau_{Exc}(\omega)$ can be substituted with the $arg\{X_{Exc}(\omega)\}$ because the periodicity does not change by differentiation or integration.

For extending the HPS to the group delay domain, it should be noted that the multiplication in magnitude spectrum domain is equivalent to a sum in the phase/group delay domains because they behave similarly to the log of the magnitude spectrum in this sense. So, the HPS should be modified to harmonic sum spectrum (HSS) as follows

$$\begin{aligned} HSS\,[k] &= \sum_{h=1}^{N_{harm}} \tau_{Exc}[h\,k] \\ F_0 &= \frac{f_s}{N_{FFT}} \underset{k_{min} \le k \le k_{max}}{argmax} \quad HSS[k] \end{aligned} \tag{4.38}$$

and the rest of the pitch estimation process would be identical to the HPS method.

Finally, SRH can be easily extended to the group delay domains as follows

$$\begin{aligned} SRH\,[k] &= \tau_{Exc}[k] + \sum_{m=2}^{N_{harm}} \left(\tau_{Exc}[m\,k] - \tau_{Exc}[(m-0.5)\,k]\right) \\ F_0 &= \frac{f_s}{N_{FFT}} \underset{k_{min} \le k \le k_{max}}{argmax} \quad SRH[k]. \end{aligned} \tag{4.39}$$

In the experiments N_{harm}, number of harmonics, was set to five. Figure 4.31 depicts the pitch, estimated using the aforementioned techniques versus ground truth values taken from [176] in the clean and noisy (5 dB) conditions. Also the employed methods are compared with their magnitude-based counterpart.

As illustrated in Figure 4.30, k_0, namely the parameter of the regression filter (4.30) has a substantial impact on the source component. To better quantify its influence the effect of different values for k_0 have been evaluated on the extracted pitch track from the source component of the phase spectrum using SRH technique in the clean and noisy conditions. As can be seen in Figure 4.33, k_0 has a significant impact on both accuracy and robustness of the phase-based pitch extraction process and can lead to a reliable phase-based F_0 estimation. As the results demonstrate, using the regression filter in comparison with a sample difference

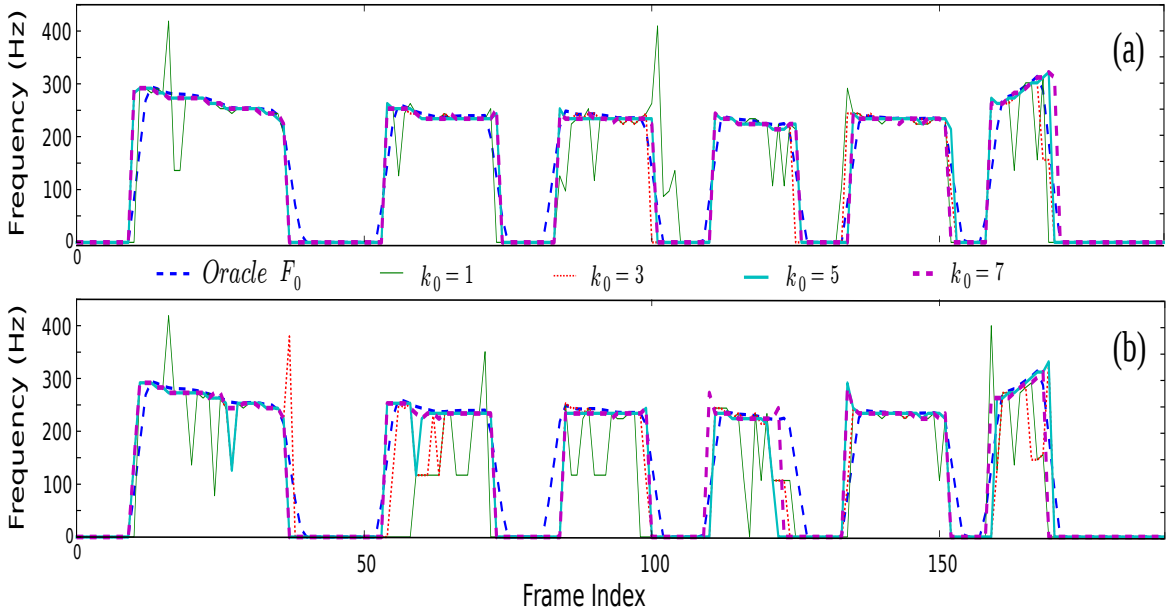

Fig. 4.33 Effect of k_0 on the accuracy/robustness of phase-based F_0 estimation using (4.39) for *sb*003.*sig* from [176] ($\alpha = 0.1$). Pitch track in (a) clean condition, (b) noisy (Gaussian white, 5 dB) condition.

returns remarkably better results in terms of both robustness and accuracy. Note that by SNR reduction, the optimal value for k_0 increases as further smoothing would be required to counter the noise effect. Optimal range for k_0 appears to be 3-7, leading to significant accuracy/robustness in both clean and noisy circumstances.

Finally, Figure 4.34 shows the results after extending the cepstrum (CEP), HPS (HSS) and SRH to the phase domain. As seen, in general, the proposed phase-based approaches in the clean condition perform as well as their magnitude-based counterparts but in the noisy conditions outperform them. For drawing a firm conclusion, however, more extensive experiments should be carried out.

4.8 Summary

This chapter presented a novel source-filter separation method for the speech signal through phase processing. It was shown that the excitation and vocal tract components are additive in the phase spectrum domain of the minimum-phase component. The source part of the phase spectrum resembles a quickly oscillating element superimposed by a slow-varying modulating filter component. Based on the additive property and the differing pace of change with respect to the frequency, phase spectrum of the minimum-phase part was morphed into the Trend-plus-Fluctuation format and decomposed into the vocal tract (Trend) and excitation

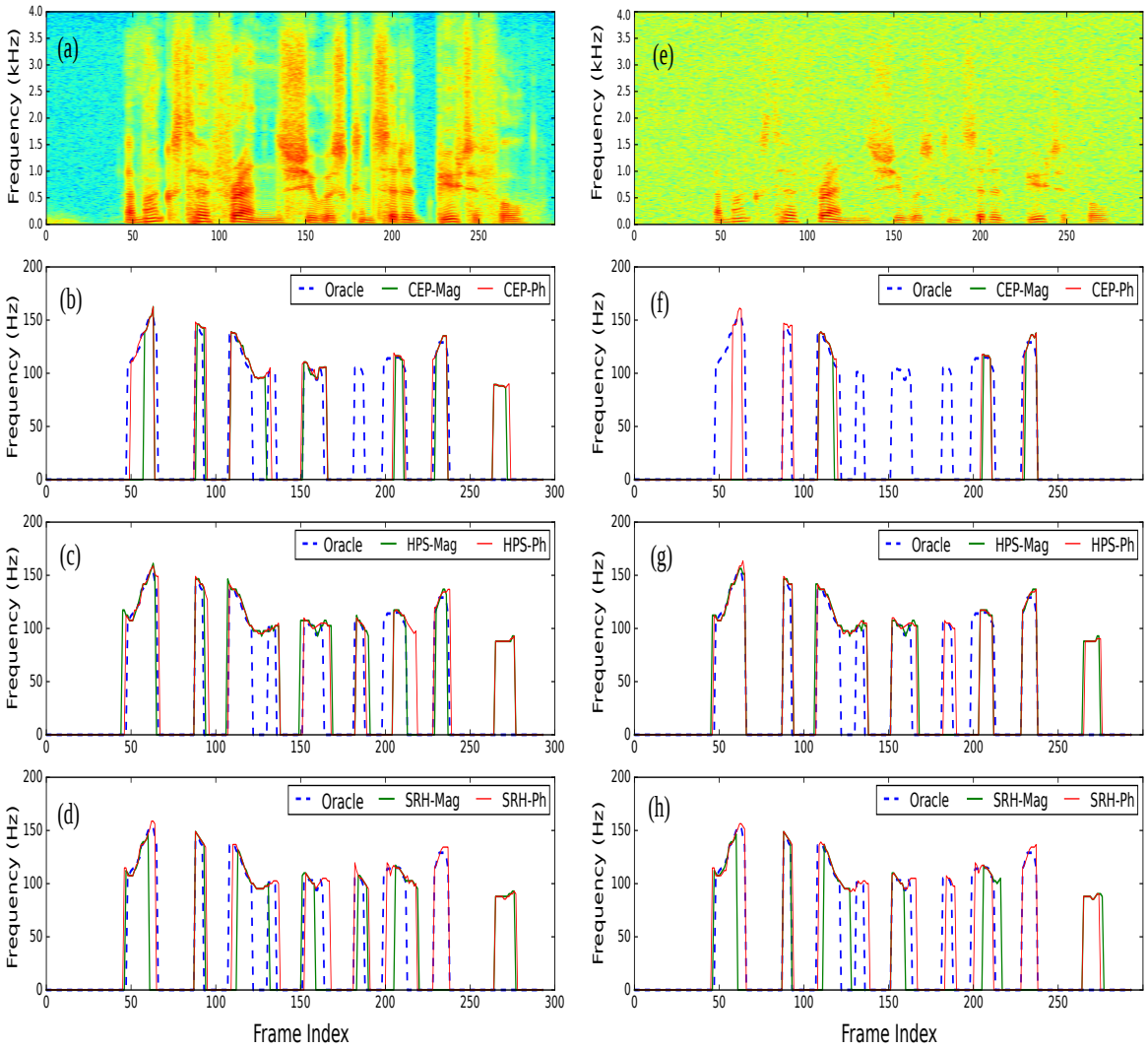

Fig. 4.34 Pitch tracking based on CEP, HPS and SRH methods using the phase and magnitude spectra in the clean and noisy conditions. (a) Spectrogram of the clean signal (*rl*026.*sig* from [176].), (b) pitch track using CEP method in the clean condition, (c) pitch track using HPS method in the clean condition, (d) pitch track using SRH method in the clean condition, (e) spectrogram of the noisy signal (White Gaussian, 5 dB), (f) pitch track using CEP method in the noisy condition, (g) pitch track using HPS in the noisy condition, (h) pitch track using SRH in the noisy condition. No post-processing on F_0 track has been applied. XX-YY indicates method XX (cepstrum, HPS or SRH) along with spectrum YY (phase or magnitude).

(Fluctuation) components through low-time linear liftering. The extracted filter and source parts where successfully employed in the feature extraction for ASR and pitch frequency extraction, respectively. Using regression filter in computing the group delay was shown to be useful in working with the source component and extracting the fundamental frequency

from it. On the other hand, replacing the log function with the generalised logarithmic function (power transformation) in the Hilbert transform as well as statistical normalisation (Gaussianisation, Laplacianisation and histogram equalisation) turned out to be useful in improving the robustness of the phase-based features extracted from the filter component. In the next chapter we aim at combining the power transformation with more advanced model-based statistical normalisation techniques to achieve higher robustness in ASR.

Chapter 5

Generalised VTS in the Phase and Group Delay Domains for Robust ASR

All generalisations are dangerous, even this one.
– Alexandre Dumas

Robust grass endures mighty winds;
loyal ministers emerge through ordeal.
– William Shakespeare, *Measure for Measure*

5.1 Introduction

Countering the effect of the noise is an important challenge in speech processing and robust ASR. There are a wide range of methods for dealing with this problem. Speech enhancement techniques could be one option, primarily aiming to elevate the quality and/or intelligibility of the speech signal. However, they are not popular in building robust speech recognition systems as the intelligibility or quality improvement does not necessarily translate into WER reduction. As shown in the previous chapter, statistical normalisation methods like HEQ or Gaussianisation are another options. They do not need stereo data and the mathematical description of the environment model. However, the scenarios under which they perform well is limited. For example, HEQ presumes that the noise contamination process is monotonic; but due to randomness of the noise this assumption does not hold perfectly. Also reliable estimation of the histogram of the clean signal and particularly the quantile function (inverse of the cumulative distribution function) is not straightforward. On the other hand, techniques

like SPLICE[1] [177], require stereo data for learning the mapping from noisy observation to clean representation along with an estimate of the noise type and its level. This limits their practicality in scenario in which only the clean data is available for training. Vector Taylor series (VTS) [17, 178] is another powerful techniques for noise compensation which requires the environment model as well as a statistical model of the clean features and an estimate of noise but does not require stereo data.

In Chapter 4 it was shown that statistical normalisation of the features and replacing the log with the power transformation in the Hilbert transform could have a considerable impact on the robustness of the extracted phase-based parameters on speech recognition performance in noisy environment. This motivates to gravitate towards more advanced statistical normalisation schemes such as VTS. It is a model-based approach that uses the Taylor series for linearising the non-linear relationship between the noisy observation, the clean signal and noise. Such linearisation facilitates estimating the statistics of the noisy observation, Y, which is required to obtain the MMSE estimate of the clean features.

The VTS method was primarily developed for enhancing features in the (real) cepstrum or the log of the filter banks energies (log-FBE) domains. For applying the VTS technique there are three prerequisites: first, an environment model in the target domain, second, probability density function (pdf) of all the involved variables, and third the partial derivative of the noisy observation with respect to the variables in the environment model. As will be outlined, taking the idea of the VTS from the power spectrum domain to the phase and group delay (GD) domains is not straightforward. In particular, the environment model consists of double the number of variables compared to the environment model in the periodogram domain. This means that instead of four, eight pdfs should be estimated. In addition, instead of computing three partial derivatives (to be more precise, Jacobian matrices), seven should be calculated. This complicates the estimation and noise compensation process in the phase-related domain.

Moreover, VTS in its original formulation assumes usage of the log function. As demonstrated in the previous chapter, replacing the log with the power transformation, leads to higher robustness and improves the flexibility of the framework. So, before extending the idea of VTS to the phase-related domain, a new formulation for VTS is developed which assumes usage of the power transformation instead of the log. However, applying the power transformation is problematic in the phase and/or group delay domains because they can be negative for some frequency. Having negative values hinder application of the power transformation because the admissible range is restricted to the positive values.

This chapter suggests solutions to the aforementioned issues and aims to develop a novel VTS formulation in a phase-related domain. In this regard, first the environment model which

[1] Stereo-based Piecewise Linear Compensation for Environments

shows how the additive and channel noise distort the clean signal in the group delay domain is derived. This helps to illustrate the effect of the additive and channel noise on the group delay function and forms the starting point in deriving the corresponding VTS equations. Then, the aforementioned challenges are addressed and some solutions are presented.

The rest of this chapter is organised as follows. In Section 2, the environment model in the group delay domain is derived and effect of the additive noise is discussed. In Section 3, the problems related to applying the power transformation to the environment model in the group delay domain are studied and some possible solutions are presented. Section 4 scrutinises the behaviour of the channel distortion in the phase (and magnitude spectrum) domain and uses the results in simplifying the environment model after applying the power transformation. In Section 5 the equations of the VTS and gVTS in the log-FBE and cepstral domains are derived. Section 6 contains experimental results along with discussion and Section 7 summarises the chapter.

5.2 Environment Model in the Group Delay Domain

To understand the response of the group delay to the noise, first the environment model which shows how the clean signal gets contaminated with the noise should be defined. The general model takes the form of $Y(\omega) = X(\omega)H(\omega) + W(\omega)$ where ω, Y, X, H and W are the radial frequency, the (short-time) Fourier transforms (FT) of the noisy observation, clean signal, channel and additive noise, respectively. Assuming speech and noise are uncorrelated and using the periodogram[2] method for the power spectrum estimation

$$|Y(\omega)|^2 = |X(\omega)|^2 |H(\omega)|^2 + |W(\omega)|^2 \tag{5.1}$$

where $|.|^2$ is the square of the magnitude spectrum and indicates the periodogram estimate of the power spectrum.

In the power spectrum domain, additive noise acts as an additive term whereas in the phase and group delay domains, it has a different effect. The group delay of the noisy observation, τ_Y, is computed as follows:

$$\tau_Y(\omega) = -\frac{d\, arg\{Y(\omega)\}}{d\omega} = \frac{Y'_R\, Y_I - Y_R\, Y'_I}{|Y|^2} \tag{5.2}$$

[2]To be more precise, it should be called *modified* periodogram as Hamming window is typically applied rather than rectangular window [116]. Also, without loss of generality, the normalisation factor ($\frac{1}{N_{FFT}}$) is set to 1 as it has no effect on the discriminability of the extracted features.

where $arg\{Y(\omega)\}$ is the unwrapped phase spectrum[3] of Y, and subscripts R and I indicate the real and imaginary parts

$$\begin{cases} Y_R(\omega) &= X_R H_R - X_I H_I + W_R \\ Y_I(\omega) &= X_R H_I + X_I H_R + W_I. \end{cases} \tag{5.3}$$

The group delay of Y can be calculated by using (5.2) and (5.3) in (5.1). However, a better way for computing τ_Y is to take advantage of the properties of the group delay as well as the independence of clean speech (X) and additive noise (W),

$$\begin{aligned} Z_1 = X + W &\Rightarrow \tau_{Z_1} = \frac{|X|^2}{|X|^2+|W|^2}\,\tau_X + \frac{|W|^2}{|X|^2+|W|^2}\,\tau_W \\ Z_2 = X\,H \quad &\Rightarrow \tau_{Z_2} = \tau_X + \tau_H \end{aligned} \tag{5.4}$$

Using these properties and some algebraic manipulation yield the environment in the group delay domain

$$\tau_Y = \frac{|X|^2|H|^2}{|Y|^2}\,(\tau_X + \tau_H) + \frac{|W|^2}{|Y|^2}\tau_W. \tag{5.5}$$

where τ_X, τ_H and τ_W are the group delay of the clean speech, channel and the additive noise, respectively.

5.2.1 Effect of Additive Noise in the Group Delay Domain

If we assume that the signal is only contaminated with additive noise, 5.5 takes the following form

$$\tau_Y = \frac{|X|^2}{|Y|^2}\,\tau_X + \frac{|W|^2}{|Y|^2}\tau_W = \frac{\xi}{1+\xi}\,\tau_X + \frac{1}{1+\xi}\tau_W \tag{5.6}$$

where ξ is *a priori* SNR [12] and is defined as follows

$$\xi = \frac{|X|^2}{|W|^2}. \tag{5.7}$$

Let $c = \frac{\xi}{1+\xi}$, as such

$$\tau_Y = c\,\tau_X + (1-c)\,\tau_W. \tag{5.8}$$

[3] Actually, the $arg\{Y\}$ is the short-time phase spectrum with two independent variables, namely time and frequency; hence, instead of derivative (d), the notation of the partial derivative (∂) should be used in (5.2). However, since for making the equations more compact the time variable (frame index) is removed, the d is used instead of the ∂.

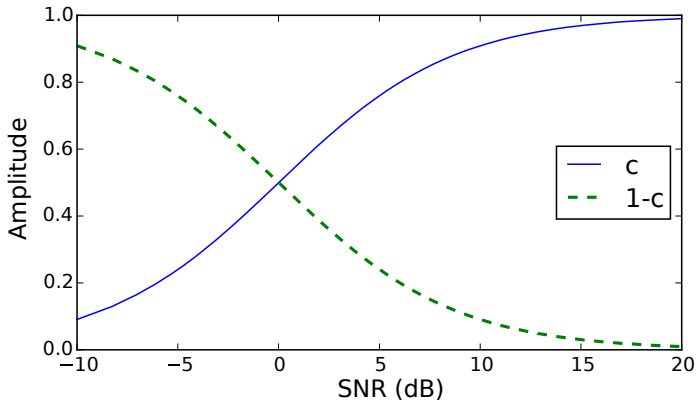

Fig. 5.1 Weight of the clean signal (c) and the additive noise ($1-c$) in the GD domain as a function of a priori SNR (ξ) in dB.

For the power spectra of the corresponding relation between the noisy observation, clean speech and the noise is

$$|Y|^2 = |X|^2 + |W|^2. \tag{5.9}$$

Figure 5.1 shows c and $1-c$ versus ξ. Note that c is a monotonically increasing function of ξ. So, by increasing the SNR, the weight of the clean part goes up and with the same rate, the weight of the noise comes down.

Comparing (5.8) with (5.9) shows the noisy observation in the group delay domain is a *weighted sum* of the clean part and the additive noise while in the periodogram domain it is just the sum of the corresponding power spectra of these two components. The weighted sum in the group delay domain takes the *convex* form and the coefficient c is proportional with a priori SNR (Figure 5.1) whereas in the periodogram domain the weights are constant regardless of the SNR.

5.3 (g)VTS in the Group Delay Domain; Problems

Having derived the environment model in the group delay domain, we now extend the idea of the VTS [17] to the group delay domain, and combine the VTS with the power transformation. However, there are some issues which should be addressed and resolved in advance. In this section, the problems are explained and some solutions are suggested.

5.3.1 Increase in the Number of Variables

For noise compensation using the VTS framework (detailed in Appendix B), as well as a mathematical expression for how the signal gets corrupted in the target domain, the statistical

distribution of all the variables involved, is needed. While in the periodogram domain, namely (5.1), there are only four variables, (5.5) shows that in the group delay domain, the environment model contains eight quantities. Hence, eight probability distribution functions (pdf) should be estimated. Considering eight variables instead of four, complicates the compensation process. If the number of variables gets reduced, the problem will be more tractable.

To decrease the number of variables, two possible cases can be taken into account: First, variables that overlap in terms of the information they carry and are added/multiplied together can be re-expressed through one variable. Second, a term containing a variable that tends to zero in the expected sense, e.g. cross-correlation of the speech and noise in (5.1), may be removed, too. We use both of these points for reducing the number of variables.

Now, one can multiply both sides of (5.5) by $|Y|^2$

$$|Y|^2\,\tau_Y = |X|^2|H|^2\,(\tau_X+\tau_H)\;+\;|W|^2\,\tau_W. \tag{5.10}$$

In general, $|Z|^2$ and τ_Z for some variable Z, are not totally independent quantities and for many signals they are closely linked together. It can be shown that

$$\begin{aligned} log|Z(\omega)| &= \mathscr{F}^{-1}\{CC_{even}[q]\} \\ \tau_Z(\omega) &= \mathscr{F}^{-1}\{q\,CC_{odd}[q]\} \end{aligned} \tag{5.11}$$

where $\mathscr{F}^{-1}$, CC and q denote the inverse Fourier transform[4], complex cepstrum and quefrency, respectively, and subscripts *even* and *odd* indicate the even and odd parts. Note that $CC = CC_{even} + CC_{odd}$ and CC_{even} equals the real cepstrum ($\hat{x}[q]$). In general, the even and odd parts of a signal are independent except for causal[5] or anti-causal[6] sequences.

For the minimum-phase (MinPh) signals, CC is causal [18] which leads to

$$CC[q]=0 \quad \forall \quad q<0 \quad \Rightarrow \quad \begin{cases} CC_{even}[q] = CC_{odd}[q] & q>0 \\ CC_{even}[q] = -CC_{odd}[q] & q<0 \end{cases} \tag{5.12}$$

This illustrates the close relation between the magnitude and group delay for the MinPh signals. Speech is not a MinPh signal since its complex cepstrum is not causal [71]. However, as demonstrated in Section 3.4.2, under short-frame decomposition the MinPh part is the dominant component. Therefore, it appears reasonable to encapsulate the multiplication of

[4]It could be inverse DCT, depending on how the cepstrum is computed.
[5]Equals zero at negative indices, namely $\hat{x}[q] = 0 \,\forall\, q<0$.
[6]Equals zero at positive indices, namely $\hat{x}[q] = 0 \,\forall\, q>0$.

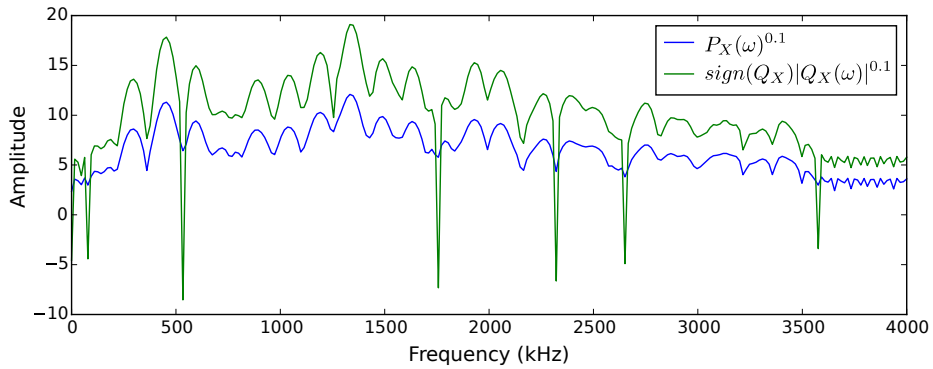

Fig. 5.2 Product spectrum ($Q_X(\omega)$) along with the periodogram power spectrum estimate ($P_X(\omega) = |X|^2$). Since the product spectrum could be negative, for a better visualisation, it was compressed through $sign(z)|z|^{\alpha}$ (here $\alpha = 0.1$). Deep valleys in the product spectrum stem from zeros placed next to the unit circle.

$|Z|^2$ τ_Z into a single variable Q_Z which is called *product spectrum* (PS) in [42]. Accordingly, (5.10) can be rewritten as

$$Q_Y = Q_X\,|H|^2 + Q_H\,|X|^2 + Q_W \tag{5.13}$$

where Q_Y, Q_X, Q_H and Q_W are the product spectra of the noisy observation, clean signal, channel and additive noise, respectively. By this trick the number of variables reduces to six from eight, although it is still larger than the periodogram domain for which there are only four variables in the environment model.

Figure 5.2 depicts the product spectrum and the periodogram. As seen, they are highly similar in terms of the information they carry, i.e., both vocal tract and excitation components have an almost identical manifestation in both of these domains. The main difference between the product spectrum and the periodogram lies in the sharp valleys in the product spectrum which stems from the sensitivity of the group delay to zeros located in the vicinity of the unit circle. Other than that, they seem to be merely two representations for almost the same information. As such encapsulating magnitude and group delay into a single quantity is unlikely to result in information loss.

5.3.2 Applying the Power Transformation

Figure 5.2 shows that the dynamic range of the product spectrum is comparable to the periodogram. This causes the corresponding histogram to have a wide support, hindering effective distribution estimation and statistical modelling. So, the dynamic range should be compressed using functions such as $log(z)$ or the power transformation (z^{α}). The valid range

for the input of both functions is restricted to positive values. Although the power spectrum is always positive, the group delay and consequently the product spectrum may take negative values (Figure 5.2). This, in turn, hinders applying the log or the power transformation for compressing the dynamic range of the group delay and the product spectrum.

5.3.3 Dealing with Negative Values in Product Spectrum Domain

To deal with the negative values one may use modulus or add a positive constant or floor the values below a pre-selected threshold. The advantages and disadvantages of each one are briefly reviewed in the following.

Absolute value

Taking the absolute value is not an appropriate solution as it makes some of the negative values larger than small positive ones. This distorts the relative order/rank of the samples which is unfavourable. The other possible solution which has been used for compressing the group delay in [40] (also in Figure 5.2), is to implement the compression using $sign(x)|x|^{\alpha}$, inspired by [161]. Although this approach preserves the relative order, it poses two problems for the VTS-based noise compensation process: first, the variable of interest, namely X, which should be estimated at the end of noise compensation process cannot be factored out

$$\begin{aligned} sign(Y)|Y|^{\alpha} &= sign(XH+W)\ |XH+W|^{\alpha} \\ &\neq \tilde{X}\ \breve{G}(\tilde{X},\tilde{H},\tilde{W}) \end{aligned} \tag{5.14}$$

where $\tilde{Z} = sign(Z)\,|Z|^{\alpha}$ for $Z \in \{X,H,W\}$, *sign* indicates the signum function and $\breve{G}$ denotes the distortion function. Second, computing the partial derivatives (Jacobians) of the noisy observation with respect to the other variables in the environment model becomes complicated due to the *sign* function and modulus (absolute value) operation.

Adding a Constant

Another option which preserves the order without complicating the Jacobian computation is to add a constant, C_0, to the product spectrum to ensure it remains positive at all time-frequency bins. The smallest value for C_0 is the negative of the minimum of Q_Z across the utterance. The main disadvantage of using this minimum is that it varies from signal to signal and consequently induces extra inter-utterance variability. An alternative option is adding a fixed large-enough constant to all the utterances guaranteeing the positiveness across all signals. The main drawback of this approach, however, is that such a big additive constant

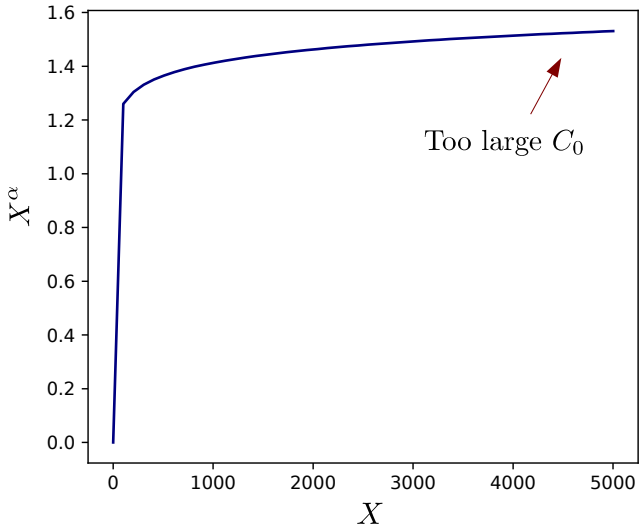

Fig. 5.3 Too large C_0 makes the compression function $((Q_Z + C_0)^\alpha)$ operate in the saturation region $((Q_Z + C_0)^\alpha \approx C_0^\alpha)$, leading to over-compression.

masks the (relatively) small values of product spectrum, and causes the compression function to operate in its saturation region, i.e., $(Q_Z + C_0)^\alpha \approx C_0^\alpha$. The optimal operating range for a compression function is around the knee point. Before this point the function behaves linearly and fails to compress the dynamic range effectively. After the knee point, compression could be too severe, leading to over-squashing the dynamic range of the variables.

Flooring

The final solution presented here to the negative value problem is to floor values below a pre-specified threshold. A potential pitfall of this technique is that it may cause information loss as it is an irreversible process. However, this is tolerable as long as the discarded data plays an insignificant role. As shown in Figure 5.2, the majority of the activity in the product spectrum domain takes place on the positive side. Therefore, flooring is unlikely to result in information loss. The floor function mathematically takes the following form

$$floor(z;\theta_z) = max(z,\theta_z), \tag{5.15}$$

where θ_z is a tunable threshold parameter. Setting θ_z to zero makes (5.15) act as a half-wave rectifier. This is in contrast to the absolute value which acts as like a full-wave rectifier. In special case of using log function for compression, setting θ_z to zero is problematic. In the work presented here we have adopted $\theta_z = 1$ in case of using the log function for compression and $\theta_z = 0$ when the power transformation is applied.

5.3.4 Environment Model after Applying Compression Function in Product Spectrum Domain

Having found a solution to the negative values issue in the product spectrum domain, the compression function can be applied. Using the log function for compression results in

$$\tilde{Q}_Y = \tilde{Q}_X + \tilde{H} + log\big(1 + exp(\tilde{Q}_H + \tilde{X} - \tilde{Q}_X - \tilde{H}) + exp(\tilde{Q}_W - \tilde{Q}_X - \tilde{H})\big) \tag{5.16}$$

where $\tilde{Q}_Z = log(floor(Q_Z;1))$ and $\tilde{Z} = log(|Z|^2)$ for any $Z \in \{Y,X,H,W\}$. Compressing (5.14) using the power transformation z^α yields

$$\breve{Q}_Y = \breve{Q}_X \breve{H} \left(1 + (\frac{\breve{Q}_H \breve{X}}{\breve{Q}_X \breve{H}})^{\frac{1}{\alpha}} + (\frac{\breve{Q}_W}{\breve{Q}_X \breve{H}})^{\frac{1}{\alpha}}\right)^\alpha \tag{5.17}$$

where $\breve{Q}_Z = (floor(Q_Z;0))^\alpha$ and $\breve{Z} = (|Z|^2)^\alpha$ for $Z \in \{Y,X,H,W\}$.

Comparing (5.16) and (5.17) with the corresponding equations in the power spectrum domain in Appendix B, shows that here there is an extra term $(\frac{\breve{Q}_H \breve{X}}{\breve{Q}_X \breve{H}})^{\frac{1}{\alpha}}$ due to the group delay of the channel distortion, τ_H. Without this term, the equations become similar to the equations in the power spectrum domain and this facilitates re-deriving the (g)VTS formulae in the product spectrum domain. As such removing this term would lead to mathematical convenience. In general, neither $|X|$ nor $|H|$ are zero for obvious reasons. However, the spectral behaviour of the group delay of the channel, τ_H, is unclear and has not been studied.

5.3.5 Phase Spectrum and Group Delay of Channel

To investigate the behaviour of the group delay of the channel, let us start by a perfect channel with a flat frequency response. Assuming the channel is minimum-phase, the phase spectrum could be computed using the Hilbert transform. It can be easily seen that

$$\begin{aligned} arg\{H(\omega)\} &= -\frac{1}{2\pi} log|H(\omega)| * cot(\frac{\omega}{2}) = -\frac{1}{2\pi} log|H_0| * cot(\frac{\omega}{2}) \\ &= -\frac{1}{2\pi} log|H_0| \left(\int_{-\pi}^{\omega-\varepsilon} cot(\frac{\omega-\theta}{2}) d\theta + \int_{\omega+\varepsilon}^{\pi} cot(\frac{\omega-\theta}{2}) d\theta \right) = 0 \end{aligned} \tag{5.18}$$

where $arg\{H(\omega)\}$ denotes the channel phase spectrum, H_0 is the constant value characterising the ideal channel across the spectrum and ε is the smallest positive number. As seen in (5.18), the minimum-phase component of the phase of the ideal channel is zero. Therefore,

the derivative of the $arg\{H(\omega)\}$, namely τ_H becomes zero, too. This allows to remove the *undesired* term in (5.17). However, the frequency response is not flat in practice.

To systematically investigate the properties of $\tau_H(\omega)$, there is a need for a database of impulse responses of different channels. Here, we make use of the test sets A and C of the Aurora-4 database [11]. Both sets include 330 utterances with an average length of 7.3 seconds. Signals in the test set A are recorded using a head-mounted close-talking microphone whereas in the test set C the same speech is simultaneously recorded by a different desktop microphone. Reportedly, 18 desktop microphones have been used in recording process [11].

For the sake of argument let us assume that the microphone used in the test set A is ideal. This allows to treat sets A and C as stereo data and facilitates channel estimation. As such one can write

$$\begin{cases} \text{Test Set A} & \Rightarrow Y^A = X \\ \text{Test Set C} & \Rightarrow Y^C = X\,H \end{cases} \quad \Rightarrow \quad H = \frac{Y^C}{Y^A} \tag{5.19}$$

where Y^A and Y^C denote the short-time Fourier transforms of two corresponding signals from test set A and test set C, respectively. By dividing Y^A and Y^C at each frame and averaging over the whole utterance, one can compute the average Fourier transform of the channel

$$H_t = \frac{Y_t^C}{Y_t^A} \Rightarrow \begin{cases} |H(t,\omega)|^2 & = \frac{|Y_t^C(t,\omega)|^2}{|Y_t^A(t,\omega)|^2} \\ arg\{H(t,\omega)\} & = arg\{Y_t^C(t,\omega)\} - arg\{Y_t^A(t,\omega)\} \\ \tau_H(t,\omega) & = \tau_{Y^C}(t,\omega) - \tau_{Y^A}(t,\omega) \end{cases} \tag{5.20}$$

where H_t, $|H_t|^2$, $arg\{H_t\}^2$ and τ_{H_t} indicate channel's short-time Fourier transform, squared magnitude spectrum, phase spectrum and the group delay at the frame t, respectively. Having computed H_t, the magnitude and phase spectra of the channel as well as its group delay can be computed at the frame level. Then, by averaging over the utterance, the $|H|^2$, $arg\{H\}$ and τ_H can be calculated

$$\begin{aligned} |H(t,\omega)|^2 &= \frac{1}{T}\sum_{t=1}^{T} |H(t,\omega)|^2 \\ arg\{H(t,\omega)\} &= \frac{1}{T}\sum_{t=1}^{T} arg\{H(t,\omega)\} \\ \tau_H(t,\omega) &= \frac{1}{T}\sum_{t=1}^{T} \tau_H(t,\omega) \end{aligned} \tag{5.21}$$

where T denotes the number of the frames of the utterance.

There are two implicit assumptions underpinning (5.19): First, the intensity of the signals does not exceed the linear range of the microphones characteristics. As a matter of fact, H_t supposedly represents the channel response in its linear operating region. Second, the length of the frame (25 ms) is longer than the effective length[7] of the impulse response of each microphone. For an ideal flat channel, the impulse response equals the impulse function which includes only one sample. As the microphone characteristics deviates from the ideal case, the effective length of the impulse response increases. Also, if the non-speech frames where $X = 0$, are filtered before estimating H using (5.19) the result can be more accurate numerically. However, this requires performing a voice activity detection (VAD) a priori and in noisy conditions, the reliability of the speech/non-speech labels could be questionable. So, we do not used VAD in the work presented here. Note that the possibility of having absolute zero ($X = 0$) is infinitesimal and also in the implementations a flooring threshold is applied to avoid this.

Figure 5.4 illustrates $|H|^2$, $arg\{H\}$, τ_H and Figure 5.5 depicts the filter bank energies (FBE) obtained by passing $|H|^2$ and τ_H through a Mel filter bank, computed for all the 330 utterances along with the overall mean. As can be observed, on average, τ_H tends to zero, both before and after the application of the filter bank, which permits the removal of $Q_H|X|^2$ from (5.13). This is specially true for frequency range below 4 kHz (sampling rate is 16 kHz, here) and the filters with index less than 17 (out of 23 filters utilised here). As a result, in the expected sense, $\tau_H \rightarrow 0$ which, in turn, leads to $Q_H \rightarrow 0$. This pushes the environment model toward its counterpart in the periodogram domain (5.1),

$$Q_Y \approx Q_X\,|H|^2 + Q_W \tag{5.22}$$

It paves the way for computing the VTS and its generalised version (Appendix B) using the corresponding equations in the periodogram domain.

