ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Signal and Information
Processing Laboratory
Institut für Signal- und
Informationsverarbeitung
ISI

Summer Semester 2003 Prof. Dr. H.-A. Loeliger

Semester Thesis

Cepstral Dereverberation

Michael Gutmann

Supervisor: Markus Hofbauer
Co-Supervisor: Sascha Korl

Abstract

The presence of reflecting walls distorts speech signals that are transmitted in a room. This room-effect is perceived as reverberation and is often undesirable. Especially for hearing-impaired listeners, reverberation can cause speech intelligibility problems. Reverberation can be represented by convolution. Enhancing speech intelligibility by means of inverting the room-effect is thus related to the blind deconvolution problem.

This document analyses the performance of a particular algorithm that inverts the room effect by means of two measured speech signals. On the road to the conclusion, this document will give a comprehensive introduction to the cepstrum and its application to deconvolution. Emphasis is put on highlighting problems that emerge under realistic conditions.

Acknowledgements

I would like to express my appreciation and gratitude to my supervisor Markus Hofbauer who helped in a variety of ways in the development of this semester thesis. I am thankful that I was given the possibility to work on such a interesting field of research like blind deconvolution. I am further much obliged to my supervisor for his scrutiny and for permitting though discussions.
I further acknowledge the help of Stefan M. Moser in LaTeXand the presence of my fellow students in the research institute.

Zurich, 11th of July 2003

Michael Gutmann

Contents

List of Symbols

x	An arbitrary signal	(2.2)
s	Speech signal	4.2
y	Convoluted signal	(2.2)
h	Channel	(2.2)
h^{-1}	Inverse of the channel h	(4.43)
w	Additive White Gaussian Noise (AWGN)	4.2
$\tilde{x}$	Cepstral modified x	(4.39)
$\hat{s}$	Estimate of s	4.2.1
$x^{\min}$	Purely min. phase	(2.39)
$x^{\max}$	Purely max. phase	(2.41)
$x_{\min}^{\mathrm{NDFT}}$	Min. phase signal from N-DFT.	4.3.1
$x_{\max}^{\mathrm{NDFT}}$	Max. phase signal from N-DFT.	4.3.1
n_0	Time shift of x	(2.37)
N_p	Spacing of an impulse train	(2.43)
$h_{\min}$	Artificial channel with min. phase signals	6.1.2
$h_{\max}$	Artificial channel with max. phase signals	6.1.2
X	z-transform of x	(2.8)
$\hat{X}$	log of X	(2.8)
Θ_x	Fourier phase of x	(3.2.1)

continued on next page

continued from previous page

$\mathfrak{Z}$	Operator z-transf.	(2.8)
$\mathfrak{D}$	Operator cepstral transf.	(2.8)
$\mathfrak{D}^{-1}$	Operator inv. cepstral transf.	(2.13)
$\mathfrak{DTFT}$	Operator DTFT	4.1.1
$\mathfrak{DFT}$	Operator DFT	4.1.1
$\mathfrak{D}_{\mathcal{N}}^{\mathfrak{DFT}}$	Operator DFT-cepstrum	4.1.1
$\mathfrak{D}_{\mathcal{N}}^{\mathfrak{DFT}}.^{-1}$	Operator inv. DFT-cepstrum	4.1.1
c_x	Cepstrum of x	(2.16)
c_{P_x}	Power cepstrum of x	(2.23)
$c_{\angle_x}$	Angle cepstrum of x	(2.25)
A	Part of c_x def. by min. phase	(2.34)
B	Part of c_x def. by max. phase	(2.35)
$c_x^{\min}$	Purely min. phase cepstrum	(2.39)
$c_x^{\max}$	Purely max. phase cepstrum	(2.39)
c_x^{NDFT}	DFT-cepstrum of x	4.2
$c_{\angle_{\hat{x}}}^{\mathrm{NDFT}}$	Angle cepstrum of x calculated by means of a DFT	(5.7)
E_p	Progress of an iterative cycle	(5.6)
$\triangle c$	Small error in the cepstrum	(4.26)
$e[n; n_i]$	Error function	(4.40)
SNR	Signal to Noise Ratio	(4.46)
e_{A}	Error (E)	4.3.3
$e_{1\mathrm{A}}$	Absolute Error (AE)	(4.8)
$e_{2\mathrm{A}}$	Absolute Squared Error (ASE)	(5.5)
$e_{2\mathrm{N}}$	Normalised Squared Error (NSE)	(4.45)

Chapter 1

Introduction

Dereverberation refers to methods which try to enhance speech that was distorted by the presence of reflecting walls. Methods that use the particular technique of cepstral transformation are said to do cepstral dereverberation. Here, we introduce their basic concepts and announce the structure of this very document.

1.1 Cepstral Dereverberation

In this section, it is explained what is meant by reverberation and what consequences reverberation has on speech intelligibility. It closes with remarks on the use of the cepstral transformation in dereverberation problems.

1.1.1 Reverberation model

When a conversation takes place inside a room, sounds travel not only the direct way from the speaker to the listener, but also reach him on a circuitous path. Speech signals are reflected off the surface of nearby objects to arrive attenuated and delayed at the listener. The listener hears thus both the original sound and the reflected sound. This "room effect" can be viewed as a mixing process of the original signal with itself at different time delays. The received speech signal y when the speaker utters s can thus be expressed in formula as follows

$$y = \sum_{k=0}^{K} h_k s(t-k).$$

Variable K is related to the reverberation time, i.e. how fast the the reflected signals are attenuated in amplitude with regards to the original signal. Hence, the "room effect" can be viewed as a convolution of the speech signal s with a room impulse response h. The above formula is sure to express improperly what happens in nature when speech travels form speaker to listener. To account for any shortcoming a noise signal w is added to y to complete the reverberation model.

$$\tilde{y} \quad = \quad \sum_{k=0}^{K} h_k s(t-k) + w.$$

1.1.2 Why dereverberation

Reverberation can cause speech intelligibility problems. As reverberation time increases, speech recognition ability decreases. Hearing-impaired listeners, however, are more affected by reverberation that are normal-hearing listeners. In contrast to persons with hale hearing abilities, hearing-impaired persons can only cope with moderate amounts of reverberation [7]. For use in hearing aids, methods that invert the effect of reverberation are therefore of importance.

Dereverberation or deconvolution stands for the mentioned inversion. When the mixing coefficients h_k are not known, which is the case in reality, the inversion process is called blind dereverberation or blind deconvolution. The task of inversion becomes then to estimate the source signal s from the observed signal y only without knowing the time delays and mixing coefficients.

1.1.3 Cepstral methods

Because of its mathematical complexity, it is not as a trivial issue to reverse a convolution like it is for example for an addition. Nevertheless, if the room impulse response was h minimum phase, the blind deconvolution problem could be solved by applying a whitening filter to the reverberated signal [9] [10]. However room impulse responses are generally mixed phase [11] which blocks the direct use of whitening filters.

The idea of cepstral dereverberation is first to choose an appropriate starting point before tackling the problem. A convolution does not have a friendly look in the time domain domain, however, when looked at it from

an other point of view, it appears as likeable as a summation. In the cepstral domain, indeed, a convolution appears as a summation. Furthermore the cepstral domain has the advantage that the involved mixed phase signals appear in it demixed. When the problem of inverting the room impulse response is handled in this particular domain, it is said that cepstral dereverberation is being done.