5.3.6 Simplified Environment Model in the Product Spectrum Domain

Compression using Log Function

Assuming the $Q_H \rightarrow 0$ allows to rewrite (5.16) as follows:

$$\tilde{Q}_Y \approx \tilde{Q}_X + \tilde{H} + log\big(1 + exp(\tilde{Q}_W - \tilde{Q}_X - \tilde{H})\big). \tag{5.23}$$

[7]By effective length, it is meant the range which outside it the impulse response, with a reasonable approximation, equals zero.

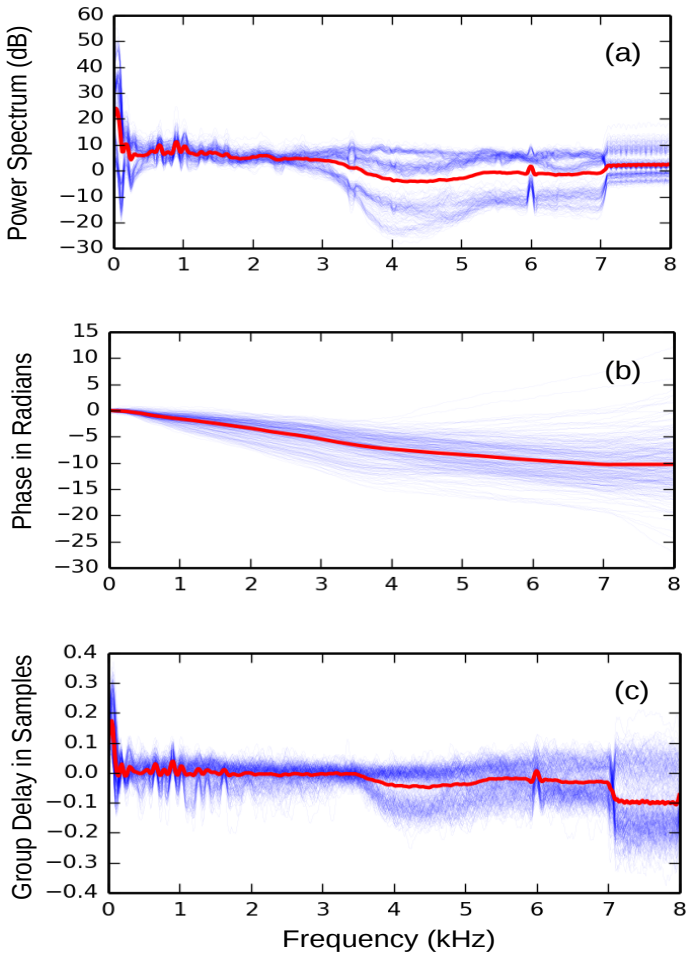

Fig. 5.4 Channel behaviour in the frequency domain. The red curve shows the average over all utterances (330 signals). (a) squared magnitude spectrum $|H|^2$, (b) unwrapped phase spectrum $arg\{H\}$, (c) group delay τ_H.

By taking DCT, the environment model in the cepstral domain can be derived as

$$\tilde{q}_y \approx \tilde{q}_x + \tilde{h} + C\, log\big(1 + exp(C^{-1}(\tilde{q}_w - \tilde{q}_x - \tilde{h}))\big), \tag{5.24}$$

where C^{-1} denotes the inverse DCT (IDCT) matrix, $\tilde{h} = C\,\tilde{H}$, $\tilde{q}_z = C\,\tilde{Q}_Z$ and $\tilde{q}_z$ indicates the cepstrum of the $\tilde{Q}_Z$, for any $Z \in \{Y, X, W\}$ and any $z \in \{y, x, w\}$. Now the noisy observation can be rewritten as the sum of the clean part, and distortion function in the frequency and quefrency domains,

$$\tilde{Q}_Y \approx \tilde{Q}_X + \tilde{G}(\tilde{Q}_X, \tilde{Q}_W, \tilde{H}), \tag{5.25}$$

$$\tilde{q}_y \approx \tilde{q}_x + \tilde{g}(\tilde{q}_x, \tilde{q}_w, \tilde{h}), \tag{5.26}$$

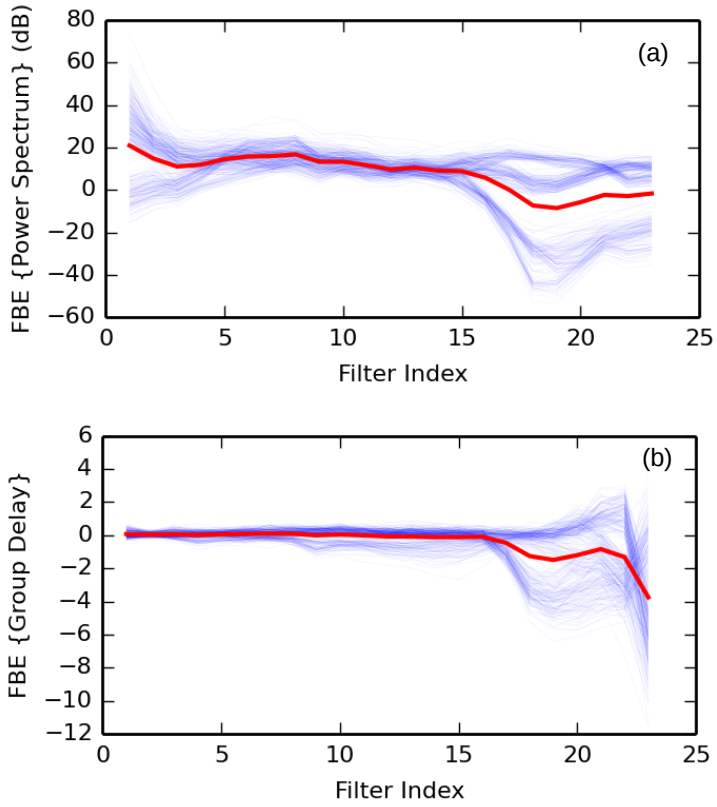

Fig. 5.5 Channel behaviour in the frequency domain after applying the filter bank, the red curve shows the average over all utterances (330 signals). Filter bank input: (a) $|H|^2$, (b) τ_H.

where $\tilde{G}$ and $\tilde{g}$ are the *distortion* functions in the frequency and quefrency domains, respectively. In the frequency domain

$$\tilde{G}(\tilde{Q}_X, \tilde{Q}_W, \tilde{H}) = \tilde{H} + log\big(1 + exp(\tilde{Q}_W - \tilde{Q}_X - \tilde{H})\big), \tag{5.27}$$

and in the quefrency domain

$$\tilde{g}(\tilde{q}_x, \tilde{q}_w, \tilde{h}) = \tilde{h} + C\, log\big(1 + exp(C^{-1}(\tilde{q}_w - \tilde{q}_x - \tilde{h}))\big). \tag{5.28}$$

As can be seen, the distortion function is additive with respect to the clean part (in the case of using log) and inversely proportional to the SNR. The higher the SNR, the closer $\tilde{G}$ and $\tilde{g}$ are to zero. The overarching goal of the noise compensation process is to counter the effect of the distortion function and estimate the clean part given the noisy observation.

Compression using Power Transformation

Assuming $Q_H \rightarrow 0$ allows for rewriting the environment model after compression through the power transformation (namely (5.17)) as follows

$$\breve{Q}_Y \approx \breve{Q}_X \breve{H} \left(1 + (\frac{\breve{Q}_W}{\breve{Q}_X \breve{H}})^{\frac{1}{\alpha}}\right)^{\alpha}. \tag{5.29}$$

Taking the DCT, provides the environment model in the cepstrum domain

$$\breve{q}_y \approx C \left((C^{-1}\, \breve{q}_x)(C^{-1}\, \breve{h}) \left(1 + (\frac{C^{-1}\breve{q}_w}{(C^{-1}\breve{q}_x)\,(C^{-1}\breve{h})})^{\frac{1}{\alpha}}\right)^{\alpha}\right) \tag{5.30}$$

where $\breve{h} = C\,\breve{H}$, $\breve{q}_z = C\,\breve{Q}_Z$ and $\breve{q}_z$ indicates the cepstrum of the $\breve{Q}_Z$ for $Z \in \{Y,X,W\}$ and $z \in \{y,x,w\}$. To be more accurate, the substitution of the log with power transformation in the cepstrum computation framework yields generalised cepstrum [48].

Applying the power transformation for the dynamic range compression changes the mechanism through which the distortion function affects the clean features. In contrast to the log case, where the distortion function was acting as an additive term, it emerges as a multiplicative term

$$\breve{Q}_Y \;\approx\; \breve{Q}_X\, \breve{G}(\breve{Q}_X, \breve{Q}_W, \breve{H}) \tag{5.31}$$

with

$$\breve{G}(\breve{Q}_X, \breve{H}, \breve{Q}_W) \;=\; \breve{H} \left(1 + (\frac{\breve{Q}_W}{\breve{Q}_X \breve{H}})^{\frac{1}{\alpha}}\right)^{\alpha} \tag{5.32}$$

Note that in this case the higher the SNR, the closer the distortion function to *unity*, not zero. Another important difference is that the distortion function in the cepstrum domain ($\breve{g}$) cannot be expressed explicitly. The reason is that the DCT of the multiplication of two signals does not lead to a clear relationship between the DCTs of each signal. This point is further explored in Section 6.2.

5.4 Noise Compensation using VTS in the Product Spectrum Domain

In this section, first the VTS approach to robust ASR is briefly reviewed. Then, the equations in the product spectrum domain at the log-FBE and the cepstral domains are derived. For a more detailed discussion about the VTS and the derivation of the equations please refer to Appendix B.

5.4.1 VTS Approach to Robust ASR

Taylor series allows expressing a function using polynomials as the basis functions. If the non-linear terms are discarded, a linear approximation of the function will be achieved. When the non-linearity function (log or power transformation) is applied, the relation between the noisy observation and other variables, namely clean part and the noise becomes non-linear. Such non-linearity becomes problematic when applying the minimum mean square error (MMSE) criterion to estimate the clean part from the noisy observation. Using the MMSE,

$$\hat{X}_{MMSE} = \mathbb{E}\{\tilde{X}|\tilde{Y}\} = \int_{\tilde{X}} \tilde{X}\, P(\tilde{X}|\tilde{Y}) d\tilde{X} \tag{5.33}$$

where $\mathbb{E}$ denote expectation, $\tilde{X}$ and $\tilde{Y}$ indicate the clean part and the noisy observation in *some*[8] domain, $\hat{X}_{MMSE}$ is the MMSE estimate of the clean part.

Solving (5.33) is not easy and involves making some simplifying assumptions. In VTS framework it is assumed that

- The noise (either additive or channel) follows a Gaussian distribution and an estimate of the mean and covariance matrix is available.
- The distribution of the clean features is assumed to be a GMM with M components, learned off-line from training data.
- The noisy observation $\tilde{Y}$ is distributed based on a GMM with M Gaussians and within each component $\tilde{X}$ and $\tilde{Y}$ are jointly Gaussian.
- The distortion function, $\tilde{G}$, which connects the $\tilde{X}$ and $\tilde{Y}$ in the $\tilde{Y} = \tilde{X} + \tilde{G}$ form, needs to be only evaluated at the means of the Gaussians. This approximation holds well if the elements on the main diagonal of the covariance matrices tends to zero, which in limit, it turns the Gaussian into impulse function at the centre. When the number of components of a GMM increases, the variance decreases and this, in turn, decreases the error associated with this assumption.

These assumptions, allow for rewriting the MMSE estimate for the X as follows

$$\hat{X}_{MMSE} \approx \tilde{Y} - \sum_{m=1}^{M} P(m|\tilde{Y})\, G(\mu^{\tilde{X}_m}, \mu^{noise}) = \tilde{Y} - \bar{\tilde{G}}(\mu^{\tilde{X}_m}, \mu^{noise}) \tag{5.34}$$

where $\bar{\tilde{G}}$ is the (weighted) mean of the distortion function and $P(m|Y)$ indicates the posterior probability of the m^{th} Gaussian. For computing the posterior probabilities the GMM of the noisy observation ($\tilde{Y}$) is required which is unknown and should be estimated.

[8] In this context, it can be either FBE domain after compression or (generalised) cepstrum domain.

The interpretation of (5.34) is important and needs to be discussed. Given that at the target domains (either frequency or quefrency) $\tilde{Y} = \tilde{X} + \tilde{G}$, the clean part $\tilde{X}$ can be computed by $\tilde{Y} - \tilde{G}$. However, the distortion function $\tilde{G}$ is unavailable. What the (5.34) does is as follows: the space is discretised using the means of the Gaussians, namely $(\mu^{\tilde{X}_m}, \mu^{noise})$ i.e., M points. Then, the distortion function is evaluated at these points and averaged using some weights. The weights are $P(m|\tilde{Y})$ which are adjusted based on the noisy observation $\tilde{Y}$ and its GMM. It bears some resemblance to the spectral subtraction in terms of clean part equals noisy observation minus some noise-related function. However, this approach has a statistical structure, benefits from the model of the clean part and does not suffer from side problems of spectral subtraction method such as over-subtraction. In addition, this approach is less sensitive to the accuracy of the noise estimate as the model of the clean part plays the major role.

What is known is the mathematical relationship between the noisy observation and the other variables (environment model) as well as the statistical distribution of the clean features and noise. However, because of the *non-linearity* of the function linking these variables together, the distribution of the noisy observation cannot be estimated *analytically*. If the relation was linear, and given that the involved variable are Gaussian, the problem of finding the GMM of the noisy observation could be easily solved.

Moreno et al [17] propose to employ Taylor series for linearisation. It allows for estimating the GMM of the $\tilde{Y}$ and consequently working out the MMSE estimation of the $\tilde{X}$ using (5.36). Figure 5.6 shows the elements of the VTS-based noise compensation process. Replacing the *log* with the power transformation or generalised logarithmic function leads to generalised VTS (gVTS). In this case the MMSE estimate takes the following form

$$\hat{X}_{MMSE} \approx \breve{Y} \sum_{m=1}^{M} P(m|\breve{Y}) \frac{1}{\breve{G}(\mu^{\breve{X}_m}, \mu^{noise})} = \frac{\breve{Y}}{\bar{\breve{G}}} \tag{5.35}$$

where $\bar{\breve{G}}$ is the (weighted) mean of the distortion function. Note that since $\breve{Y} = \breve{G}\breve{X}$, the noisy observation is divided by the $\bar{\breve{G}}$ while in the case of using the logarithm it was subtracted. Using power transformation increases the flexibility of the framework and leads to higher robustness. A detailed derivation of the VTS and gVTS is presented in Appendix B.

5.4.2 Noise Estimation

As Figure 5.6 illustrates, (g)VTS technique requires an estimate of the involved noise. Estimating the additive noise is well-studied due to its relevance in the speech enhancement and there are a wide range of methods for estimating it [12]. Here we utilise the simplest

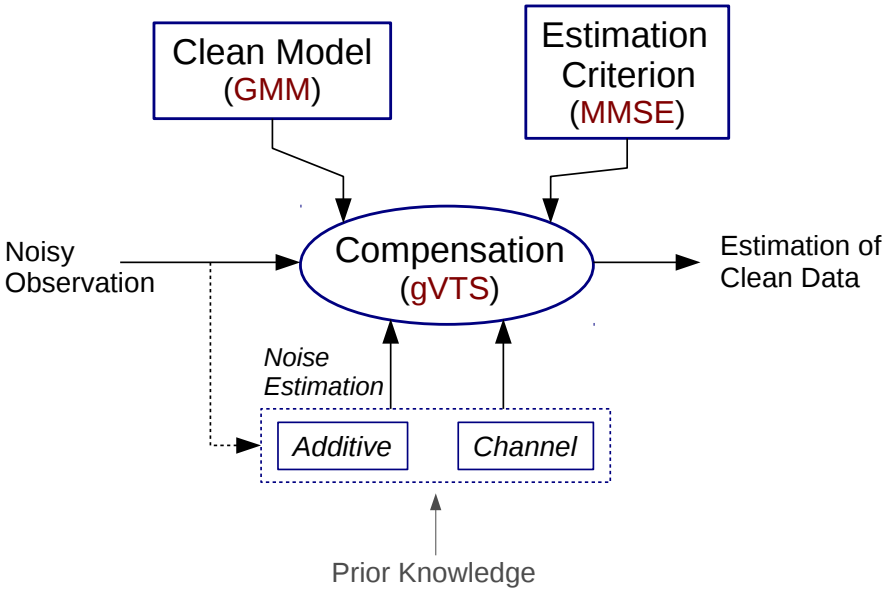

Fig. 5.6 Elements of the model-based noise compensation process. Clean model, noise estimate, estimation criterion and the compensation mechanism are the main parts of this process.

approach, namely using the first and last m frames of an utterance, assuming they are purely noise. Typically, the m is in range of 10-30 frames.

There is also a need for an estimate of the channel noise. A simple yet effective approach to deal with the channel effect is cepstral mean normalisation (CMN). Assuming the log function is used for compression, the channel and the clean part are additive in the cepstral domain. Assuming the channel is stationary and the mean of the cepstral coefficients of the clean speech at each quefrency over the whole utterance is zero, CMN removes the channel distortion

$$\tilde{y}[t,q] = \tilde{x}[t,q] + \tilde{h}[q] \Rightarrow \tilde{y}[t,q] - \frac{1}{T}\sum_{t=1}^{T} \tilde{y}[t,q] \approx \tilde{x}[t,q] \tag{5.36}$$

where $\tilde{z}[t,q]$ denotes the (real) cepstrum of z at frame t and quefrency q for $z \in \{y,x,h\}$ and T is the total number of frames. CMN is simple and easy to implement but the underlying argument is not perfect as the zero mean assumption may not hold well (especially for low quefrency components), presence of the additive noise makes the effect of the channel non-linear in the cepstral domain and ideally the non-speech segments should not be considered in the channel mean computation and subtraction. Having said that, cepstral mean normalisation (or subtraction) is widely used in feature extraction and has a positive effect on the performance in the noisy conditions.

Mean normalisation in the cepstral domain and log-FBE domain are identical because the transform which links them together is linear. It can be shown that arithmetic mean normalisation in the log-FBE domain is equivalent to geometric mean normalisation (GMN) in the FBE domain

$$\begin{aligned} \tilde{y}[t,q] - \frac{1}{T}\sum_{t=1}^{T}\tilde{y}[t,q] &= C(\tilde{Y}[t,q] - \frac{1}{T}\sum_{t=1}^{T}\tilde{Y}[t,q]) \\ \tilde{Y}[t,q] - \frac{1}{T}\sum_{t=1}^{T}\tilde{Y}[t,q] &= log(Y / \sqrt[T]{\prod_{t=1}^{T} Y[t,k]}) \\ (\prod_{t=1}^{T} Y[t,k])^{\alpha} &= \sqrt[T]{\prod_{t=1}^{T} \check{Y}[t,k]} \end{aligned} \tag{5.37}$$

where C is the DCT matrix and k indicates discrete frequency. Therefore, GMN similar to CMN could be helpful in dealing with the channel distortion when the power transformation is used for compression.

However, relying only on the CMN or GMN to counter the channel mismatch is not sufficient in practice. In the (g)VTS framework, there is a room for coping with the channel distortion, if an estimate of the channel becomes available. However, (blind) channel noise estimation is not straightforward. In this regard, we developed a novel model-based channel estimation technique which uses the model of the clean data and (g)VTS framework. This method is iterative and converges quickly, mostly after 2 to 4 iterations. It can be used in both periodogram and product spectrum domains and also along with applying either log or power transformation as the compression function. For a detailed explanation of this technique please refer to Appendix C.

5.4.3 Deriving the VTS Equations in the Frequency Domain

Now we are at a point in which the VTS equations can be derived. In particular, the environment model in the target domain after applying the compression function is known along with the clean model as well as an estimate of both additive and channel noise. What should be done is linearising the function which links the noisy observation to other variables and working out the GMM of the noisy observation. The VTS equations are first derived in the log-FBE (frequency) domain and then they will be derived in the cepstral (quefrency) domain.

In this regard, (5.23) should be expanded using the Taylor series and then only the linear term should be preserved to linearise the non-linear relationship between $\tilde{Q}_Y$ and other

variables in (5.23). The first-order Taylor series expansion leads to the following

$$\tilde{Q}_Y \approx \tilde{Q}_{Y_0} + J^{\tilde{Q}_X}(\tilde{Q}_X - \tilde{Q}_{X_0}) + J^{\tilde{Q}_W}(\tilde{Q}_W - \tilde{Q}_{W_0}) + J^{\tilde{H}}(\tilde{H} - \tilde{H}_0) \tag{5.38}$$

where J^Z denotes the Jacobian matrix (partial derivatives) of the $\tilde{Q}_Z$ with respect to Z for $Z \in \{\tilde{Q}_X, \tilde{Q}_W, \tilde{H}\}$ and $(\tilde{Q}_{X_0}, \tilde{Q}_{W_0}, \tilde{H}_0)$ is the point about which the function is linearised. In practice, the linearisation is performed around the mean vectors of the Gaussians, namely $(\mu_m^{\tilde{X}}, \mu^{\tilde{H}}, \mu^{\tilde{W}})$ i.e., M points. With some algebraic manipulation the Jacobians can be computed as follows

$$J_m^{\tilde{Q}_X} = \frac{\partial \tilde{Q}_Y}{\partial \tilde{Q}_X}\bigg|_{(\mu_m^{\tilde{Q}_X}, \mu^{\tilde{H}}, \mu^{\tilde{Q}_W})} = diag\{\frac{1}{1+V_m}\} \tag{5.39}$$

$$J_m^{\tilde{H}} = \frac{\partial \tilde{Q}_Y}{\partial \tilde{H}}\bigg|_{(\mu_m^{\tilde{Q}_X}, \mu^{\tilde{H}}, \mu^{\tilde{Q}_W})} = J_m^{\tilde{X}} = diag\{\frac{1}{1+V_m}\} \tag{5.40}$$

$$J_m^{\tilde{Q}_W} = \frac{\partial \tilde{Q}_Y}{\partial \tilde{Q}_W}\bigg|_{(\mu_m^{\tilde{Q}_X}, \mu^{\tilde{H}}, \mu^{\tilde{Q}_W})} = J_m^{\tilde{X}} = I - J_m^{\tilde{X}} = diag\{\frac{V_m}{1+V_m}\} \tag{5.41}$$

where $diag\{.\}$ indicates the operator which turns a vector to a diagonal matrix, I is Identity matrix and

$$V_m = exp(\mu^{\tilde{Q}_W} - \mu_m^{\tilde{Q}_X} - \mu^{\tilde{H}}). \tag{5.42}$$

We want to estimate the $\tilde{Q}_X$ from the noisy observation, $\tilde{Q}_Y$. As was shown in Figure 5.6, the (g)VTS-based framework consists of four main building blocks, namely the model of the clean features, an estimate of the additive/channel noise, estimation criterion and a compensation scheme. For modelling the clean features, a GMM with M Gaussians is employed and each noise type is modelled by a single Gaussian

$$\begin{cases} \tilde{Q}_X \sim \sum_{m=1}^{M} p_{\tilde{Q}_X}(m)\, \mathcal{N}(\tilde{Q}_X; \mu_m^{\tilde{Q}_X}, \Sigma_m^{\tilde{Q}_X}) \\ \tilde{Q}_W \sim \mathcal{N}(\mu^{\tilde{Q}_W}, \Sigma^{\tilde{Q}_W}) \\ \tilde{H} \sim \mathcal{N}(\mu^{\tilde{H}}, \Sigma^{\tilde{H}}) \end{cases} \tag{5.43}$$

where M, $p_{\tilde{Q}_X}(m)$, μ and Σ denote the mixture's components, weight, mean vector and covariance matrix, respectively. As an estimation criterion, MMSE[9] is used and finally, VTS

[9]The advantages of MMSE over maximum a posteriori (MAP) for this particular problem are discussed in Appendix B.

approach is utilised for carrying the noise compensation out. Using MMSE leads to

$$\hat{Q}_X^{MMSE} = \mathbb{E}[\tilde{Q}_X|\tilde{Q}_Y] = \tilde{Q}_Y - \sum_{m=1}^{M} P(m|\tilde{Q}_Y)\, \tilde{G}(\mu_m^{\tilde{Q}_X}, \mu^{\tilde{H}}, \mu^{\tilde{Q}_W}). \tag{5.44}$$

The only missing part in (5.44) is the $P(m|\tilde{Q}_Y)$. As mentioned, it is assumed that $\tilde{Q}_Y$, similar to $\tilde{Q}_X$, follows a GMM distribution with M components

$$\tilde{Q}_Y \sim \sum_{m=1}^{M} p_{\tilde{Q}_Y}(m)\, \mathcal{N}(\tilde{Q}_Y; \mu_m^{\tilde{Q}_Y}, \Sigma_m^{\tilde{Q}_Y}). \tag{5.45}$$

This assumption translates the problem into estimating the statistics of $\tilde{Q}_Y$. Using the linear relationship supplied by the Taylor series, namely (5.38), the GMM of $\tilde{Q}_Y$, i.e. $\{p_{\tilde{Q}_Y}(m), \mu_m^{\tilde{Q}_Y}, \Sigma_m^{\tilde{Q}_Y}\}$ can be computed as follows

$$\begin{cases} p_{\tilde{Q}_Y}(m) & \approx\ p_{\tilde{Q}_X}(m) \\ \mu_m^{\tilde{Q}_Y} & \approx\ \mu_m^{\tilde{Q}_X} + \mu^{\tilde{H}} + log(1+V_m) \\ \Sigma_m^{\tilde{Q}_Y} & \approx\ J_m^{\tilde{Q}_X} \Sigma_m^{\tilde{Q}_X} {J_m^{\tilde{Q}_X}}^T + J^{\tilde{H}} \Sigma^{\tilde{H}} {J_m^{\tilde{H}}}^T + J_m^{\tilde{Q}_W} \Sigma^{\tilde{Q}_W} {J_m^{\tilde{Q}_W}}^T \end{cases} \tag{5.46}$$

Having estimated the GMM of $\tilde{Q}_Y$, $P(m|\tilde{Q}_Y)$ can be computed using Bayes' rule

$$p(m|\tilde{Q}_Y) = \frac{p_{\tilde{Q}_Y}(m)\, p(\tilde{Q}_Y|m)}{p(\tilde{Q}_Y)} = \frac{p_{\tilde{Q}_Y}(m)\, \mathcal{N}(\tilde{Q}_Y; \mu_m^{\tilde{Q}_Y}, \Sigma_m^{\tilde{Q}_Y})}{\sum_{m'=1}^{M} p_{\tilde{Q}_Y}(m')\, \mathcal{N}(\tilde{Q}_Y; \mu_{m'}^{\tilde{Q}_Y}, \Sigma_{m'}^{y})} \tag{5.47}$$

which allows for calculating $\hat{Q}_X^{MMSE}$ using (5.44).

5.4.4 Deriving the VTS Equations in the Quefrency Domain

The noise compensation may be also carried out in the cepstrum domain. In this case, first the DCT of (5.38) should be calculated

$$\tilde{q}_y \approx \tilde{q}_{y_0} + J^{\tilde{q}_x}(\tilde{q}_x - \tilde{q}_{x_0}) + J^{\tilde{q}_w}(\tilde{q}_w - \tilde{q}_{w_0}) + J^{\tilde{h}}(\tilde{h} - \tilde{h}_0) \tag{5.48}$$

The second step is to compute the Jacobians. In the previous case, compensation in the frequency domain, $J_m^{\tilde{Z}}$ for $Z \in \{\tilde{Q}_X, \tilde{Q}_W, \tilde{H}\}$ was computed. Although one may derive the partial derivatives from scratch, the following equation helps in computing $J_m^{\tilde{z}}$ using $J_m^{\tilde{Z}}$

$$J_m^{\tilde{z}} = C\, J_m^{\tilde{z}}\, C^{-1} \tag{5.49}$$

where C^{-1} denotes the IDCT matrix and for any $z \in \{\tilde{q}_x, \tilde{q}_w, \tilde{h}\}$. As such

$$J_m^{\tilde{q}_x} = C\ diag\{\frac{1}{1+V_m}\}\ C^{-1} \tag{5.50}$$

$$J_m^{\tilde{h}} = J_m^{\tilde{x}} = C\ diag\{\frac{1}{1+V_m}\}\ C^{-1} \tag{5.51}$$

$$J_m^{\tilde{q}_w} = I - J_m^{\tilde{x}} = C\ diag\{\frac{V_m}{1+V_m}\}\ C^{-1}, \tag{5.52}$$

and

$$V_m = exp(C^{-1}\ (\mu^{\tilde{q}_w} - \mu_m^{\tilde{q}_x} - \mu^{\tilde{h}})). \tag{5.53}$$

Consequently the MMSE estimate for $\tilde{q}_x$ can be calculated as follows

$$\hat{q}_x^{MMSE} = \mathbb{E}[\tilde{q}_x|\tilde{q}_y] = \tilde{q}_y - \sum_{m=1}^{M} P(m|\tilde{q}_y)\ \tilde{g}(\mu_m^{\tilde{q}_x}, \mu^{\tilde{h}}, \mu^{\tilde{q}_w}) \tag{5.54}$$

where $\tilde{g}$ is the distortion function in the cepstrum domain, as defined in (5.28). Now the GMM parameters of $\tilde{q}_y$ can be derived

$$\begin{cases} p_{\tilde{q}_y}(m) & \approx\ p_{\tilde{q}_x}(m) \\ \mu_m^{\tilde{q}_y} & \approx\ \mu_m^{\tilde{q}_x} + \mu^{\tilde{H}} + log(1+V_m) \\ \Sigma_m^{\tilde{q}_y} & \approx\ J_m^{\tilde{q}_x}\Sigma_m^{\tilde{q}_x}{J_m^{\tilde{q}_x}}^T + J_m^{\tilde{h}}\Sigma^{\tilde{h}}{J_m^{\tilde{h}}}^T + J_m^{\tilde{q}_w}\Sigma^{\tilde{q}_w}{J_m^{\tilde{q}_w}}^T \end{cases} \tag{5.55}$$

The main advantage of performing the compensation process in the cepstrum domain is that the decorrelation provided by the DCT better matches the GMM's diagonal covariance matrices. However, as described in Appendix B, since the number of Gaussians (M) is usually high (say 256 or 512 components), a GMM with diagonal covariance matrices is capable of modelling the correlated data like log-FBEs. Therefore, assuming the mixture has sufficient components, there should be no significant difference between these two domains, practically. However, elevating the number of components increases the number of parameters of the GMM and this runs the risk of over-fitting unless enough training data is provided.

5.5 Generalised VTS in the Product Spectrum Domain

5.5.1 gVTS in the Frequency Domain

Based on the success of using the statistical normalisation (Section 4.5.3) and the power transformation (Section 4.6.1), one of the goals of this chapter was to couple the VTS

technique with the power transformation. The generalised VTS is the embodiment of this combination. Replacing the log with the power transformation results in the environment model expressed in (5.30). Combining this equation with the general formula of the MMSE estimate, namely (5.35), leads to the following estimator for $\breve{Q}_X$

$$\begin{aligned}\hat{Q}_X^{MMSE} &= \mathbb{E}[\breve{Q}_X|\breve{Q}_Y] = \int \breve{Q}_X\, P(\breve{Q}_X|\breve{Q}_Y) d\breve{Q}_X \\ &= \breve{Y} \sum_{m=1}^{M} P(m|\breve{Q}_Y)\, \frac{1}{\breve{G}(\mu_m^{\breve{Q}_X}, \mu^{\breve{H}}, \mu^{\breve{Q}_W})},\end{aligned} \tag{5.56}$$

where the distortion function $\breve{G}$ is defined in (5.32). The only unknown element in (5.56) is $P(m|\breve{Q}_Y)$ and to compute it, the statistics of $\breve{Q}_Y$ should be estimated. Similar to the VTS, it is assumed that $\breve{Q}_Y$ follows a GMM distribution with M components

$$\breve{Q}_Y \sim \sum_{m=1}^{M} p_{\breve{Q}_Y}(m)\, \mathcal{N}(\mu_m^{\breve{Q}_Y}, \Sigma_m^{\breve{Q}_Y}) \tag{5.57}$$

where $p_{\breve{Q}_Y}(m)$, μ and Σ denote the weight, mean vector and (diagonal) covariance matrix of $\breve{Q}_Y$, respectively. Expanding (5.30) using the Taylor series and keeping only the linear term results in

$$\breve{Q}_Y \approx \breve{Q}_{Y_0} + J^{\breve{Q}_X}(\breve{Q}_X - \breve{Q}_{X_0}) + J^{\breve{Q}_W}(\breve{Q}_W - \breve{Q}_{W_0}) + J^{\breve{H}}(\breve{H} - \breve{H}_0) \tag{5.58}$$

where J^Z is the Jacobian matrix of $\breve{Q}_Z$ with respect to Z for $Z \in \{\breve{Q}_X, \breve{Q}_W, \breve{H}\}$ and $(\breve{Q}_{X_0}, \breve{Q}_{W_0}, \breve{H}_0)$ is the point about which the function is linearised. As in previous cases, linearisation is performed around the mean values of all the Gaussians, namely $(\mu_m^{\breve{X}}, \mu^{\breve{H}}, \mu^{\breve{W}})$ which are M points altogether. With some algebraic manipulation the Jacobians can be computed as follows

$$J_m^{\breve{Q}_X} = \left.\frac{\partial \breve{Q}_Y}{\partial \breve{Q}_X}\right|_{(\mu_m^{\breve{Q}_X}, \mu^{\breve{H}}, \mu^{\breve{Q}_W})} = diag\{\frac{\mu^{\breve{H}}}{(1+\breve{V}_m)^{1-\alpha}}\} \tag{5.59}$$

$$J_m^{\breve{H}} = \left.\frac{\partial \breve{Q}_Y}{\partial \breve{H}}\right|_{(\mu_m^{\breve{Q}_X}, \mu^{\breve{H}}, \mu^{\breve{Q}_W})} = diag\{\frac{\mu_m^{\breve{Q}_X}}{(1+\breve{V}_m)^{1-\alpha}}\} \tag{5.60}$$

$$J_m^{\breve{Q}_W} = \left.\frac{\partial \breve{Q}_Y}{\partial \breve{Q}_W}\right|_{(\mu_m^{\breve{Q}_X}, \mu^{\breve{H}}, \mu^{\breve{Q}_W})} = diag\{(\frac{\breve{V}_m}{1+\breve{V}_m})^{1-\alpha}\}, \tag{5.61}$$

where

$$\breve{V}_m = (\frac{\mu^{\breve{Q}_W}}{\mu_m^{\breve{Q}_X}\, \mu^{\breve{H}}})^{\frac{1}{\alpha}}. \tag{5.62}$$

Having computed the Jacobians, the GMM parameters of $\breve{Q}_Y$ will be

$$\begin{cases} P_{\breve{Q}_Y}(m) & \approx P_{\breve{Q}_X}(m) \\ \mu_m^{\breve{Q}_Y} & \approx \mu_m^{\breve{Q}_X}\ \mu^{\breve{H}}(1+(\frac{\mu^{\breve{Q}_W}}{\mu_m^{\breve{Q}_X}\mu^{\breve{H}}})^{\frac{1}{\alpha}})^{\alpha} \\ \Sigma_m^{\breve{Q}_Y} & \approx J_m^{\breve{Q}_X}\Sigma_m^{\breve{Q}_X}\ {J_m^{\breve{Q}_X}}^T + J_m^{\breve{Q}_W}\ \Sigma^{\breve{Q}_W}\ {J_m^{\breve{Q}_W}}^T + J_m^{\breve{H}}\ \Sigma^{\breve{H}}\ {J_m^{\breve{H}}}^T. \end{cases} \tag{5.63}$$

5.5.2 gVTS in the Cepstrum Domain

As mentioned earlier, in case of using the power transformation the distortion function can not be computed explicitly in the frequency domain. Note that since a large M is used in practice, the decorrelation supplied by the DCT is not a necessary requirement to have a good modelling using a GMM with diagonal covariance matrices. Therefore, one can conduct the compensation in the frequency domain and gets the full advantage of the gVTS approach. However, for completeness, the equations are also derived in the cepstral domain.

Let us first rewrite (5.54) as follows

$$\begin{aligned} \hat{Q}_X^{MMSE} &= \mathbb{E}[\breve{Q}_X|\breve{q}_y] = \int \breve{Q}_X\, P(\breve{Q}_X|\breve{q}_y)\, d\breve{Q}_X \\ &= \left[C^{-1}\,\breve{q}_y\right] \sum_{m=1}^{M} P(m|\breve{q}_y)\, \frac{1}{\breve{G}(C^{-1}\mu_m^{\breve{q}_x}, C^{-1}\mu^{\breve{h}}, C^{-1}\mu^{\breve{q}_w})}, \end{aligned} \tag{5.64}$$

where $\mathbb{E}[\breve{Q}_X|\breve{q}_y]$ means that the observation is made in the cepstrum domain but the compensation takes place in the frequency domain. This allows for taking advantage of the statistical modelling in the cepstrum domain and at the same time bypassing the problem of unavailability of the distortion function in the cepstrum domain. The next step is to compute the $P(m|\breve{q}_y)$. Note that since the DCT is a linear transform, the likelihood and consequently the posterior probabilities, for a GMM model, do not change

$$\begin{cases} p_{\breve{z}}(m) = p_{\breve{Z}}(m) \\ P(\breve{z}|m) = P(\breve{Z}|m) \end{cases} \Rightarrow P(m|\breve{z}) = P(m|\breve{Z}) \tag{5.65}$$

where $\breve{z} = C\,\breve{Z}$ and C is the DCT matrix. Therefore, $p(m|\breve{q}_y) = p(m|\breve{Q}_Y)$.