1.2 Document structure

The purpose of this document is to analyse an algorithm that effects cepstral blind deconvolution on the basis of two measurements, i.e. two-channel dereverberation. Note that the performance will be measured from a mathematical point of view not taking into consideration the human hearing system. Furthermore, in light of the differences in speech recognition between normal-hearing and hearing-impaired persons, in [14] [23], it has been recommended to focus on hearing-impaired listeners when evaluating an algorithm being considered for implementation in hearing aids.

The evaluation approach is based on viewing the algorithm as a scheme where a few self-contained tools are applied severals times in succession. Performance of the algorithm will be explained in terms of the performance of the underlying tools and their interaction. Hence, the document is structured into three parts. The first part gives background information on cepstral deconvolution. The second part discusses each of the tools needed in the two-channel blind deconvolution task. The third part compiles the gained insights for the evaluation of the blind deconvolution algorithm.

- Part I summarises the *papers* that have introduced the cepstral transformation. It further explains a particular publication that is needed in the algorithm that solves the two-channel blind deconvolution problem. Chapter 2 defines the cepstral transformation that maps a convolution into a summation. It further discusses the properties of this application that are of use in dereverberation. Chapter 3 gives the conditions that a signal has to fulfill so that the magnitude of its Fourier transform is function of the Fourier phase. As in cepstral dereverberation, we merely look differently at the deconvolution problem, we do not have more information on the channel h than in the time domain. The theory in Chapter 3 is used in combination with

the cepstral transformation to do two-channel blind dereverberation in the cepstral domain.

- Part II discusses the practical realisation of the *tools* given in theory in part I and later used by the two-channel deconvolution algorithm. Chapter 4 analyses to what extent the cepstral transformation calculated by numerical methods, i.e. the Discrete Fourier Transform, differs from the definition given in Part I. It further discusses the implications of that deviation on dereverberation concepts . Chapter 5 analyses the performance of several algorithms that reconstruct a signal from its Fourier phase only. These algorithms are used in the mentioned two-channel blind deconvolution algorithm to reveal the room impulse responses.

- Part III details the structure of the *two-channel blind deconvolution algorithm*. The results established in the previous chapters are used to comment on the dereverberation ability of the algorithm.

Part I

Cepstral Dereverberation – On Paper

Chapter 2

The Cepstrum

Foundation of any cepstral dereverberation algorithm is the cepstral transformation, i.e. the application that maps the time domain into the cepstral domain. In the cepstral domain, a convolution becomes an addition and a mixed phase signal appears in separated form. This chapter reviews the definition of the cepstrum and its properties as mostly given in [22].

2.1 The cepstral domain

In this section, it is shown how a space is found where a convolution becomes a summation. Furthermore, mapping D is defined that transforms a time domain signal in that space.

2.1.1 Problem formulation

Consider the discrete convolution of two signals denoted by $*$

$$\begin{aligned} y[n] &= (s * h)[n] && (2.1) \\ &= \sum_{k=-\infty}^{k=+\infty} s[k]h[n-k] && (2.2) \end{aligned}$$

We like to find now an application $\mathcal{D}$,

$$\mathcal{D} : y \rightarrow c_y \tag{2.3}$$

such that

$$c_y = c_x + c_h \tag{2.4}$$

where

$$c_y = \mathcal{D}[y] \tag{2.5}$$
$$c_x = \mathcal{D}[x] \tag{2.6}$$
$$c_h = \mathcal{D}[h]. \tag{2.7}$$

That is, we like the application $\mathcal{D}$ to transform a convolution into a summation.

2.1.2 Defining the cepstral transformation

It is known that the z-transform of two convoluted signals is the product of the z-transforms of the underlying signals. Further, the log-function has the property to transform a product into a sum.
This leads to the following application $\mathcal{D}$:

$$\mathcal{D}[\,] = \mathcal{Z}^{-1} \circ \log \circ \mathcal{Z}\,[\,] \tag{2.8}$$

In this document, we denote the z-transform $\mathcal{Z}[x]$ of the signal x by $X(z)$ and its logarithm $\log[X]$ by $\hat{X}$.
As the z-transformed signal is a complex function, we have to use the complex logarithm in (2.8). The complex logarithm of a complex number is actually not a function, but a set of numbers:

$$\log(z) = \log(z) + i\arg(z) \tag{2.9}$$
$$\arg(z) \ni \{\arg(z)\} \tag{2.10}$$

To resolve the ambiguities, its imaginary part has to be fixed to some argument. Then, the log-function is an analytic function because the exp-function is complex differentiable in the whole complex plane and in virtue of the mentioned fixation also bijectif. Note that the standard procedure of taking the principal value of the log does not work here as the principal value of a sum is generally not equal to the sum of the principal values violating requirement (2.4).
Instead, we impose the constraint that x is a real function. As the Discrete Time Fourier Transform (DTFT) $X\left(e^{i\Omega}\right)$ of x is in that case conjugate symmetric,

$$X\left(e^{-i\Omega}\right) = \overline{X}\left(e^{i\Omega}\right) \tag{2.11}$$

we require in order to make the complex logarithm bijective that the imaginary part of $\hat{X}$ is an odd function of Ω.
In virtue of that requirement, the imaginary part of $\hat{X}$ is a continuous odd periodic function of Ω. Therefore, c_x is real as well. Consequently, the application $\mathcal{D}$, as defined by (2.8) and the above stated choice of the complex logarithm, becomes a mapping between real numbers:

$$\mathcal{D} : \mathbb{R} \quad \rightarrow \quad \mathbb{R} \tag{2.12}$$

Furthermore, it is invertible, because the complex logarithm has been fixed to be so and because the z-transform is it either.

$$\mathcal{D}^{-1}[\,] \quad = \quad \mathcal{Z} \circ \exp \circ \mathcal{Z}^{-1}\,[\,] \tag{2.13}$$

2.2 The cepstrum

According to the above discussion, we define the cepstrum c_x of x as

$$\begin{aligned} c_x[n] &= \mathcal{D}[x] && (2.14)\\ &= \mathcal{Z}^{-1} \circ \log \circ \mathcal{Z}\,[x] && (2.15)\\ &= \frac{1}{2\pi i} \oint_C \log[X(z)] z^{n-1} dz && (2.16) \end{aligned}$$

An alternative expression for c_x is given in the Appendix A.1.1. An implicit relation between the cepstrum c_x and the signal x in Appendix A.1.2.