First-order Taylor series expansion of the environment model in the cepstrum domain after applying power transformation, namely (5.30), results in

$$\breve{q}_y \approx \breve{q}_y(\breve{q}_{x_0}, \breve{q}_{w_0}, \breve{h}_0) + J^{\breve{q}_x}(\breve{q}_x - \breve{q}_{x_0}) + J^{\breve{q}_w}(\breve{q}_w - \breve{q}_{w_0}) + J^{\breve{h}}(\breve{h} - \breve{h}_0). \tag{5.66}$$

With some algebraic manipulation the Jacobians can be computed as follows

$$J_m^{\check{q}_x} = \frac{\partial \check{q}_y}{\partial \check{q}_x} = C\, diag\{\frac{\mu^{\check{H}}}{(1+\check{V}_m)^{1-\alpha}}\}\, C^{-1} \tag{5.67}$$

$$J_m^{\check{h}} = \frac{\partial \check{q}_y}{\partial \check{h}} = C\, diag\{\frac{\mu_m^{\check{X}}}{(1+\check{V}_m)^{1-\alpha}}\}\, C^{-1} \tag{5.68}$$

$$J_m^{\check{q}_w} = \frac{\partial \check{q}_y}{\partial \check{q}_w} = C\, diag\{(\frac{\check{V}_m}{1+\check{V}_m})^{1-\alpha}\}\, C^{-1} \tag{5.69}$$

where

$$\check{V}_m = (\frac{C^{-1}\mu^{\check{q}_w}}{(C^{-1}\mu_m^{\check{q}_x})\ (C^{-1}\mu^{\check{h}})})^{\frac{1}{\alpha}}. \tag{5.70}$$

Having computed the Jacobians, the GMM parameters of $\check{q}_y$ will be

$$\begin{cases} P_{\check{q}_y}(m) & \approx\ P_{\check{q}_x}(m) \\ \mu_m^{\check{q}_y} & \approx C\left[\left(C^{-1}\mu_m^{\check{q}_x}\right)\ \left(C^{-1}\mu^{\check{h}}\right)(1+\check{V}_m)^{\alpha}\right] \\ \Sigma_m^{\check{q}_y} & \approx\ J_m^{\check{q}_x}\Sigma_m^{\check{q}_x}\ {J_m^{\check{q}_x}}^T + J_m^{\check{q}_w}\ \Sigma^{\check{q}_w}\ {J_m^{\check{q}_w}}^T + J_m^{\check{h}}\ \Sigma^{\check{h}}\ {J_m^{\check{h}}}^T. \end{cases} \tag{5.71}$$

After estimating the GMM of the $\check{q}_y$, the posterior probabilities and finally the MMSE estimate of $\check{Q}_x$ can be calculated.

In practice, for mathematical convenience, the covariance matrices of the noisy observation are forced to be diagonal. In this regard, all the off-diagonal elements of the covariance matrices are set to zero through Hadamard (element-wise) multiplication of the covariance matrix by the Identity matrix. For instance, for gVTS in the cepstral domain

$$\Sigma_m^{\check{q}_y} \leftarrow \Sigma_m^{\check{q}_y}\ \odot I \tag{5.72}$$

where $\odot$ denotes the Hadamard multiplication.

5.6 Experimental Results

In this section the experimental results are presented and discussed. First, the ASR set up and the parametrisation process are explained and in the next subsection the ASR results for the clean and multi-style (a.k.a. multi-condition) training modes in a standard GMM-HMM system and a bottleneck DNN-based system are presented and discussed.

5.6.1 Parametrisation and ASR Setup

ASR experiments were conducted on the Aurora-4 [11] database which is a medium to large vocabulary noisy speech recognition task based on the Wall Street Journal (WSJ0) corpus. HMMs were trained with 16 components per mixture and all the acoustic models were standard phonetically state-clustered triphones trained from scratch using a standard HTK recipe [162]. Decoding was performed with the standard 5k-word WSJ0 bigram language model. The evaluation set consists of 14 subsets, grouped into 4 test sets: clean, (additive) noisy, clean with channel mismatch and noisy with channel mismatch, referred to as A, B, C and D, respectively. SNR of noisy test sets is ranged between 5 to 15 dB with average of 10 dB. As well as clean training data (*CL*), Aurora-4 has two extra sets for multi-style training, namely $Multi1(M1)$ and $Multi2(M2)$. Training data in the former is contaminated with only additive noise whereas in the latter both additive noise and channel distortion are present. In both cases, SNR of training samples ranged between 10 to 20 dB with average of 15 dB. For the DNN part, the network consists of four hidden layers with 1300 nodes, followed by a bottleneck (BN) [172] layer containing 26 nodes placed just before the output layer. The network was trained using TNET [179] and standard HMM-GMM models were trained on the BN features. For more details about the employed DNN set up please refer to Appendix D.

All the static features are post-processed using cepstral mean normalisation (CMN). Geometric mean normalisation (GMN) is only employed for the gVTS noise compensation where the power transformation is used for compression. Note that it is not applicable (N/A) when the log function is used for compression. The feature vector is augmented by c_0, delta and acceleration coefficients. Number of components of the GMM of the clean features, M, was set to 512 and additive noise was estimated using the first/last 30 frames. The product spectrum (PS) was parametrised in an MFCC-like framework through replacing the periodogram with the product spectrum [42]. A generalised PS (gPS) feature was calculated by replacing the log with the power transformation. Please refer to Appendix F for more details. (g)VTS1 indicates the compensation is performed only for the additive noise whereas (g)VTS2 denotes that the compensation has been carried out for both additive and channel distortion. For the (g)VTS2, the channel was estimated based on the method explained in Appendix C using three iterations. Finally, the average WER was computed in two ways:

$$\begin{aligned} Ave_1 &= \frac{A+6B+C+6D}{14} \\ Ave_2 &= \frac{A+B+C+D}{4} \end{aligned} \tag{5.73}$$

where in the Ave_1 the weight of each test sets is determined based on the number of its subsets.

5.6.2 Discussion

Training with Clean Data

Table 5.1 shows the word error rate (WER) for different test sets. On average, gVTS in the product spectrum domain performs as well as in the power spectrum domain (Appendix B). It provides a significant accuracy improvement in the noisy conditions (test sets B, C and D) along with some WER reduction in the clean-matched conditions (test set A). This is a test which most of the noise robustness methods fail to pass. This means that gVTS enhances both robustness and discriminability of the features.

The optimal value for parameter α depends on the SNR and the noise type. In the presence of the additive noise, on average, the larger the α, the higher the robustness. It can be justified given that increasing α takes the FBEs to a space with a higher SNR, as explained in Section 4.6.1. Although, increasing α provides some SNR gain, it has a negative influence on the distribution of the FBEs and fitting the distribution with a GMM or a single Gaussian accompanies with higher error. In Appendix B (Figure B.4), the effect of α on the histograms of the FBEs is illustrated.

On the other hand, increasing this parameter intensifies the channel effect and in particular highlights any channel mismatch between the test and train data. In general, $0.05 - 0.1$ appears to be an optimal range which resembles the power spectrum domain. This stems from the fact that the dynamic range of both domains are similar. As can be observed in Table 5.1, the compensation in both cepstral and FBE domains leads to almost identical results.

Effect of the geometric mean normalisation is shown in the Table 5.2. Clearly, it improves the results when there is a channel mismatch (test sets C and D), specially when there is no additive noise in the background (test sets C). However, in the presence of additive noise (test set B) it slightly worsens the performance.

Table 5.3 shows the WER for the case where the additive noise is estimated using the median instead of the mean of the first/last m frames. Note that increasing m has a dual effect, it leads to having more data for noise estimation which may translate into more reliable estimation, but at the same time, it runs the risk of including speech frames in the noise estimation process. In noise estimation problem, speech frames act as outliers. Since the median is less sensitive to the outliers than the mean, its use permits to set m to a larger value which potentially leads to a better noise estimate and simultaneously controls the effect of

Table 5.1 *WER of the proposed method along the baselines for the Aurora-4 (HMMs trained on clean data).*

Feature	α	GMN	A	B	C	D	Ave_1	Ave_2
MFCC-Clean	log	N/A	7.0	33.7	23.6	49.9	38.0	28.6
MFCC-Multi1	log	N/A	9.1	18.4	23.4	35.9	25.6	21.7
MFCC-Multi2	log	N/A	10.7	17.0	19.1	31.3	22.8	19.5
PS-log	log$\leftrightarrow$ 0	N/A	7.1	33.7	23.7	49.9	38.1	28.6
gPS-0.05	0.05	-	7.0	25.3	23.2	42.9	31.4	24.6
gPS-0.075	0.075	-	7.5	22.6	23.4	41.2	29.5	23.7
gPS-0.1	0.1	-	8.1	22.1	25.6	40.8	29.4	24.1
VTS1-log-FBE	log$\leftrightarrow$ 0	N/A	6.1	21.9	22.3	39.4	28.3	22.4
gVTS1	0.05	-	6.7	19.9	21.9	37.9	26.8	21.6
gVTS1	0.075	-	7.2	18.9	22.9	37.4	26.3	21.6
gVTS1	0.1	-	7.6	18.9	24.0	37.6	26.5	22.0
gVTS1-log-CEP	log$\leftrightarrow$ 0	N/A	6.9	21.3	22.4	37.8	27.4	22.3
gVTS1	0.05	-	6.5	19.7	22.4	37.5	26.6	21.5
gVTS1	0.075	-	7.2	19.1	23.0	37.0	26.2	21.6
gVTS1	0.1	-	7.3	19.6	23.5	37.7	26.7	22.0

the outliers. In case of the Aurora-4 ASR task, using the median of the first/last 50 frames resulted in about 0.6% absolute WER reduction on average, compared with using the mean (Table 5.2). The * in the gVTS* in Tables 5.3-5.8 means that the additive noise is estimated using the median ($m = 50$) instead of the mean ($m = 30$).

Tables 5.4 and 5.5 show the effect of employing the proposed channel estimation method, presented in Appendix C. As can be seen in Table 5.4, the proposed iterative channel estimation technique requires 2-4 iterations to converge. Performance-wise as Tables 5.4 and Table 5.5 demonstrate, this technique leads to a significant performance gain (above 20% relative) in dealing with the channel mismatch (test set C). Another important point is that, if the Ave_2 (in which all the test sets have the same weight) is considered for comparison, the system trained only on the clean data outperforms the one trained on multi-style training data using MFCC. Performance-wise and generally speaking, in noisy test conditions a system trained on a multi-style data, forms the upper bound for a system trained on only clean data. To the best of our knowledge, gVTS reinforced with the proposed channel estimation method, is among a few techniques which can outperform a HMM/GMM set up trained by multi-style data without using special extra knowledge or imposing heavy constraints.

As mentioned earlier, the purpose of applying the GMN is to alleviate the channel mismatch effect. Comparing the Table 5.5 with Table 5.2 shows that estimating the channel leads to remarkably higher performance improvement in dealing with the channel mismatch

Table 5.2 *Effect of Applying GMN on the WER.*

Feature	α	GMN	A	B	C	D	Ave_1	Ave_2
VTS1-FBE	log	N/A	6.1	21.9	22.3	39.4	28.3	22.4
gVTS1	0.05	✓	6.6	20.3	20.8	37.2	26.6	21.2
gVTS1	0.075	✓	6.8	19.8	21.1	36.4	26.1	21.0
gVTS1	0.1	✓	7.4	19.5	21.9	36.5	26.1	21.3
VTS1-CEP	log	N/A	6.9	21.3	22.4	37.8	27.4	22.3
gVTS1	0.05	✓	6.6	20.0	21.2	36.6	26.2	21.1
gVTS1	0.075	✓	7.1	19.2	21.4	36.0	25.7	20.9
gVTS1	0.1	✓	7.4	18.9	22.1	36.2	25.7	21.2

Table 5.3 *Effect of estimating the additive noise using median on the WER.*

Feature	α	GMN	A	B	C	D	Ave_1	Ave_2
VTS1*-FBE	log	N/A	6.5	21.6	21.6	39.4	28.1	22.3
gVTS1*	0.05	✓	6.6	19.2	20.7	36.9	26.0	20.9
gVTS1*	0.075	✓	7.0	18.6	21.0	36.4	25.6	20.8
gVTS1*	0.1	✓	7.1	18.3	21.4	36.4	25.5	20.8
VTS1*-CEP	log	N/A	6.5	21.0	22.4	37.6	27.2	21.9
gVTS1*	0.05	✓	6.4	19.4	21.4	36.4	25.9	20.9
gVTS1*	0.075	✓	6.7	18.6	21.2	36.4	25.6	20.7
gVTS1*	0.1	✓	7.4	18.5	22.4	36.3	25.6	21.2

problem. One question arises at this point is that whether the GMN is needed after applying the channel estimation technique or not. Table 5.6 presents the WER when channel estimate is available and the GMN is not applied and Table 5.5 demonstrates the results in similar case where the GMN is utilised. Comparing these two tables shows that after estimating the channel with the proposed technique there is no need to the GMN.

Multi-style Training

Although the (g)VTS approach is originally designed for constructing robust systems when only the clean data is available, it still appears to be helpful in the multi-style training condition where the noisy data is available for training the system. Table 5.7 and Table 5.8 show the results when only additive noise is available for training (M1) and when both additive and channel noise is available for training (M2), respectively. As can be seen, in both M1 and M2 cases employing the (g)VTS leads to consistent performance improvement.

Table 5.4 *Effect of number of iterations in the proposed channel estimation method on the WER.*

Feature	α	GMN	iterations	A	B	C	D	Ave_1	Ave_2
gVTS2	0.05	✓	1	6.6	20.7	15.7	34.8	25.4	19.5
gVTS2	0.05	✓	2	6.6	20.7	14.3	34.5	25.1	19.0
gVTS2	0.05	✓	3	6.6	21.1	14.3	35.0	25.6	19.3
gVTS2	0.05	✓	4	6.6	21.2	14.5	35.1	25.6	19.3
gVTS2	0.05	✓	5	6.8	21.1	13.8	34.6	25.3	19.1
gVTS2	0.05	✓	6	7.5	22.4	16.2	35.5	26.5	20.4

Table 5.5 *WER of gVTS after adding channel estimation block to the noise compensation process. Effect of estimating the additive noise with the mean and median is shown.*

Feature	α	GMN	A	B	C	D	Ave_1	Ave_2
MFCC-Clean	log	N/A	7.0	33.7	23.6	49.9	38.0	28.6
MFCC-Multi1	log	N/A	9.1	18.4	23.4	35.9	25.6	21.7
MFCC-Multi2	log	N/A	10.7	17.0	19.1	31.3	22.8	19.5
VTS2	log	N/A	6.7	22.5	15.2	35.3	26.4	19.9
gVTS2	0.05	✓	6.6	21.1	14.3	35.0	25.6	19.3
gVTS2	0.075	✓	6.9	20.4	14.6	34.9	25.2	19.0
gVTS2	0.1	✓	7.0	19.8	15.3	34.3	24.7	19.1
VTS2*	log	N/A	6.5	21.7	14.6	35.4	26.0	19.5
gVTS2*	0.05	✓	6.5	20.2	13.9	34.3	24.8	18.7
gVTS2*	0.075	✓	7.1	19.8	15.0	34.0	24.7	19.0
gVTS2*	0.1	✓	7.4	19.6	15.4	33.9	24.5	19.1

Table 5.6 *WER of gVTS after adding channel estimation block to the noise compensation process and removing the GMN. Effect of estimating the additive noise with mean and median is shown.*

Feature	α	GMN	A	B	C	D	Ave_1	Ave_2
VTS2	log	N/A	6.7	22.5	15.2	35.3	26.4	19.9
gVTS2	0.05	-	6.7	20.3	15.1	34.6	25.1	19.2
gVTS2	0.075	-	6.8	20.2	15.0	34.6	25.0	19.1
gVTS2	0.1	-	7.1	19.2	15.7	34.5	24.6	19.1
VTS2*	log	N/A	6.5	21.7	14.6	35.4	26.0	19.5
gVTS2*	0.05	-	6.6	19.9	15.5	34.3	24.8	19.1
gVTS2*	0.075	-	7.1	19.4	15.7	34.5	24.7	19.2
gVTS2*	0.1	-	7.2	18.9	15.9	34.4	24.5	19.1

Table 5.7 *WER of gVTS for Aurora-4 in Multi1 (M1) training mode. In this case the training data consists of clean speech and additive noise.*

Feature	α	GMN	A	B	C	D	Ave_1	Ave_2
PS-log	$\log \leftrightarrow 0$	N/A	8.9	18.5	23.4	36.2	25.7	21.8
gPS-0.05	0.05	-	8.8	16.7	24.4	34.6	24.3	21.1
gPS-0.075	0.075	-	9.5	16.4	24.9	34.3	24.2	21.3
gPS-0.1	0.1	-	10.0	16.0	25.4	34.4	24.1	21.5
VTS2*	log	N/A	8.6	16.6	15.4	31.4	22.3	18.0
gVTS2*	0.05	-	8.6	15.8	15.1	31.1	21.8	17.7
gVTS2*	0.075	-	9.3	16.1	16.2	32.2	22.5	18.4
gVTS2*	0.1	-	9.8	16.1	17.0	32.1	22.6	18.8

Table 5.8 *WER of gVTS for Aurora-4 in Multi2 (M2) training mode. In this case the training data consists of clean speech, additive noise and channel distortion.*

Feature	α	GMN	A	B	C	D	Ave_1	Ave_2
PS-log	log	N/A	10.5	17.6	18.6	31.3	23.0	19.5
gPS-0.05	0.05	-	11.1	16.7	19.3	30.4	22.1	19.2
gPS-0.075	0.075	-	11.4	16.2	19.3	30.6	22.2	19.4
gPS-0.1	0.1	-	11.6	16.3	21.5	31.1	22.7	20.1
VTS2*	log	N/A	8.7	17.07	14.6	30.2	21.9	17.7
gVTS2*	0.05	-	9.7	16.3	13.9	29.6	21.3	17.4
gVTS2*	0.075	-	10.2	15.8	14.6	30.1	21.4	17.7
gVTS2*	0.1	-	10.4	16.3	15.0	29.8	21.6	17.9

5.6.3 Post-processing the gVTS with Statistical Normalisation Techniques

Table 5.10 illustrates the performance when the HMMs are trained by only clean data and features are post processed with statistical normalisation techniques such as CMVN and Gaussianisation. The statistical normalisation was applied to the static (13) and also to the whole feature vector including the static, delta and delta-delta coefficients (39). As can be seen, coupling the gVTS with the Gaussianisation and CMVN normalisation techniques can further improve the robustness. The best performance was achieved when the whole feature vector (39 elements) was Gaussianised. This led to 1.9% absolute and 9.7% relative WER reductions in average (Ave_2) in comparison with the MFCC system trained in multi-style (M2) mode which is a substantial gain. Another eye-capturing performance improvement is in case of the test set C in which the WER decreases down to 11.9%.

Table 5.9 *WER of gVTS for Aurora-4 in Clean training mode after post-processing the statistic (13) and all the features (39) with CMVN and Gaussianisation (Gauss) [13] statistical normalisation (stat-norm) techniques. GMN did not applied.*

Feature	α	stat-norm	features	A	B	C	D	Ave_1	Ave_2
PS-log	log	CMVN	static	6.5	28.1	20.3	45.8	33.6	25.2
gPS-0.05	0.05	CMVN	static	6.4	22.0	19.7	40.8	28.8	22.2
gPS-0.075	0.075	CMVN	static	6.4	21.4	20.0	40.8	28.5	22.1
gPS-0.1	0.1	CMVN	static	6.8	20.2	21.5	39.8	27.7	22.1
PS-log	log	CMVN	all	6.4	26.8	20.0	44.6	32.5	24.5
gPS-0.05	0.05	CMVN	all	6.3	21.7	19.4	41.5	28.8	22.1
gPS-0.075	0.075	CMVN	all	6.7	20.7	22.3	40.5	28.3	22.5
gPS-0.1	0.1	CMVN	all	7.0	19.4	22.5	39.4	27.2	22.1
PS-log	log	Gauss	static	7.3	26.6	21.7	41.0	31.0	24.1
gPS-0.05	0.05	Gauss	static	7.0	22.2	21.3	37.7	27.7	22.0
gPS-0.075	0.075	Gauss	static	7.4	21.2	21.0	37.7	27.3	21.8
gPS-0.1	0.1	Gauss	static	7.1	20.7	21.8	36.7	26.7	21.6
PS-log	log	Gauss	all	6.6	24.5	20.0	40.5	29.8	22.9
gPS-0.05	0.05	Gauss	all	6.3	20.4	19.9	37.4	26.7	21.0
gPS-0.075	0.075	Gauss	all	6.8	20.5	19.9	37.2	26.3	21.1
gPS-0.1	0.1	Gauss	all	6.7	18.8	20.7	37.2	26.0	20.9
VTS2*	log	CMVN	static	6.1	22.8	12.3	36.4	26.7	19.4
gVTS2*	0.05	CMVN	static	6.6	19.8	11.9	34.8	24.7	18.3
gVTS2*	0.075	CMVN	static	7.0	19.3	13.3	34.7	24.6	18.6
gVTS2*	0.1	CMVN	static	7.0	18.7	13.9	34.7	24.4	18.6
VTS2*	log	CMVN	all	6.3	22.1	12.2	35.8	26.1	19.1
gVTS2*	0.05	CMVN	all	6.7	19.4	11.9	34.8	24.6	18.2
gVTS2*	0.075	CMVN	all	6.7	18.8	12.4	34.4	24.1	18.1
gVTS2*	0.1	CMVN	all	7.0	18.4	13.6	34.7	24.2	18.4
VTS2*	log	Gauss	static	6.7	21.9	12.6	35.0	25.8	19.1
gVTS2*	0.05	Gauss	static	6.8	20.4	12.6	34.1	24.7	18.5
gVTS2*	0.075	Gauss	static	7.3	19.3	13.0	33.5	24.1	18.3
gVTS2*	0.1	Gauss	static	7.4	19.5	13.5	34.0	24.4	18.6
VTS2*	log	Gauss	all	6.1	20.6	11.8	34.6	24.9	18.3
gVTS2*	0.05	Gauss	all	**6.5**	**18.7**	**11.9**	**33.2**	**23.5**	**17.6**
gVTS2*	0.075	Gauss	all	6.8	18.7	12.1	33.7	23.8	17.8
gVTS2*	0.1	Gauss	all	7.2	18.4	13.2	34.3	24.1	18.3

5.6.4 Combining the gVTS with the DNN

Table 5.10 shows the effect of combining the (g)VTS noise compensation with the Bottleneck DNN-based system. In this case, the noisy observation first gets processed through (g)VTS

and then is sent to the Bottleneck (BN) DNN-based system. In the back-end, a standard GMM-HMM system is employed. As can be observed in Table 5.10, the gain in performance depends on the training conditions and such combination could be *superadditive*[10] or *subadditive*[11].

Generally speaking, DNN-based set ups cannot handle the structural mismatch well. For example, when only the clean data is available for training the DNN, it can deal with the clean unseen data effectively, however, it suffers in handling the noisy data. On the other hand, if the noisy data becomes available during the training, the DNNs can cope with the noise effectively, as far as the noise type is similar to the noisy data provided in the training phase. For instance, if the training data is only contaminated with the additive noise, the DNN-based system can deal with the additive noise well. However, if there is a channel mismatch, it can not successfully cope with the channel distortion. Having said that, if the training data includes (sufficient amount of) both additive noise and channel distortion, the DNNs will be well capable of handling both noise types and this leads to the state-of-the-art performance. Therefore, the comfort zone of the DNN where it handles the data variability reliably and effectively, depends on the training data. Any structural mismatch pushes the data outside the comfort zone of the DNN and results in poor performance.

On the other hand, the signal processing-based methods compensate for the lack of data using knowledge about how the noise contaminates the clean signal and also using the properties of the noise and clean signal. Since they mostly do not involve learning, they are less sensitive to the training data. However, the prior knowledge and the way which such knowledge are embedded in the parametrisation process plays a key role in success of such techniques.

Based on the aforementioned argument, if there is enough data covering the variabilities needed to be dealt with in the test stage, the DNN-based system can learn the optimal representation of the data. On the other hand, if the training data notably mismatch with the test condition, for example only additive noise is available for training and the system is going to be tested in a condition with channel mismatch, the gain in using the DNNs will be limited. In this circumstance, the signal processing techniques like gVTS can (to a certain extent) play a complementary role, alleviate the mismatch effect and map the data onto the comfort zone of the DNN. Figure 5.7 illustrates this point.

Table 5.10 shows the results of a combined gVTS/DNN ($BN\{gVTS\}$) system in the clean and multi-style conditions. When only the clean data is available for training, DNN on

[10] Superadditive combination (here) means the performance of the combined system is higher than each system individually.

[11] Subadditive combination refers to the case where the performance of the combined system is poorer than each system individually.

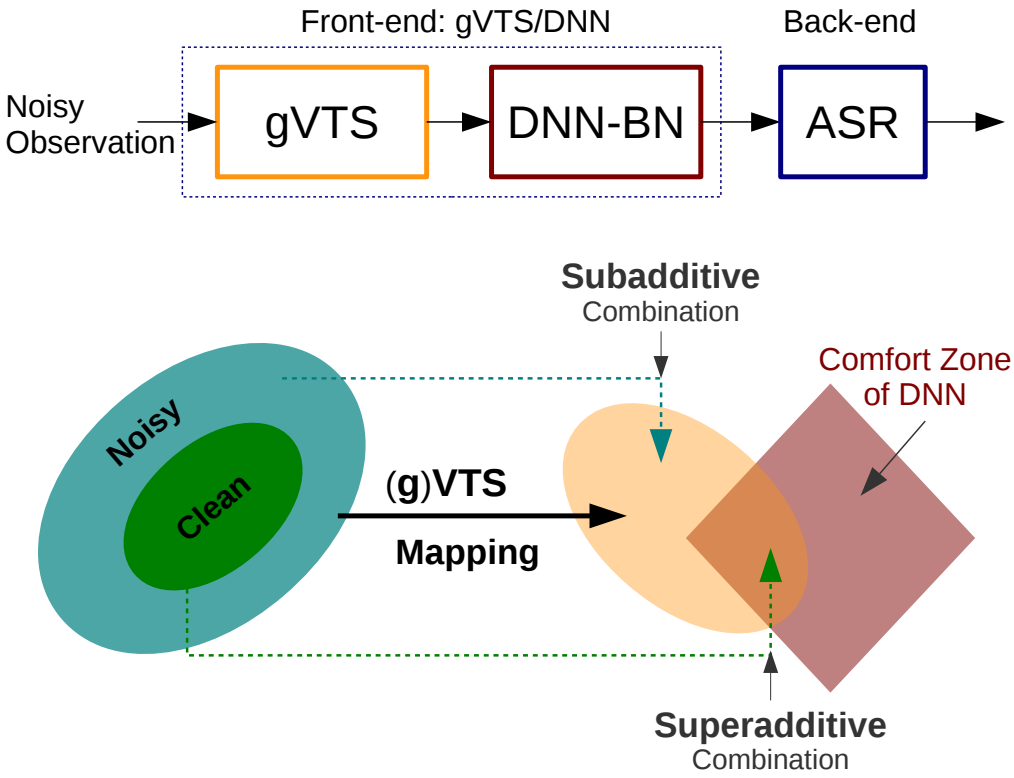

Fig. 5.7 Combination of the gVTS and the DNN system for ASR. The combination could be superadditive or subsadditive.

its own cannot deal with the variability induced by noise. However, combining it with the gVTS allows for keeping the performance similar to the conventional GMM-HMM system in the mismatch condition while benefiting from the DNN in the matched condition. In multi-style training, when only additive noise is available (M1), although DNN (on its own) leads to a significant performance improvement in dealing with the additive noise, it fails in coping with the channel mismatch. In this case, the gVTS can well play a complementary role. Finally, if the DNN is trained on both additive and channel noise (M2), although still its combination with gVTS could be useful in the test sets A and C, on average, the DNN system outperforms the gVTS/DNN system. Quantitatively, for example when the $\alpha = 0.1$, combining the gVTS with bottleneck DNN-based system led to average absolute (relative) performance improvements of 6.0 (23.5) when training on clean data; 2.5 (13.8) when using multi-style training with additive noise; but -0.5% (-3.7%) when using multi-style training with both additive and channel noise.

Table 5.10 *WER of the combined gVTS/DNN and the DNN(-alone) for Aurora-4 in clean and multi-style training modes.*

Feature	α	GMN	A	B	C	D	Ave_1	Ave_2
gPS-BN-CL	0.1	-	5.5	24.2	26.8	45.4	32.1	25.5
gVTS2-BN-CL	0.1	-	4.6	20.6	16.0	36.7	26.0	19.5
gPS-BN-M1	0.1	-	5.5	11.1	23.5	32.3	20.7	18.1
gVTS2-BN	0.1	-	5.3	12.4	14.3	30.6	19.8	15.6
gPS-BN-M2	0.1	-	5.7	10.8	13.0	24.7	16.6	13.6
gVTS2-BN-M2	0.1	-	5.6	11.9	12.3	26.5	17.8	14.1

5.7 Summary

This chapter explored the combination of the VTS method with the power transformation in a phase-related domain. The VTS is a well-principled noise compensation technique with the capability of dealing with both additive noise and channel distortion. However, in its original formulation, it assumes using the log function for compression. So, first a novel formulation for the VTS was developed when the power transformation is used instead of the log and it was called generalised VTS (gVTS). In the next step, the extension of this approach to the group delay domain was studied. In this regard, the environment model was derived in the group delay domain. It was discussed that deriving the (g)VTS formulation in the group delay domain is more complicated than the periodogram domain. The problems which this presents were discussed, some solutions were presented and the corresponding equations were derived. The effect of the additive and channel noise in the group delay domain were examined, too. It was shown that the group delays of the additive noise and clean part mix in a convex combination form and the group delay of the channel, in the expected sense, tends to zero. The latter was used in simplifying the environment model and deriving the (g)VTS equations. Experimental results on the Aurora-4 ASR task showed that a system trained only on the clean data using the proposed feature, on average, outperforms an MFCC-based system trained using multi-style data. Combination of the gVTS features with the bottleneck framework in clean training mode resulted in significant WER reductions in the clean-match test condition with minor performance loss in unmatched conditions. This potentially allows robust systems to be built using DNNs even when only clean training data is available. In multi-style training with only additive noise, combination of the gVTS with the DNN lead to significant performance improvement, especially in handling the channel mismatch. However, in multi-style training with both additive and channel noise, the DNN system slightly outperformed the hybrid gVTS/DNN system.

Chapter 6

Conclusion and Scope for Future Work

If I have seen further, it is by standing upon the shoulders of giants.
– Isaac Newton

Now this is not the end. It is not even the beginning of the end.
But it is, perhaps, the end of the beginning.
– Winston Churchill

6.1 Short Review of the Previous Chapters

Our relatively long journey is approaching the end. The first chapter, namely Introduction, laid out the foundation of this thesis, the goals, the contributions and the thesis structure. Chapter 2 provided a review of the related works and applications of the phase spectrum in speech processing. Chapter 3 aimed to study the signal's information regions. Using a Venn diagram, it was illustrated which pieces of information are unique to the magnitude and phase spectra and what fraction of information is shared by them. Also the relative importance of each part in short and long term were discussed and demonstrated. Chapter 4 targeted filling three black boxes, the goal of the first one was to extract the source and filter components from the phase spectrum and the second and third black boxes aimed at extracting feature from the filter element for ASR and estimating the pitch frequency from the source part. The effectiveness of the proposed framework was improved by replacing the log by generalised logarithmic function in the Hilbert transform and also via employing the regression function rather than sample difference in computing the group delay. In addition, the statistical behaviour of the phase spectrum and its representations along the parametrisation pipeline was studied. Furthermore, the influence of statistical normalisation techniques on the filter-based feature was examined. In Chapter 5, the effect of additive

and channel noise in the phase and group delay domains was studied. In addition, novel formulations for the (generalised) VTS and channel noise estimation in the group delay-power product spectrum domain were developed with substantial robustness in dealing with both additive noise and channel distortion.

6.2 Take-home Messages

6.2.1 Chapter 3

Chapter 3 investigated the information content of the phase spectrum. The main findings were as follows

– In general, the information content of a mixed-phase signal is decomposable into three disjoint regions, namely

- the all-pass part exclusively captured by phase
- scale information, uniquely residing in the magnitude spectrum
- the minimum-phase-scale-excluded part which is shared by both magnitude and phase spectra

The information content of the signal is constant regardless of the frame length but by changing the frame length, the relative importance of each part varies.

– In *short-term* processing (frame length: 20-40 ms), the minimum-phase component plays the central role and the all-pass element has a marginal significance. The importance of the minimum-phase part means that both phase and magnitude spectra are almost equally informative (apart from scale information which is unique to the magnitude). As such the fraction of signal information which is shared by phase and magnitude spectra is dominant. Consequently, the redundancy between the phase and magnitude information becomes maximum. In this condition, employing the phase spectrum along with an algorithm which already benefits from the magnitude spectrum information does not provide a noteworthy quality/intelligibility/performance improvement. This stems from the fact that what is unique about the phase, namely the all-pass component, has a minor role to play in the short-term analysis and the other type of information encoded in the phase, namely the minimum-phase, is already captured by the magnitude spectrum. However, this should not be misinterpreted as unimportance of the phase, or that the phase is devoid of perceptually important information (Ohm's Acoustic Law). As a matter of fact, this point could be stated the other way around, namely adding the magnitude information to an algorithm that has already deployed the

phase information (apart from the scale information), does not affect the performance, either. This, likewise, should not be misread as the unimportance of the magnitude spectrum.

– In *long-term* analysis the importance of the all-pass part increases and the significance of the minimum-phase component and the scale information decreases. As such the phase spectrum will include a larger fraction of the signal information and will become more informative whereas the the magnitude spectrum will lose its importance.

– The importance of the all-pass part in the long-term analysis stems from the fact that it contains the timing information of the signal. By frame length extension, the temporal resolution decreases. Therefore, localising the events in the time domain becomes more critical and that is what the all-pass part can help with as it carries the timing information. This elevates the importance of the phase spectrum and simultaneously leads to unimportance of the minimum-phase and the magnitude spectrum which are blind to the timing information that the all-pass part affords.

– The significance of the scale information, which the phase is blind to, decreases as frame length increases. That is why the quality/intelligibility of the phase-only reconstructed signal improves when expanding the frames because phase contains all the signal information except for the scale part. Note that in the extreme case in which the frame length equals the signal length, the scale information loses its importance from a perceptual standpoint. It should be emphasised that the scale information is important inter-frame-wise not intra-frame-wise. Therefore, its role is significant only when the frames should be joined together, for example in overlap-add synthesis.

– The reason why windows like rectangular and Chebyshev (25-35 dB) lead to quality improvement in case of the phase-only signal reconstruction was also explained.

6.2.2 Chapter 4

Chapter 4 aimed to develop a source-filter model in the phase domain. The key deliveries were as follows

– Source and filter components are additive in the unwrapped phase domain of the minimum-phase component. For separating them, first the unwrapped phase of the minimum-phase component was computed using the Hilbert transform. Then, the unwrapped phase was modelled using the Trend-plus-Fluctuation structure. The Trend is the slowly varying (modulating) component and is associated with the vocal tract. The Fluctuation part varies with a higher rate with respect to the independent variable (frequency) and is linked to the excitation element. These two elements were successfully dissociated through Trend Extraction filtering which is a low-pass filter (actually, low-time lifter) in essence. In comparison with the magnitude-based source-filter separation, the proposed phase-based

approach leads to better frequency resolution, lower spectral leakage and higher noise robustness.

– A set of features were extracted from the filter part and their discriminability and robustness were investigated in both a connected-digit recognition task (Aurora-2) and medium to large vocabulary continuous speech recognition task (Aurora-4), in the clean and multi-style training modes along with GMM-HMM and DNN-based set ups. The proposed approach returned superior performance when only clean data was available for training.

– The applicability and usefulness of the source component in fundamental frequency extraction was studied, too. Three magnitude-based techniques, namely HPS, Cepstrum and SRH were modified and employed for extracting the fundamental frequency from the group delay of the source component of the phase spectrum. The accuracy and robustness were evaluated. It was shown that in the clean condition, both phase-based techniques and their magnitude-based counterparts act as well as each other. However, in the noisy conditions the phase-based approach outperformed the magnitude-based methods.

– The evolution of the statistical behaviour of the phase and magnitude spectra and their representations along the parametrisation pipeline were investigated in the clean condition through estimating the histograms using more than 1.4 M frames (about 244 minutes) of speech signal. The true distribution of the unwrapped phase spectrum of the minimum-phase part was shown to be bell-shaped, contrary to the uniform assumption usually made for the phase spectrum. The uniform density was demonstrated to be correct only for the principle (wrapped) phase. It was explained that the uniform distribution is not a structural property of the phase spectrum but a consequence of the phase wrapping phenomenon. If the magnitude spectrum gets wrapped similarly to the phase spectrum, its distribution becomes uniform, too. We argued that assuming the uniform density for the phase spectrum is essentially paradoxical and cannot be true.

– The influence of conducting statistical normalisation across the proposed phase-based feature extraction pipeline using mean-variance normalisation, Gaussianisation, Laplacianisation and histogram equalisation (HEQ) was examined. The results showed that statistical normalisation of the phase-based features, similar to the magnitude-based representations, can enhance the noise robustness.

– Using the generalised logarithmic function (Box-Cox transformation) instead of the log in the Hilbert transform along with applying a regression filter instead of sample difference lead to a more robust source-filter separation. In particular, the former resulted in robustness improvement for the phase filter-based features and the latter was helpful in extracting the pitch frequency from the source part with higher accuracy.

– The reason behind the (useful) similarity of the group delay to the magnitude spectrum was investigated, too. Broadly speaking, the information gets encoded in the phase spectrum in a frequency modulation (FM) format, namely in the slopes rather than the amplitude. This is in contrast to the magnitude spectrum in which the information is encoded in an amplitude modulation (AM) format. Similar to the FM demodulation using *discriminator* circuit (also known as *slope detector*), taking the derivative demodulates the information and moves it to the amplitude, thus generating an AM signal out of an FM one. That is why in the group delay (result of the phase derivative), the information is encoded in the amplitude, analogous to the magnitude spectrum, resulting in the aforementioned similarity between them. Note that the frequency modulation structure of the phase spectrum could provide a relatively higher noise robustness than the amplitude modulation format of the magnitude spectrum.

6.2.3 Chapter 5

Chapter 5 was dedicated to developing the (generalised) VTS formulation in the group delay-power product spectrum domain. The main findings were as follows

– The environment model which mathematically underpins the relation between the noisy observation with the clean signal and the noise (both additive and channel) was derived in the group delay domain. It was shown that the combination of the clean signal and the additive noise in the group delay domain takes a *convex* format or a weighted sum form. The weight of the clean part is proportional to the SNR while the weight of the additive noise is inversely proportional to SNR. In the power spectrum domain, the periodograms of the clean signal and the additive noise are just summed together and the weights are constant irrespective of the SNR level.