2.2.1 The power and angle cepstrum

Recall that $\mathcal{D}$ is a mapping between real numbers. Thus, the following property of the Fourier transformation $\mathcal{F}$

$$\mathcal{F}^{-1}[\overline{f}] \quad = \quad \overline{\mathcal{F}^{-1}[f]}(-t) \tag{2.17}$$

degenerates to

$$\mathcal{F}^{-1}[\overline{f}] \quad = \quad \mathcal{F}^{-1}[f](-t), \tag{2.18}$$

and we can write the following identities:

$$\begin{aligned} \mathcal{F}^{-1}[\mathrm{Re} f] &= \frac{1}{2}(\mathcal{F}^{-1}[f](t) + \mathcal{F}^{-1}[f](-t)) && (2.19)\\ \mathcal{F}^{-1}[\mathrm{Im} f] &= \frac{1}{2i}(\mathcal{F}^{-1}[f](t) - \mathcal{F}^{-1}[f](-t)) && (2.20) \end{aligned}$$

These identities will be used in the following definitions of the angle and power cepstrum to link them to the cepstrum. The power cepstrum c_{P_x} is defined as

$$c_{P_x}[n] = \frac{1}{2\pi i}\oint_C \log[|X(z)|]z^{n-1}dz \quad (2.21)$$

$$= \mathrm{Z}^{-1}\left[\mathrm{Re}\hat{X}(z)\right]. \quad (2.22)$$

We can use the above stated identities to derive the power cepstrum c_{P_x} from the cepstrum c_x:

$$c_{P_x}[n] = \frac{c_x[n] + c_x[-n]}{2} \quad (2.23)$$

The angle cepstrum $c_{\angle_x}$ is defined as

$$c_{\angle_x}[n] = \frac{1}{2\pi i}\oint_C i\arg X(z)z^{n-1}dz \quad (2.24)$$

$$= \mathrm{Z}^{-1}\left[i\mathrm{Im}\hat{X}(z)\right], \quad (2.25)$$

and again, we can write it as function of the cepstrum c_x:

$$c_{\angle_x}[n] = \frac{c_x[n] - c_x[-n]}{2} \quad (2.26)$$

We see that $\arg X(z)$, as defined by the above choice of the complex algorithm, is given by the z-transform of the angle cepstrum.

2.3 Rational z-transforms

In this section, an explicit expression of the cepstrum is derived for sequences with rational z-transforms. In this document, all signals x can be presented by such a z-transform.

2.3.1 Basic case

Consider a signal $x[n]$ with the z-transform

$$X(z) = (1 - az^{-1}) \quad (2.27)$$

where $a < 1$.
When we develop the log-function in its Laurent-series

$$\log[1 - z] = -\left[\sum_{k=1}^{\infty}\frac{z^k}{k}\right] \quad (2.28)$$

assuming $|z < 1|$ we find the cepstrum c_x of x by coefficient comparison. $\hat{X}$ is given by:

$$\hat{X}(z) = \log[1 - az^{-1}] \tag{2.29}$$

$$= -\left[\sum_{k=1}^{\infty} \frac{a^k}{k} z^{-k}\right] \tag{2.30}$$

Therefore:

$$c_x[n] = \begin{cases} -\frac{a^n}{n} & n > 0 \\ 0 & n \leq 0 \end{cases} \tag{2.31}$$

2.3.2 General case

It follows from the above result, that the cepstrum of a signal x with a z-transform given by

$$X(z) = A\frac{\prod_{k=1}^{M_i}(1 - a_k z^{-1}) \prod_{k=1}^{M_o}(1 - b_k z)}{\prod_{k=1}^{N_i}(1 - c_k z^{-1}) \prod_{k=1}^{N_o}(1 - d_k z)} \tag{2.32}$$

is

$$c_x[n] = \begin{cases} \sum_{k=1}^{N_i} \frac{c_k^n}{n} - \sum_{k=1}^{M_i} \frac{a_k^n}{n} & n > 0 \\ \log[A] & n = 0 \\ \sum_{k=1}^{M_o} \frac{b_k^{-n}}{n} - \sum_{k=1}^{N_o} \frac{d_k^{-n}}{n} & n < 0 \end{cases} \tag{2.33}$$

where the quantities a_k, b_k, c_k and d_k are all less than 1.
To increase the readability, the following abbreviations are introduced:

$$A[n] = \sum_{k=1}^{N_i} a_k^n - \sum_{k=1}^{M_i} c_k^n \tag{2.34}$$

$$B[n] = \sum_{k=1}^{M_o} b_k^n - \sum_{k=1}^{N_o} d_k^{-n}, \tag{2.35}$$

which permits to write (2.33) in more compact form:

$$c_x[n] = \begin{cases} -\frac{A[n]}{n} & n > 0 \\ \log[A] & n = 0 \\ \frac{B[-n]}{n} & n < 0 \end{cases} \tag{2.36}$$

We note that, in general, the cepstrum is nonzero and of infinite extent for both positive and negative n. Furthermore, it decays at least as 1/n.

Time delay terms

Equation (2.32) excludes a term like z^{-n_0}. This is the norm in cepstral computing. The reason is the following.
Let x be defined by

$$X(z) \quad = \quad z^{-n_0} \tag{2.37}$$

Then its cepstrum is given by

$$c_x[n] = \begin{cases} 0 & n = 0 \\ \frac{-n_0}{n}(-1)^n & n \neq 0 \end{cases} \tag{2.38}$$

It changes sign at each sample and although it decays, it may be quite large depending on n_0. This is inconvenient for some applications. That is why time shifts are removed before the cepstrum is calculated. Note that the value of n_0 has to be traced in order to make the cepstral transformation invertible.

2.3.3 Minimum phase sequences

If $X(z)$ has no poles or zeros outside the unit circle, e.g. $b_k = d_k = 0$, then the signal $x^{\min}$, obtained by stable inverse z-transformation, is said to be minimum phase. We note that it verifies $x^{\min}[n] = 0$ for $n < 0$ as in (2.32) no linear phase part is present.
From (2.33), we find for the cepstrum of a minimum phase sequence $x^{\min}$

$$c_x^{\min}[n] = \begin{cases} \sum_{k=1}^{N_i} \frac{c_k^n}{n} - \sum_{k=1}^{M_i} \frac{a_k^n}{n} & n > 0 \\ \log[A] & n = 0 \\ 0 & n < 0 \end{cases} \tag{2.39}$$

or equivalently,

$$c_x^{\min}[n] = \begin{cases} -\frac{A[n]}{n} & n > 0 \\ \log[A] & n = 0 \\ 0 & n < 0 \end{cases} \tag{2.40}$$

Thus, both $x^{\min}[n]$ and $c_x^{\min}$ are zero for $n < 0$.

2.3.4 Maximum phase sequences

The case of maximum phase signals is symmetric to the case of minimum phase signals. If $X(z)$ has no poles or zeros inside the unit circle, e.g. $a_k =$

$b_k = 0$, then the signal x, obtained by stable inverse z-transformation, is said to be maximum phase. We note that the signal $x^{\max}$ verifies $x^{\max}[n] = 0$ for $n > 0$ as in (2.32) no linear phase part is present.
From (2.33), it follows that the cepstrum of $x^{\max}$ is given by

$$c_x^{\max}[n] = \begin{cases} 0 & n > 0 \\ \log[A] & n = 0 \\ \sum_{k=1}^{M_o} \frac{b_k^{-n}}{n} - \sum_{k=1}^{N_o} \frac{d_k^{-n}}{n} & n < 0 \end{cases} \tag{2.41}$$

or equivalently by,

$$c_x^{\max}[n] = \begin{cases} 0 & n > 0 \\ \log[A] & n = 0 \\ \frac{B[-n]}{n} & n < 0 \end{cases} \tag{2.42}$$

Thus, both $x^{\max}$ and $c_x^{\max}$ are zero for $n < 0$.

2.3.5 Train of impulses

Consider the signal x, defined as

$$x[n] = \delta[n] + \alpha\delta[n - N_p] \tag{2.43}$$

with $|\alpha| < 1$.
Its z-transform is

$$X(z) = 1 + \alpha z^{-N_p}, \tag{2.44}$$

and taking the log of $X(z)$, we have:

$$\hat{X}(z) = \log[1 + \alpha z^{-N_p}] \tag{2.45}$$

$$= \sum_{k=1}^{\infty} (-1)^{k+1} \frac{\alpha^k}{n} z^{-kN_p} \tag{2.46}$$

The cepstrum is thus a infinitely long impulse train spaced by N_p

$$c_x[n] = \sum_{k=1}^{\infty} (-1)^{k+1} \frac{\alpha^k}{k} \delta[n - kN_p]. \tag{2.47}$$

For signals of more general form

$$x[n] = \sum_{k=1}^{k=M} \alpha_k \delta[n - kN_p] \tag{2.48}$$

the logarithm of its z-transform is a sum of terms like in (2.45), thus the complex cepstrum is also given by a impulse train spaced by N_p.

2.4 Conclusion & References

. In this chapter, we have reviewed the basics of the cepstrum as mostly given in [22]. Especially, we have highlighted the underlying motivation for its definition: Transforming a convolution into a summation. Furthermore, we have seen that the cepstrum is of infinite extent and, by construction, that it does not contain any information on the absolute position of the signal in time. Additionally, we have noticed that the Fourier transform of the angle cepstrum corresponds to the Fourier phase of the time domain signal up to a linear phase shift. Of further interest will be that a mixed phase signals appears in the cepstral domain demixed.

As mentioned, this review of the cepstrum is based on [22] which introduces the cepstrum from the point of view of homomorphic filtering. It discusses further the use of the cepstrum to process echoed signal. [18] is a short variant of [22]. An introduction to the cepstrum in form of a tutorial is given in [17]. It includes an exhaustive reference list. A collection of papers related to speech analysis containing articles on the cepstrum is found in [21]. Textbooks that deal with digital processing of speech signals provide also an introduction to the cepstrum, see for example [19].

Chapter 3

Signal Reconstruction from Fourier Phase

In dereverberation problems, the convolutive channel cannot be assumed to be known. In one way or the other, the channel is estimated during the deconvolution process. For the estimation of the channel, as much information on it should be included as possible. This is the motivation for summarising the condition on a signal for its Fourier phase is function of the magnitude of the Fourier transform.
In case of minimum phase signals, the Kramers-Krönig provides such a link [20]. In this chapter, conditions given in [2] are reviewed that do not exclude mixed phase signals.

3.1 Preliminaries

Before we cite the main result of [2], we present some mathematical background that will make the theorem easier to understand. Refer to Appendix B for more details.

- For real valued signals, zeros occur in complex conjugate pairs and for symmetrical signals in reciprocal pairs, respectively.

$$x[n] \in \mathbb{R} \quad \Rightarrow \quad \overline{X}(z) = X(\overline{z}) \tag{3.1}$$

$$x[-n] = x[n] \quad \Rightarrow \quad X(\frac{1}{z}) = X(z) \tag{3.2}$$

  Thus, for real valued, symmetric sequences, we have

$$z_0 \in \{\text{zeros}\} \quad \Rightarrow \quad \overline{z_0},\ \frac{1}{z_0},\ \frac{1}{\overline{z_0}} \in \{\text{zeros}\}. \tag{3.3}$$

- For signals x with rational z-transforms as in (2.32), a zero at $z = z_0$ and a pole at $z = \frac{1}{\bar{z}_0}$ contribute the same amount of phase to the spectrum.

- If the phase of the spectrum of a finite-length signal x is zero or π for all Ω, then, x is an even sequence.

3.2 Main theorem

The section states the conditions, as given in [2], for that a signal is fully identical to an other signal when they are merely equal in Fourier phase. Respectively, it shows in which cases the magnitude of the Fourier transform is function of the phase of the Fourier transform.

3.2.1 Statement

All involved sequences are supposed to be real valued. We denote the phase of the Fourier transform of $x_1[n]$ by $\theta_{x_1}(\Omega)$ and of $x_2[n]$ by $\theta_{x_2}(\Omega)$, respectively.

Theorem 3.1 *Let $x_1[n]$ and $x_2[n]$ be two real valued, finite length sequences whose z-transforms have no zeros in reciprocal pairs or on the unit circle. If $\theta_{x_1}(\Omega) = \theta_{x_2}(\Omega)$ for all Ω, then $x_1[n] = \beta x_2[n]$ for some positive constant β. If $\tan[\theta_{x_1}(\Omega)] = \tan[\theta_{x_2}(\Omega)]$ for all Ω, then $x_1[n] = \beta x_2[n]$ for some real constant β.*

This theorem can be extended to the case when only sampled Fourier phase information is available. The conditions remain the same, however, the length N of the signal x must additionally be known. Refer to [2].

3.2.2 Proof

To proof the theorem, we consider the signal $y[n]$ defined by

$$y[n] \quad = \quad x_1[n] * x_2[-n] \tag{3.4}$$

If $\theta_{x_1}(\Omega) = \theta_{x_2}(\Omega)$, then the Fourier phase of $y[n]$ is zero, if $\tan[\theta_{x_1}(\Omega)] = \tan[\theta_{x_2}(\Omega)]$, then the phase values zero or π because of the periodicity of the tan function. From Section 3.1, we conclude that $y[n]$ is not only a real valued, finite length sequence, but also even. From Section 3.1, it follows

further that the zeros of $Y(z)$ occur in reciprocal pairs. Let z_0 be such that $Y(z_0) = 0$, then we have as well $Y(1/z_0) = 0$. As the zeros of $Y(z)$ are collectively the zeros of $X_1(z)$ and $X_2(1/z)$, the zero at z_0 is either due to $X_1(z)$ or $X_2(1/z)$. Let it be a zero of $X_1(z)$. As $Y(1/z_0) = 0$, we have either $X_1(1/z_0) = 0$ or $X_2(z_0) = 0$. Because of the assumption that the zeros do not occur in reciprocal pairs, $Y(1/z_0)$ is zero because of $X_2(z_0)$. Thus, if z_0 is a zero of $X_1(z)$, it follows that z_0 is a zero of $X_2(z)$ as well and vice versa. Since $x_1[n]$ and $x_2[n]$ are all zero sequences

$$X_1(z) \quad = \quad \beta X_2(z) \tag{3.5}$$

Or

$$x[n] \quad = \quad \beta y[n] \tag{3.6}$$

$y[n]$ can therefore be written as a convolution of $x_1[n]$ with $\beta x_1[-n]$. The Fourier phase of $x_1[n] * x_1[-n]$ is always zero. Since the Fourier phase of $y[n]$ is zero if $\theta_{x_1}(\Omega) = \theta_{x_2}(\Omega)$, β must be positive. Since the phase of $y[n]$ is zero or π if $\tan[\theta_{x_1}(\Omega)] = \tan[\theta_{x_2}(\Omega)]$, β must be real.

3.2.3 Interpretation

The problem is to determine $x[n] \in \mathbb{R}$ with a rational z-transform from its Fourier phase alone.

As explained in Section 3.1, a zero at $z = z_0$ and a pole at $z = \frac{1}{\overline{z_0}}$ contribute the same amount of phase to the spectrum. Thus, given phase information alone, it is not possible to determine if the phase is due to a zero or due to an appropriate pole. That is the reason why in the above theorem, the finite length condition was imposed restricting the z-transform to have zeros only, except possibly at $z = 0$ or $z = \infty$.