– The channel behaviour in the phase and group delay domains was evaluated empirically. In this regard, the Fourier transform (FT) of the channel was estimated through comparing the FT of the clean signal with the FT of the noisy signal simultaneously recorded by a different microphone. The behaviour of the channel in terms of its magnitude spectrum, phase spectrum and group delay was investigated for 330 signals (40 minutes of speech all-together, recorded by 18 microphones). In particular, it was empirically observed that in the expected sense, the group delay of the channels tends to zero. This became instrumental in the later stages in which we explored the extension of the VTS idea to the group delay domain.

– The generalised VTS (gVTS) framework (Appendix B) which is a combination of the power transformation with VTS, was extended to the group delay-power product spectrum domain. The problems that this presents were discussed and some solutions were proposed. Finally, the equations were derived in both the filter bank energy and cepstral

domains. In addition, we proposed a novel method for blind channel estimation (Appendix C), successfully extended it to the aforementioned domain and integrated it into the gVTS framework. Remarkable robustness to additive noise and channel distortion was achieved without performance loss in the clean-matched condition.

6.3 Scope for Future Work

Our imagination is the only limit to what we can hope to have in the future.
– Charles Kettering

There are several lines of research arising from this work and are worthwhile for further investigation. In this section a number of suggestions for future directions are listed.

- Given the centrality of the source-filter model in speech processing, the proposed source-filter model in the phase domain, as well as removing the doubts about the usefulness of the phase and its information content, opens up the possibility of applying the phase spectrum in a wide range of applications in speech processing such as speech analysis, speech coding, speech synthesis and phase-based feature/pattern extraction for a wide range of speech-related classification tasks.

- The relative importance of the source and filter components primarily depends on the application. However, in many cases this does not mean that the less important part is absolutely useless. For example in ASR for tonal languages, although the filter element plays the central role, the source component also includes useful information. Having separated the source and filter one may either generate a new representation by constructing a weighted mean of them which includes both components but with more weight given to the more relevant part or use them separately in a multi-stream framework and fuse the results in the later stages.

- The proposed gVTS formulation could be equally well implemented in both magnitude and product spectra domains. By simultaneously conducting the noise compensation in both domains and dividing the enhanced product spectrum by the enhanced magnitude spectrum, the enhanced group delay and consequently the enhanced phase spectrum can be computed. This could be useful in speech enhancement and also for denoising the phase-based features.

- The proposed generalised VTS approach rests upon the Taylor Series for linearising the involved non-linearity. Rather than using the Taylor Series, the linearisation can

instead be carried out using the Unscented Transform [180]. This can lead to a more accurate linearisation and statistical characterisation of the noisy observation paving the way for achieving a higher level of enhancement/robustness.

- Merging the magnitude-based and phase-based features in a DNN-based feature extraction scheme such as the bottleneck framework could be another avenue for future work. Although the end-to-end paradigm which directly takes the waveform in the time domain as input appears to benefit from both the phase and magnitude spectra, using the phase and magnitude-based features allow for integrating our knowledge about the spectro-temporal properties of the speech/noise into parametrisation pipeline. There is no room to embed such a priori knowledge into the end-to-end framework.

- As shown the true distribution of the phase spectrum has a bell-shaped form, in contrast to the uniform assumption made by mainstream speech enhancement techniques. Assuming that the phase spectrum has a bell-shaped distribution and conducting further investigations to parametrise its density appropriately may be helpful in deriving more powerful estimators for speech enhancement.

- Studying the relative importance of the signal information regions and their sensitivity to the noise is another direction for future work. Using a collection of *stereo data*[1], one can replace the minimum-phase, all-pass and scale information of the noisy signal by their clean counterparts. Measuring the quality improvement of the synthesised speech relative to the noisy one gives an estimate of the relative importance of each part. This will shed light on the importance of minimum-phase, all-pass and scale information and quantify their significance/usefulness in dealing with the noise.

[1] Data that consists of simultaneous recordings of both the clean and noisy speech.

References

[1] S.B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *ACOUSTICS, SPEECH AND SIGNAL PROCESSING, IEEE TRANSACTIONS ON*, pp. 357–366, 1980.

[2] G. S. Ohm, "Ueber die definition des tones, nebst daran geknüpfter theorie der sirene und ähnlicher tonbildender vorrichtungen," 1843.

[3] R.S. Turner, "The ohm-seebeck dispute, hermann von helmholtz, and the origins of physiological acoustics," *The British Journal for the History of Science*, vol. 10, no. 1, pp. 1–24, 1977.

[4] H.V. Helmholtz and A.J. Ellis, *On the sensations of tone as a physiological basis for the theory of music / by Herman L.F. Helmholtz*, Longmans, Green London, 1885.

[5] J.F. Schouten, R.J. Ritsma, and B. Lopes Cardozo, "Pitch of the residue," *The Journal of the Acoustical Society of America*, vol. 34, no. 9B, pp. 1418–1424, 1962.

[6] A.V. Oppenheim and J.S. Lim, "The importance of phase in signals," *Proceedings of the IEEE*, vol. 69, no. 5, pp. 529–541, May 1981.

[7] L. Liu, J. He, and G. Palm, "Effects of phase on the perception of intervocalic stop consonants," *Speech Communication*, vol. 22, no. 4, pp. 403–417, 1997.

[8] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 32, no. 6, pp. 1109–1121, Dec 1984.

[9] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 2, pp. 443–445, Apr 1985.

[10] D. Pearce and H.G. Hirsch, "The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions.," in *INTERSPEECH*. 2000, pp. 29–32, ISCA.

[11] N. Parihar and J. Picone, "Aurora working group: Dsr front end lvcsr evaluation au/384/02," *Inst. for Signal and Information Process, Mississippi State University, Tech. Rep*, vol. 40, pp. 94, 2002.

[12] P.C. Loizou, *Speech Enhancement: Theory and Practice*, CRC Press, Inc., Boca Raton, FL, USA, 2nd edition, 2013.

[13] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *2001: A Speaker Odyssey - The Speaker Recognition Workshop*, Crete, Greece, 2001, pp. 213–218, International Speech Communication Association (ISCA).

[14] A. Torre, A.M. Peinado, J.C. Segura, J.L. Perez-Cordoba, M.C. Benitez, and A.J. Rubio, "Histogram equalization of speech representation for robust speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 3, pp. 355–366, May 2005.

[15] G.E.P. Box and D.R. Cox, "An analysis of transformations," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 26, no. 2, pp. 211–252, 1964.

[16] S. Furui, "Speaker independent isolated word recognition using dynamic features of speech spectrum," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 1, pp. 52–59, 1986.

[17] P.J. Moreno, B. Raj, and R.M. Stern, "A vector taylor series approach for environment-independent speech recognition," in *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*. IEEE, 1996, vol. 2, pp. 733–736.

[18] A.V. Oppenheim and R.W. Schafer, *Discrete-Time Signal Processing*, Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edition, 2009.

[19] J.S. Lim and A.V. Oppenheim, Eds., *Advanced Topics in Signal Processing*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1987.

[20] Wikipedia, "Uncertainty principle — wikipedia, the free encyclopedia," 2017, [Online; accessed 30-October-2017].

[21] J.R.J. Deller, J.H.L. Hansen, and J.G. Proakis, *Discrete-Time Processing of Speech Signals*, An IEEE Press classic reissue. Wiley, 2000.

[22] Y. Hu and P.C. Loizou, "Subjective comparison and evaluation of speech enhancement algorithms," *Speech Communication*, vol. 49, no. 7-8, pp. 588–601, July 2007.

[23] G.E.P. Box, G.M. Jenkins, and G.C. Reinsel, *Time Series Analysis: Forecasting and Control*, Wiley Series in Probability and Statistics. Wiley, 2013.

[24] S.M. Kay, *Modern Spectral Estimation: Theory and Application/Book and Disk*, Prentice-Hall Signal Processing Series: Advanced monographs. PTR Prentice Hall, 1988.

[25] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 27, no. 2, pp. 113–120, Apr 1979.

[26] S.S. Haykin, *Communication systems*, Wiley, 2001.

[27] D.C. Ghiglia and M.D. Pritt, *Two-Dimensional Phase Unwrapping: Theory, Algorithms and Software*, WileyBlackwell, May 1998.

[28] S. Chavez, Qing-San Xiang, and L. An, "Understanding phase maps in mri: a new cutline phase unwrapping method," *IEEE Transactions on Medical Imaging*, vol. 21, no. 8, pp. 966–977, Aug 2002.

[29] D. Danudirdjo and A. Hirose, "Anisotropic phase unwrapping for synthetic aperture radar interferometry," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 7, pp. 4116–4126, July 2015.

[30] J.A. Quiroga, M. Servin, and F. Cuevas, "Modulo 2π fringe orientation angle estimation by phase unwrapping with a regularized phase tracking algorithm," *J. Opt. Soc. Am. A*, vol. 19, no. 8, pp. 1524–1531, Aug 2002.

[31] J. Tribolet, "A new phase unwrapping algorithm," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 25, no. 2, pp. 170–177, Apr 1977.

[32] T. Drugman and Y. Stylianou, "Fast and accurate phase unwrapping," in *INTERSPEECH*, 2015.

[33] M. Gdeisat and F. Lilley, "One-dimensional phase unwrapping problem," *signal*, vol. 4, pp. 6, 2012.

[34] Messaoud Benidir, "On the root distribution of general polynomials with respect to the unit circle," *Signal Processing*, vol. 53, no. 1, pp. 75 – 82, 1996.

[35] L.H. Keel and S.P. Bhattacharyya, "Root counting, phase unwrapping, stability and stabilization of discrete time systems," *Linear Algebra and its Applications*, vol. 351, no. Supplement C, pp. 501 – 518, 2002, Fourth Special Issue on Linear Systems and Control.

[36] A.V. Oppenheim, A.S. Willsky, and S.H. Nawab, *Signals &Amp; Systems (2Nd Ed.)*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1996.

[37] John Makhoul, "Linear prediction: A tutorial review," *Proceedings of the IEEE*, vol. 63, no. 4, pp. 561–580, 1975.

[38] B. Yegnanarayana and H.A. Murthy, "Significance of group delay functions in spectrum estimation," *IEEE Transactions on Signal Processing*, vol. 40, no. 9, pp. 2281–2289, 1992, cited By (since 1996)52.

[39] L. Rabiner and R. Schafer, *Theory and Applications of Digital Speech Processing*, Pearson Education, 2011.

[40] H.A. Murthy and V. Gadde, "The modified group delay function and its application to phoneme recognition," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International conference on*, April 2003, vol. 1, pp. I–68–71 vol.1.

[41] R.M. Hegde, H.A. Murthy, and V.R.R. Gadde, "Significance of the modified group delay feature in speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 1, pp. 190–202, Jan 2007.

[42] D. Zhu and K.K. Paliwal, "Product of power spectrum and group delay function for speech recognition," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International conference on*, May 2004, vol. 1, pp. I–125–8 vol.1.

[43] B. Bozkurt, L. Couvreur, and T. Dutoit, "Chirp group delay analysis of speech signals," *Speech Communication*, vol. 49, no. 3, pp. 159 – 176, 2007.

[44] L. Rabiner, R. Schafer, and C. Rader, "The chirp z-transform algorithm," *IEEE Transactions on Audio and Electroacoustics*, vol. 17, no. 2, pp. 86–92, June 1969.

[45] B. Yegnanarayana, "Formant extraction from linear prediction phase spectra," *Journal of the Acoustical Society of America (JASA)*, vol. 63, no. 5, pp. 1638 – 1640, May 1978.

[46] H.A. Murthy and B. Yegnanarayana, "Formant extraction from group delay function," *Speech Communication*, vol. 10, no. 3, pp. 209 – 221, 1991.

[47] H.A. Murthy and B Yegnanarayana, "Speech processing using group delay functions," *Signal Processing*, vol. 22, no. 3, pp. 259 – 267, 1991.

[48] J. Lim, "Spectral root homomorphic deconvolution system," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 3, pp. 223–233, Jun 1979.

[49] B. Yegnanarayana, H.A. Murthy, and V.R. Ramachandran, "Processing of noisy speech using modified group delay functions," in *[Proceedings] ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing*, Apr 1991, pp. 945–948 vol.2.

[50] R. Rajan and H.A Murthy, "Group delay based melody monopitch extraction from music," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 186–190.

[51] R. Rajan, M. Misra, and H.A. Murthy, "Melody extraction from music using modified group delay functions," *I. J. Speech Technology*, vol. 20, no. 1, pp. 185–204, 2017.

[52] R. Rajan and H.A. Murthy, "Two-pitch tracking in co-channel speech using modified group delay functions," *Speech Communication*, vol. 89, pp. 37 – 46, 2017.

[53] T. Drugman, B. Bozkurt, and T. Dutoit, "Complex cepstrum-based decomposition of speech for glottal source estimation," in *Proc. of Interspeech 09*, 2009.

[54] R. Smits and B. Yegnanarayana, "Determination of instants of significant excitation in speech using group delay function," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 5, pp. 325–333, Sep 1995.

[55] K.S. Rao, S.R.M. Prasanna, and B. Yegnanarayana, "Determination of instants of significant excitation in speech using hilbert envelope and group delay function," *IEEE Signal Processing Letters*, vol. 14, no. 10, pp. 762–765, Oct 2007.

[56] J. L. Flanagan and R. M. Golden, "Phase vocoder," *Bell System Technical Journal*, vol. 45, no. 9, pp. 1493–1509, 1966.

[57] F.J. Charpentier, "Pitch detection using the short-term phase spectrum," in *Proceedings of ICASSP'86*, 1986, pp. 113–116.

[58] B. Boashash, "Estimating and interpreting the instantaneous frequency of a signal. i. fundamentals," *Proceedings of the IEEE*, vol. 80, no. 4, pp. 520–538, Apr 1992.

[59] B. Boashash, "Estimating and interpreting the instantaneous frequency of a signal. ii. algorithms and applications," *Proceedings of the IEEE*, vol. 80, no. 4, pp. 540–568, Apr 1992.

[60] D. Friedman, "Instantaneous-frequency distribution vs. time: An interpretation of the phase structure of speech," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '85.*, Apr 1985, vol. 10, pp. 1121–1124.

[61] A.P. Stark and Kuldip K.K. Paliwal, "Speech analysis using instantaneous frequency deviation," in *INTERSPEECH'08*, 2008, pp. 2602–2605.

[62] S. Kay, "A fast and accurate single frequency estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, 1989.

[63] A.P. Stark and K.K. Paliwal, "Group-delay-deviation based spectral analysis of speech," in *INTERSPEECH 2009, 10th Annual Conference of the International Speech Communication Association, Brighton, United Kingdom, September 6-10, 2009*, 2009, pp. 1083–1086.

[64] I. Saratxaga, I. Hernáez, D. Erro, E. Navas, and J. Sanchez, "Simple representation of signal phase for harmonic speech models," *Electronics Letters*, vol. 45, no. 7, pp. 381–383, March 2009.

[65] I. Saratxaga, I. Hernáez, I. Odriozola, E. Navas, I. Luengo, and D. Erro, "Use of harmonic phase information for polarity detection in speech signals," in *Interspeech*, Brighton, UK, 2009, ISCA, number September, p. 1075–1078, ISCA.

[66] I. Saratxaga, I. Hernáez, I. Odriozola, E. Navas, I. Luengo, and D. Erro, "Using harmonic phase information to improve asr rate," in *Interspeech*, Makuhari, Japan, 2010, ISCA, number September, p. 1185–1188, ISCA.

[67] I. Hernáez, I. Saratxaga, J. Sánchez, E. Navas, and I. Luengo, "Use of the harmonic phase in speaker recognition," in *Interspeech*. 2011, pp. 2757–2760, ISCA.

[68] I. McCowan, D. Dean, M. McLaren, R. Vogt, and S. Sridharan, "The delta-phase spectrum with application to voice activity detection and speaker recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2026–2038, Sept 2011.

[69] A.V. Oppenheim, J.S. Lim, G. Kopec, and S.C. Pohlig, "Phase in speech and pictures," in *Acoustics, Speech, and Signal Processing, IEEE International conference on ICASSP '79.*, Apr 1979, vol. 4, pp. 632–637.

[70] M.H. Hayes, J.S. Lim, and A.V. Oppenheim, "Signal reconstruction from phase or magnitude.," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-28, no. 6, pp. 672–680, 1980.

[71] E. Loweimi, S.M. Ahadi, and H. Sheikhzadeh, "Phase-only speech reconstruction using very short frames.," in *INTERSPEECH*. 2011, pp. 2501–2504, ISCA.

[72] P. Van Hove, M. Hayes, J. Lim, and A. Oppenheim, "Signal reconstruction from signed fourier transform magnitude," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 5, pp. 1286–1293, Oct 1983.

[73] B. Yegnanarayana, D. Saikia, and T. Krishnan, "Significance of group delay functions in signal reconstruction from spectral magnitude or phase," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 32, no. 3, pp. 610–623, Jun 1984.

[74] K. Paliwal and K. Wojcicki, "Effect of analysis window duration on speech intelligibility," *IEEE Signal Processing Letters*, vol. 15, pp. 785–788, 2008.

[75] H.J.M. Steeneken and T. Houtgast, "A physical method for measuring speechtransmission quality," *The Journal of the Acoustical Society of America*, vol. 67, no. 1, pp. 318–326, 1980.

[76] K.L. Payton and L.D. Braida, "A method to determine the speech transmission index from speech waveforms," *The Journal of the Acoustical Society of America*, vol. 106, no. 6, pp. 3637–3648, 1999.

[77] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *Proceedings of the Acoustics, Speech, and Signal Processing, 200. On IEEE International Conference - Volume 02*, Washington, DC, USA, 2001, ICASSP '01, pp. 749–752, IEEE Computer Society.

[78] Y. Hu and P.C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 16, no. 1, pp. 229–238, 2008.

[79] K.K. Paliwal and L.D. Alsteris, "Usefulness of phase spectrum in human speech perception," in *INTERSPEECH*. 2003, ISCA.

[80] K.K. Paliwal and L.D. Alsteris, "On the usefulness of stft phase spectrum in human listening tests," *Speech Communication*, vol. 45, no. 2, pp. 153–170, 2005.

[81] L.D. Alsteris and K.K. Paliwal, "Importance of window shape for phase-only reconstruction of speech," in *ICASSP, IEEE International conference on Acoustics, Speech and Signal Processing - Proceedings*, 2004, vol. 1, pp. I573–I576.

[82] D.A. Leigh and K.K. Paliwal, "Short-time phase spectrum in speech processing: A review and some experimental results," *Digital Signal Processing*, vol. 17, no. 3, pp. 578 – 616, 2007.

[83] L.D. Alsteris and K.K. Paliwal, "Further intelligibility results from human listening tests using the short-time phase spectrum," *Speech Communication*, vol. 48, no. 6, pp. 727–736, 2006.

[84] J.B. Allen and L.R. Rabiner, "A unified approach to short-time fourier analysis and synthesis," *Proceedings of the IEEE*, vol. 65, no. 11, pp. 1558–1564, Nov 1977.

[85] K. Wojcicki and K. Paliwal, "Importance of the dynamic range of an analysis windowfunction for phase-only and magnitude-only reconstruction of speech," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, April 2007, vol. 4, pp. IV–729–IV–732.

[86] F.J. Harris, "On the use of windows for harmonic analysis with the discrete fourier transform," *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–83, Jan. 1978.

[87] G. Shi, M.M. Shanechi, and P. Aarabi, "On the importance of phase in human speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 5, pp. 1867–1874, Sept 2006.

[88] D.W. Griffin and J.S. Lim, "Signal estimation from modified short-time fourier transform.," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-32, no. 2, pp. 236–243, 1984.

[89] M. Kazama, S. Gotoh, M. Tohyama, and T. Houtgast, "On the significance of phase in the short term fourier spectrum for speech intelligibility," *The Journal of the Acoustical Society of America*, vol. 127, no. 3, pp. 1432–1439, 2010.

[90] E. Loveimi and S.M. Ahadi, "Objective evaluation of magnitude and phase only spectrum-based reconstruction of the speech signal," in *Communications, Control and Signal Processing (ISCCSP), 2010 4th International Symposium on*, March 2010, pp. 1–4.

[91] D. Klatt, "Prediction of perceived phonetic distance from critical-band spectra: A first step," in *ICASSP '82. IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 1982, vol. 7, pp. 1278–1281.

[92] S. R. Quackenbush, T. P. Barnwell, and M. A. Clements, *Objective Measures of Speech Quality*, Prentice-Hall, Englewood Cliffs, NJ, 1988.

[93] E. Loveimi and S.M. Ahadi, "Objective evaluation of phase and magnitude only reconstructed speech: New considerations," in *Information Sciences Signal Processing and their Applications (ISSPA), 2010 10th International conference on*, May 2010, pp. 117–120.

[94] L. Rabiner, "James l. flanagan: A scholar and a true gentleman [reflections]," *IEEE Signal Processing Magazine*, vol. 33, no. 1, pp. 141–143, Jan 2016.

[95] H. Dudley, "Remaking speech," *The Journal of the Acoustical Society of America*, vol. 11, no. 2, pp. 169–177, 1939.

[96] G. Fant, *Acoustic Theory of Speech Production: With Calculations based on X-Ray Studies of Russian Articulations*, Description and Analysis of Contemporary Standard Russian. De Gruyter, 1971.

[97] M. Portnoff, "Implementation of the digital phase vocoder using the fast fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 3, pp. 243–248, Jun 1976.

[98] J. Tribolet and R. Crochiere, "Frequency domain coding of speech," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 5, pp. 512–530, October 1979.

[99] J.A. Moorer, "The use of the phase vocoder in computer music applications," *J. Audio Eng. Soc*, vol. 26, no. 1/2, pp. 42–45, 1978.

[100] J. Laroche and M. Dolson, "Improved phase vocoder time-scale modification of audio," May 1999, vol. 7, pp. 323–332.

[101] R.J. McAulay and T.F. Quatieri, "Sinusoidal coding.," Tech. Rep., MASSACHUSETTS INST OF TECH LEXINGTON LINCOLN LAB, 1995.

[102] M. Schroeder and B. Atal, "Code-excited linear prediction(celp): High-quality speech at very low bit rates," in *ICASSP '85. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Apr 1985, vol. 10, pp. 937–940.

[103] H. Pobloth and W.B. Kleijn, "On phase perception in speech," in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*, Mar 1999, vol. 1, pp. 29–32 vol.1.

[104] D.S. Kim, "Perceptual phase quantization of speech," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 4, pp. 355–364, July 2003.

[105] Wikipedia, "Just-noticeable difference — wikipedia, the free encyclopedia," 2017.

[106] Y. Agiomyrgiannakis and Y. Stylianou, "Wrapped gaussian mixture models for modeling and high-rate quantization of phase data of speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 775–786, May 2009.

[107] S.V. Andersen, W.B. Kleijn, R. Hagen, J. Linden, M.N. Murthi, and J. Skoglund, "ilbc - a linear predictive coder with robustness to packet losses," in *Speech Coding, 2002, IEEE Workshop Proceedings.*, Oct 2002, pp. 23–25.

[108] P.J. Moreno, "Sinusoidal coding of speech for voice over-ip," Tech. Rep., Ph. D. Dissertation, Dept. of Computer Sci., University of Crete, Crete, Greece, 2007.

[109] T. Dutoit, *An Introduction to Text-to-speech Synthesis*, Kluwer Academic Publishers, Norwell, MA, USA, 1997.

[110] H. Zen, K. Tokuda, and A.W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039 – 1064, 2009.

[111] Y. Stylianou, "Applying the harmonic plus noise model in concatenative speech synthesis," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 1, pp. 21–29, Jan 2001.

[112] R. Maia, M. Akamine, and M.J.F. Gales, "Complex cepstrum as phase information in statistical parametric speech synthesis," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 4581–4584.

[113] A. Bayya and B. Yegnanarayana, "Noise-invariant representation for speech signals.," in *EUROSPEECH*. 1999, ISCA.

[114] R.M. Hegde, H.A. Murthy, and V.R.R. Gadde, "Significance of joint features derived from the modified group delay function in speech processing," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2007, no. 1, pp. 079032, Dec 2006.

[115] E. Loweimi and S.M. Ahadi, "A new group delay-based feature for robust speech recognition," in *Multimedia and Expo (ICME), 2011 IEEE International conference on*, July 2011, pp. 1–5.

[116] M.H. Hayes, *Statistical Digital Signal Processing and Modeling*, John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1996.

[117] E. Loweimi, S.M. Ahadi, and T. Drugman, "A new phase-based feature representation for robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International conference on*, May 2013, pp. 7155–7159.

[118] E. Loweimi, J. Barker, and T. Hain, "Compression of model-based group delay function for robust speech recognition," *The University of Sheffield Engineering Symposium Conference Proceedings Vol. 1*, vol. 1, 2014.

[119] R. Schluter and H. Ney, "Using phase spectrum information for improved speech recognition performance," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, 2001, vol. 1, pp. 133–136 vol.1.

[120] Y. Wang, J. Hansen, G.K. Allu, and R. Kumaresan, "Average instantaneous frequency (AIF) and average log-envelopes (ALE) for ASR with the aurora 2 database," in *8th European Conference on Speech Communication and Technology, EUROSPEECH 2003 - INTERSPEECH 2003, Geneva, Switzerland, September 1-4, 2003*. 2003, ISCA.

[121] A. Potamianos and P. Maragos, "Time-frequency distributions for automatic speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 3, pp. 196–200, Mar 2001.

[122] K.K. Paliwal and B.S. Atal, "Frequency-related representation of speech," in *8th European Conference on Speech Communication and Technology, EUROSPEECH 2003 - INTERSPEECH 2003, Geneva, Switzerland, September 1-4, 2003*, 2003.

[123] R.M. Hegde, H.A. Murthy, and G.V.R. Rao, "Application of the modified group delay function to speaker identification and discrimination," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International conference on*, May 2004, vol. 1, pp. I–517–20 vol.1.

[124] R. Padmanabhan, S.H.K. Parthasarathi, and H.A. Murthy, "Robustness of phase based features for speaker recognition," in *INTERSPEECH*. 2009, pp. 2355–2358, ISCA.

[125] S.R. Madikeri, A. Talambedu, and H.A. Murthy, "Modified group delay feature based total variability space modelling for speaker recognition," *I. J. Speech Technology*, vol. 18, no. 1, pp. 17–23, 2015.

[126] N. Dehak, P.J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[127] K. Vijayan and K.S.R. Murty, "Analysis of phase spectrum of speech signals using allpass modeling," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 12, pp. 2371–2383, Dec 2015.

[128] C.-Y. Chi and J.-Y. Kung, "A new identification algorithm for allpass systems by higher-order statistics," *Signal Processing*, vol. 41, no. 2, pp. 239 – 256, 1995.

[129] K. Vijayan, R.R. Pappagari, and K. Sri Rama, "Significance of analytic phase of speech signals in speaker verification," *Speech Communication*, vol. 81, no. C, pp. 54–71, July 2016.

[130] M. Athineos and D.P.W. Ellis, "Frequency-domain linear prediction for temporal features," in *2003 IEEE Workshop on Automatic Speech Recognition and Understanding (IEEE Cat. No.03EX721)*, Nov 2003, pp. 261–266.

[131] L. Wang, S. Ohtsuka, and S. Nakagawa, "High improvement of speaker identification and verification by combining mfcc and phase information," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, April 2009, pp. 4529–4532.

[132] V. Sethu, E. Ambikairajah, and J. Epps, "Group delay features for emotion detection.," in *INTERSPEECH*. 2007, pp. 2273–2276, ISCA.

[133] J. Deng, X. Xu, Z. Zhang, S. Frühholz, and B. Schuller, "Exploitation of phase-based features for whispered speech emotion recognition," *IEEE Access*, vol. 4, pp. 4299–4309, 2016.

[134] Z. Wu, X. Xiao, E.S. Chng, and H. Li, "Synthetic speech detection using temporal modulation feature," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 7234–7238.

[135] P.L. De Leon, M. Pucher, J. Yamagishi, I. Hernaez, and I. Saratxaga, "Evaluation of speaker verification security and detection of hmm-based synthetic speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 8, pp. 2280–2290, Oct 2012.

[136] J. Sanchez, I. Saratxaga, I. Hernáez, E. Navas, D. Erro, and T. Raitio, "Toward a universal synthetic speech spoofing detection using phase information," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 810–820, April 2015.

[137] P. Vary, "Noise suppression by spectral magnitude estimation—mechanism and theoretical limits—," *Signal Processing*, vol. 8, no. 4, pp. 387–400, 1985.

[138] D. Wang and J.S. Lim, "The unimportance of phase in speech enhancement," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 30, no. 4, pp. 679–681, Aug 1982.

[139] B. Shannon and K.K. Paliwal, "Role of phase estimation in speech enhancement," in *INTERSPEECH 2006 - ICSLP, Ninth International Conference on Spoken Language Processing, Pittsburgh, PA, USA, September 17-21, 2006*. 2006, ISCA.

[140] W. Yang, *Enhanced Modified Bark Spectral Distortion (Embsd): An Objective Speech Quality Measure Based on Audible Distortion and Cognition Model*, Ph.D. thesis, Philadelphia, PA, USA, 1999, AAI9938714.

[141] K. Wojcicki, M. Milacic, A. Stark, J. Lyons, and K. Paliwal, "Exploiting conjugate symmetry of the short-time fourier spectrum for speech enhancement," *Signal Processing Letters, IEEE*, vol. 15, pp. 461–464, 2008.

[142] A.P. Stark, K.K. Wójcicki, J.G. Lyons, and K.K. Paliwal, "Noise driven short time phase spectrum compensation procedure for speech enhancement," Brisbane, QLD, Australia, Sept. 2008, pp. 549–552.

[143] E. Loweimi, S.M. Ahadi, and S. Loveymi, "On the importance of phase and magnitude spectra in speech enhancement," in *Electrical Engineering (ICEE), 2011 19th Iranian conference on*, May 2011, pp. 1–1.

[144] T. Gerkmann and M. Krawczyk, "Mmse-optimal spectral amplitude estimation given the stft-phase," *IEEE Signal Processing Letters*, vol. 20, no. 2, pp. 129–132, Feb 2013.

[145] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase processing for single-channel speech enhancement: History and recent advances," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 55–66, March 2015.

[146] T. Gerkmann, M. Krawczyk, and R. Rehr, "Phase estimation in speech enhancement – unimportant, important, or impossible?," in *Electrical Electronics Engineers in Israel (IEEEI), 2012 IEEE 27th Convention of*, Nov 2012, pp. 1–5.

[147] P. Mowlaee, J. Kulmer, J. Stahl, and F. Mayer, *Single Channel Phase-Aware Signal Processing in Speech Communication: Theory and Practice*, Wiley, 2016.

[148] J. Moura, "What is signal processing? [president's message]," *IEEE Signal Processing Magazine*, vol. 26, no. 6, pp. 6–6, Nov 2009.

[149] I. James, "Claude elwood shannon. 30 april 1916 — 24 february 2001," *Biographical Memoirs of Fellows of the Royal Society*, 2009.

[150] C.E. Shannon, "A mathematical theory of communication," *Bell system technical journal*, vol. 27, 1948.

[151] T. Dontchev, A.and Zolezzi, *Hadamard and tykhonov well-posedness*, pp. 38–80, Springer Berlin Heidelberg, Berlin, Heidelberg, 1993.

[152] R. Martin and R. V. Cox, "New speech enhancement techniques for low bit rate speech coding," in *1999 IEEE Workshop on Speech Coding Proceedings. Model, Coders, and Error Criteria (Cat. No.99EX351)*, 1999, pp. 165–167.

[153] M. Krawczyk and T. Gerkmann, "Stft phase reconstruction in voiced speech for an improved single-channel speech enhancement," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 22, no. 12, pp. 1931–1940, Dec. 2014.

[154] P. Lieberman and S.E. Blumstein, *Speech Physiology, Speech Perception, and Acoustic Phonetics*, Cambridge Studies in Speech Sc. Cambridge University Press, 1988.

[155] T. Chiba and M. Kajiyama, *The vowel, its nature and structure*, Phonetic Society of Japan, 1958.

[156] K.N. Stevens, *Acoustic Phonetics*, Current Studies in Linguistics Series. CogNet, 2000.

[157] A. Oppenheim and R. Schafer, "Homomorphic analysis of speech," *IEEE Transactions on Audio and Electroacoustics*, vol. 16, no. 2, pp. 221–226, Jun 1968.

[158] R. Hodrick and E. Prescott, "Postwar u.s. business cycles: An empirical investigation," *Journal of Money, Credit and Banking*, vol. 29, no. 1, pp. 1–16, 1997.

[159] A. Savitzky and M.J.E. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Analytical Chemistry*, vol. 36, pp. 1627–1639, 1964.

[160] M. Ross, H. Shaffer, A. Cohen, R. Freudberg, and H. Manley, "Average magnitude difference function pitch extractor," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 22, no. 5, pp. 353–362, Oct 1974.

[161] P.J. Bickel and K.A. Doksum, "An analysis of transformations revisited," *Journal of the American Statistical Association*, vol. 76, no. 374, pp. 296–311, 1981.

[162] S.J. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *The HTK Book Version 3.4*, Cambridge University Press, 2006.

[163] D. Freedman and P. Diaconis, "On the histogram as a density estimator:L2 theory," *Probability Theory and Related Fields*, vol. 57, no. 4, pp. 453–476, Dec. 1981.

[164] D.J.C. MacKay, *Information Theory, Inference & Learning Algorithms*, Cambridge University Press, New York, NY, USA, 2002.

[165] Lawrence T. Decarlo, "On the meaning and use of kurtosis," *Psychological Methods*, pp. 292–307, 1997.

[166] P.G. Hoel, S.C. Port, and C.J. Stone, *Introduction to Probability Theory*, Houghton Mifflin series in statistics. Houghton Mifflin, 1971.

[167] S. Dharanipragada and M. Padmanabhan, "A nonlinear unsupervised adaptation technique for speech recognition," in *Proc. ICSLP'00*, 2000, pp. 556–559.

[168] T. Kobayashi and S. Imai, "Spectral analysis using generalized cepstrum," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 5, pp. 1087–1089, Oct 1984.

[169] S.S. Haykin and M. Moher, *Communication Systems*, Wiley, 2010.

[170] S. Hawking and L. Mlodinow, *The Grand Design*, Bantam Books. Bantam Books, 2010.

[171] K. Tokuda, T. Kobayashi, T. Masuko, and S. Imai, "Unified approach to mel-generalized cepstral analysis," in *Proc. ICSLP-94*, 1994, pp. 1043–1046.

[172] F. Grezl, M. Karafiat, S. Kontar, and J. Cernocky, "Probabilistic and bottle-neck features for lvcsr of meetings," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, April 2007, vol. 4, pp. IV–757–IV–760.

[173] A.M. Noll, *Pitch determination of human speech by the harmonic product spectrum, the harmonic sum spectrum, and maximum likelihood estimate*, vol. XIX of *Proceedings of the Symposium on Computer Processing in Communications*, Polytechnic Press: Brooklyn, New York, April 1969.

[174] A.M. Noll, *Short-time Spectrum and "cepstrum" Techniques for Vocal-pitch Detection*, Bell telephone system technical publications. Bell Telephone Laboratories, 1964.

[175] T. Drugman and A. Alwan, "Joint robust voicing detection and pitch estimation based on residual harmonics.," in *INTERSPEECH*. 2011, pp. 1973–1976, ISCA.

[176] P.C. Bagshaw, S.M. Hiller, and M.A. Jack, "Enhanced pitch tracking and the processing of f0 contours for computer aided intonation teaching," in *EUROSPEECH*, 1993.

[177] J. Droppo, A. Acero, and L. Deng, "Evaluation of the splice algorithm on the aurora 2 database," in *Proc. Eurospeech Conference*. September 2001, International Speech Communication Association.

[178] P.J. Moreno, "Speech recognition in noisy environments," Tech. Rep., Ph. D. Dissertation, ECE Department, CMU, 1996.

[179] K. Vesely, L. Burget, and F. Grezl, "Parallel training of neural networks for speech recognition," in *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, 2010, pp. 2934–2937.

[180] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, Mar 2004.

[181] A.B. Carlson, P. Crilly, and P.B. Crilly, *Communication Systems*, McGraw-Hill Education, 2009.

[182] K. Ogata, *Modern Control Engineering*, Instrumentation and controls series. Prentice Hall, 2010.

[183] J.W. Tukey, *Exploratory Data Analysis*, Addison-Wesley, 1977.

[184] L. Deng, J. Droppo, and A. Acero, "Enhancement of log mel power spectra of speech using a phase-sensitive model of the acoustic environment and sequential estimation of the corrupting noise," March 2004, vol. 12, p. 133–143, Institute of Electrical and Electronics Engineers, Inc.

[185] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G.J. McLachlan, A. Ng, B. Liu, P.S. Yu, Z-H. Zhou, M. Steinbach, D.J. Hand, and D. Steinberg, "Top 10 algorithms in data mining," *Knowl. Inf. Syst.*, vol. 14, no. 1, pp. 1–37, Dec. 2007.

[186] G.E. Hinton, S. Osindero, and Y-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, July 2006.

[187] Y. Bengio, "Learning deep architectures for ai," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[188] V. Nair and G.E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, Johannes Fürnkranz and Thorsten Joachims, Eds. 2010, pp. 807–814, Omnipress.

[189] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

[190] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," *arXiv preprint arXiv:1302.4389*, 2013.

[191] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[192] Y. LeCun and Y. Bengio, "The handbook of brain theory and neural networks," chapter Convolutional Networks for Images, Speech, and Time Series, pp. 255–258. MIT Press, Cambridge, MA, USA, 1998.

[193] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 2672–2680. Curran Associates, Inc., 2014.

[194] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd International Conference on Machine Learning*, New York, NY, USA, 2006, ICML '06, pp. 369–376, ACM.

[195] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 328–339, Mar 1989.

[196] K.J. Lang, A.H. Waibel, and G.E. Hinton, "A time-delay neural network architecture for isolated word recognition," *Neural Networks*, vol. 3, no. 1, pp. 23 – 43, 1990.

[197] H. Bourlard and N. Morgan, "Hybrid hmm/ann systems for speech recognition: Overview and new research directions," in *Adaptive Processing of Sequences and Data Structures, International Summer School on Neural Networks, "E.R. Caianiello"-Tutorial Lectures*, London, UK, UK, 1998, pp. 389–417, Springer-Verlag.

[198] A. Mohamed, G. Dahl, and G. Hinton, "Deep belief networks for phone recognition," *NIPS 22 workshop on deep learning for speech recognition*, 2009.