The additional condition that the z-transform has no zeros in reciprocal pairs eliminates the possibility of zero phase components which could of course not be recovered from phase information alone. From Section 3.1, it can be seen how the phase terms introduced by the zeros z_0, $\overline{z_0}$, $1/z_0$ and $1/\overline{z_0}$ cancel each other contributing effectively nothing to the phase of the spectrum.

3.3 Closed form solution

A closed form solution for reconstructing a sequence $x[n]$ of length N from samples of its phase, $\theta_x(\Omega)$, follows from the definition of $\theta_x(\Omega)$.
For a complex number $w = a + ib$, a and b $\in \mathbb{R}$, the phase can be found by $\tan[\theta] = b/a$. As $x[n]$ is real, we have

$$\tan[\theta_x(\Omega)] = \frac{-\sum_{n=0}^{N-1} x[n] \sin n\Omega}{\sum_{n=0}^{N-1} x[n] \cos n\Omega} \tag{3.7}$$

Using the definition of the tan function and the fact that $\sin(a + b) = \sin a \cos b + \cos a \sin b$ we have for $\theta_x(\Omega) \neq k\frac{\pi}{2}, k \in \mathbb{Z}$

$$\sum_{n=0}^{N-1} \sin(\theta_x(\Omega) + n\Omega)x[n] = 0 \tag{3.8}$$

Respectively,

$$\sum_{n=1}^{N-1} \sin(\theta_x(\Omega) + n\Omega)x[n] = -x[0] \sin(\theta_x(\Omega)) \tag{3.9}$$

In case $\theta_x(\Omega) = k\frac{\pi}{2}, k \in \mathbb{Z}$, we obtain instead

$$\sum_{n=0}^{N-1} x[n] \cos n\Omega = 0 \tag{3.10}$$

Respectively,

$$\sum_{n=1}^{N-1} x[n] \cos n\Omega = -x[0] \tag{3.11}$$

Sampling the phase at $N - 1$ distinct frequencies Ω_k, we have therefore

$$\sum_{n=1}^{N-1} \sin(\theta_x(\Omega_k) + n\Omega_k)x[n] = -x[0] \sin(\theta_x(\Omega_k)) \tag{3.12}$$

As there are $N - 1$ equations for the $N - 1$ unknowns of $x[n]$, we can write the system in matrix form.

$$\mathbf{A}x = x[0]\mathbf{b} \tag{3.13}$$

where A and b are given by

$$(\mathbf{A})_{lm} = \sin(\theta_x(\Omega_l) + m\Omega_l) \tag{3.14}$$

$$(\mathbf{b})_l = \sin(\theta_x(\Omega_l)) \tag{3.15}$$

Note that $x[0]$ takes the role of the real scaling factor β. We know from the theorem in Section 3.2 that a solution exists if the sequence x fullfills the stated requirements. Hence, we can conclude that in that case the inverse of $\mathbf{A}$ can be calculated.

3.4 Conclusion & References

In this chapter, we have dealt with the problem to reconstruct a signal from its Fourier phase only. If the signal is of finite extent and it does not have zeros in reciprocal pairs or on the unit circle, then, as we have seen, is the magnitude of its Fourier transform function of the Fourier phase. Furthermore, a closed form solution in form of a linear equation system has been presented.

Main reference is [2]. Of further interest is also [13] as it discusses iterative procedures for signal reconstruction from Fourier transform phase. In [12], a proof of convergence for the iterative algorithms of [13] is given. An iterative algorithm based on 3.3 is found in [1].

Part II

Cepstral Dereverberation – In Practice

Chapter 4

The Cepstrum

The definition of the cepstrum as given in part I is based on some assumptions that are not met when the signals are digitally processed. The use of the z-transform implicitly assumes infinite storage capacity as well as infinite observation time. In this chapter, the consequences on the use of the cepstrum in dereverberation problems are discussed when the cepstrum is calculated under the realistic assumption of finite storage capacity. First, it is investigated how much the two cepstra differ. Then the basic properties of the cepstrum, e.g. the transformation of a convolution into a summation and its ability to separate mixed phase signals in the cepstral domain, are reconsidered from the standpoint of finite storage capacity.

4.1 Aliasing

In Chapter 2, we have seen that the cepstrum is of infinite extent. In practice, i.e. in case of numerical processing, it is however not possible to store all that data. For the same reasons, it is neither possible to completely calculate a z-transform.
In this section, is is analysed in what the cepstrum calculated under the constraint of finite storage capacity differs from the cepstrum defined in Chapter 2.

4.1.1 How to calculate the cepstrum

Let x be a signal of finite extent. This assumption is valid for the rest of the document. Then x is stable and it has a rational z-transform. Further,

the region of convergence of x will include the unit circle. Choosing for the contour C in (2.16) just the unit circle, we can then use the Discrete Time Fourier Transform (DTFT) interchangeably with the z-transform when calculating the cepstrum.
Consequently, the calculation of the cepstrum $c_x = \mathcal{D}[x]$ consists of the following three steps.

1. Take the DTFT of $x[n]$ and remove any linear phase term to obtain $X(e^{i\Omega})$.

$$X(e^{i\Omega}) \quad = \quad e^{i\Omega n_0}\mathcal{DTFT}\,[x]$$

   Keep the time shift n_0.

2. Take the complex logarithm of $X(e^{i\Omega})$ to obtain $\hat{X}(e^{i\Omega})$.

$$\hat{X}(e^{i\Omega}) \quad = \quad \log\left[X(e^{i\Omega})\right]$$

3. Take the Inverse DTFT (IDTFT) to obtain the cepstrum $c_x[n]$.

$$c_x[n] \quad = \quad \mathcal{IDTFT}\,\left[\hat{X}(e^{i\Omega})\right]$$

For the choice of the complex algorithm, see Section 2.1.

If only finite storage capacity is available, the Discrete Fourier Transform (DFT) is normally used to calculate numerically the DTFT of x. Then the above listed three steps become:

1. Take the N-DFT of $x[n]$ and remove any linear phase term to obtain $X[k]$.

$$X[k] \quad = \quad e^{i\frac{2\pi}{N}kn_0}\;\mathcal{DFT}\,[x]$$

   Keep the time shift n_0.

2. Take the complex logarithm of $X[k]$ to obtain $\hat{X}[k]$.

$$\hat{X}[k] \quad = \quad \log\left[X[k]\right]$$

3. Take the Inverse DFT (IDFT) to obtain the cepstrum $c_x^{\mathrm{NDFT}}[n]$.

$$c_x^{\mathrm{NDFT}}[n] \quad = \quad \mathcal{IDFT}\left[\hat{X}[k]\right]$$

Equivalence

As shown in various textbooks on Digital Signal Processing, e.g. [16], sampling the DTFT of x at N distinct frequency points has in the time domain the effect of periodising x. In concrete, if the DTFT $X(e^{i\Omega}) = \mathcal{DTFT}\,[x]$ is transformed back into the time domain by use of a N-DFT, we obtain for $x^{\mathrm{NDFT}} = \mathcal{IDFT}\,[X]$

$$x^{\mathrm{NDFT}} \quad = \quad \sum_m x[n+Nm] \tag{4.1}$$

Because c_x is of infinite extent, this implies for the calculation of the cepstrum by means of the DFT, that $c_x^{\mathrm{NDFT}} = \mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}\,[x]$ is given by

$$c_x^{\mathrm{NDFT}}[n] \quad = \quad \sum_m c_x[n+Nm] \tag{4.2}$$

This relation indicates that c_x^{NDFT} is obtained from c_x by adding an infinite number of shifted replicas of c_x to c_x, with each replica shifted by an integer multiple of N, and observing the sum only for the interval $0 \leq n \leq N-1$. If c_x decays too slowly, there is a time-domain aliasing effect in generating c_x^{NDFT}, and in that case the two cepstra will not correspond. Hence, in order to distinguish the two cepstra, we name c_x^{NDFT} "DFT-cepstrum", and c_x "z-cepstrum".
In Chapter 2, we have seen that the application $\mathcal{D}$ is invertible, i.e.

$$\mathcal{D}^{-1} \circ \mathcal{D}\,[x] \quad = \quad x. \tag{4.3}$$

This uniqueness-property of the z-cepstrum is preserved, i.e.

$$\mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}^{-1}} \circ \mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}\,[x] \quad = \quad x. \tag{4.4}$$

4.1.2 Illustration of aliasing

Here, we like to illustrate (4.2) and (4.4). From (4.2), we see that the deviation of the DFT-cepstrum from the z-cepstrum depends on the DFT length N and the decay rate of the z-cepstrum. From (2.33), we notice that the position of the zeros of x dictates the decay rate of c_x. Zeros close to the unit circle cause the z-cepstrum to decay slowly. Consider a signal x that has two zeros, the first at $x = x_1$, $x_1 = 2.5$. The second at $x = x_2$,

$x_2 = 0.999999$. Its z-cepstrum is given by

$$c_x[n] = \begin{cases} -\frac{x_2^n}{n} & n > 0 \\ \frac{x_1^{-n}}{n} & n < 0 \\ 0 & n = 0 \end{cases} \tag{4.5}$$

Figure 4.1 shows c_x and c_x^{NDFT}, $N = 64$, as well the summation process of (4.2).

It has been noticed that the application $\mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}$ is invertible, e.g. that the DFT-cepstrum calculated by the mapping $\mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}$ is unique. Figure 4.2 shall emphasise that the inverse must in general be calculated with the same DFT-length for the latter statement to be true. Indeed, if c_x^{NDFT} has been affected by aliasing, then for $N' \neq N$

$$\mathcal{D}_{\mathcal{N}'}^{\mathcal{DFT}^{-1}} \circ \mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}\,[x] \neq x. \tag{4.6}$$

However, if it has not been affected, then

$$\mathcal{D}_{\mathcal{N}'}^{\mathcal{DFT}^{-1}} \circ \mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}\,[x] = x. \tag{4.7}$$

4.1.3 Quantitative analysis

We have seen that the location of the zeros of x and the DFT-length N have an influence on the amount the DFT-cepstrum deviates from the z-cepstrum. This will be analysed in more detail in the following paragraph.

An upper bound for the error

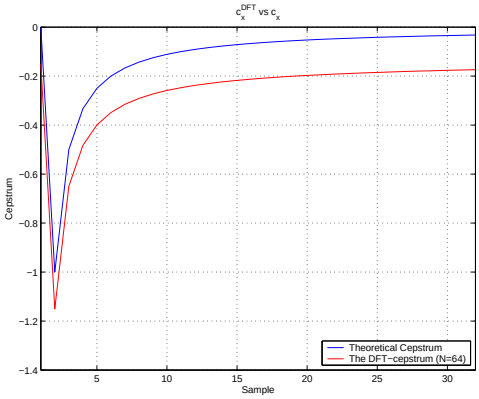
Assuming that x is mixed phase , we like to determine an upper bound for the error Absolute Error (AE) $e_{1\mathrm{A}}$ defined by

$$e_{1\mathrm{A}}[n] = |(c_x[n] - c_x^{\mathrm{NDFT}}[n]|. \tag{4.8}$$

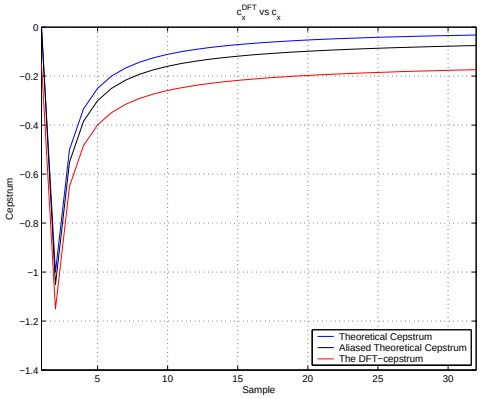
Further, we will indicate how the error behaves in function of the sample index n.

$$\max_n e_{1\mathrm{A}}[n] = \max_n \left|c_x^{\mathrm{NDFT}}[n] - c_x[n]\right| \tag{4.9}$$

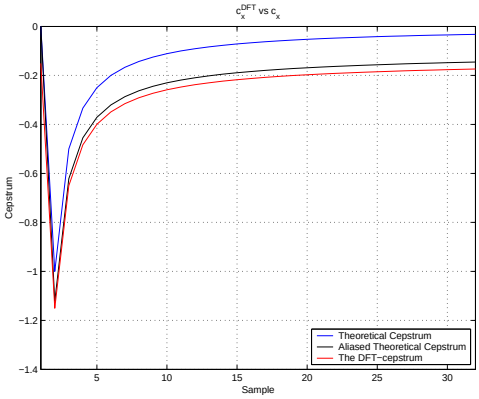
$$= \max_n \left|\sum_{m\neq 0} c_x[n + mN]\right| \tag{4.10}$$



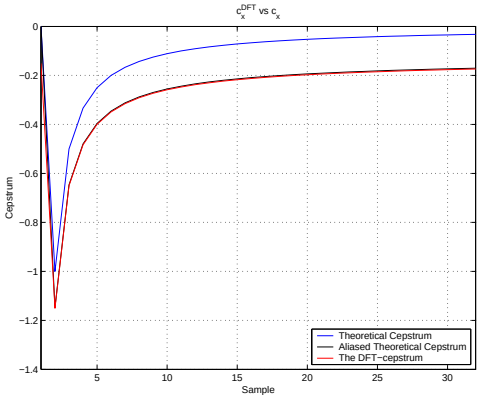
(A) DFT-cepstrum vs. z-cepstrum



(B) Adding back 10^3 shifted replicas

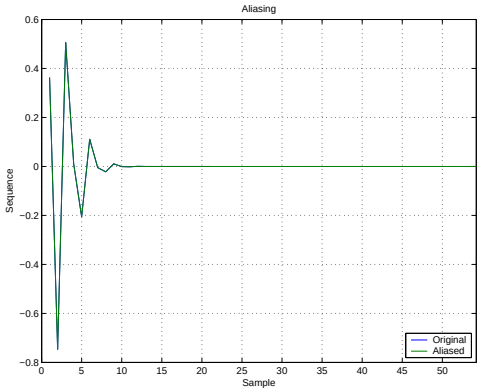


(C) Adding back 10^5 shifted replicas

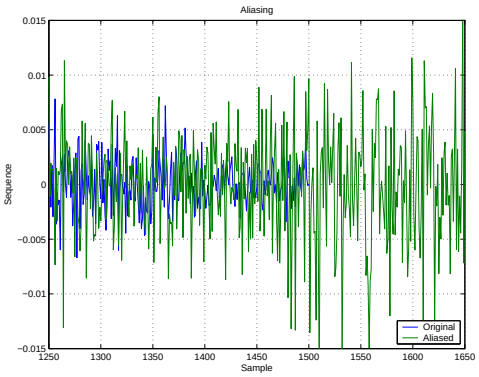


(D) Adding back 10^6 shifted replicas

Figure 4.1: The DFT-cepstrum and the z-cepstrum. For $N = 64$, the DFT-cepstrum c_x^{NDFT} differs remarkably from the z-cepstrum. At $n = 0$, the deviation $\triangle c$ amounts to $\triangle c = 0.15$, at $n = 5$ to $\triangle c = 0.149$ (60%) and at $n = 25 = \triangle c = 0.1432$ (344%). The absolute deviation is biggest at the origin. Note also that the number of replicas we have to add to c_x in order to obtain c_x^{NDFT} is a measure for the DFT length N that must be used to avoid aliasing.