[199] A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 6645–6649.

[200] G. Hinton, L. Deng, D. Yu, G.E. Dahl, A.r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T.N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, Nov 2012.

[201] O. Abdel-Hamid, A. r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 10, pp. 1533–1545, Oct 2014.

[202] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling.," in *INTERSPEECH*, 2014, pp. 338–342.

[203] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, "Recurrent neural network based language model.," in *Interspeech*, 2010, vol. 2, p. 3.

[204] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, pp. 1137–1155, Mar. 2003.

[205] H. Hermansky, D.P.W. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional hmm systems," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, 2000, vol. 3, pp. 1635–1638 vol.3.

[206] F. Grezl, M. Karafiat, S. Kontar, and J. Cernocky, "Probabilistic and bottle-neck features for lvcsr of meetings," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, April 2007, vol. 4, pp. IV–757–IV–760.

[207] F. Grezl and P. Fousek, "Optimizing bottle-neck features for lvcsr," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2008, pp. 4729–4732.

[208] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Ng, "Deep speech: Scaling up end-to-end speech recognition," *CoRR*, vol. abs/1412.5567, 2014.

[209] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning*, Eric P. Xing and Tony Jebara, Eds., Bejing, China, 22–24 Jun 2014, vol. 32 of *Proceedings of Machine Learning Research*, pp. 1764–1772, PMLR.

[210] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving Human Parity in Conversational Speech Recognition," *ArXiv e-prints*, Oct. 2016.

[211] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L.-L. Lim, B. Roomi, and P. Hall, "English Conversational Telephone Speech Recognition by Humans and Machines," *ArXiv e-prints*, Mar. 2017.

[212] Y. Liu, P. Zhang, and T. Hain, "Using neural network front-ends on far field multiple microphones based speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International conference on*, Florence, Italy, May 2014.

[213] R. Leonard, "A database for speaker-independent digit recognition," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.*, Mar 1984, vol. 9, pp. 328–331.

[214] G. Hirsch, "Fant - filtering and noise adding tool," http://dnt.kr.hsnr.de/download.html, 2005.

[215] "Ieee recommended practice for speech quality measurements," *IEEE No 297-1969*, pp. 1–24, June 1969.

[216] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *The Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, Apr. 1990.

Appendix A

Hilbert Transform

A.1 Introduction

Hilbert transform, $\mathscr{H}$, is a linear transformation which contrary to the Fourier transform, $\mathscr{F}$, does not change the domain. One of its important applications is in computing the analytic signal, x_a, which is a complex-valued sequence that its real and imaginary parts forms a Hilbert transform pair. That is, the real part equals the original signal and the imaginary part equals the corresponding Hilbert transform

$$x_a[n] = x[n] + j\,\mathscr{H}\{x[n]\} \Rightarrow X_a[k] = \begin{cases} X[k] & k = 0 \\ 2X[k] & 0 \leq k \leq N_{FFT}/2 - 1 \\ X[k] & k = \frac{N_{FFT}}{2} \\ 0 & k > \frac{N_{FFT}}{2} \end{cases} \tag{A.1}$$

where k and N_{FFT} denote the discrete frequency symbol and the FFT size, respectively. The special advantage of the analytic signal is that its Fourier transform is causal[1], i.e. it is zero at the negative frequencies and in the positive frequencies equals (twice) the spectral content of the original signal. This practically could be desirable in applications like single-sideband (SSB) AM modulation/demodulation [181].

So, the causality of the Fourier transform of a signal means that its real and imaginary part form a Hilbert transform pair. This puts a clear constraint on the signals which their Fourier

[1] Causality means the impulse response equals zero at negative time samples. Here the concept is extended to the case which a function is zero at negative values of its independent variable.

transform is causal. Due to the time/frequency duality of the Fourier transform[2] causality in the time domain, would put a clear and similar constraint on the real and imaginary parts of its Fourier transform, $X_{Re}(\omega)$ and $X_{Im}(\omega)$, respectively. By the same token, causality in the complex cepstrum domain implies the same constrain on the real and imaginary parts of its Fourier transform, namely $log|X(\omega)|$ and $arg\{X(\omega)\}$, respectively.

In all of the aforementioned causal sequences, the imaginary part equals the Hilbert transform of the real part. So, for computing the imaginary part, only partial information of the Fourier transform, namely the real part suffices. Having computed the imaginary part from the real part, the whole sequence can be recovered. How about computing the real part from the imaginary part? One can compute the inverse Hilbert transform of the imaginary part but there is an issue with the Hilbert transform which does not allow to recover the real part perfectly. That is, the Hilbert transform of a constant is zero

$$x[n]+c \xrightarrow{\mathscr{H}} \underbrace{\mathscr{H}\{x[n]\}}_{H_x[n]} + \underbrace{\mathscr{H}\{c\}}_{0} \xrightarrow{\mathscr{H}} -x[n] \tag{A.2}$$

where c denotes a constant. This is because the kernel of the Hilbert transform is an odd function. As such after performing this transform, the constant term is lost and cannot be recovered in the reverse process any more. Therefore, by taking the inverse Hilbert transform the signal can be recovered within an additive constant. In addition, the Hilbert transform of a Hilbert transform of a sequence equals the minus of the original one (constant excluded). As a result, the inverse of the Hilbert transform is nothing other than the negative of the Hilbert transform itself.

Backing to the main track, for recovering the real part from the imaginary part, based on the aforementioned argument, the real part and consequently the signal is recoverable upto an additive error. Likewise, for a sequence with causal complex cepstrum, the $log|X(\omega)|$ can be calculated within an additive error from the $arg\{X(\omega)\}$ which means that the $|X(\omega)|$ and consequently the signal (in the time domain) can be calculated upto a scale error.

In this Appendix we derive the Hilbert transform relations between the real and imaginary parts of the Fourier transform of the causal signals and use that to calculate the relation between the magnitude and the unwrapped phase spectra for the minimum-phase[3] sequences.

[2]Duality of the DFT means

$$\begin{aligned} &x[n] \Leftrightarrow X[k] \\ &X[n] \Leftrightarrow Nx[-k \bmod N], \quad 0 \le k \le N-1 \end{aligned}$$

[3]Minimum-phase system is a system which all of its poles and zeros are located inside the unit circle. It can be also defined as a stable and causal system with a stable and causal inverse. The term minimum-phase in the

The later is used in Sections 3.2 and 4.3, in defining signal's information regions and the proposed source-filter separation. In addition, the Hilbert transform relations for the maximum-phase signals is derived. Finally, a brief comparison between the Hilbert transform kernel in the continuous case and discrete case is carried out. The material are mainly taken from [18] and [26].

A.2 Real and Imaginary Parts Relationship for Causal Signals

A.2.1 Preliminaries

For a discrete-time sequence $x[n]$ one can write

$$x[n] = x_{even}[n] + x_{odd}[n] \tag{A.3}$$

$$x_{even}[n] = \frac{x[n] + x[-n]}{2} \tag{A.4}$$

$$x_{odd}[n] = \frac{x[n] - x[-n]}{2} \tag{A.5}$$

$$x[n] = 2x_{even}[n]\,u[n] - x[0]\delta[n] \tag{A.6}$$

$$x[n] = 2x_{odd}[n]\,u[n] + x[0]\delta[n] \tag{A.7}$$

$$z[n] = x[n]\,y[n] \quad \underset{\mathscr{F}^{-1}}{\overset{\mathscr{F}}{\rightleftharpoons}} \quad Z(\omega) = \frac{1}{2\pi} X(\omega) * Y(\omega) \tag{A.8}$$

where $\mathscr{F}$ and $\mathscr{F}^{-1}$ indicate the Fourier transform and inverse Fourier transform, respectively.

control theory means a stable system which has the minimum phase lag than any stable system with the same magnitude spectrum [182]. In [18] the term minimum-phase is described as historical and the minimum-group delay is suggested as a more accurate term because it is more meaningful since such systems have minimum group delay among all the possible systems with the same magnitude spectrum. Also note that although the above definition is for the systems, without loss of generality, it can be used for signals. In this case, one may consider the signal as the impulse response of the system.

$$\mathscr{F}\{x[n]\} = \mathscr{F}\{x_{even}[n]\} + \mathscr{F}\{x_{odd}[n]\} = X_{Re}(\omega) + jX_{Im}(\omega) \tag{A.9}$$
$$\Rightarrow \begin{cases} \mathscr{F}\{x_{even}[n]\} & = X_{Re}(\omega) \\ \mathscr{F}\{x_{odd}[n]\} & = jX_{Im}(\omega) \end{cases}$$

$$\mathscr{F}\{u[n]\} = U(\omega) = \frac{1}{1-e^{-j\omega}} + \pi \sum_{k=-\infty}^{\infty} \delta(\omega - 2\pi k) \tag{A.10}$$

$$\mathscr{F}\{\delta[n]\} = 1 \tag{A.11}$$

$$\begin{aligned} \frac{1}{1-e^{-jw}} &= \frac{1}{1-cos(\omega)+j\,sin(\omega)} = \frac{1}{2sin^2(\frac{\omega}{2}) + j\,2sin(\frac{\omega}{2})cos(\frac{\omega}{2})} \\ &= \frac{1}{2sin(\frac{\omega}{2})}\,\frac{1}{sin(\frac{\omega}{2})+j\,cos(\frac{\omega}{2})} \times \frac{sin(\frac{\omega}{2})-j\,cos(\frac{\omega}{2})}{sin(\frac{\omega}{2})-j\,cos(\frac{\omega}{2})} \\ &= \frac{sin(\frac{\omega}{2})-j\,cos(\frac{\omega}{2})}{2sin(\frac{\omega}{2})} = \frac{1}{2}(1-j\,cot(\frac{\omega}{2})) \end{aligned} \tag{A.12}$$

A.2.2 Imaginary Part as a Function of Real Part

$$\begin{aligned} \mathscr{F}\{x[n]\} &= \mathscr{F}\{2x_{even}[n]\,u[n] - x[0]\delta[n]\} \\ &= \frac{1}{2\pi}\mathscr{F}\{2x_{even}[n]\} * \mathscr{F}\{u[n]\} - x[0]\mathscr{F}\{\delta[n]\} = \frac{1}{\pi}X_{Re}(\omega) * U(\omega) - x[0] \\ &= \frac{1}{\pi}\int_{-\pi}^{\pi} X_{Re}(\theta)\,U(\omega-\theta)d\theta - x[0] \\ &= \frac{1}{\pi}\int_{-\pi}^{\pi} X_{Re}(\theta)\,\Big[\frac{1}{2} - \frac{j}{2}cot(\frac{\omega-\theta}{2}) + \pi\sum_{k=-\infty}^{\infty}\delta(\omega-\theta+2\pi k)\Big]d\theta - x[0] \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} X_{Re}(\theta)d\theta - \frac{1}{\pi}\int_{-\pi}^{\pi} X_{Re}(\theta)\,\frac{j}{2}cot(\frac{\omega-\theta}{2})d\theta \\ &+ \frac{\pi}{\pi}\int_{-\pi}^{\pi} X_{Re}(\theta)\,\delta(\omega-\theta)d\theta - x[0] \end{aligned} \tag{A.13}$$

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} X_{Re}(\theta)d\theta = x_{even}[0] = x[0] \quad \& \quad \int_{-\pi}^{\pi} X_{Re}(\theta)\,\delta(\omega-\theta)d\theta = X_{Re}(\omega) \tag{A.14}$$

$$\begin{aligned}\mathscr{F}\{x[n]\} &= X_{Re}(\omega)+jX_{Im}(\omega)\\ &= x[0]-\frac{j}{2\pi}\int_{-\pi}^{\pi}X_{Re}(\theta)\,cot(\frac{\omega-\theta}{2})d\theta+X_{Re}(\omega)-x[0]\\ &= X_{Re}(\omega)-\frac{j}{2\pi}\int_{-\pi}^{\pi}X_{Re}(\theta)\,cot(\frac{\omega-\theta}{2})d\theta \end{aligned} \tag{A.15}$$

$$X_{Im}(\omega) = -\frac{1}{2\pi}\int_{-\pi}^{\pi}X_{Re}(\theta)\,cot(\frac{\omega-\theta}{2})\,d\theta = -\frac{1}{2\pi}X_{Re}(\omega)*cot(\frac{\omega}{2}) \tag{A.16}$$

To be more precise,

$$\begin{aligned}X_{Im}(\omega) &= -\frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi}X_{Re}(\theta)\,cot(\frac{\omega-\theta}{2})\,d\theta\\ &= -\frac{1}{2\pi}\int_{-\pi}^{\omega-\varepsilon}X_{Re}(\theta)\,cot(\frac{\omega-\theta}{2})\,d\theta-\frac{1}{2\pi}\int_{\omega+\varepsilon}^{\pi}X_{Re}(\theta)\,cot(\frac{\omega-\theta}{2})\,d\theta\end{aligned} \tag{A.17}$$

where $\mathscr{P}$ denotes Cauchy principle value of the integral.

A.2.3 Real Part as a Function of Imaginary Part

$$\begin{aligned}\mathscr{F}\{x[n]\} &= \mathscr{F}\{2x_{odd}[n]\,u[n]+x[0]\delta[n]\}\\ &= \frac{1}{2\pi}\mathscr{F}\{2x_{odd}[n]\}*\mathscr{F}\{u[n]\}+x[0]\mathscr{F}\{\delta[n]\}\\ &= \frac{1}{\pi}j\,X_{Im}(\omega)*U(\omega)+x[0]=\frac{j}{\pi}\int_{-\pi}^{\pi}X_{Im}(\theta)\,U(\omega-\theta)d\theta+x[0]\\ &= \frac{j}{\pi}\int_{-\pi}^{\pi}X_{Im}(\theta)\,\Big[\frac{1}{2}-\frac{j}{2}cot(\frac{\omega-\theta}{2})+\pi\sum_{k=-\infty}^{\infty}\delta(\omega-\theta+2\pi k)\Big]d\theta+x[0]\\ &= \frac{j}{2\pi}\int_{-\pi}^{\pi}X_{Im}(\theta)d\theta-\frac{j}{\pi}\int_{-\pi}^{\pi}X_{Im}(\theta)\,\frac{j}{2}cot(\frac{\omega-\theta}{2})\,d\theta\\ &+ \frac{j\pi}{\pi}\int_{-\pi}^{\pi}X_{Im}(\theta)\,\delta(\omega-\theta)d\theta+x[0]\\ &= \frac{j}{2\pi}\int_{-\pi}^{\pi}X_{Im}(\theta)d\theta+\frac{1}{\pi}\int_{-\pi}^{\pi}X_{Im}(\theta)\,\frac{1}{2}cot(\frac{\omega-\theta}{2})d\theta\\ &+ j\int_{-\pi}^{\pi}X_{Im}(\theta)\,\delta(\omega-\theta)d\theta+x[0]\end{aligned} \tag{A.18}$$

$$\int_{-\pi}^{\pi} X_{Im}(\theta)d\theta = 0 \quad \& \quad \int_{-\pi}^{\pi} X_{Im}(\theta)\ \delta(\omega-\theta)d\theta = X_{Im}(\omega) \tag{A.19}$$

$$\begin{aligned}\mathscr{F}\{x[n]\} &= X_{Re}(\omega) + j\,X_{Im}(\omega) \\ &= 0 + \frac{1}{2\pi}\int_{-\pi}^{\pi} X_{Im}(\theta)\ cot(\frac{\omega-\theta}{2})d\theta + jX_{Im}(\omega) + x[0] \\ &= x[0] + \frac{1}{2\pi}\int_{-\pi}^{\pi} X_{Im}(\theta)\ cot(\frac{\omega-\theta}{2})d\theta + jX_{Im}(\omega)\end{aligned} \tag{A.20}$$

$$\begin{aligned}X_{Re}(\omega) &= x[0] + \frac{1}{2\pi}\int_{-\pi}^{\pi} X_{Im}(\theta)\ cot(\frac{\omega-\theta}{2})d\theta \\ &= x[0] + \frac{1}{2\pi}X_{Im}(\omega) * cot(\frac{\omega}{2})\end{aligned} \tag{A.21}$$

$$\begin{aligned}X_{Re}(\omega) &= x[0] + \frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} X_{Im}(\theta)\ cot(\frac{\omega-\theta}{2})d\theta \\ &= x[0] + \frac{1}{2\pi}\int_{-\pi}^{\omega-\varepsilon} X_{Im}(\theta)\ cot(\frac{\omega-\theta}{2})d\theta + \frac{1}{2\pi}\int_{\omega+\varepsilon}^{\pi} X_{Im}(\theta)\ cot(\frac{\omega-\theta}{2})d\theta\end{aligned} \tag{A.22}$$

A.3 Magnitude and Phase Relationship for the Minimum-Phase Signals

$$\begin{aligned}\tilde{x}[q] &\triangleq \mathscr{F}^{-1}\{log(\mathscr{F}\{x[n]\}) = \mathscr{F}^{-1}\{log(X(\omega))\} \\ &= \mathscr{F}^{-1}\{log(|X(\omega)|\ e^{jarg[X(\omega)]})\} = \mathscr{F}^{-1}\{log|X(\omega)| + j\ arg[X(\omega)]\}\end{aligned} \tag{A.23}$$

$$\mathscr{F}\{\tilde{x}[q]\} = \tilde{X}(\omega) = \tilde{X}_{Re}(\omega) + j\tilde{X}_{Im}(\omega) = log|X(\omega)| + jarg[X(\omega)] \tag{A.24}$$

$$\Rightarrow \begin{cases}\tilde{X}_{Re}(\omega) = log|X(\omega)| \\ \tilde{X}_{Im}(\omega) = arg[X(\omega)]\end{cases}$$

$$
\begin{aligned}
arg[X(\omega)] &= -\frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} log|X(\theta)|cot(\frac{\omega-\theta}{2})\,d\theta \\
log|X(\omega)| &= \tilde{x}[0]+\frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} arg[X(\theta)]cot(\frac{\omega-\theta}{2})\,d\theta
\end{aligned}
\tag{A.25}
$$

A.3.1 Anti-causal and Maximum-Phase Signals

In case of maximum-phase signals the complex cepstrum becomes anti-causal i.e equals to zero for positive qufrencies. So, first we develop the formula for any anti-casusal sequence and then look at a particular case where the sequence is the complex cepstrum. For an anti-causal sequence

$$x[n] = 2x_{even}[n]\,u[-n] - x[0]\,\delta[n] \tag{A.26}$$

$$x[n] = 2x_{odd}[n]\,u[-n] + x[0]\,\delta[n] \tag{A.27}$$

Let's define the $y[n] = x[-n]$. The real and imaginary parts of $x[n]$ and $y[n]$ relate to each other in the following way

$$y[n] = x[-n] \Rightarrow \begin{cases} X_{Re}(\omega) = Y_{Re}(\omega) \\ X_{Im}(\omega) = -Y_{Im}(\omega) \end{cases} \tag{A.28}$$

Given that $x[n]$ is assumed to be anti-causal, it is straightforward to see that $y[n]$ is a causal sequence. As a result, its real and imaginary parts satisfy (A.17) and (23). Using these formula along with relations between the real and imaginary parts of $x[n]$ and $y[n]$, namely (A.28), the equivalent relationships can be derived between the real and imaginary parts of an anti-causal sequence and also between the magnitude and phase spectra of a maximum-phase sequences with anti-causal complex cepstrum.

A.3.2 Real and Imaginary Parts Relationship

$$
\begin{aligned}
X_{Re}(\omega) &= x[0] - \frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} X_{Im}(\theta)\,cot(\frac{\omega-\theta}{2})d\theta \\
&= x[0] - \frac{1}{2\pi}X_{Im}(\omega) * cot(\frac{\omega}{2})
\end{aligned}
\tag{A.29}
$$

$$X_{Im}(\omega) = \frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} X_{Re}(\theta)\, cot(\frac{\omega-\theta}{2})d\theta = \frac{1}{2\pi}X_{Re}(\omega) * cot(\frac{\omega}{2}) \tag{A.30}$$

A.3.3 Magnitude and Phase Relationship

$$\begin{aligned} log|X(\omega)| &= \tilde{x}[0] - \frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} arg[X(\theta)]\, cot(\frac{\omega-\theta}{2})d\theta \\ &= \tilde{x}[0] - \frac{1}{2\pi}arg[X(\omega)] * cot(\frac{\omega}{2}) \end{aligned} \tag{A.31}$$

$$\begin{aligned} arg[X(\omega)] &= \frac{1}{2\pi}\mathscr{P}\int_{-\pi}^{\pi} log|X(\theta)|\, cot(\frac{\omega-\theta}{2})d\theta \\ &= \frac{1}{2\pi}log|X(\omega)| * cot(\frac{\omega}{2}) \end{aligned} \tag{A.32}$$

A.4 Hilbert Transform in Continuous and Discrete Domains

This is a bit of a personal story. My first contact with the Hilbert transform was in the Communication Theory course during my undergraduate. On that course, the Hilbert transform was defined as the convolution with $\frac{1}{\pi t}$, where t denotes the time and the corresponding Fourier transform was $-j\,sign(\omega)$ [26] where *sign* indicates the sign function. However, in the DSP course, the Hilbert transform was defined as the convolution with $\frac{1}{2\pi}cot(\frac{\omega}{2})$ [18]. Since the Hilbert transform does not change the domain, the t or ω are not important. So, the Hilbert transform kernel was $\frac{1}{\pi z}$ in one case and $\frac{1}{2\pi}\,cot(\frac{z}{2})$ in another case. I was confused for some years and could not build a relationship between them. In fact, the $\frac{1}{t}$ Homographic function apparently had nothing to do with the *cot* trigonometric function. Figure A.1 shows that, they highly resemble each other, though, except for the fact that the Homographic function asymptotically tends to zero whereas the $cot(\frac{\omega}{2})$ becomes zero at $\omega = \pi$.

The point which I was missing was that the apparent difference between the kernels gets back to the properties of the Fourier transform in the discrete and continuous case. The kernel of the Hilbert transform is such that its Fourier transform equals $-j\ sign(\omega)$. By taking the inverse Fourier transform the kernel in the domain of interest could be achieved. In the continuous case, one ends up with the Homographic function and in the discrete

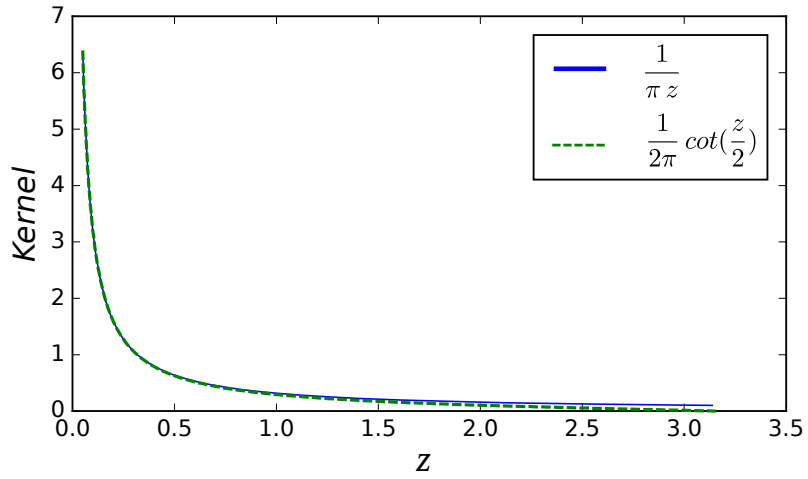

Fig. A.1 Kernels of the Hilbert transform in continuous ($\frac{1}{\pi z}$) and discrete ($\frac{1}{2\pi} cot(\frac{z}{2})$) domains.

case cotangent function shows up. Mathematically speaking, one may think of the Hilbert transform, $\mathscr{H}$ of $x(z)$ in the unknown domain z as a transform which satisfies the following

$$x_a(z) = x(z) + j\,\mathscr{H}\{x(z)\}$$

$$x(z) \underset{\mathscr{F}^{-1}}{\overset{\mathscr{F}}{\rightleftharpoons}} X(Z) \quad , \quad x_a(z) \underset{\mathscr{F}^{-1}}{\overset{\mathscr{F}}{\rightleftharpoons}} X_a(Z)$$

$$X_a(Z) = \begin{cases} X(Z) & Z = 0 \\ 2X(Z) & 0 \leq Z < max\{Z\} \\ X(Z) & Z = max\{Z\} < \infty \\ 0 & Z < 0 \end{cases} \tag{A.33}$$

$$\Rightarrow \quad \mathscr{H}\{x(z)\} = Im\{\mathscr{F}^{-1}\{X_a(Z)\}\} \tag{A.34}$$

where z and Z are the domain before and after taking Fourier transform ($\mathscr{F}$), respectively, e.g time and frequency. Having computed the $\mathscr{H}\{x(z)\}$ from the above equation, the Hilbert transform kernel, $\mathscr{H}_{kernel}(z)$, can be computed as follows

$$\mathscr{H}_{kernel}(z) = \mathscr{F}^{-1}\{\frac{\mathscr{F}\{\mathscr{H}\{x(z)\}\}}{\mathscr{F}\{x(z)\}}\} \tag{A.35}$$

and

$$\mathscr{H}\{x(z)\} = x(z) * \mathscr{H}_{kernel}(z). \tag{A.36}$$

Appendix B

Generalised Vector Taylor Series (gVTS) Approach to Robust ASR

In this Appendix the theoretical basics of the vector Taylor series (VTS) technique and its generalised version, gVTS, for additive and channel noise compensation are reviewed and discussed. In Chapter 5, the (g)VTS idea is extended to the group delay-power product spectrum domain.

B.1 Introduction

Vector Taylor series (VTS) is among the powerful methods for robust speech recognition. It is mathematically well-principled and rests upon reasonable assumptions. This technique allows for estimating the statistics of the noisy observation using an environment model along with the statistical distribution of the clean data and noise. Having estimated the statistics of the noisy observation, estimation of the clean features can be carried out. Before dealing with the VTS technique for robust ASR, let us first briefly review the Taylor series expansion.

B.2 Review of the VTS Basics

Expanding a function $f(x)$ using a weighted sum of a set of (usually orthogonal) basis functions (ϕ) is a well-established topic in mathematics and often allows for re-expressing the function in a more useful way. It takes the following general form

$$f(x) = \sum_{k=1}^{K} w_k \phi_k \tag{B.1}$$

where x, w_k and K indicate the independent variable, weights and number of basis functions, respectively. Weights are mainly computed by deploying the orthogonality of the basis functions. Taylor series expansion, Fourier and Wavelet transforms are examples of such decomposition. The main difference, however, lies in the basis functions themselves: in Fourier transform complex exponentials (sinusoids), in wavelet the scaled and shifted version of the mother wavelet and in the Taylor series polynomials are used. Each one has its own pros, cons and consequently applications.

Taylor series helps in approximating a function around a specific point using polynomials which basically forms the basis functions. If infinite terms are used, (in theory) the series converges to the exact value of the function. The practical usefulness of the method is that in many applications and within the range of interest for the independent variable(s), a few terms could return a reasonably good approximation. Therefore, the function could be parametrised in a relatively simple way which is easier to handle. Definition of Taylor series for a function $f(x)$ where x is a 1-D variable around point x_0 is as follows

$$\begin{aligned} f(x) &= \sum_{n=0}^{\infty} \frac{f^{(n)}(x_0)}{n!}(x-x_0)^n \\ &= f(x_0) + f'(x_0)(x-x_0) + \frac{f''(x_0)}{2!}(x-x_0)^2 + \ldots \end{aligned} \tag{B.2}$$

where $f^{(n)}(x_0)$ is the n^{th} derivative of $f(x)$ evaluated at point x_0. In special case which $x_0 = 0$, it is called *Maclaurin* series.

The extension to the multivariate variable, $\mathbf{x} \in \mathbb{R}^d$ (d-dimensional space), leads to *Vector* Taylor series. In practice, we are mainly interested in approximating the function with the linear (1st-order VTS, $n = 1$) or quadratic (2nd order VTS, $n = 2$) approximations, as follows

$$f(\mathbf{x}) \approx \begin{cases} f(\mathbf{x_0}) + (\mathbf{x}-\mathbf{x_0})^{\mathbf{T}} \nabla f(\mathbf{x_0}) & n = 1 \\ f(\mathbf{x_0}) + (\mathbf{x}-\mathbf{x_0})^{\mathbf{T}} \nabla f(\mathbf{x_0}) + \frac{1}{2}(\mathbf{x}-\mathbf{x_0})^{\mathbf{T}} \nabla^{\mathbf{2}} f(\mathbf{x_0})\,(\mathbf{x}-\mathbf{x_0}) & n = 2 \end{cases} \tag{B.3}$$

where $\nabla f(\mathbf{x_0})$ and $\nabla^2 f(\mathbf{x_0})$ denote the gradient vector ($\in R^d$) and Hessian matrix ($\in R^{d \text{ x } d}$) of the scalar function $f(x)$ evaluated at $\mathbf{x_0}$, respectively,

$$\nabla f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f(\mathbf{x})}{\partial x_1} \\ \frac{\partial f(\mathbf{x})}{\partial x_2} \\ . \\ . \\ . \\ \frac{\partial f(\mathbf{x})}{\partial x_d} \end{bmatrix} \tag{B.4}$$

and

$$\nabla^2 f(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & . & . & . & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_d} \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2^2} & . & . & . & \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_d} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_d \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_d \partial x_2} & . & . & . & \frac{\partial^2 f(\mathbf{x})}{\partial x_d^2} \end{bmatrix}. \tag{B.5}$$

Component-wise they take the following forms

$$\nabla f_i(x) = \frac{\partial f(\mathbf{x})}{\partial x_i}, \quad i = 1, 2, ..., d \tag{B.6}$$

$$\nabla^2 f_{i,j}(\mathbf{x}) = \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} \tag{B.7}$$

The overarching goal of the noise compensation process is to counter the effect of the noise. So, before dealing with the noise compensation process, let us review the process based on which the clean signal gets contaminated by the noise.

B.3 Environment Model

Environment model mathematically underpins the process based on which the clean signal gets contaminated with noise and out of which the noisy observation generated. Here we assume that the speech, in general, is corrupted by two types of noise, namely additive and channel. Additive noise is mainly the noise exists in the background and added to the signal in the time and frequency domains. Channel distortion reflects the effect of the channel (microphone or acoustical environment between the sound source and the microphone) which is convolutional in the time domain and multiplicative in the frequency domain. As such the environment model takes the following form

$$y[n] = x[n] * h[n] + w[n] \tag{B.8}$$

where $y[n]$, $x[n]$, $h[n]$ and $w[n]$ denote the noisy observation, clean signal, impulse response of the channel and additive noise, respectively, in the time. Computing the power spectrum yields

$$Y[k] = X[k]\,H[k] + W[k] \tag{B.9}$$

where k, Y, X and W denote discrete frequency, the short-time power spectra of the noisy observation, clean signal and additive noise, respectively and H is the square of the magnitude spectrum (frequency response) of the channel. Usually the power spectrum is estimated through the simplest method namely *periodogram*,

$$Z[k] = \frac{1}{N_{FFT}} |\mathscr{F}\{z[n]\}|^2 \tag{B.10}$$

where $z \in \{y,x,h,w\}$ and N_{FFT} is the FFT size. N_{FFT} can be dropped without loss of generality because it does not change the discriminability of the features and consequently classification results. Also to be more accurate, since in practice the speech frames are windowed using tapered windows like Hamming, (B.10) returns the $modified$ periodogram. However, it is usually referred to as periodogram.

Equation (B.9) is computed using the *Wiener-Khinchin* theorem and assuming the additive noise and the clean signal are uncorrelated. Based on this theorem for wide-sense stationary (WSS) stochastic processes, the power spectrum equals the Fourier transform of the autocorrelation sequence. Therefore, the clean speech and additive noise are assumed to be WSS. Since the channel noise is not considered to be a stochastic process, H in (B.9) should be referred to as square of the frequency response and not the power spectrum. If the channel properties changes over time, e.g. due to change in the relative position of the sound source (speaker) to the microphone, it can be considered as a stochastic process and H becomes power spectrum.

The next step is to take the log from both sides and factorising the power spectrum of the clean signal

$$
\begin{aligned}
Y[k] &= X[k]\,H[k]\,(1+\frac{W[k]}{X[k]\,H[k]}) \\
log\{Y[k]\} &= log\{X[k]\}+log\{H[k]\}+log\{1+\frac{W[k]}{X[k]H[k]}\} \\
\Rightarrow \tilde{Y}[k] &= \tilde{X}[k]+\tilde{H}[k]+log(1+exp(\tilde{W}[k]-\tilde{X}[k]-\tilde{H}[k])).
\end{aligned} \tag{B.11}
$$

which yields the environment model in the log of the power spectrum domain. Taking discrete cosine transform (DCT) of both sides provides the environment model in the cepstrum domain

$$
\tilde{y}[q] = \tilde{x}[q]+\tilde{h}[q]+C\,log\,(1+exp\,(C^{-1}\,(\tilde{w}[q]-\tilde{x}[q]-\tilde{h}[q]))) \tag{B.12}
$$

where q, C, C^{-1}, $\tilde{y}$, $\tilde{x}$, $\tilde{h}$ and $\tilde{w}$ denote the quefrency, DCT matrix, inverse DCT matrix, (real) cepstrum of the noisy observation, clean signal, channel distortion and additive noise, respectively. From now onwards, we drop the k and q for simplicity.

As seen in both (B.11) and (B.12) the noisy observation could be written as a sum of the clean representation and an extra term which could be considered as a *distortion function*, $\tilde{G}$ or $\tilde{g}$, depending on the domain,

$$
\begin{cases}
\tilde{Y} &= \tilde{X}+\tilde{G}(\tilde{X},\tilde{H},\tilde{W}) \\
\tilde{y} &= \tilde{x}+\tilde{g}(\tilde{x},\tilde{h},\tilde{w})
\end{cases} \tag{B.13}
$$

where

$$
\begin{cases}
\tilde{G} &= \tilde{X}+\tilde{H}+log(1+exp\,(\tilde{W}-\tilde{X}-\tilde{H})) \\
\tilde{g} &= \tilde{h}+C\,log(1+exp\,(C^{-1}\,(\tilde{w}-\tilde{x}-\tilde{h}))
\end{cases} \tag{B.14}
$$

The distortion function in both frequency and quefrency domains is reciprocally proportional with the signal to noise ration (SNR) and tends to zero as the SNR goes to infinity.

B.4 Model-based Noise Compensation in Feature Domain

Figure B.1 illustrates the workflow and the components of the model-based noise compensation approach. As seen, it consists of four main building blocks, namely a model for the clean features, a model for each type of noise, estimation criterion and compensation part.

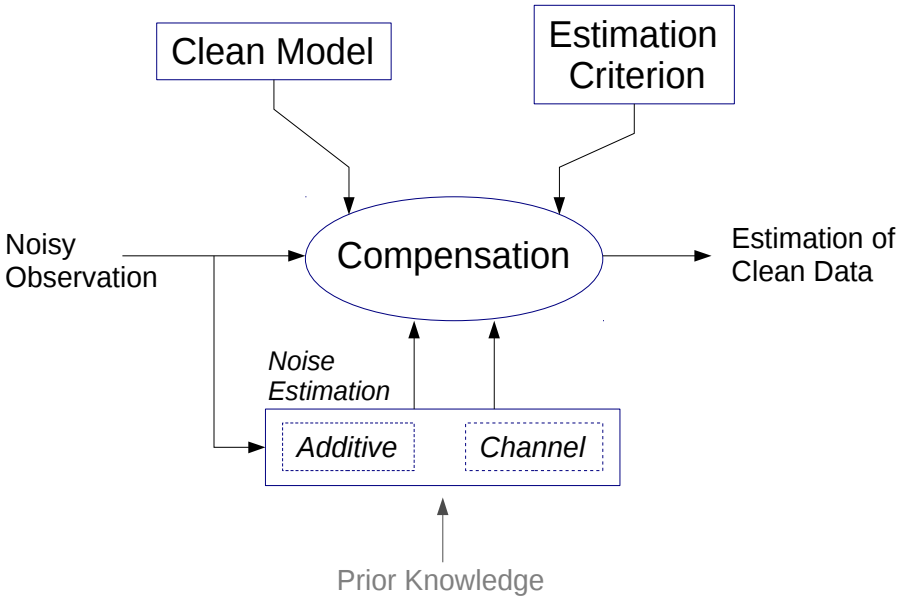

Fig. B.1 Workflow of the model-based noise compensation process.

B.4.1 Statistical Models for the Clean Features and Noise

For modelling the clean feature representation, x, usually a Gaussian mixture model (GMM) with M components is employed

$$x \sim \sum_{m=1}^{M} P_x(m)\,\mathcal{N}(x;\mu_m^x,\Sigma_m^x) \tag{B.15}$$

where $\mathcal{N}$, $P_x(m)$, μ_m^x and Σ_m^x denote the Gaussian distribution, component weight, mean vector and covariance matrix of the m^{th} Gaussian, respectively. The additive noise is usually modelled by a single Gaussian

$$w \sim \mathcal{N}(w;\mu^w,\Sigma^w) \tag{B.16}$$

where μ^w and Σ^w denote its mean vector and covariance matrix, respectively. Likewise, for modelling the channel noise a single Gaussian is used

$$h \sim \mathcal{N}(h;\mu^h,\Sigma^h) \tag{B.17}$$

As mentioned earlier, channel distortion is not a stochastic process; hence one may treat it as a deterministic variable and set the covariance matrix to zero. Since channel noise should be estimated and most of the times estimation accompanied with some uncertainty, having non-zero covariance matrix, in theory, allows for better handling of the uncertainty. However,

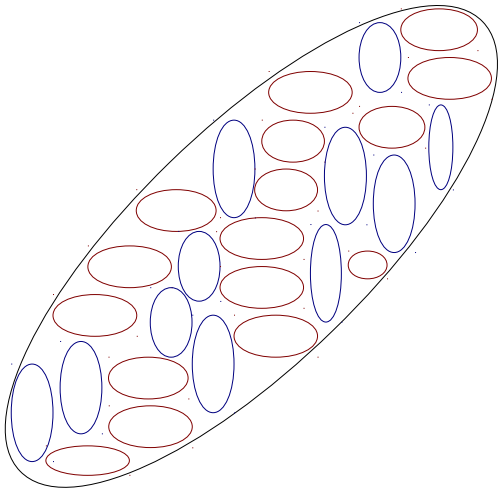

Fig. B.2 Vertical (blue) and horizontal (red) ellipses indicate Gaussians with diagonal covariance matrix and big rotated ellipse (black) illustrates a Gaussian with full covariance matrix. A GMM with diagonal covariance matrices can model data with correlated dimensions if enough components are provided.

reliable estimation is not straightforward. To keep the framework generic, here the equations are derived in the general case where an estimate of the channel mean and covariance matrix are assumed to be available.