(A) A nice minimum phase signal, length $N_x = 20$



(B) A signal with zeros near the unit circle, length $N_x = 1500$

Figure 4.2: Invertability of the DFT-cepstrum. Both sequences have been transformed into the cepstral domain using a DFT of length $N = 2048$. After zero padding, their inverse was calculated. It is seen that we cannot rely on $\mathcal{D}_{\mathcal{N}'}^{\mathcal{DFT}^{-1}} \circ \mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}[x] = x$ unless aliasing plays no role.

From (2.33), we get for c_x

$$c_x[n] = \begin{cases} \sum_{k=1}^{M_i} -\frac{a_k^n}{n} & n > 0 \\ \sum_{k=1}^{M_o} \frac{b_k^{-n}}{n} & n < 0 \end{cases} \tag{4.11}$$

Therefore:

$$\max_n e_{1\mathrm{A}}[n] = \max_n \left| \sum_{\substack{m \neq 0 \\ n+mN>0}} \sum_{k=1}^{M_i} -\frac{a_k^{n+mN}}{n+mN} + \ldots \right. \\ \left. \ldots \sum_{\substack{m \neq 0 \\ n+mN<0}} \sum_{k=1}^{M_o} \frac{b_k^{-(n+mN)}}{n+mN} \right| \tag{4.12}$$

Respectively

$$\begin{aligned}\max_n e_{1\mathrm{A}}[n] \leq \max_n \left| \sum_{\substack{m\neq 0 \\ n+mN>0}} \sum_{k=1}^{M_i} -\frac{a_k^{n+mN}}{n+mN} + \ldots \right| \\ \ldots \max_n \left| \sum_{\substack{m\neq 0 \\ n+mN<0}} \sum_{k=1}^{M_o} \frac{b_k^{-(n+mN)}}{n+mN} \right|\end{aligned} \tag{4.13}$$

The sum is maximised by maximising the addends. For the first addend, we obtain:

$$\max_n \left| \sum_{\substack{m\neq 0 \\ n+mN>0}} \sum_{k=1}^{M_i} -\frac{a_k^{n+mN}}{n+mN} \right| \leq \max_n \sum_{\substack{m\neq 0 \\ n+mN>0}} \sum_{k=1}^{M_i} \left| -\frac{a_k^{n+mN}}{n+mN} \right| \tag{4.14}$$

$$\leq \sum_{m>0} \sum_{k=1}^{M_i} \frac{|a_k^N|^m}{mN} \tag{4.15}$$

$$\leq \frac{M_i}{N} \sum_{m>0} \frac{\left(\max_k(|a_k^N|)\right)^m}{m} \tag{4.16}$$

With the short hand notation

$$a^* = \max_k(|a_k^N|) \tag{4.17}$$

we obtain for the upper bound B_1 of the first addend:

$$B_1 \leq \frac{M_i}{N} \sum_{m>0} \frac{a^*}{m}. \tag{4.18}$$

It is possible to express the sum in the above equation in closed form:

$$\sum_{m>0} \frac{a^*}{m} = \int_0^{a^*} \left[\sum_{m>0} (a)^{m-1} \right] \mathrm{d}a \tag{4.19}$$

$$= \int_0^{a^*} \left[a^{-1} \sum_{m>0} (a)^m \right] \mathrm{d}a \tag{4.20}$$

$$= \int_0^{a^*} \frac{1}{1-a} \mathrm{d}a \tag{4.21}$$

$$= -\log(1-a^*) \tag{4.22}$$

Hence, the upper bound of the first addend is

$$B_1 \leq -\frac{M_i}{N} \log(1-a^*). \tag{4.23}$$

The same arguments are valid for the second addend, therefore, we obtain the following upper bound for the AE $e_{1\mathrm{A}}$ caused by aliasing:

$$\begin{aligned} \max_n e_{1\mathrm{A}}[n] &= e_{1\mathrm{A}}[0] && (4.24) \\ &\leq -\frac{M_i}{N}\log(1-a^*) - \frac{M_o}{N}\log(1-b^*) && (4.25) \end{aligned}$$

From 4.25 and its derivation, it is noticed:

- The closer the zeros are to the unit circle, the bigger is the error. The dependency is sub-linear. Therefore, the error can in theory be suppressed by choosing the DFT-length N big enough. In practice, however, the required N becomes easily too big.

- The different zeros act additively on the error. This implies that a few zeros close to the unit circle can make the DFT-cepstrum remarkably different from the z-cepstrum.

- The smaller the sample index, the bigger the error. The maximum error occurs at sample index $n = 0$. As the scaling of the signal x in the time domain depends exponentially on the value of the cepstrum at $n = 0$, a big error at $n = 0$ may cause the signal to "explode" in the time domain.

- The error e_{1A} decays fast if n increases. Compare to Figure 4.3.

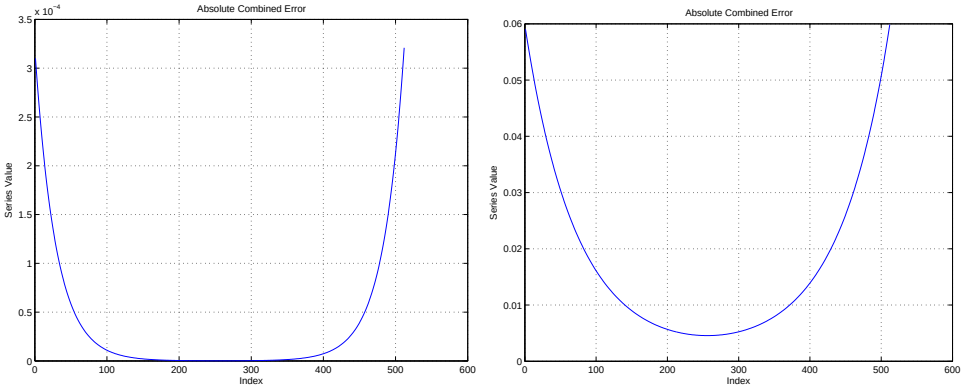
Error propagation

In this paragraph, it is investigated how a change in the cepstrum c_x manifests itself in the signal x.
Assuming a cepstrum given by

$$\tilde{c}[n] = c_x[n] + \triangle c\delta[n - n_0], \quad (4.26)$$

we will determine the relation between $\tilde{x}$ and x by a calculation of $\mathcal{D}^{-1}[\tilde{c}]$. With reference to Subsection 4.1.1, the DTFT of $\tilde{c}_x$ is first calculated for that purpose

$$\begin{aligned} \hat{\tilde{X}}(e^{i\Omega}) &= \sum_n c_x[n]e^{-i\Omega n} + \triangle c e^{-i\Omega n_0} && (4.27) \\ &= \hat{X}(e^{i\Omega}) + \triangle c e^{-i\Omega n_0}, && (4.28) \end{aligned}$$