Covariance matrices of both clean data and noise are assumed to be diagonal for mathematical convenience. It should be noted that this assumption is reasonable in the cepstrum (quefrency) domain thanks to the decorrelation provided by DCT, but in the frequency domain accompanies with some error. Note that a GMM with diagonal covariance matrix is still capable of efficiently modelling the probability density function (pdf) of correlated data. Figure B.2 shows this point. The inclined ellipse indicates the non-zero correlation between the dimensions which requires full covariance matrix to be modelled. On the other hand, the vertical and horizontal ellipses represent the diagonal covariance matrix and number of ellipses equals number of mixture components. As seen, by increasing the number of components, the data correlated data can be modelled using Gaussians with diagonal covariance matrix. The cost which should be paid would be increasing number of components of the model and consequently its parameters which runs the risk of over-fitting unless the amount of training data be sufficient to avoid this issue. So in summary, under having a large M and enough training data, compensation can be equally well performed in both domains, otherwise cepstrum domain better matches with the diagonal assumption and should lead to better performance.

B.4.2 Estimation Criterion

The goal of the noise compensation process is to counter the noise effect and estimate the clean features. So, in essence, it is an estimation problem, estimating the clean part given noisy observation. In any estimation process the output is computed in such a way that it is optimal in a particular sense or criterion. Two popular options are minimum mean square square (MMSE) and maximum a posteriori (MAP). In the problem of estimating x from observation y, optimal solution based on the MMSE and MAP criteria are as follows

$$\begin{cases} \hat{x}_{MMSE} & = \mathbb{E}[x|y] = \int_x x\, P(x|y)\, dx \\ \hat{x}_{MAP} & = \arg\max_x P(x|y) \end{cases} \tag{B.18}$$

where $\mathbb{E}\{\ \}$, $\hat{x}_{MMSE}$, $\hat{x}_{MAP}$ and $P(x|y)$ indicate expected value operator, estimate of unknown x based on MMSE, estimate based on MAP methods and the posterior probability of x given (noisy) observation y, respectively, and

$$\operatorname*{argmax}_x f(x) = \{x \mid f(x) = \max_{x'} f(x')\}. \tag{B.19}$$

MAP

In the MAP technique, after formulating the posterior density, the argmax of the conditional density which is called *mode*, should be computed. This recasts the estimation problem into an optimisation problem. The challenging issue is that the *argmax* means *global* maximum and in practice calculating the global optimum point is not straightforward. In addition, when the distribution becomes multi-modal the mode (or argmax) of a pdf is not necessarily a good representative. As demonstrated in Figure B.3, argmax may be related to a component with a very small variance which models the statistical behaviour of the data lies in a very limited subspace. As such it is not a good representative for the whole data and is uncharacteristic of the majority of the distribution.

MMSE

In MMSE the goal is to minimise the mean square error (MSE),

$$MSE_x = \mathbb{E}\{\|\hat{x} - x\|^2 | y\} = \int_x \|\hat{x} - x\|^2 P(x|y) dx \tag{B.20}$$

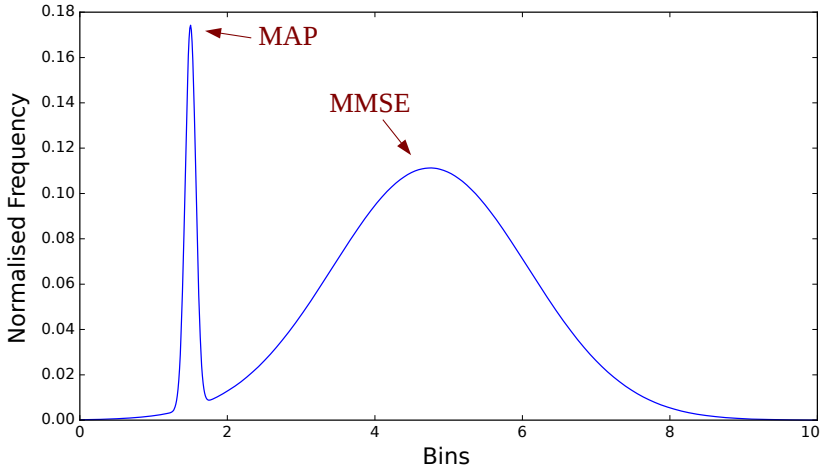

Fig. B.3 Maximum a posteriori (MAP) estimate of a bimodal distribution may not be a good representative for the distribution.

where $\hat{x}$ indicates the estimate and x denotes the true value. By setting the derivative of MSE with to zero

$$\frac{\partial MSE_x}{\partial \hat{x}} = 0 \Rightarrow \; 2\int_x (\hat{x}-x)\, P(x|y)dx = 0, \tag{B.21}$$

the optimal estimator could be computed as follows

$$\hat{x}_{MMSE} = \frac{\int_x xP(x|y)dx}{\int_x P(x|y)dx} = \int_x xP(x|y)dx \tag{B.22}$$

From practical standpoint, working out the integral in (B.22) is easier than computing the *argmax* in the MAP method especially in multidimensional space. That is why in a wide range of applications including VTS, MMSE is employed as estimation criterion.

One shortcoming of both MAP and MMSE is that they are *point estimators*. It basically means that they just provide an estimate, $\hat{x}$, of the unknown x, without any confidence measure about the estimate. The alternative methods are called *interval estimators* which along with an estimate for the unknown, supply a measure which reflects the associated uncertainty of the estimator.

B.4.3 Noise Compensation through VTS

By choosing MMSE, the starting point for carrying out the noise compensation process will be (B.22). Unknown, x, could be either in $\tilde{X}$ or $\tilde{x}$ which corresponds to frequency or quefrency domains, respectively.

Let's begin with the frequency domain by combining (B.14) with (B.22)

$$\hat{X}_{MMSE} = \int \tilde{X}\, P(\tilde{X}|\tilde{Y})\, d\tilde{X} = \int \left[\tilde{Y} - \tilde{G}(\tilde{X},\tilde{H},\tilde{W})\right] P(\tilde{X}|\tilde{Y})\, d\tilde{X}$$
$$= \tilde{Y} - \int \tilde{G}(\tilde{X},\tilde{H},\tilde{W})\, P(\tilde{X}|\tilde{Y})\, d\tilde{X} \tag{B.23}$$

The distribution of $\tilde{X}$, is modelled using a GMM with M components, based on (B.15). Using that, the posterior probability can be rewritten as follows

$$P(\tilde{X}|\tilde{Y}) = \sum_{m=1}^{M} P(\tilde{X},m|\tilde{Y}) = \sum_{m=1}^{M} P(\tilde{X}|m,\tilde{Y})P(m|\tilde{Y}) \tag{B.24}$$

where m denotes the m^{th} Gaussian of the GMM of $\tilde{X}$. For mathematical convenience, it is assumed that $\tilde{X}$ and $\tilde{Y}$ are conditionally independent, given m

$$P(\tilde{X}|m,\tilde{Y}) \approx P(\tilde{X}|m) \Rightarrow P(\tilde{X}|\tilde{Y}) \approx \sum_{m=1}^{M} P(\tilde{X}|m)P(m|\tilde{Y}). \tag{B.25}$$

This approximation results in

$$\int \tilde{G}(\tilde{X},\tilde{H},\tilde{W})\, P(\tilde{X}|\tilde{Y})\, d\tilde{X} \approx \int \tilde{G}(\tilde{X},\tilde{H},\tilde{W}) \sum_{m=1}^{M} P(\tilde{X}|m)P(m|\tilde{Y})\, d\tilde{X}$$
$$= \sum_{m=1}^{M} P(m|\tilde{Y}) \int \tilde{G}(\tilde{X},\tilde{H},\tilde{W})\, P(\tilde{X}|m)\, d\tilde{X}. \tag{B.26}$$

Since $\tilde{X}$'s distribution is a GMM, $P(\tilde{X}|m)$ is a Gaussian. The next assumption is that the majority of the probability mass of these Gaussians is located at their centre, namely at $\mu_m^{\tilde{X}}$. This implicitly means that the Gaussians are assumed to behave like Dirac delta function which could be reasonable specially when the variances become small enough. For Dirac delta function

$$f(x)\delta(x-x_0) = f(x_0)\delta(x-x_0) \Rightarrow \int f(x)\delta(x-x_0)dx = f(x_0) \tag{B.27}$$

Based on this property and assuming the Gaussians to behave like Dirac delta, one can rewrite (B.26) as follows

$$\sum_{m=1}^{M} P(m|\tilde{Y}) \int \tilde{G}(\tilde{X},\tilde{H},\tilde{W})\, P(\tilde{X}|m)\, d\tilde{X} \approx \sum_{m=1}^{M} P(m|\tilde{Y}) \int \tilde{G}(\mu_m^{\tilde{X}},\mu^{\tilde{H}},\mu^{\tilde{W}})\, P(\tilde{X}|m)\, d\tilde{X}$$
$$= \sum_{m=1}^{M} P(m|\tilde{Y}) \tilde{G}(\mu_m^{\tilde{X}},\mu^{\tilde{H}},\mu^{\tilde{W}}) \int P(\tilde{X}|m)\, d\tilde{X} = \sum_{m=1}^{M} P(m|\tilde{Y})\, \tilde{G}(\mu_m^{\tilde{X}},\mu^{\tilde{H}},\mu^{\tilde{W}}) \qquad \text{(B.28)}$$

In other words, each single Gaussian could be assumed to represent a cluster in the feature space. As such the space can be decomposed into a number of subspaces (equal to number of Gaussians) and discretised through representing each subspace by the mean of the corresponding Gaussian.

Now, the final equation for MMSE estimation can be derived

$$\hat{X}_{MMSE} = \tilde{Y} - \sum_{m=1}^{M} P(m|\tilde{Y})\, \tilde{G}(\mu_m^{\tilde{X}},\mu^{\tilde{H}},\mu^{\tilde{W}}). \qquad \text{(B.29)}$$

The only missing part in (B.29) is the posterior probability[1] $P(m|\tilde{Y})$ and for computing it, the pdf of the $\tilde{Y}$ should be estimated first.

For estimating the statistics of $\tilde{Y}$, a number of assumptions are made to make the problem tractable. The first assumption is that $\tilde{Y}$ follows a GMM distribution with M Gaussians, similar to $\tilde{X}$,

$$\tilde{Y} \sim \sum_{m=1}^{M} P_{\tilde{Y}}(m)\, \underbrace{\mathcal{N}(\tilde{Y};\mu_m^{\tilde{Y}},\Sigma_m^{\tilde{Y}})}_{P(\tilde{Y}|m)} \qquad \text{(B.30)}$$

where $P_{\tilde{Y}}(m)$, $\mu_m^{\tilde{Y}}$ and $\Sigma_m^{\tilde{Y}}$ indicate m^{th} component's weight, mean vector and covariance matrix, respectively. Again covariance matrix is assumed to be diagonal for mathematical convenience. Second assumption is that $\tilde{X}$ and $\tilde{Y}$ (as two random variables) are jointly Gaussian within each mixture component (m)

$$\begin{pmatrix} \tilde{X} \\ \tilde{Y} \end{pmatrix}_m \sim \mathcal{N}\left[\begin{pmatrix} \mu_m^{\tilde{X}} \\ \mu_m^{\tilde{Y}} \end{pmatrix}, \begin{pmatrix} \Sigma_{\tilde{X}\tilde{X}} & \Sigma_{\tilde{X}\tilde{Y}} \\ \Sigma_{\tilde{Y}\tilde{X}} & \Sigma_{\tilde{Y}\tilde{Y}} \end{pmatrix} \right] \qquad \text{(B.31)}$$

If two RVs are jointly Gaussian, the conditional probability would be Gaussian, too. This facilitates computing the integral in (B.22). These two assumption along with assuming the Gaussians behave like Dirac delta function in approximating the value of the distortion

[1] Also known as *responsibility* of the m^{th} component for observation $\tilde{Y}$.

function are the three fundamental assumptions facilitates deriving the MMSE estimate for clean feature.

So, the problem is computing the GMM of the noisy observation $\tilde{Y}$. Note that the direct estimation of the statistics of $\tilde{Y}$ from the given utterance does not work becuase the amount of the data is not enough for estimating the corresponding GMM. The available pieces of information are as follows: the function links the noisy observation to other variables, namely the clean signal and noise (both additive and channel), also the statistics of the clean features and noise are assumed to be available. However, still estimating the statistics of $\tilde{Y}$ is complicated which primarily stems from the non-linear relationship between the noisy observation with other variables in both frequency and quefrency domains. If the relation was linear and given that the involved variables has a Guassian distribution, the statistics of the $\tilde{Y}$ could be easily computed. At this point which the problem is the non-linearity, the VTS comes into the scene to linearise the non-linear function and paves the way for estimating the GMM of $\tilde{Y}$.

B.4.4 VTS in the Frequency Domain

As mentioned earlier, one of the applications of (vector) Taylor series is to express a function through a line around the point of interest which is referred to as linearisation. Here, the goal is to derive a linear relation between $\tilde{Y}$ and $\{\tilde{X},\tilde{H},\tilde{W}\}$. Linearisation through Taylor series yields

$$\tilde{Y} \approx \tilde{Y}(\tilde{X}_0,\tilde{W}_0,\tilde{H}_0)+J^{\tilde{X}}(\tilde{X}-\tilde{X}_0)+J^{\tilde{W}}(\tilde{W}-\tilde{W}_0)+J^{\tilde{H}}(\tilde{H}-\tilde{H}_0) \tag{B.32}$$

where $J^{\tilde{Z}}$ is the Jacobian matrix of $\tilde{Y}$ with respect to $\tilde{Z}$ ($\tilde{Z}\in\{\tilde{X},\tilde{H},\tilde{W}\}$) and $(\tilde{X}_0,\tilde{W}_0,\tilde{H}_0)$ denotes the point around which $\tilde{Y}$ is linearised. By definition,

$$J^{\tilde{Z}}=\frac{\partial\tilde{\mathbf{Y}}}{\partial\tilde{\mathbf{Z}}}=\begin{bmatrix} \frac{\partial\tilde{Y}_1}{\partial\tilde{Z}_1} & \frac{\partial\tilde{Y}_1}{\partial\tilde{Z}_2} & \cdot & \cdot & \cdot & \frac{\partial\tilde{Y}_1}{\partial\tilde{Z}_D} \\ \frac{\partial\tilde{Y}_2}{\partial\tilde{Z}_1} & \frac{\partial\tilde{Y}_2}{\partial\tilde{Z}_2} & \cdot & \cdot & \cdot & \frac{\partial\tilde{Y}_2}{\partial\tilde{Z}_D} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \frac{\partial\tilde{Y}_i}{\partial\tilde{Z}_j} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \frac{\partial\tilde{Y}_D}{\partial\tilde{Z}_1} & \frac{\partial\tilde{Y}_D}{\partial\tilde{Z}_2} & \cdot & \cdot & \cdot & \frac{\partial\tilde{Y}_D}{\partial\tilde{Z}_D} \end{bmatrix} \tag{B.33}$$

where D is number of the frequency bins and without loss of generality, it could also be number of the filters of the filter bank. Note that Jacobian is the general form of the gradient. Gradient is a vector which its elements are the partial derivatives of a scaler function to a

vector and Jacobian is a matrix which reflects the partial derivatives of a vector with respect to another vector.

Linearisation is performed around the mean values of all the Gaussians, namely $(\mu_m^{\tilde{X}}, \mu^{\tilde{H}}, \mu^{\tilde{W}})$ which are M points altogether. With some algebraic manipulation the Jacobians can be computed as follows

$$J_m^{\tilde{X}} = diag\{\frac{1}{1+V_m}\} \tag{B.34}$$

$$J_m^{\tilde{H}} = J_m^{\tilde{X}} = diag\{\frac{1}{1+V_m}\} \tag{B.35}$$

$$J_m^{\tilde{W}} = I - J_m^{\tilde{X}} = diag\{\frac{V_m}{1+V_m}\} \tag{B.36}$$

where $diag\{.\}$ is the operator which turns a vector into a diagonal matrix, I denotes a $D-by-D$ Identity matrix and

$$V_m = exp(\mu^{\tilde{W}} - \mu_m^{\tilde{X}} - \mu^{\tilde{H}}). \tag{B.37}$$

Using (B.32) and the Jacobian matrices, namely $J^{\tilde{X}}$, $J^{\tilde{H}}$ and $J^{\tilde{W}}$, the statistics of $\tilde{Y}$, can be computed as follows

$$\begin{cases} P_{\tilde{Y}}(m) & \approx & P_{\tilde{X}}(m) \\ \mu_m^{\tilde{Y}} & \approx & \mu_m^{\tilde{X}} + \mu^{\tilde{H}} + log(1+V_m) \\ \Sigma_m^{\tilde{Y}} & \approx & J_m^{\tilde{X}} \Sigma_m^{\tilde{X}} J_m^{\tilde{X}^T} + J_m^{\tilde{H}} \Sigma^{\tilde{H}} J_m^{\tilde{H}^T} + J_m^{\tilde{W}} \Sigma^{\tilde{W}} J_m^{\tilde{W}^T} \end{cases} \tag{B.38}$$

Having estimated the statistics of $\tilde{Y}$, namely $\{P_{\tilde{Y}}(m), \mu_m^{\tilde{Y}}, \Sigma_m^{\tilde{Y}}\}$, $\hat{X}_{MMSE}$ can be calculated using (B.29). Since the Jacobians and the covariance matrices are diagonal, $\Sigma_m^{\tilde{Y}}$ will be diagonal, too.

B.4.5 VTS in the Quefrency Domain

The statistical modelling and noise compensation may also be carried out in the cepstral (quefrency) domain. As mentioned earlier, the quefrency domain is slightly preferable over the frequency domain due to the decorrelation supplied by DCT which better match the diagonal covariance matrices. In order to derive the equations in the quefrency domain, (B.32) should be rewritten in the cepstrum domain

$$\tilde{y} \approx \tilde{y}(\tilde{x}_0, \tilde{w}_0, \tilde{h}_0) + J^{\tilde{x}}(\tilde{x} - \tilde{x}_0) + J^{\tilde{w}}(\tilde{w} - \tilde{w}_0) + J^{\tilde{h}}(\tilde{h} - \tilde{h}_0) \tag{B.39}$$

where $\tilde{z} = C\tilde{Z}$ for $\tilde{z} \in \{\tilde{y}, \tilde{x}, \tilde{h}, \tilde{w}\}$ and $\tilde{Z} \in \{\tilde{Y}, \tilde{X}, \tilde{H}, \tilde{W}\}$. The next step, as before, is computing the Jacobians. They can be computed from scratch but the following trick allows for using the Jacobians already computed in the frequency domain

$$J^{\tilde{z}} = \frac{\partial \tilde{\mathbf{y}}}{\partial \tilde{\mathbf{z}}} = \frac{\partial\, \mathbf{C}\tilde{\mathbf{Y}}}{\partial\, \mathbf{C}\tilde{\mathbf{Z}}} = C\, \frac{\partial \tilde{\mathbf{Y}}}{\partial \tilde{\mathbf{Z}}}\, C^{-1} \Rightarrow J^{\tilde{z}} = C\, J^{\tilde{Z}}\, C^{-1} \tag{B.40}$$

where C and C^{-1} are DCT and IDCT matrices and $C\,C^{-1} = I$. Using (B.40) and (B.34)-(B.36), the Jacobians in the cepstral domain would be

$$J_m^{\tilde{x}} = C\; diag\{\frac{1}{1+V_m}\}\, C^{-1} \tag{B.41}$$

$$J_m^{\tilde{h}} = J_m^{\tilde{x}} = C\; diag\{\frac{1}{1+V_m}\}\, C^{-1} \tag{B.42}$$

$$J_m^{\tilde{w}} = C\, C^{-1} - J_m^{\tilde{x}} = C\; diag\{\frac{V_m}{1+V_m}\}\, C^{-1} \tag{B.43}$$

where

$$V_m = exp(C^{-1}(\mu^{\tilde{w}} - \mu_m^{\tilde{x}} - \mu^{\tilde{h}})). \tag{B.44}$$

Note that

$$\begin{cases} z = CZ & \Rightarrow \ \frac{\partial z}{\partial Z} = C \\ Z = C^{-1} z & \Rightarrow \ \frac{\partial Z}{\partial z} = C^{-1} = C^T \end{cases} \tag{B.45}$$

Finally, statistics of $\tilde{y}$ can be computed as follows

$$\begin{cases} P_{\tilde{y}}(m) & \approx & P_{\tilde{x}}(m) \\ \mu_m^{\tilde{y}} & \approx & \mu_m^{\tilde{x}} + \mu^{\tilde{h}} + C\; log(1+V_m) \\ \Sigma_m^{\tilde{y}} & \approx & J_m^{\tilde{x}} \Sigma_m^{\tilde{x}} J_m^{\tilde{x}\,T} + J_m^{\tilde{h}} \Sigma^{\tilde{h}} J_m^{\tilde{h}\,T} + J_m^{\tilde{w}} \Sigma^{\tilde{w}} J_m^{\tilde{w}\,T} \end{cases} \tag{B.46}$$

Due to multiplication in C and C^{-1}, the Jacbians and consequently the $\Sigma_m^{\tilde{y}}$ are not diagonal. To force $\Sigma_m^{\tilde{y}}$ to be diagonal, it is multiplied in I as follows

$$\Sigma_m^{\tilde{y}} \leftarrow \Sigma_m^{\tilde{y}} \odot I \tag{B.47}$$

where $\odot$ denotes the element-wise (Hadamard) multiplication. Having computed the statsitics of $\tilde{y}$, $\hat{x}_{MMSE}$ can be estimated as follows

$$\hat{x}_{MMSE} = \tilde{y} - \sum_{m=1}^{M} P(m|\tilde{y})\ \tilde{g}(\mu_m^{\tilde{x}}, \mu^{\tilde{h}}, \mu^{\tilde{w}}). \tag{B.48}$$

where $\tilde{g}$ is the distortion function in the cepstrum domain (B.14).

B.4.6 Advantages of VTS

VTS method has a remarkable capability in improving the performance of the speech recognition system in noisy condition. Also it does not worsen the performance of the system in the clean/matched condition. In fact, contrary to many robust methods which their advantage is highlighted in low SNRs, VTS works well in all SNRs.

In addition, it does not need to *stereo data*[2] which may be unavailable in practice. As a matter of fact, if such data becomes available one may resort to multi-style training or model adaptation based on the amount of such data. However, VTS only needs clean data. It should be noted that even in multi-style mode VTS results in some performance improvement. The reason backs to the fact that in this algorithm the features are sort of normalised using the background model and this alleviate the mismatch induced by any nuisance factor. So, in the case which the system is trained only using clean data VTS can be though of as feature enhancement technique and in multi-style case in which the model of the clean data is estimated using the multi-style data it should be considered as the feature normaliser.

Another advantage of the VTS is that it does not need an accurate estimate of noise and with even a poor estimate of the noise, still it leads to significant performance improvement. This is due to its general structure and the model of the clean data which plays a central role and can be reliably learned offline.

Like many other methods, VTS rests upon some assumptions which implicitly puts constraint on the conditions which it can work well within. Discarding the non-linear terms after Taylor series expansion accompanies with some errors. However, this error goes down by increasing number of Gaussians, because the non-linear terms are proportional with the variances of the Guassians and by increasing the M the variances goes down. On the other hand, increasing M, raises the number of the parameters of the models and in this case, more training data is required for efficient and reliable model training. In addition the noise and

[2]Data that consists of simultaneous recordings of both the clean and noisy speech.

speech are assumed to be uncorrelated, which is true in expected sense but on a frame basis this is not the case. Furthermore, in a wide range of features such as generalised MFCC, PLP, PNCC and phase-based representations, instead of *log*, power transformation is used. The current VTS formulation cannot be extended to these techniques. In the next section, we develop a novel formulation for VTS assuming that instead of the log, power transformation is used for compression of the power spectrum or the filter bank energies (FBE). From computational viewpoint, it is more costly than the trivial feature extraction techniques, but still is easily affordable given the capabilities of the modern computers.

B.5 Generalised VTS

Conventional VTS assumes that the log function is used for compression. In a wide range of features like PLP, PNCC and phase/group delay-based features, power transformation is used which helps in improving the robustness of the features. In this section we first review the advantages of replacing the log with the generalised logarithmic function and then re-derive the equations for a generalised VTS (gVTS) which assumes usage of power transformation or generalised logarithmic function (*GenLog*) instead of logarithmic function.

B.5.1 Generalised Logarithmic Function

Definition

The generalised logarithmic function, for the first time, was introduced in 1964 in Statistics literature as Box-Cox transformation [15]. It is defined as follows

$$\begin{cases} GenLog(x;\alpha) = \frac{1}{\alpha}(x^{\alpha}-1), \quad x>0 \quad \alpha \neq 0 \\ \lim_{\alpha\to 0} GenLog(x;\alpha) = log(x), \end{cases} \tag{B.49}$$

where α is its parameter and when α approaches zero, *GenLog* converges to log. Historically, it was put forward as an extension to Tukey's ladder of powers [183], which was discontinuous at 0. This transformation, resolves this issue and unifies the log and power transformation (x^{α}).

Both -1 in the numerator and α in the denominator can be discarded without loss of generality because they are identical for all the classes and does not change the discriminability of the features and the classification results. As such, *GenLog* becomes equivalent to the power transformation, namely x^{α} and both would have similar statistical effect. This shows that log transformation is a special case of the power transformation where α tends to 0.

Taking *GenLog* from both sides of the environment model yields

$$\breve{Y} \;=\; \breve{X}\,\breve{H}\,(1+(\frac{\breve{W}}{\breve{X}\breve{H}})^{\frac{1}{\alpha}})^{\alpha} \tag{B.50}$$

where $\breve{Z} = Z^{\alpha}$ for $Z \in \{Y,X,H,W\}$.

Statistical Effects

From statistical standpoint, it is claimed that this transform can potentially be helpful in enhancing the linearity (regression), Gaussianity and homoscedasticity (variance stabilisation) [15], if its parameter set properly.

In order to investigate the effect of α in speech processing context, the histograms of the FBEs were computed using the test set A of the Aurora-4 database which includes 330 signals supplying 40 minutes of clean speech. Figure B.4 shows the results and the influence of α on the distribution. As can be seen, the support of the histogram and consequently the variance of the distribution go up by increasing this parameter. On the other hands, by decreasing this parameter the distribution becomes more Gaussian in term of having a smaller skewness and (excess) kurtosis [3]. However, this should not be confused with Gaussianisation, which results in a sequence with an exact normal distribution. It should be also noted that although Gaussian distribution is desirable, it is not the ultimate goal. The main target is to improve the performance measure, e.g. word error rate (WER) in ASR.

It may be argued that by taking *log* from both sides of the (B.50), one ends up with equations similar to the conventional VTS. Let us compute the log to clarify this point

$$log\breve{Y} \;=\; log(\breve{X}) + log(\breve{H}) + \alpha\, log(1+(\frac{\breve{W}}{\breve{X}\breve{H}})^{\frac{1}{\alpha}}). \tag{B.51}$$

In this case, the variable which should be statistically modelled is $log(\breve{X}) = \alpha\, log(X)$ whereas in gVTS, the variable which the algorithm is going to compensate the effect of noise on is X^{α}. In the former the target variable is a linear function of α, namely $\alpha\, log(X)$, while in the latter, it is a non-linear function of α, i.e. X^{α}. Linear effect of α means that it only affects the first and second order statistics but has no influence on the higher-order statistics. This means it has no effect on the *overall* shape and to be more precise the family which the random variable belongs too. However, non-linear effect of α means that by changing it the overall shape of the density function and the family which the distribution belongs to will be

[3] For Gaussian distribution both skewness (third order statistics) and excess kurtosis (forth-order statistics) are zero.

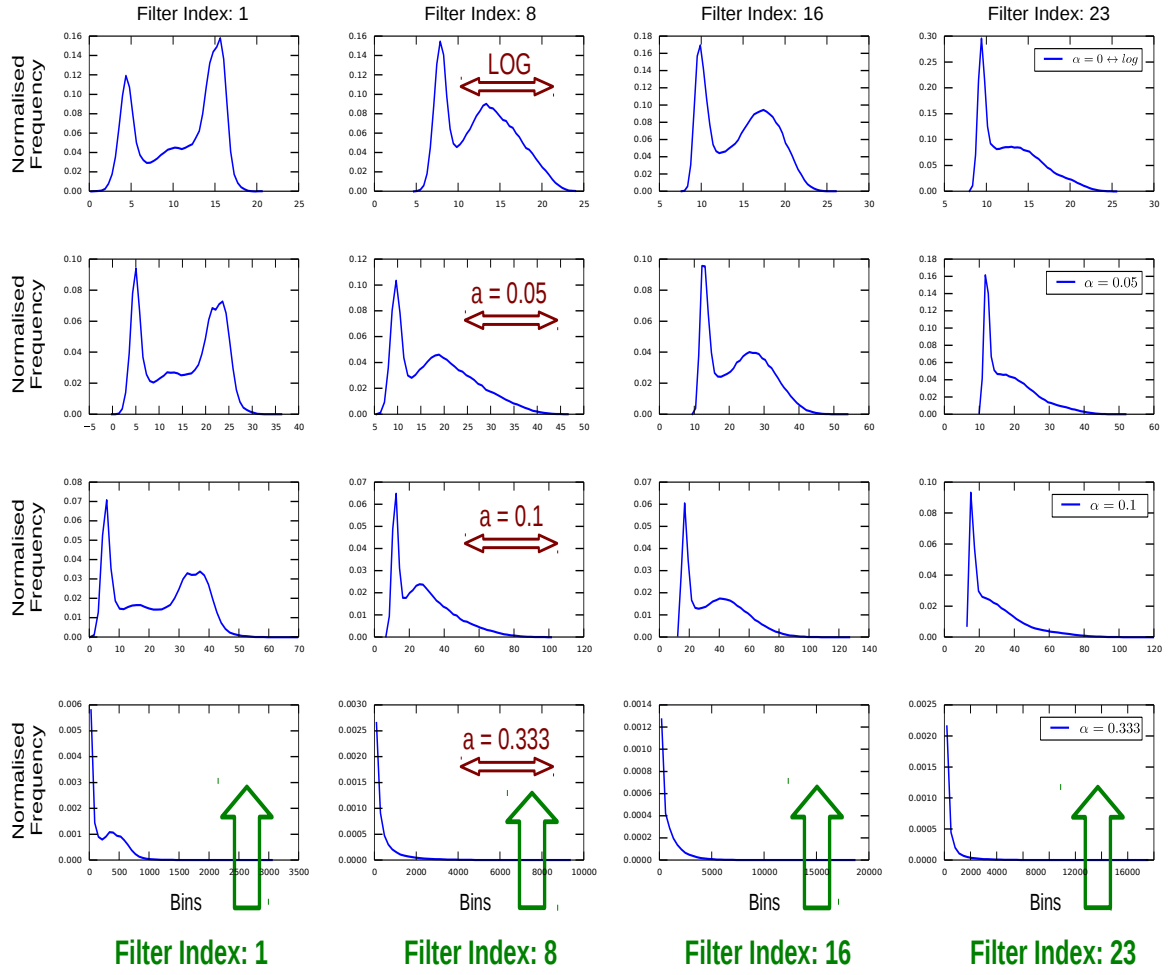

Fig. B.4 Distribution of the FBEs after compressing the FBEs of the m^{th} filter through log and GenLog in clean condition (sampling rate: 16 kHz, number of filters: 23, scale: Mel). By Increasing the α fitting a GMM on the distribution would be more difficult. Database: test set A of Aurora-4, number of signals: 330, number of frames which have been used in estimating the histograms: $\approx$ 241000 ($\approx$ 40 minutes).

changed. As such, by taking the log of (B.50) the gVTS gets reduced to conventional VTS and the effect of α will be neutralised.

Potential Usefulness in Robustness

As will be shown in Table B.1 (and also Table 4.7) at the end of this appendix , the mere replacement of the *log* with *GenLog* in the MFCC framework, which yields gMFCC [171], can lead to a substantial robustness improvement. This is, to some extent, related to the modification of the statistical distribution of the features but there is another important reason which is the SNR boost provided by applying the *GenLog* or power transformation. This

potentially implies more robustness in ASR. In order to demonstrate how using *GenLog* (or power transformation) improves the SNR, two justifications are put forward.

Let us compute the SNR as the fraction of the power spectra of the clean to additive noise after applying the power transformation. Figure B.5 shows the *GenLog* for two different αs where $\alpha_2 > \alpha_1$. The relation between the SNRs would be as follows

$$\begin{cases} \breve{X}_1 = X^{\alpha_1} \\ \breve{X}_2 = X^{\alpha_2} \\ \breve{W}_1 = W^{\alpha_1} \\ \breve{W}_2 = W^{\alpha_2} \end{cases} \Rightarrow \begin{cases} SNR_1 = \frac{\breve{X}_1}{\breve{W}_1} \\ SNR_2 = \frac{\breve{X}_2}{\breve{W}_2} = \frac{\breve{X}_1 + \Delta\breve{X}_{1\to 2}}{\breve{W}_1 + \Delta\breve{W}_{1\to 2}} \\ X > W \end{cases} \xrightarrow{\alpha_2 > \alpha_1} SNR_2 > SNR_1 \tag{B.52}$$

In general, it is assumed that the system is operating in positive SNR_{dB} regime. In this case, for $\alpha_2 > \alpha_1$ the $\Delta X_{1\to 2}$ is larger than $\Delta W_{1\to 2}$ and $SNR_2 > SNR_1$. Therefore, increasing α leads to SNR improvement.

However, as mentioned there is an upper bound on the values which α can take. As illustrated in Figure B.4 by increasing the α the support of the histogram expanded and the distribution becomes more peaky. This means that the mean, variance and more importantly skewness and kurtosis of the distribution would increase. As a result, fitting the distribution with a GMM would be more difficult because for reaching a reasonable fit many components and consequently huge amount of data would be required.

The other way of justifying the advantage of the *GenLog* with respect to the log in highlighting the role the clean part relative to the noise is through using the notion of *sensitivity* [182]. It is mainly applied in control engineering and is meant to reflect the sensitivity of a function (systems response) w.r.t a variable. Sensitivity is defined as follows

$$S_x^y = \frac{\partial\, logy}{\partial\, logx} = \frac{\frac{\partial y}{y}}{\frac{\partial x}{x}} = \frac{\frac{\partial y}{\partial x}}{\frac{y}{x}} \tag{B.53}$$

where y is the response and x is the variable of interest. The higher the value of S, the higher the sensitivity of the system to x. In addition, positive and negative values for S indicate positive and negative correlation, respectively. Now, let us compute the sensitivity of the noisy observation with respect to the power spectra of the clean signal and additive noise

$$S_W^{\breve{Y}} = \frac{\frac{\partial \breve{Y}}{\partial W}}{\frac{\breve{Y}}{W}} = \alpha\frac{W}{X+W} \qquad S_X^{\breve{Y}} = \frac{\frac{\partial \breve{Y}}{\partial X}}{\frac{\breve{Y}}{X}} = \alpha\frac{X}{X+W} \tag{B.54}$$

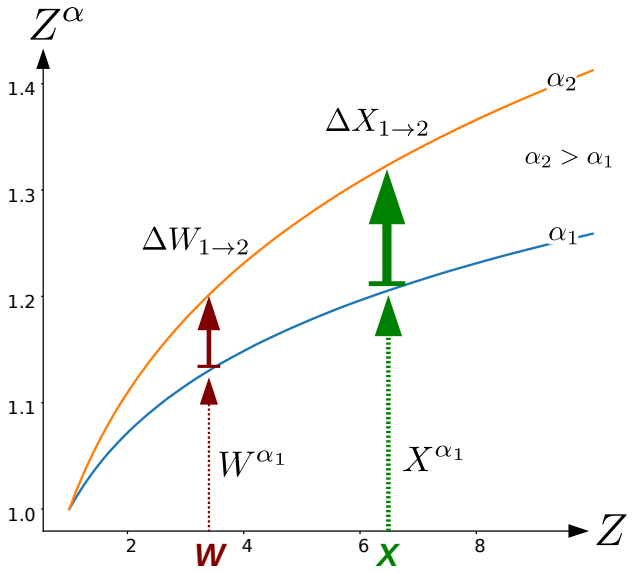

Fig. B.5 Increasing α in the power transformation boosts the relative ratio of the clean to the noise power spectra and consequently the SNR.

where $S_X^{\check{Y}}$ and $S_W^{\check{Y}}$ are sensitivities of $\check{Y}$ w.r.t X and W, respectively. Both $S_X^{\check{Y}}$ and $S_X^{\check{Y}}$ are positive and proportional with α. This means the higher the α the more the liability of Y to W and the higher the influence of X on Y, too, and vice versa. However, what is important is not the individual sensitivities but the difference in the sensitivities, ΔS,

$$\Delta S = S_X^{\check{Y}} - S_W^{\check{Y}} = \alpha \frac{X-W}{X+W}. \tag{B.55}$$

which could be thought of as the relative sensitivity. As mentioned, in general it is assumed that (on average) SNR is positive, so $X - W > 0$ and consequently $\Delta S > 0$. This means that although by increasing α, the sensitivity of Y to both X and W goes up, the relative contribution of X would be higher and proportional with α. Therefore, higher α further asserts the impact of the clean part relative to the additive noise and consequently boosts the SNR. In other words, the net effect of X on Y would be higher than the net effect of W on Y. As a result, an increase in α leads to SNR improvement. However, as mentioned, too large α has de-constructive influence on the statistical distribution of the features.

So the higher the α the better the performance in dealing with additive noise. However, it turns out that increasing α has a negative effect when the channel mismatch exists. The reason backs to the role the channel plays in the environment model. It is multiplied in the X and no matter how strong X is, the higher the α, the larger the $\check{H}$ effect and consequently the higher the influence of H on $\check{Y}$. As such if there is a channel mismatch between test and train conditions, it will be further pronounced by larger α. Mathematically, it runs as follows

$$S_H^{\breve{Y}} = \frac{\frac{\partial \breve{Y}}{\partial H}}{\frac{\breve{Y}}{H}} = \alpha. \tag{B.56}$$

For mathematical convenience, in this case it is assumed that there is no additive noise. As (B.56) shows, the higher the α, the higher the sensitivity to the channel.

B.6 Deriving the Generalised VTS Equations

B.6.1 gVTS in the Frequency Domain

The aforementioned arguments provide pre-requisite incentive for reformulating the VTS after substituting the logarithm with the generalised logarithmic function. Taking *GenLog* from both sides of the environment model yields (B.50) and as seen, the clean part ($\breve{X}$) is distorted by a distortion function, $\breve{G}$,

$$\begin{aligned} \breve{Y} &= \breve{X}\, \breve{G}(\breve{X}, \breve{H}, \breve{W}) \\ \breve{G}(\breve{X}, \breve{H}, \breve{W}) &= \breve{H}\,(1 + (\frac{\breve{W}}{\breve{X}\breve{H}})^{\frac{1}{\alpha}})^{\alpha}, \end{aligned} \tag{B.57}$$

where the higher the SNR, the closer the $\breve{G}$ to unity. Similar to the conventional VTS, the ultimate goal of the noise compensation through gVTS is to counter this function and extract an estimate of $\breve{X}$ given $\breve{Y}$.

To this end, the statistical distributions of the clean features and noise should be estimated

$$\begin{cases} \breve{X} \sim \sum_{m=1}^{M}\ p_{\breve{X}}(m)\ \mathcal{N}(\mu_m^{\breve{X}}, \Sigma_m^{\breve{X}}) \\ \breve{W} \sim \mathcal{N}(\mu^{\breve{W}}, \Sigma^{\breve{W}}), \\ \breve{H} \sim \mathcal{N}(\mu^{\breve{H}}, \Sigma^{\breve{H}}), \end{cases} \tag{B.58}$$

where M, $p_{\breve{x}}(m)$, μ and Σ denote the number of components, weight, mean vector and covariance matrix, respectively. The statistical models could be learned either in the frequency domain (*GenLog* of FBEs) or in the cepstrum domain (DCT of the *GenLog* of FBEs). Let us

start with simpler case, namely frequency domain. Using MMSE as estimation criterion

$$\begin{aligned}\hat{X}_{MMSE} &= \mathbb{E}[\check{X}|\check{Y}] = \int \check{X}\, p(\check{X}|\check{Y}) d\check{X} \\ &\approx \int \frac{\check{Y}}{G(\check{X},\check{H},\check{W})} \sum_{m=1}^{M} p(\check{X}|m)\; p(m|\check{Y})\; d\check{X} \\ &\approx \check{Y} \sum_{m=1}^{M} p(m|\check{Y})\; \frac{1}{G(\mu_m^{\check{X}},\mu^{\check{H}},\mu^{\check{W}})},\end{aligned} \tag{B.59}$$

and the underlying assumptions are exactly the same as the conventional VTS formulation.

The only missing part in B.59 is $p(m|\check{Y})$. Using the first-order Taylor series, the non-linear relationship in the environment model between the $\check{Y}$ and other variables after applying *GenLog* in (B.57) can be linearised as follows

$$\check{Y} \approx \check{Y}(\check{X}_0,\check{W}_0,\check{H}_0) + J^{\check{X}}(\check{X}-\check{X}_0) + J^{\check{W}}(\check{W}-\check{W}_0) + J^{\check{H}}(\check{H}-\check{H}_0) \tag{B.60}$$

where J^Z is the Jacobian matrix of $\check{Y}$ with respect to Z ($Z \in \{\check{X},\check{H},\check{W}\}$) and $(\check{X}_0,\check{W}_0,\check{H}_0)$ denotes the point around which $\check{Y}$ is linearised. Linearisation is performed around the mean values, namely $(\mu_m^{\check{X}},\mu^{\check{H}},\mu^{\check{W}})$ which will be M points altogether. Therefore, the Jacobians should be evaluated at each point.

Since this time the terms are not added together, computing the partial derivative or the Jacobians is a bit more complicated. In this regard, one can make use of the chain rule to break the problem into smaller sub-problems and also take advantage of the following formula

$$\check{Z} = Z^{\alpha} \Rightarrow \begin{cases} \frac{\partial \check{Z}}{\partial Z} = \alpha\, Z^{\alpha-1} = \alpha \check{Z}^{1-\frac{1}{\alpha}} \\ \frac{\partial Z}{\partial \check{Z}} = \frac{1}{\alpha}\check{Z}^{\frac{1}{\alpha}-1} \end{cases} \tag{B.61}$$

With some algebraic manipulation it can be shown that

$$J_m^{\check{X}} = \frac{\partial \check{Y}}{\partial \check{X}}\Big|_{(\mu_m^{\check{X}},\mu^{\check{H}},\mu^{\check{W}})} = \frac{\partial \check{Y}}{\partial Y}\frac{\partial Y}{\partial X}\frac{\partial X}{\partial \check{X}}\Big|_{(\mu_m^{\check{X}},\mu^{\check{H}},\mu^{\check{W}})} = diag\{\frac{\mu^{\check{H}}}{(1+\check{V}_m)^{1-\alpha}}\} \tag{B.62}$$

$$J_m^{\check{H}} = \frac{\partial \check{Y}}{\partial \check{H}}\Big|_{(\mu_m^{\check{X}},\mu^{\check{H}},\mu^{\check{W}})} = \frac{\partial \check{Y}}{\partial Y}\frac{\partial Y}{\partial H}\frac{\partial H}{\partial \check{H}}\Big|_{(\mu_m^{\check{X}},\mu^{\check{H}},\mu^{\check{W}})} = diag\{\frac{\mu_m^{\check{X}}}{(1+\check{V}_m)^{1-\alpha}}\} \tag{B.63}$$

$$J_m^{\check{W}} = \frac{\partial \check{Y}}{\partial \check{W}}\Big|_{(\mu_m^{\check{X}},\mu^{\check{H}},\mu^{\check{W}})} = \frac{\partial \check{Y}}{\partial Y}\frac{\partial Y}{\partial W}\frac{\partial W}{\partial \check{W}}\Big|_{(\mu_m^{\check{X}},\mu^{\check{H}},\mu^{\check{W}})} = diag\{(\frac{\check{V}_m}{1+\check{V}_m})^{1-\alpha}\} \tag{B.64}$$

where

$$\check{V}_m = (\frac{\mu^{\check{W}}}{\mu_m^{\check{X}}\,\mu^{\check{H}}})^{\frac{1}{\alpha}}. \tag{B.65}$$

Having computed the Jacobians, the statistics of $\check{Y}$ can be computed as follows

$$\begin{cases} P_{\check{Y}}(m) & \approx\ P_{\check{X}}(m) \\ \mu_m^{\check{Y}} & \approx\ \mu_m^{\check{X}}\,\mu^{\check{H}}(1+(\frac{\mu^{\check{W}}}{\mu_m^{\check{X}}\mu^{\check{H}}})^{\frac{1}{\alpha}})^{\alpha} \\ \Sigma_m^{\check{Y}} & \approx\ J_m^{\check{X}}\Sigma_m^{\check{X}}{J_m^{\check{X}}}^T + J_m^{\check{W}}\,\Sigma^{\check{W}}\,{J_m^{\check{W}}}^T + J_m^{\check{H}}\,\Sigma^{\check{H}}\,{J_m^{\check{H}}}^T. \end{cases} \tag{B.66}$$

B.6.2 gVTS in the Quefrency Domain

In case of the conversational VTS, since the clean part and the distortion function are additive, formulating the problem in the cepstral domain highly resembles the frequency domain. However, by using *GenLog*, instead of addition, the clean part and the distortion function become multiplicative and DCT of multiplication of two terms does not have a clear-cut relationship with the corresponding DCTs of each one

$$\check{y} = C\,\check{Y}\ = C\ \left[\check{X}\,\check{H}\,(1+(\frac{\check{W}}{\check{X}\check{H}})^{\frac{1}{\alpha}})^{\alpha}\right]. \tag{B.67}$$

In other words, we should first compute the $\check{Y}$ and then multiply the results in C. Operations inside $[\,.\,]$ in (B.67) are element-wise (Hadamard) whereas multiplication of C in the resultant vector is based on multiplications of the matrices. So, we cannot write the $\check{y}$ as multiplication of $\check{x}$ and distortion function, $\check{g}$. As such in case of gVTS we cannot directly carry out the compensation in the quefrency domain.

As mentioned earlier the main advantage of the cepstrum domain is the decorrelation of the features which better complies with the diagonal covariance matrices of the Gaussians. It should be noted that since DCT is a linear operation (C) the following holds

$$\begin{cases} \check{z} = C\,\check{Z} \\ \check{Z} \sim \sum_{m=1}^{M}\ p_{\check{Z}}(m)\ \mathcal{N}(\mu_m^{\check{Z}}, \Sigma_m^{\check{Z}}) \\ \check{z} \sim \sum_{m=1}^{M}\ p_{\check{z}}(m)\ \mathcal{N}(\mu_m^{\check{z}}, \Sigma_m^{\check{z}}) \end{cases} \Rightarrow \begin{cases} p_{\check{z}}(m) = p_{\check{Z}}(m) \\ \mu_m^{\check{z}} = C\,\mu_m^{\check{Z}}, \\ \Sigma_m^{\check{z}} = C\,\Sigma_m^{\check{Z}}\,C^T \end{cases} \tag{B.68}$$

In addition, it can be shown that after applying a linear transformation, the likelihood and consequently the posterior probabilities, for a GMM, do not change

$$\begin{cases} p_{\check{z}}(m) = p_{\check{Z}}(m) \\ P(\check{z}|m) = P(\check{Z}|m) \end{cases} \Rightarrow P(m|\check{z}) = P(m|\check{Z}) \tag{B.69}$$

Now for doing the compensation in the cepstrum domain, in the first step (B.59) should be rewritten as follows

$$\begin{aligned} \hat{X}_{MMSE} &= \mathbb{E}[\check{X}|\check{y}] = \int \check{X}\,P(\check{X}|\check{y})d\check{X} \\ &= \int \frac{\left[C^{-1}\,\check{y}\right]}{G(\check{X},\check{H},\check{W})} \sum_{m=1}^{M} P(\check{X}|m)\,P(m|\check{y})\,d\check{X} \\ &= \left[C^{-1}\,\check{y}\right] \sum_{m=1}^{M} P(m|\check{y})\,\frac{1}{G(C^{-1}\mu_m^{\check{x}}, C^{-1}\mu^{\check{h}}, C^{-1}\mu^{\check{w}})}, \end{aligned} \tag{B.70}$$

The next step is computing the posterior probabilities, $p(m|\check{y})$. Using the first-order Taylor series

$$\check{y} \approx \check{y}(\check{x}_0, \check{w}_0, \check{h}_0) + J^{\check{x}}(\check{x}-\check{x}_0) + J^{\check{w}}(\check{w}-\check{w}_0) + J^{\check{h}}(\check{h}-\check{h}_0) \tag{B.71}$$

where $J^{\check{z}} = \frac{\partial \check{y}}{\partial \check{z}}$ denotes the Jacobian matrix for $\check{z} \in \{\check{x}, \check{w}, \check{h}\}$ and $(\check{x}_0, \check{h}_0, \check{w}_0)$ is the point around which the linearisation is carried out. With some algebraic manipulation the Jacobians can be worked out

$$J_m^{\check{x}} = \frac{\partial \check{y}}{\partial \check{x}} = \frac{\partial \check{y}}{\partial \check{Y}} \underbrace{\frac{\partial \check{Y}}{\partial Y} \frac{\partial Y}{\partial X} \frac{\partial X}{\partial \check{X}}}_{J^{\check{X}}} \frac{\partial X}{\partial \check{x}} = C\, diag\{\frac{\mu^{\check{H}}}{(1+\check{V}_m)^{1-\alpha}}\}\, C^{-1} \tag{B.72}$$

$$J_m^{\check{h}} = \frac{\partial \check{y}}{\partial \check{h}} = \frac{\partial \check{y}}{\partial \check{Y}} \underbrace{\frac{\partial \check{Y}}{\partial Y} \frac{\partial Y}{\partial H} \frac{\partial H}{\partial \check{H}}}_{J^{\check{H}}} \frac{\partial H}{\partial \check{h}} = C\, diag\{\frac{\mu_m^{\check{X}}}{(1+\check{V}_m)^{1-\alpha}}\}\, C^{-1} \tag{B.73}$$

$$J_m^{\check{w}} = \frac{\partial \check{Y}}{\partial \check{W}} = \frac{\partial \check{y}}{\partial \check{Y}} \underbrace{\frac{\partial \check{Y}}{\partial Y} \frac{\partial Y}{\partial W} \frac{\partial W}{\partial \check{W}}}_{J^{\check{W}}} \frac{\partial W}{\partial \check{w}} = C\, diag\{(\frac{\check{V}_m}{1+\check{V}_m})^{1-\alpha}\}\, C^{-1} \tag{B.74}$$

where

$$\check{V}_m = (\frac{C^{-1}\mu^{\check{w}}}{C^{-1}\mu_m^{\check{x}}\ C^{-1}\mu^{\check{h}}})^{\frac{1}{\alpha}}. \tag{B.75}$$

Having computed the Jacobians, the statistics of $\check{y}$ can be computed as follows

$$\begin{cases} P_{\check{y}}(m) & \approx\ P_{\check{x}}(m) \\ \mu_m^{\check{y}} & \approx C\left[\left(C^{-1}\mu_m^{\check{x}}\right)\left(C^{-1}\mu^{\check{h}}\right)(1+\check{V}_m)^{\alpha}\right] \\ \Sigma_m^{\check{y}} & \approx\ J_m^{\check{x}}\Sigma_m^{\check{x}}\,{J_m^{\check{x}}}^T + J_m^{\check{w}}\,\Sigma^{\check{w}}\,{J_m^{\check{w}}}^T + J_m^{\check{h}}\,\Sigma^{\check{h}}\,{J_m^{\check{h}}}^T. \end{cases} \tag{B.76}$$

B.7 Phase Factor in the VTS and gVTS Techniques

In the formulations derived so far for both VTS and gVTS, it was assumed that the noise and the clean speech are uncorrelated. Although this is a relatively reasonable assumption in the macro-structure expected sense, in a frame-wise micro-structure it does not hold. Let us rewrite the environment model without setting the cross term to zero

$$Y_k\ =\ X_k H_k\ +\ W_k\ +\ 2\sqrt{X_k H_k W_k}\, cos(\phi_{X_k} + \phi_{H_k} - \phi_{W_k}). \tag{B.77}$$

where Z_k denotes the power spectrum of z in the frequency bin k and ϕ_{Z_k} denotes the respective phase spectrum. Let

$$\lambda_k = cos(\phi_{X_k} + \theta_{H_k} - \theta_{W_k}) = \frac{Y_k\ -\ X_k H_k\ -\ W_k}{2\sqrt{X_k H_k W_k}} \tag{B.78}$$

where λ_k is called *phase factor*. Although Y_k, X_k, H_k, W_k are the power spectra (k denotes the frequency index), without loss of generality, similar relations hold between the outputs of the filter bank, namely Y_l, X_l, H_l, W_l, too, where l indicates the index of the filter in the filter bank. In order to compute the equivalence of λ_k, after applying the filterbank, (B.78) may be rewritten as follows

$$\lambda_l = \frac{Y_l - X_l H_l - W_l}{2\sqrt{X_l H_l W_l}} \tag{B.79}$$

Figure B.6, shows the histograms of λ_l for different l. As seen, almost all are zero-mean which is compatible with the fact that speech and noise are uncorrelated in expected sense. However, as seen, the variance is not zero and it is higher for low frequency filters. For mathematical convenience and with good approximation, especially for high frequency filters with a wider bandwidth, the phase factor is assumed to have a Gaussian distribution with zero mean [184]

$$\lambda_l \sim \mathcal{N}(0, \sigma_l^2) \Rightarrow \lambda \sim \mathcal{N}(0, \Sigma^\lambda) \tag{B.80}$$

where σ_l^2 and Σ^λ indicate the variance and the (diagonal) covariance matrix of the phase factor of the l^{th} filter in the filter bank.

Backing to the compensation process, since the mean of this factor is zero, the mean of $\tilde{Y}$ in VTS or $\check{Y}$ in gVTS does not change. However, a new term will be added to the covariance matrix of the noisy observation. In order to compute that term, the Jacobian should be computed first. For the conventional VTS

$$J_m^\lambda \overset{VTS}{=} \frac{\partial \tilde{Y}}{\partial \lambda} = \frac{\partial \tilde{Y}}{\partial Y}\frac{\partial Y}{\partial \lambda} = 2\frac{\sqrt{V_m}}{1+V_m} \tag{B.81}$$

and consequently (B.38) should be rewritten as follows

$$\Sigma_m^{\check{Y}} \leftarrow \Sigma_m^{\check{Y}} + J_m^\lambda \, \Sigma^\lambda \, {J_m^\lambda}^T. \tag{B.82}$$

By the same token, for the gVTS

$$J_m^\lambda \overset{gVTS}{=} \frac{\partial \check{Y}}{\partial \lambda} = \frac{\partial \check{Y}}{\partial Y}\frac{\partial Y}{\partial \lambda} = 2\alpha\frac{\mu_m^{\check{X}} \mu^{\check{H}} \sqrt{V_m}}{(1+V_m)^{1-\alpha}} \tag{B.83}$$

which results in

$$\Sigma_m^{\check{Y}} \leftarrow \Sigma_m^{\check{Y}} + J_m^\lambda \, \Sigma^\lambda \, {J_m^\lambda}^T. \tag{B.84}$$

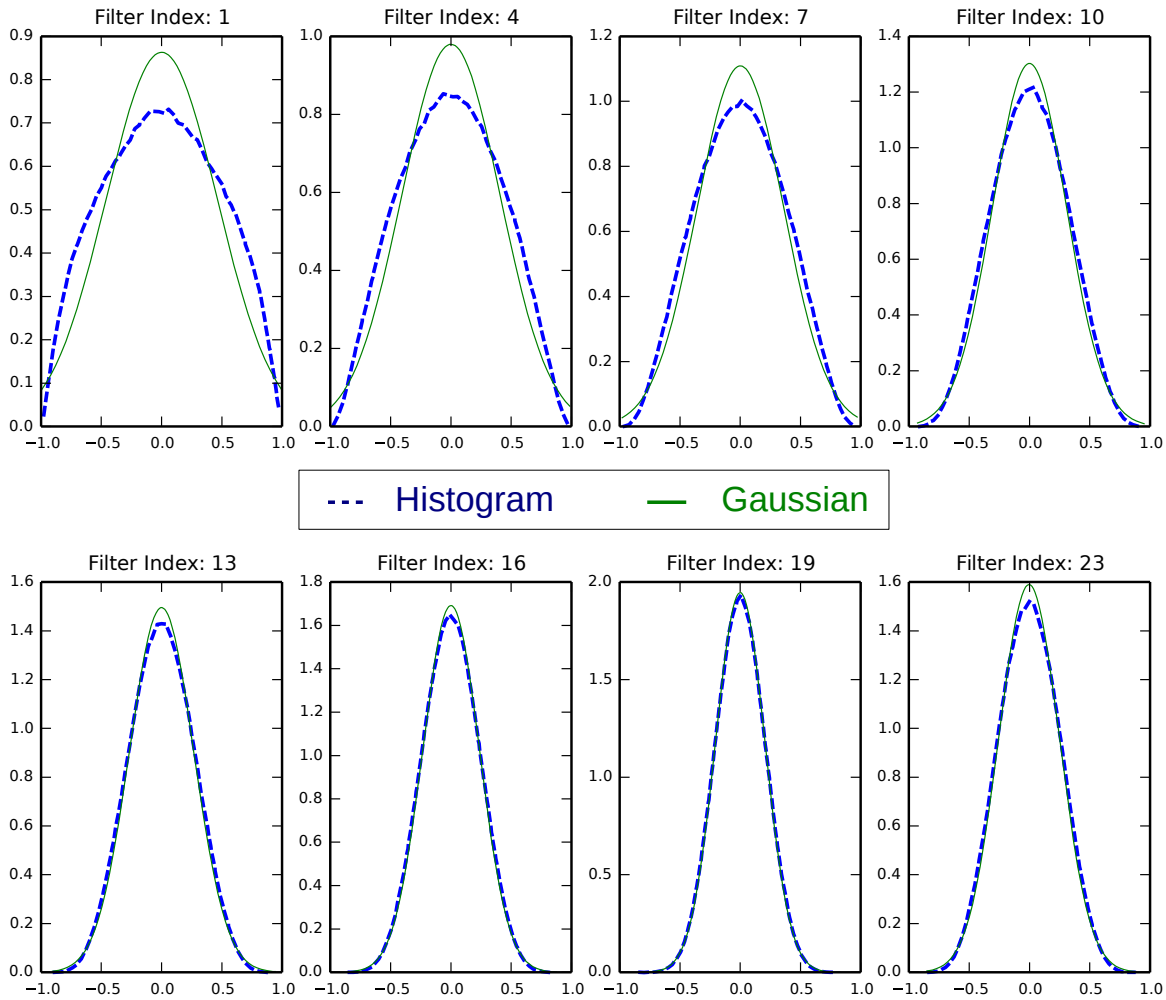

Fig. B.6 Histogram of the phase factor, λ, for different filters of the Mel filter bank. The navy dashed curve shows the histogram of the phase factor and the green curve shows the Gaussian distribution with the same mean and variance. The additive noise is Babble and 1206 signals of the *devset* of the Aurora-4 database have been used (about 134 minutes of speech) to compute the histograms. As seen, the Gaussian distribution is a reasonable approximation for the distribution of the phase factor, especially at high frequencies.

B.8 Experimental Results

In this part, the ASR results are presented. Since the utilised ASR system, effects of the parameters and the trends in the results are entirely similar to the material presented in Section 5.6, for a detailed description of the setup and discussion about the results please refer to Section 5.6.2. In Table B.1, Clean refers to training the system using only clean data, Multi1 denotes multi-style (a.k.a. multi-condition) training using additive noise and Multi2 indicates multi-style training with both additive and channel noise. The (g)VTS results are

Table B.1 *WER of the VTS and gVTS for Aurora-4 (HMMs are trained on clean data).*

Feature	α	A	B	C	D	Ave_1	Ave_2
MFCC-Clean	log	7.0	33.7	23.6	49.9	38.0	28.6
MFCC-Multi1	log	9.1	18.4	23.4	35.9	25.6	21.7
MFCC-Multi2	log	10.7	17.0	19.1	31.3	22.8	19.5
gMFCC	0.05	6.9	25.5	23.7	43.1	31.6	24.8
gMFCC	0.075	7.7	22.9	24.3	40.7	29.6	23.9
gMFCC	0.1	7.9	22.2	25.7	40.4	29.2	24.0
VTS1-FBE	log	6.5	21.5	22.0	38.6	27.8	22.2
gVTS1-0.05	0.05	6.4	20.2	20.8	37.4	26.6	21.2
gVTS1-0.075	0.075	6.7	19.7	21.2	37.0	26.3	21.2
gVTS1-0.1	0.1	7.0	18.7	20.8	36.3	25.6	20.7
VTS1-CEP	log	6.7	21.6	22.2	37.9	27.6	22.1
gVTS1	0.05	6.5	19.5	21.1	36.1	25.8	20.8
gVTS1	0.075	6.8	19.3	21.0	35.9	25.6	20.7
gVTS1	0.1	7.7	19.1	21.4	36.0	25.7	21.0
VTS2	log	6.5	22.2	14.9	35.5	26.3	19.8
gVTS2	0.05	6.5	20.3	14.4	34.2	24.9	18.9
gVTS2	0.075	7.4	20.3	15.4	34.5	25.2	19.4
gVTS2	0.1	7.6	20.3	15.4	34.5	25.1	19.4

$$Ave_1 = \frac{A+6B+C+6D}{14} \quad Ave_2 = \frac{A+B+C+D}{4}$$

computed using only the clean data. (g)VTS1 means the noise compensation is performed only for the additive noise and gVTS2 indicates that the compensation has been carried out for both additive and channel distortion. In this case, the channel was estimated based on a method detailed in Appendix C, using three iterations. Also, FBE means the compensation was carried out in the frequency domain and CEP indicates the compensation was done in the cepstral domain. Number of Gaussians in the GMM of the clean model, namely M, was set to 512 and additive noise was estimated using the first/last 30 frames.

Appendix C

Channel Noise Estimation Using (Generalised) Vector Taylor Series

C.1 Introduction

Broadly speaking, noise may be defined as any interfering signal or distortion which negatively affects the signal of interest. Mathematically speaking, in the Fourier domain it could take two forms, namely additive or multiplicative (a.k.a. convolutional). The former is called additive noise and the latter is called channel distortion. In either case, for countering the noise effect on the signal of interest, it is often helpful to estimate the noise. Estimating the additive noise is well studied due to its importance in speech enhancement. However, the channel noise estimation has received less attention. In this appendix, we proposed a novel channel estimation method which is based on (g)VTS and leads to substantial WER reduction in the presence of the channel noise.

C.2 Channel Noise Estimation

Environment model, as mentioned in Appendix B, shows how the clean signal is contaminated with noise and generally takes the following form

$$Y = XH + W \tag{C.1}$$

where X, W and Y are the power spectra (periodogram) of the clean signal, additive noise and noisy observation, respectively, and H is the square of the magnitude spectrum of the channel. As demonstrated in Appendix B, in a model-based noise compensation process, there is a need for an estimate of the noise. For mathematical convenience, it is assumed that

the noise follows a Gaussian distribution, and the goal of the noise estimation process is to estimate the corresponding mean vector and the covariance matrix.

In the simplest way, the additive noise can be estimated using the first and last 10 to 30 frames of the utterance. If a reliable VAD results become available, one can estimate the additive noise and update its parameters using the non-speech segments. An important difference between the channel and additive noise estimation is that ideally the channel distortion should be only estimated from the speech segments. The reason backs to the environment model in which the contribution of H in Y is zero in the non-speech frames where $X = 0$. Thus, if the speech/non-speech labels become available, in the non-speech frames only the additive noise exists whereas in the speech part the clean, additive and channel noise are present.

It is reasonable to assume that the frequency response of the channel is stationary and does not change over the utterance recording. As a matter of fact, the channel is related to the physical properties of the microphone as well as the environment between speaker and the microphone and could be assumed to be fixed unless the speaker deliberately changes his position, relative to the microphone.

As mentioned earlier, for mathematical convenience, the distribution of the noise is considered to be Gaussian. If the variance is set to zero, it means that there is no variability in the characteristic of the microphone and the channel could be expressed by only the mean vector. This implies the channel is no longer a random variable and is treated like a deterministic variable. On the other hand, non-zero covariance matrix, allows for embedding uncertainty in the estimated mean. This potentially could be useful because the estimation process returns an approximate of the variable of interest, not its exact value; hence, entering the uncertainty into the modelling process, to some extent, makes up for the probable errors occur during estimation and/or modelling. However, estimating the covariance matrix in practice is not straightforward. Therefore, here it is assumed that the covariance matrix of the channel is zero and the channel is characterised only by its mean.

C.2.1 Channel Noise Estimation in the Absence of the Additive Noise

As will be explained later in this appendix, the additive noise overshadows the channel noise estimation process. For the sake of argument, let us start off with the case in which there is no additive noise in the background. As such at the frame t of the utterance u one can write

$$\breve{Y}_t = \breve{H}_t \, \breve{X}_t \Rightarrow \breve{H}_t = \frac{\breve{Y}_t}{\breve{X}_t} \Rightarrow \mu^{\breve{H}} = \mathbb{E}\{\breve{H}\} = \mathbb{E}\{\frac{\breve{Y}_u}{\breve{X}_u}\} \tag{C.2}$$

where $\mathbb{E}$, $\breve{Y}$, $\breve{X}$, $\breve{H}$ and $\mu^{\breve{H}}$ denote the expected value operator, Y^{α}, X^{α}, H^{α} and the mean of the channel, respectively. The channel estimation process aims to estimate the $\mu^{\breve{H}}$. Although $\breve{Y}_t$ is the noisy observation and is available, $\breve{X}_t$ is not accessible and actually, the ultimate goal of the noise compensation process is to find it.

For making the channel estimation problem tractable a number of suboptimal assumptions are made here. These approximations of course accompany with some error, but make the problem tractable[1]. In order to compute $\mathbb{E}\{\frac{\breve{Y}_u}{\breve{X}_u}\}$, it is supposed that $\breve{Y}_u$ and $\frac{1}{\breve{X}_u}$ are uncorrelated. Note that it does not mean $\breve{Y}$ and $\breve{X}$ are uncorrelated. Of course assuming the noisy observation and the clean part to be uncorrelated is a crude assumption unless the SNR becomes very low.

To investigate the independence of the $\breve{Y}_u$ and $\frac{1}{\breve{X}_u}$, let us first revisit the Homographic function $f(x) = \frac{1}{x}$ which is depicted in Figure C.1. When x is small the derivative is quite large and the function is very sensitive to any change in the independent variable x. This indicates high level of correlation between $f(x)$ and $\frac{1}{x}$. However, when the x increases the function asymptotically tends to a constant (zero). When x becomes large enough, $\frac{df(x)}{dx}$ tends 0 which happens approximately for $x \geq 8$ based on Figure C.1). In this case, no matter how much the x is, $\frac{1}{x}$ remains constant and no longer covaries with x. What is of our interest is that x and $\frac{1}{x}$ do not covary which is the case when x is large enough. So, when the x is sufficiently large, it becomes uncorrelated with $\frac{1}{x}$. By the same token, $\breve{Y}$ and $\frac{1}{\breve{X}}$ become uncorrelated, too, because when $\breve{X}$ is large enough $\frac{1}{\breve{X}}$ behaves like $f(x) = c$ which is uncorrelated with any varying quantity. However, how we can get ensured that in practice $\breve{X}$ is adequately large, at least most of the times?

To answer this question, we need to look at the histogram of x. If the support of the histogram shows that most of the probability mass occurs in the region which the variable x could be assumed large enough, one can say x and $\frac{1}{x}$ are uncorrelated. Figure B.4 shows the histograms after applying *GenLog* for different values of α. It can be easily verified that based on the aforementioned range for x, when $0 \leq \alpha \leq 0.1$, the variable $\breve{X}$ is sufficiently large. This allows for assuming $\breve{X}$ and $\frac{1}{\breve{X}}$ and consequently $\breve{Y}$ and $\frac{1}{\breve{X}}$ are uncorrelated, because as far as $\frac{1}{\breve{X}}$ behaves like a constant, it is uncorrelated with any varying function. As such

$$\mu^{\breve{H}} = \mathbb{E}\{\frac{\breve{Y}_u}{\breve{X}_u}\} \approx \mathbb{E}\{\breve{Y}_u\}\,\mathbb{E}\{\frac{1}{\breve{X}_u}\} \tag{C.3}$$

Now, the problem gets divided into two subproblems, namely estimating $\mathbb{E}\{\breve{Y}_u\}$ and $\mathbb{E}\{\frac{1}{\breve{X}_u}\}$. Estimation of $\mathbb{E}\{\breve{Y}_u\}$ can be carried out using the sample mean of $\breve{Y}_t$ over the

[1] *Essentially, all models are wrong, but some are useful.* – George Box

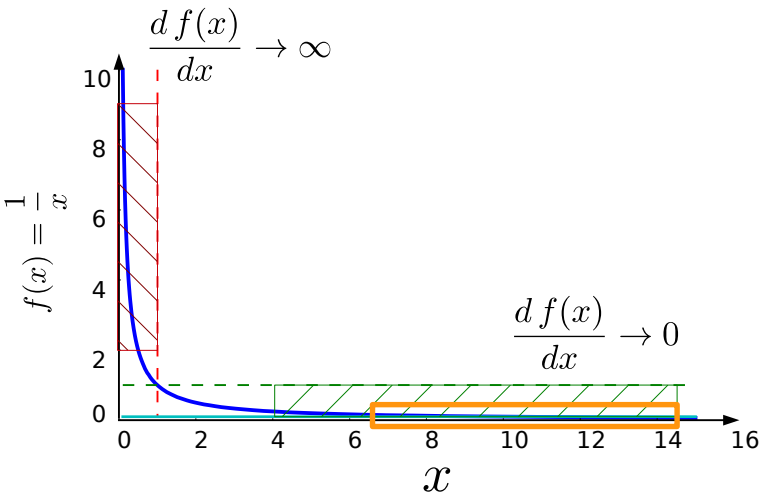

Fig. C.1 Homographic function, $f(x) = \frac{1}{x}$. When x becomes large enough, $f(x)$ asymptomatically tends to a constant (zero) and (approximately) does not covary with x.

utterance

$$\mathbb{E}\{\breve{Y}_u\} \approx \frac{1}{T}\sum_{t=1}^{T}\breve{Y}_t \tag{C.4}$$

where T indicates the number of frames of the utterance. Based on the *law of large numbers* [166], the larger the T, the better the estimate.

Now, $\mathbb{E}\{\frac{1}{\breve{X}_u}\}$ should be estimated. In the (g)VTS framework, the model of the clean data, namely the GMM of $\breve{X}$, is available which helps computing the $\mathbb{E}\{\breve{X}\}$

$$\mathbb{E}\{\breve{X}\} = \sum_{m=1}^{M} p_{\breve{x}}(m)\,\mu_m^{\breve{X}}\,. \tag{C.5}$$

where $p_{\breve{x}}(m)$ and $\mu_m^{\breve{X}}$ denote the weight and the mean vector of the m^{th} component of the mixture, respectively, and M indicates number of Gaussians. If the utterance is long enough with adequate phonetic diversity, the mean of its clean version would be close to the global mean of the clean speech model

$$\mathbb{E}\{\breve{X}_u\} \approx \mathbb{E}\{\breve{X}\}. \tag{C.6}$$

However, what is required for estimating the channel through (C.2) is $\mathbb{E}\{\frac{1}{\breve{X}}\}$, not $\mathbb{E}\{\breve{X}\}$. Given that the Homographic function $\frac{1}{x}$ is convex (curves up), Jensen's inequality underpins the relation between $\mathbb{E}\{\frac{1}{\breve{X}}\}$ and $\mathbb{E}\{\breve{X}_u\}$ in the following way

$$\mathbb{E}\{\frac{1}{\breve{X}_u}\} \geq \frac{1}{\mathbb{E}\{\breve{X}_u\}}. \tag{C.7}$$

Such inequality provides a lower bound for $\mathbb{E}\{\frac{1}{\breve{X}}\}$ whereas what is needed for the channel estimation is equality.

The question which arises at this point is that when does Jensen's inequality tend to equality? The answer is not difficult, note that when the function becomes concave (curves down) the direction of the inequality changes; hence, when the function of interest locates in between the concave and convex functions the inequality could be approximated with equality. The only function in the convex-concave border is the constant function. Backing to our Homographic function, although it is convex, based on the histograms of the $\breve{X}$ and assuming that $\breve{X}$ is large enough, $\frac{1}{\breve{X}}$ behaves like a constant, as shown in C.1. As such one can approximately replace the inequality in (C.7) with equality

$$\mathbb{E}\{\frac{1}{\breve{X}_u}\} \approx \frac{1}{\mathbb{E}\{\breve{X}_u\}}. \tag{C.8}$$

Having said that, this approximation runs the risk of underestimation of channel, $\breve{H}$, because the true value is approximated with a smaller quantity. The extent of the error will be illustrated soon. Finally, using (C.2)-(C.8), an estimate of the frequency response of the channel can be formulated as follows

$$\mu^{\breve{H}} \approx \frac{\frac{1}{T}\sum_{t=1}^{T}\breve{Y}_t}{\sum_{m=1}^{M} P_{\breve{x}}(m)\ \mu_m^{\breve{X}}}. \tag{C.9}$$

Figure C.2 illustrates the estimated frequency response versus the target values. As seen, the proposed approach shows a great potential in blindly capturing the trend and local shape of the channel. However, in some cases e.g. Figure C.2(b) and Figure C.2c), despite capturing the overall trend, local estimates are inexact. The possible causes of error are discussed in Section C.2.4.

C.2.2 Channel Estimation in the Presence of Additive Noise using gVTS

In practice, we rarely encounter with the case in which there is no additive noise in the background. In order to raise the practicality of the proposed approach, the algorithm should be able to estimate the channel noise in the presence of additive noise, too. Rewriting (C.2) by taking the additive noise into account yields

$$\mathbb{E}\{\frac{\breve{Y}_u}{\breve{X}_u}\} = \mathbb{E}\{\big(H + \frac{W_u}{X_u}\big)^{\alpha}\} \approx \mu^{\breve{H}} + \mathbb{E}\{\frac{\breve{W}_u}{\breve{X}_u}\} \tag{C.10}$$

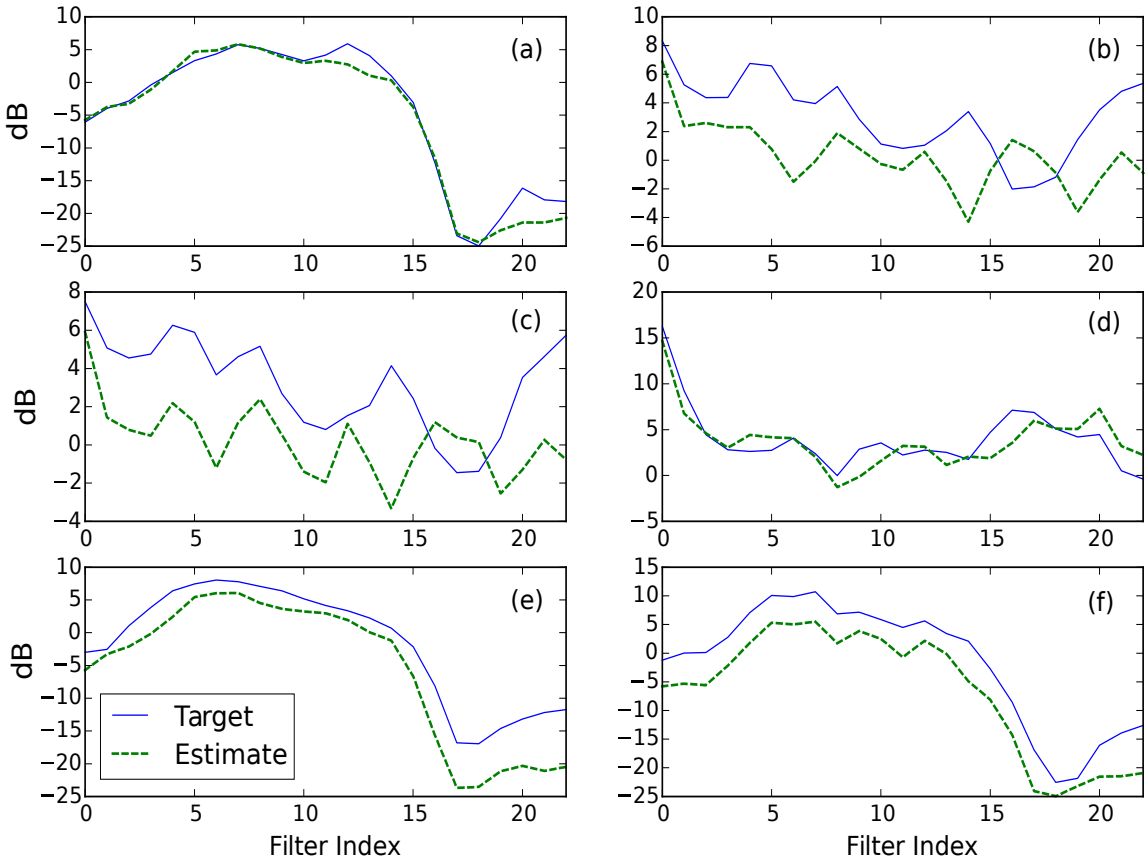

Fig. C.2 Blind channel estimation based on the proposed method for six waves from the test set C of the Aurora-4 [11]. Target channel response was computed through comparing the noisy wave ($Y = XH$) from test set C with its clean counterpart (X) from test set A. Underestimation is due to Jensen's inequality and (C.8).

where H and $\frac{W}{X}$ are assumed to be uncorrelated. As seen, the presence of additive noise introduces an error term, $\mathbb{E}\{\frac{\check{W}_u}{\check{X}_u}\}$, which is inversely proportional to a priori SNR and leads to overestimation (since it is always positive). This error has an opposite effect to (C.8) which gives rise to underestimation. This is just from theoretical standpoint and one should not rely on the errors to cancel each other.

To deal with this extra term, the additive noise should be attenuated/suppressed. Speech enhancement techniques are mainly designed for dealing with the additive noise and may seem useful in this regard. However, they bring about the problem of distorting the speech in sense that the enhanced signal does not necessarily remain consistent with the statistical model of the clean speech ($GMM_{\check{X}}$). Given that the VTS framework and also the proposed channel estimation approach rely on the GMM of the clean speech, the compatibility with this

model should not be undermined. Therefore, in ideal scenario the additive noise should be filtered out but simultaneously the compatibility with the clean model should be maintained.

To this end, an iterative algorithm illustrated in Figure C.3 is suggested. First, the channel estimate is initialised. Since at this stage the channel noise is not available, the gVTS aims at suppressing only the additive noise ($gVTS_1$). Let $Z = \breve{X}\breve{H}$, which encapsulate $\breve{X}$ and $\breve{H}$ into one variable. Now $gVTS_1$ aims at alleviating the additive noise and finding $\hat{Z}$. In this regard, GMM_Z should be computed through adapting the $GMM_{\breve{X}}$ using $\breve{H}$ as follows

$$Z \sim \sum_{m=1}^{M} p_{\breve{X}}(m)\, \mathcal{N}(z; \breve{H}_{diag}\, \mu_m^{\breve{X}}, \breve{H}_{diag}\, \Sigma_m^{\breve{X}}\, \breve{H}_{diag}^T). \tag{C.11}$$

where $\breve{H}_{diag}$ is $diag\{\mu^{\breve{H}}\}$. The adaptation could include both mean and covariance or only the mean. It was observed that both approaches return almost the same results.

This process attenuates the additive noise and also pushes the utterance closer to the background clean model in a statistical sense (i.e. likelihood under the clean model increases). It allows for a better channel noise estimation even in the clean condition because (C.6) will hold more closely. The gVTS1 output, namely $\hat{Z}$, ideally should be additive-noise-free to form an approximation for $\breve{X}\,\breve{H}$. As such an estimate for the channel frequency response can be formed using (C.9) for the the next iteration. Number of iterations should be set empirically. Experimental results showed that 2-4 iterations suffice. Initialisation can be carried out either by setting the initial channel (H_0) to unity or $\frac{\breve{Y}}{\breve{X}}$. Figure C.4 shows that in both cases the algorithm converges. In the experiments we used $\frac{\breve{Y}}{\breve{X}}$ for initialisation since it resulted in slightly better results.

C.2.3 Extension of the Proposed Approach to the Conventional VTS

The equations derived for estimating the channel noise were tailored for the gVTS in which the power spectrum is compressed by power transformation, $\breve{Z} = Z^{\alpha}$. However, they can be easily extended to the VTS in which the power spectrum is compressed by log function. In this regard, (C.9) should be rewritten in the following way

$$\mu^{\tilde{H}} \approx \frac{1}{T}\sum_{t=1}^{T} \tilde{Y}_t - \sum_{m=1}^{M} P_{\tilde{x}}(m)\, \mu_m^{\tilde{x}}. \tag{C.12}$$

where $\tilde{Z} = logZ$ for $Z \in \{Y, X, H\}$. The adaptation step in (C.11) takes the following form

$$Z \sim \sum_{m=1}^{M} P_{\tilde{X}}(m)\, \mathcal{N}(z; \tilde{H}_{diag} + \mu_m^{\tilde{X}},\, \Sigma_m^{\breve{X}}). \tag{C.13}$$

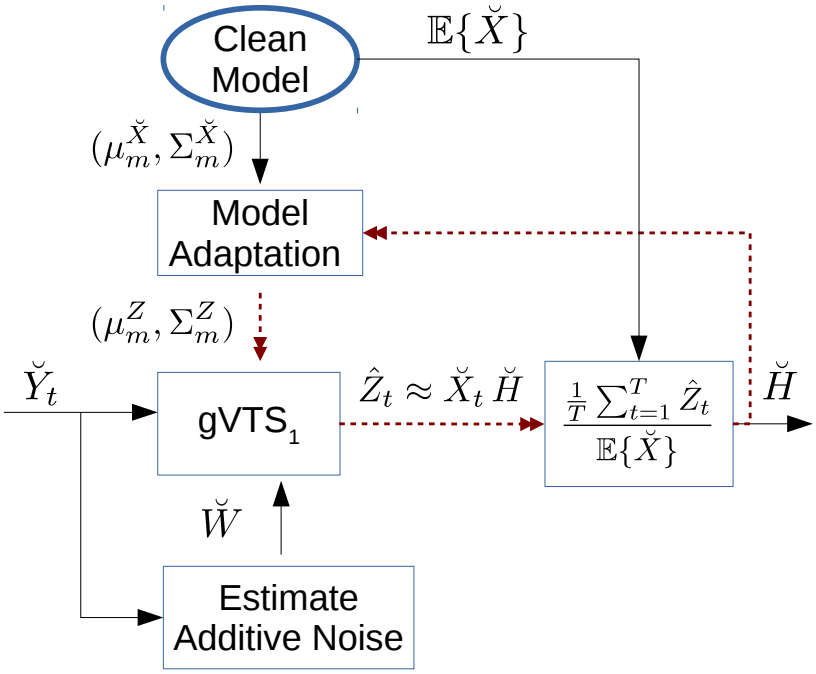

Fig. C.3 Workflow of the proposed channel estimation method using *generalised* VTS. gVTS$_1$ indicates the noise compensation is carried out only for additive noise.

where $\tilde{H}_{diag}$ is $diag\{\tilde{H}\}$. Note that the adaptation here, contrary to (C.11), only operates on the mean. Figure C.5 shows the channel noise estimation process for the case of compressing the power spectrum through log along with VTS framework.

C.2.4 Difficulties with the Proposed Approach

As mentioned earlier, utterance length and phonetic diversity need to be sufficient for (C.4) and (C.6) to hold and (C.8) and (C.10) cause under and over estimation, respectively. In addition, (C.5) and (C.6) implicitly assumes that the channel used in recording the training data has a flat frequency response and this is not necessarily the case. Moreover, the frame length (typically 25 ms) may not be longer than the effective length of the impulse response of the channel in the time domain. As such the frequency resolution will be insufficient for resolving the channel frequency response. In such case even target (ground truth) values in Figure C.2 could be inaccurate. Sub-sampling the spectrum through filter bank could also introduce some error, especially in the high frequencies where the band width of the filters becomes wider and the power spectrum of many bins get added together.

On the other hand, averaging in (C.4) is performed across all frames, both speech and non-speech ones. At the non-speech segments where $X = 0$, the channel contribution is zero, based on the environment model. As a result, such frames not only do not provide any useful

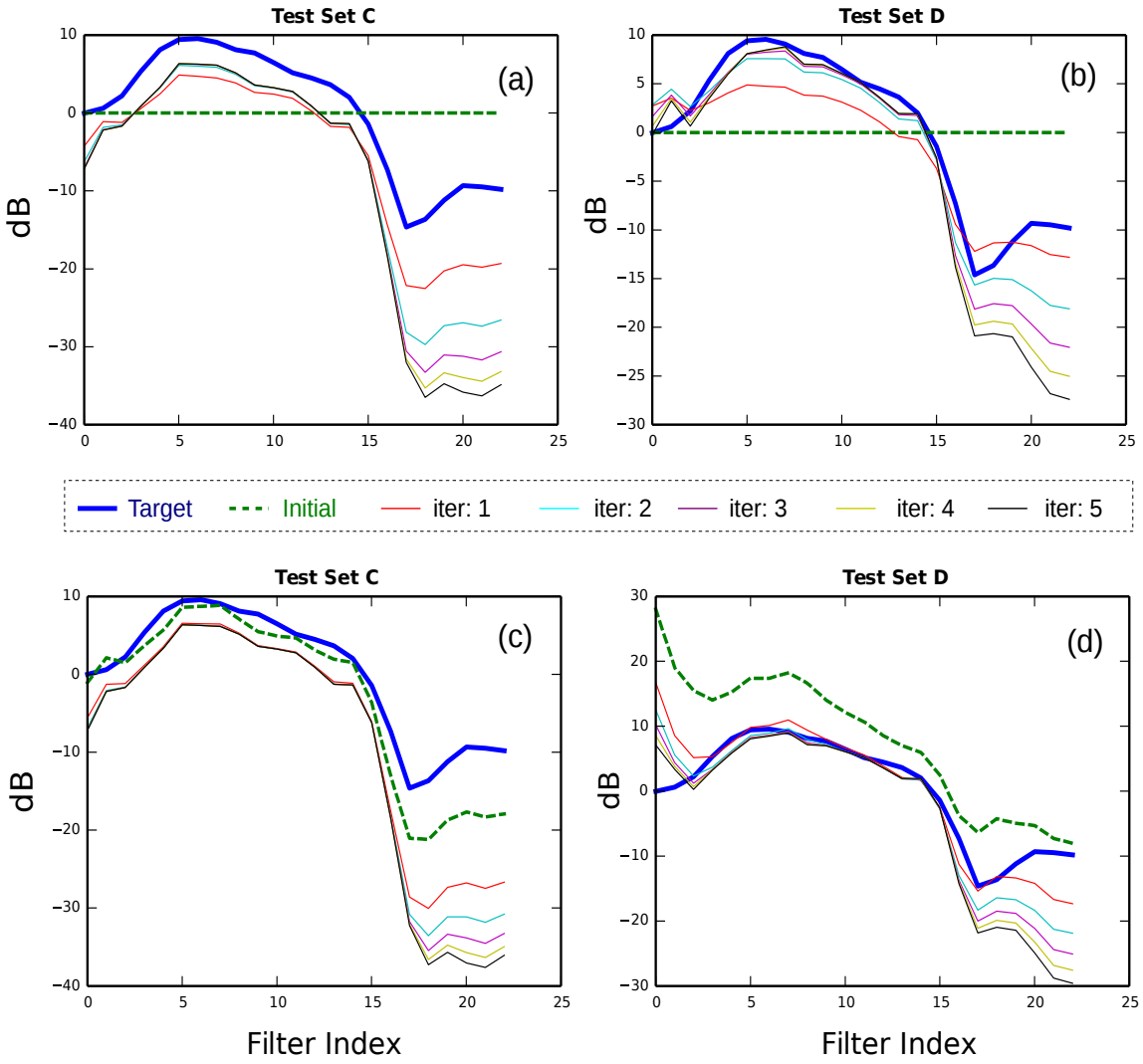

Fig. C.4 Effect of initialisation, number of iterations (iter) and presence of the additive noise on the performance of the proposed channel estimation method. (a) Unit channel initialisation, additive-noise-free (a signal from Test Set C, Aurora-4), (b) Unit channel initialisation, additive noise is present (a signal from Test Set D, Aurora-4), (c) initialisation: $\frac{Y}{X}$, additive-noise-free (a signal from Test Set C, Aurora-4) (d) initialisation: $\frac{Y}{X}$, additive noise is present (a signal from Test Set D, Aurora-4). For both unit and $\frac{Y}{X}$ initialisations the algorithm converges after 2-4 iterations.

information as far as the channel estimation is concerned but also allow the additive noise to further overshadow the channel estimation process because they only contain additive noise. So, ideally, for the channel estimation only the speech segments should be considered. A simple solution could be using a voice activity detection (VAD) block. However, building a reliable and robust VAD in noisy conditions is not straightforward. So, in the current

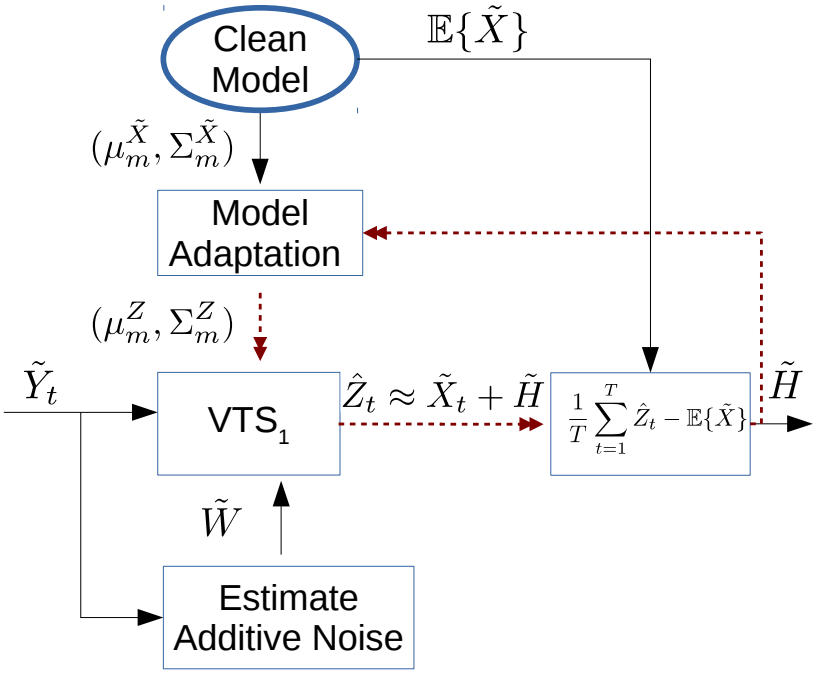

Fig. C.5 Workflow of the proposed channel estimation method using *conventional* VTS. VTS_1 indicates the noise compensation is carried out only for additive noise.

implementation, we have computed the averaged over all the frames of the utterance. Finally, the output of the $(g)VTS_1$ in Figure C.3 and Figure C.5 is not additive-noise-free.

Experimental results in Section 5.6 on the Aurora-4 ASR task show that the proposed method leads to remarkable robustness improvement in dealing with the channel noise.

Appendix D

Deep Neural Networks for ASR

D.1 AI Sprint and Deep Neural Networks

We live in the golden age of automatic speech recognition (ASR) systems thanks to availability of huge amount of data for training, powerful computers and a modelling technique with massive learning capability that can make the most out of the these: deep neural networks (DNN). Artificial Neural Networks (ANN) are not new methods and were around from the mid of the 20^{th} century. However, up until one decade ago they were not considered among the top 10 algorithms for processing the data [185]. The breakthrough happened in 2006 [186] and for the first time it was shown that the neural network with deep structure can learn well. Before this, one could hardly make the neural network with more than 2 hidden layers work. The main advantage of the deep structure compared with other techniques, which are described as shallow, is that it has a higher potential in extracting abstract representations of the data which lead to more effective learning. Such models, however, are data hungry and require much data and computing power to realise such potential.

The main novelty of [186] was so-called greedy layer-wise pre-training in which instead of random initialisation of the network's weights before training through back propagation, the weights were learned using unsupervised techniques like restricted Boltzmann machine (RBM) (and in later work auto-encoders [187]) in a layer-by-layer style. At the time it was thought that the pre-training is the major missing piece of the puzzle for making the NNs work. However, by time it turned out that techniques belong to 80s (vanilla feed-forward network trained by back-propagation) in light of using enough data can still return very good results.

Having said that, technical improvements also have had some role to play. For example, using rectified linear unit (ReLU) [188] as activation function speeds up the training process and more importantly enhances the learning capabilities of the network. Drop-out technique

[189] was also an important technical improvement which made the NNs generalise better and less vulnerable to over-fitting the training data (at the cost of higher computational load). The other useful activation function was *Maxout* which well suits the training with the drop-out [190].

Other NN architectures which contributed to the success of the deep learning are recurrent neural networks (RNN) and in particular long-short term memories (LSTM) [191], convolutional neural networks (CNN) [192] and generative adversarial networks (GAN) [193]. Sequence-to-sequence models such as connectionist temporal classification (CTC) [194] also turned out to be a very powerful structure with applications in deep end-to-end approach.

D.2 Deep Neural Networks for ASR

Applying the neural networks for speech recognition is not new. The first applications could be traced back to the late 1980s using architectures like time delay neural networks (TDNN) which was employed in phoneme recognition (with two hidden layers) [195] and isolated word recognition (with two/three hidden layers) [196]. The intuition was that using the NN allows for learning non-linear decision borders and using the time-delay arrangement enables the network to discover effective acoustic-phonetic features along with optimal temporal relationships between them. However, the main short-coming of NNs such as TDNN, was that they could not deal with the variable length patterns and long time-sequences.

Bourlard and Morgan [197] employed the neural networks instead of GMM in the so-called hybrid HMM/ANN systems for modelling the emission probabilities of the HMM states. In such combination, the temporal variability was handled with HMM. The advantages of using ANNs was that since they are non-parametric universal function approximator and do not make any particular assumption, potentially could learn more complicated functions/manifolds/surfaces/decision borders. Also compared with GMMs which are generative models, ANNs are discriminative and potentially better fit the classification tasks. Efficient computation of the outputs at the recognition time along with hardware implementability were two other advantages [197]. However, effective training was a challenge and that is why in the 90s and early 2000s, HMM/GMM remained the dominant approach in ASR.

After the breakthrough in 2006 [186], Mohamed et al [198] used the DNN [1] for phoneme recognition in the TIMIT database and managed to get 23.0% phoneme error rate (PER). In [199] Grave et al achieved 17.7% PER using deep LSTM network trained with CTC.

[1] They called it Deep Belief Network (DBN) at the time. This term did not become popular. One reason is that the DBN is also an acronym for Dynamic Bayesian Networks.

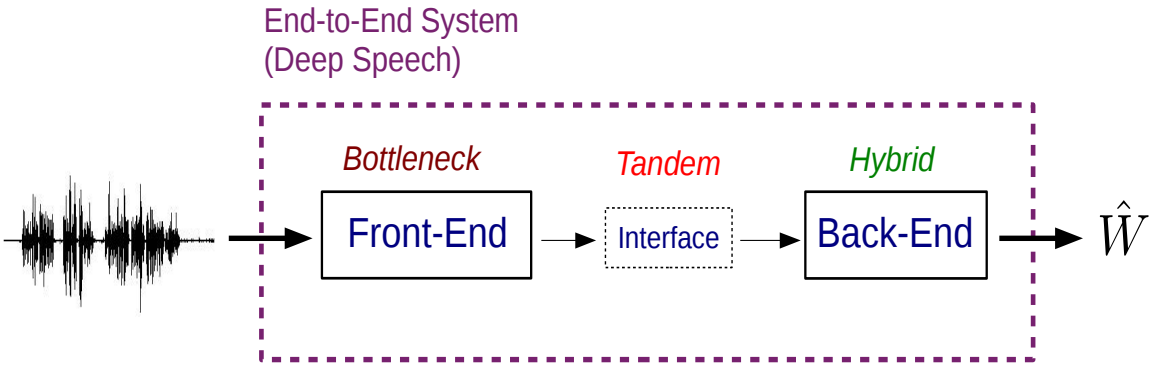

Fig. D.1 Applications of DNNs in ASR. Bottleneck: DNN in the front-end, Tandem: DNN as an interface between the conventional front-end and the back-end, Hybrid: DNN in the back-end (DNN-HMM for acoustic modelling), Deep Speech: integrating all the process (front-end and back-end) in a big DNN.

Nowadays DNNs are widely used in acoustic modelling (feedforward [200], CNN [201] and LSTM [202]) as well as the language modelling [203] [2].

The neural networks can be also employed in the front-end. An important application of the NNs for feature extraction from speech was the Tandem [205] approach in which the NN aims at extracting the posterior probabilities of the phonemes. Such vector of probabilities (which could be post-processed by log non-linearity and also decorrelated using DCT) is sent as feature vector to the standard HMM/GMM recogniser. Note that since the number of phonemes is around 40 the length of tandem-based feature vector is similar to the conventional 39 (13 x 3) feature vectors. Tandem approach belongs to the era before DNN spring and the utilised neural network was a conventional multi layer perceptron (MLP) with one hidden layer. Its counterpart in the DNN era is called *Bottleneck* (BN) approach [172, 206, 207]. In case of the BN features, there are more layers (say six) and number of the nodes of the output layer equals the number of state-clustered triphones. The BN feature is the output of the BN layer after using the linear activation function while in case of the Tandem, the features are the posteriori probabilities of the output layer. BN layer could be just before the output layer or at any other places. Its dimension is usually in the range of the conventional feature vectors.

Last but not least, DNNs can be employed in the end-to-end regime (also known as deep speech [208]) in which the input is the waveform and the output is the corresponding transcription [209]. Figure D.1 shows the possible applications of the DNNs in the automatic speech recognition.

[2] One of the first significant applications of the neural networks in language modelling was in 2003 in [204].

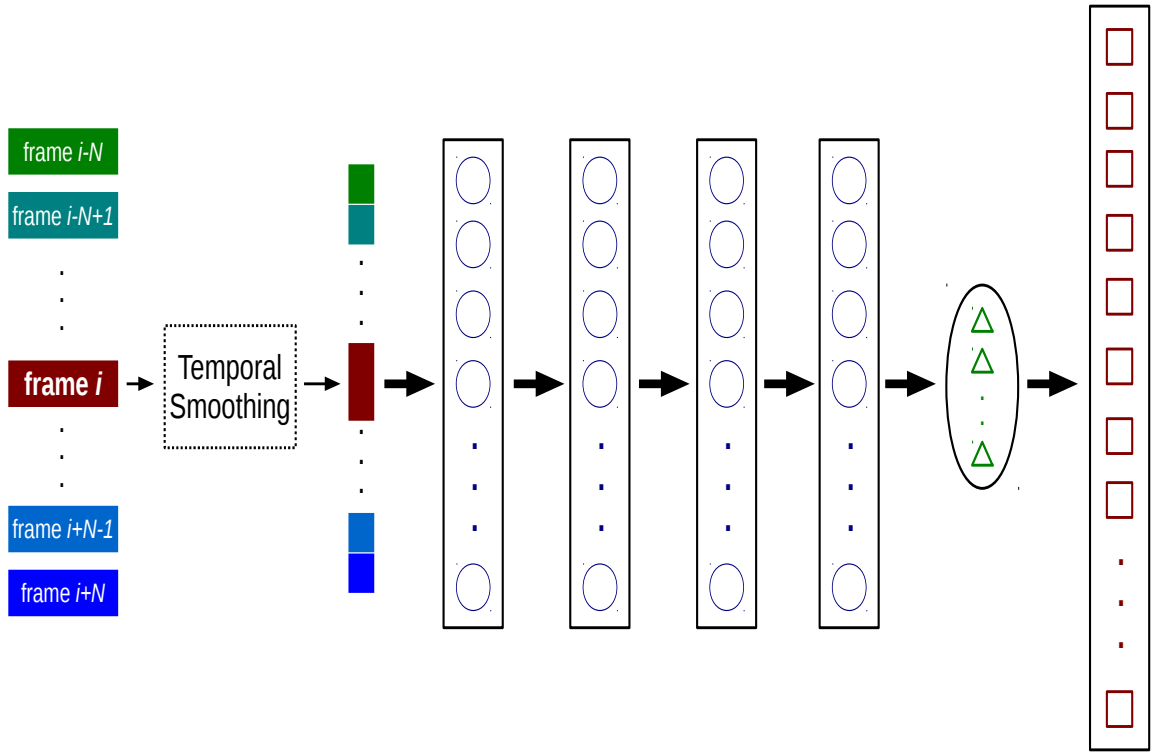

Fig. D.2 The Bottleneck DNN-based system utilised in this thesis. Number of nodes (#nodes) in the output layer (red squares): $\approx$ 2000 (is equal to the number of the state-clustered triphones), #nodes in the bottleneck layer (green triangles): 26, #nodes in other hidden layers (blue circles): 1300. N was set to 15.

Thanks to DNNs learning capabilities, availability of big data and the computing resources the performance kept improving over the last years and now machines are getting closer to human parity level in the conversational speech recognition task [210, 211].

D.3 The Employed DNN Setup

In this thesis, for the DNN-based experiments, a feed-forward neural network consisting of four hidden layers with 1300 nodes, followed by a bottleneck layer containing 26 nodes placed just before the output layer was used. The network was trained using TNet [179] and standard HMM/GMM models were trained on the BN features. Figure D.2 shows the topology of the utilised network.

As seen the network consists of 5 hidden layers. In the input layer, first a context matrix is built by stacking the previous N frames, current frame and the next N frames. Assuming that the feature vector length for each frame is 39, the dimension of such matrix is $(2N+1)$ x 39. Then a DCT of length n is taken on a each column. The advantage of such approach

over using a context of $2n+1$ is that a longer contextual information is captured with less number of samples [212]. In our system N equals 15 and n is 5. Each of the first four hidden layers consists of 1300 nodes, succeeded by a bottleneck layer with 26 neurons placed just before the output layer. Number of the nodes of the output layer equals the number of state-clustered triphones which in the employed setup are around 2000. Training was done layer-by-layer. The training data was split into training and dev: 85% for training and 15% for cross validation (CV). Learning rate was initially set to 0.003 and maximum number of iterations were set to 20 for each layer. When the frame accuracy improvement in comparison with the previous iteration become less than 0.5%, the learning rate gets halved. If the frame accuracy improvement is lower than 0.1%, the training is stopped before completing the 20 iterations (early stop) to avoid over-fitting.

Appendix E

Description of the Utilised Databases

E.1 Aurora-2

The Aurora2 [10] corpus is a small vocabulary, speaker independent connected-digit database. The speech data are derived from the clean TI-Digits database [213] that contains recordings of male and female US-American adults and is comprised of 11 words (oh, zero, one, two, three, four, five, six, seven, eight and nine). The 20 kHz signals, originally collected at Texas Instruments, Inc., were downsampled to 8 kHz.

Three different test sets (A, B and C) are constructed by artificially adding noise to the clean samples at SNRs ranging from 20 to -5 dB in steps of 5 dB using FaNT[1] [214] software. The set A and B consist of 4004 utterances from the TI-Digits test data, pre-filtered according to the G.712 [215] frequency characteristic. They are divided into 4 subsets of 1001 utterances each, to which four different noises are added at seven SNR-levels (-5, 0, 5, 10, 15, 20, clean). For set A these are subway, babble, car and exhibition hall noise; set B includes restaurant, street, airport and train-station noise. Test set C is designed to simulate channel distortions or convolutional noise. In this case, a Modified Intermediate Reference System (MIRS) filter [215] is applied to the original data. The MIRS filter models the behaviour of a telecommunication terminal that meets the technical specifications of the GSM 03.50. In addition, test set C includes two additive noises: subway and street. In total, 32883 words ($\approx$ 290 minutes of speech data) are to be recognized at each SNR-level.

The Aurora-2 database has two training sets: a clean set and a multi-conditions set. Both sets are composed of 8440 utterances selected from the TI-Digits training data, filtered with the G.712 characteristic and containing approximately 244 minutes of data. The four noise

[1] Filtering and Noise adding Tool

types of test set A are used in the multi-style set, at SNRs ranging from 5 to 20 dB in steps of 5 dB.

E.2 Aurora-4

Aurora-4 database [11], is a medium to large vocabulary continuous speech recognition task which is derived from the SI-84 WSJ0[2] 5k-word read-speech (dictation) task. The SI-84 set contains a mixture of utterances with and without verbalised punctuation[3]. It is a closed-loop vocabulary task, so there is no out of vocabulary (OOV) words in the evaluation set.

The database is available in two sampling rates: 8 kHz and 16 kHz. The former data is filtered by G.712 and the later is filtered by P.341 to simulate the telephone speech quality. The training set consists of 7138 utterances with average length of 7.6 seconds, 83 speakers and 14 hours of speech data. Average number of words per utterance is 17.8 and average speaking rate is 2.4 words per second. All recordings are made with the head-mounted close-talking microphone (Seinheiser HMD-414) and no noise was added in the clean-training mode. Aurora-4 has two extra training sets for multi-style training, namely *noisy* and *multi* consisting of the same utterances in the clean case, distorted by noise. Training data in the former is contaminated with only additive noise and in the latter by both additive noise and channel distortion. In both cases, SNR of the training samples ranged between 10 to 20 dB with average of 15 dB. In Chapter 5, the *noisy* and *multi* are referred to as $Multi1\ (M1)$ and $Multi2\ (M2)$, respectively.

The test set is taken from Nov'92 WSJ0 dataset with average length of 7.3 seconds, 8 speakers, 330 utterances and 40 minutes speech. FaNT software was used for artificially adding noise or telephone characteristics to the speech signals. Six noise types has been used: airport, babble, car, restaurant, street, train-station. The test or evaluation set of Aurora-4 consists of 14 test sets, grouped into four subsets: clean, (additive) noisy, clean with channel distortion and noisy with channel distortion, referred to as A, B, C and D, respectively. SNR of noisy test sets is ranged between 5 to 15 dB with average of 10 dB. Data of the test sets C and D were recorded using a desktop microphone of a group of 18.

The database comes with a bigram (2-gram) language model provided by the MIT Lincoln Lab which its perplexity is 147, reportedly.

[2]WSJ name is taken from the Wall Street Journal newspaper in which the sentences from this newspaper are read. Different versions of this database have been released over time. WSJ0 is the one which released in 1992.

[3]Example with verbalised punctuation: John COMMA who came home early COMMA decided to read the newspaper PERIOD

E.3 NOIZEUS

NOIZEUS [22] database is composed of thirty IEEE gender and phonetically balanced utterances including all phonemes in the American English language. Speech signals are produced by three male and three female speakers. The sounds were recorded in a sound-proof booth. The speeches were originally sampled at 25 kHz and downsampled to 8 kHz with a precision of 16 bits per sample. The clean signals are corrupted by eight different real-world noise types at four SNR levels, namely 0, 5, 10 and 15 dB. The noise was taken from the Aurora database and includes suburban train, babble, car, exhibition hall, restaurant, street, airport and train-station.

To simulate the receiving frequency characteristics of the telephone handsets, the speech and noise signals were filtered by the modified Intermediate Reference System (IRS) [215] filters. It is a bandpass filter with with cut-off frequencies at 300 Hz and 3400 Hz. Noise was added artificially similar to the FaNT [214] framework.

Appendix F

Feature Extraction Techniques Review

F.1 Mel Filter Bank

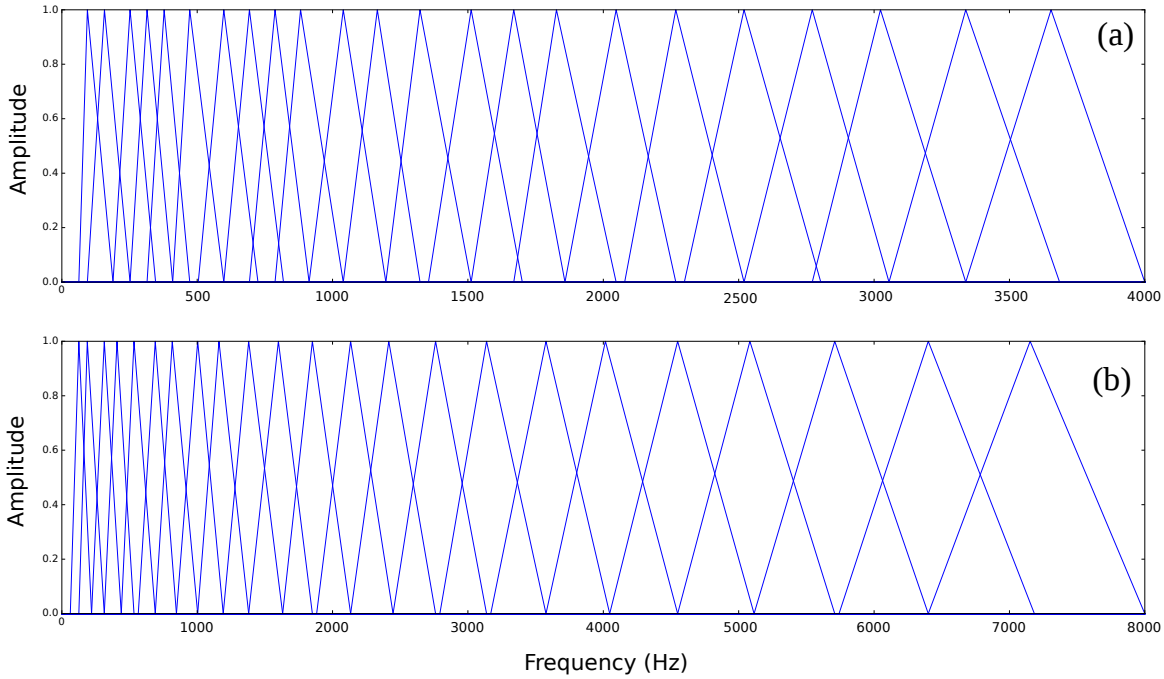

Fig. F.1 The employed filter bank in the feature extraction process. Number of filters: 23, scale: Mel, $f_{min} = 64$ Hz, $f_{max} = \frac{f_s}{2}$ where f_s is the sampling rate in Hz. Sampling rate: (a) 8 kHz, (b) 16 kHz.

F.2 (generalised) MFCC

1. Frame blocking (frame length: 25 ms, frame shift: 10 ms) and windowing (Hamming)
2. Compute the Fourier transform, $X(\omega)$
3. Compute the periodogram estimate of the power spectrum, $|X(\omega)|^2$
4. Apply Mel-scaled triangular filter bank to the power spectrum to get filter bank energies (FBEs)
5. Apply the (generalised) log[1] non-linearity
6. Apply the discrete cosine transform (DCT)

For more details please refer to [1] and [171].

F.3 PLP

1. Frame blocking (frame length: 25 ms, frame shift: 10 ms) and windowing (Hamming)
2. Compute the Fourier transform, $X(\omega)$
3. Compute the periodogram estimate of the power spectrum, $|X(\omega)|^2$
4. Apply Bark-scale PLP filter bank to the power spectrum to get the FBEs
5. Apply Equal Loudness Curve (ELC) to the FBEs
6. Apply cubic root ($x^{0.333}$) non-linearity (for intensity to loudness conversion)
7. Take inverse Fourier Transform and pick up the first 13 autocorrelation coefficients
8. Compute perceptual linear coefficients using the Yule-walker equations (through Levinson-Durbin method)
9. Convert the perceptual linear prediction coefficients to cepstral coefficients (for decorrelation)

For more details please refer to [216].

[1] For all the features, before applying the log as non-linearity, the FBEs were floored by 1, namely FBE $\leftarrow$ maximum(FBE,1).

F.4 (generalised) Product Spectrum (PS)

1. Frame blocking (frame length: 25 ms, frame shift: 10 ms) and windowing (Hamming)
2. Compute the Fourier transform, $X(\omega)$
3. Compute the product spectrum, $Q_X(\omega)$
4. Apply Mel-scale triangular filter bank to the product spectrum to get FBEs
5. Apply the (generalised) log non-linearity
6. Apply the discrete cosine transform (DCT)

For more details please refer to [42].

F.5 Modified Group Delay (MODGD)

1. Frame blocking (frame length: 25 ms, frame shift: 10 ms) and windowing (Hamming)
2. Compute the Fourier transform of $x[n]$, $X(\omega)$
3. Compute the Fourier transform of $nx[n]$, $Y(\omega)$
4. Compute the cepstrally smoothed magnitude spectrum of $x[n]$, $S(\omega)$
5. Compute the modified group delay function ($\alpha = 0.3$, $\gamma = 0.9$)
6. Apply the discrete cosine transform (DCT)

For more details please refer to [40] and [41].

F.6 Chirp Group Delay (CGD)

1. Frame blocking (frame length: 25 ms, frame shift: 10 ms) and windowing (Hamming)
2. Compute the Fourier transform, $X(\omega)$
3. Compute the zero-phase signal (Inverse Fourier transform of $|X(\omega)|$)
4. Compute the Chirp Fourier transform (ρ: the radius of the circle on which the Z-transform is evaluated was set to 1.12)

5. Extract the phase spectrum and compute the group delay
6. Apply Mel-scale triangular filter bank to the chirp group delay to get the FBEs
7. Take discrete cosine transform (DCT)

For more details please refer to [43].

F.7 ARGDF

1. Frame blocking (frame length: 25 ms, frame shift: 10 ms) and windowing (Hamming)
2. Perform linear prediction (LP) of order p (p should be set around $1.5\frac{f_s}{1000}$ where f_s is the sampling frequency in Hz)
3. Compute the Fourier transform of the extracted all-pole model
4. Extract the phase spectrum and compute the group delay
5. Apply Mel-scale triangular filter bank to the group delay
6. Apply the discrete cosine transform (DCT)

For more details please refer to [117].