(A) 200 zeros at $z = 0.97$ and at $z = 1/0.97$ (B) 200 zeros at $z = 0.99$ and at $z = 1/0.99$

Figure 4.3: Absolute Error e_{A1} in the cepstrum in function of the index n. Note that a DFT of length $N = 512$ has been used. That is why the indices $n = -1, \ldots, -255$ appear at 511 to 256.

followed by the application of the exp-function

$$\tilde{X}(e^{i\Omega}) = \exp[\hat{X}] \exp\left[\triangle c e^{-i\Omega n_0}\right] \quad (4.29)$$

$$= X(e^{i\Omega}) \exp\left[\triangle c e^{-i\Omega n_0}\right] \quad (4.30)$$

and determining the inverse DTFT

$$\tilde{x}[n] = x[n] * \mathcal{IDTFT}\left[\exp\left[\triangle c e^{-i\Omega n_0}\right]\right] \quad (4.31)$$

The IDTFT of

$$E(\Omega) = \exp\left[\triangle c e^{-i\Omega n_0}\right] \quad (4.32)$$

is easily calculated when the exp-function is developed in a series, i.e

$$\exp[x] = 1 + \sum_{n>0} \frac{x^n}{n!}, \quad (4.33)$$

because for $n_0 \neq 0$

$$E(\Omega) = 1 + \sum_{m>0} \frac{\triangle c^m}{m!} e^{-i\Omega m n_0} \quad (4.34)$$

which is just the Fourier series representation of the 2π periodic function $E(\Omega)$. For $n_0 = 0$, we would have

$$E(\Omega) = \exp[\triangle c]. \quad (4.35)$$

Therefore, the IDTFT $e[n]$ of $E(\Omega)$ is given by

$$e[n] = \begin{cases} \sum_{m>0} \frac{\triangle c^m}{m!}\delta[n-mn_0] & n \neq 0 \quad n_0 \neq 0 \\ 1 & n = 0 \quad n_0 \neq 0 \\ e^{\delta c}\delta[n] & \forall n \quad\quad n_0 = 0 \end{cases} \tag{4.36}$$

and the time domain signal $\tilde{x}[n]$ relates to the signal $x[n]$ as follows:

$$\tilde{x}[n] = x[n] * e[n] \tag{4.37}$$

An error in the cepstrum turns out to cause the signal to be convoluted with a weighted Dirac train.

Assuming now that the cepstrum has been modified at several indices n_i and let the cepstrum be given by

$$\tilde{c}[n] = c_x[n] + \sum_{i=0}^{I} \triangle c[n_i]\delta[n-n_i] \tag{4.38}$$

Then the signal $\tilde{x}[n]$ that we obtain when we transform $\tilde{c}[n]$ back into the time domain, is given by

$$\tilde{x}[n] = x[n] \coprod_{i=0}^{i=I} e[n;n_i] \tag{4.39}$$

where $e[n;n_i]$ is

$$e[n;n_i] = \begin{cases} \sum_{m>0} \frac{\triangle c[n_i]^m}{m!}\delta[n-mn_i] & n \neq 0 \quad n_i \neq 0 \\ 1 & n = 0 \quad n_i \neq 0 \\ e^{\triangle c[n_i]}\delta[n] & \forall n \quad\quad n_i = 0 \end{cases} \tag{4.40}$$

and $\coprod$ means multiple convolutions. Note that the above derivation was made under the hypothesis that the DTFT is used. In case, we use the DFT instead, it turns out that the only change is that $e[n]$ equals zero for $n > N-1$ and $n < 0$.
The consequences of (4.39) are the following:

- The cepstral coefficient at $n = 0$ is responsible for scaling. An absolute change $\triangle c$ at $n = 0$ causes the signal to be multiplied by a factor of $\exp[\triangle c]$.

- Each absolute error in the cepstrum accounts for a convolutive changing of the previous sequence.

- Provided the original sequence decays fast, changes at small indices n_i account for alterations in form while larger indices for replication phenomena.

Figure 4.4 visualises the effects of a cepstral change as described by (4.39).

4.1.4 Summary

The main points of this section are listed below.

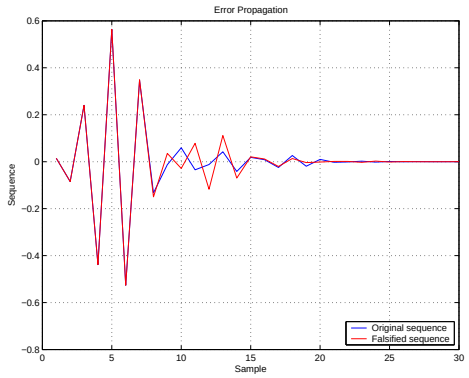
- Time domain aliasing due to the infinite extent of c_x causes the DFT-cepstrum to be in general different from the z-cepstrum.
- The deviation from the z-cepstrum becomes considerable whenever the sequence x has zeros that are too close to unit circle. Deviation is biggest at cepstral coefficients around the origin.
- Each zero acts additively on the deviation from the z-cepstrum. Hence, a few zeros close to the unit circle can flaw the calculation of the z-cepstrum by means of the DFT.

4.2 Deconvolution

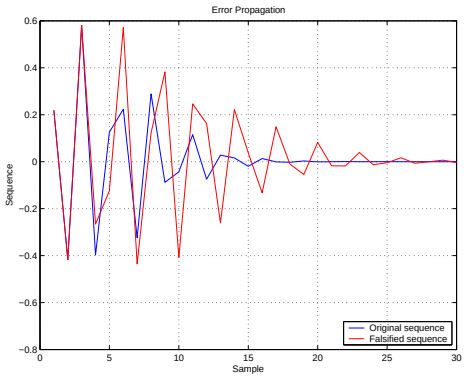
In this section, we show how the cepstrum can be used to invert a mixed phase channel. The problem is stated as follows and shall referred to as "one-channel deconvolution problem".

- Let h and s be mixed phase sequences of finite extent.
- Observed is $\tilde{y} = h * s + w$ whereas the channel h is known and w is supposed to be White Gaussian Noise (WGN).
- The goal is to estimate the sequence s.

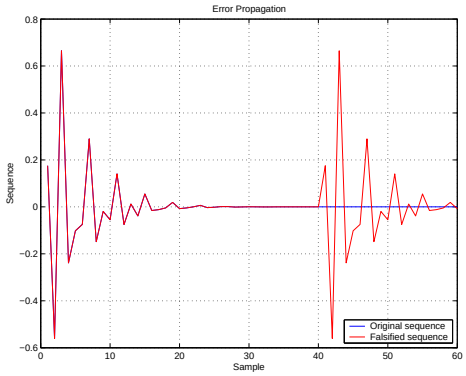
First, it is explained how cepstral deconvolution works, then it is illustrated with some noise–free examples. The noisy case is afterwards analysed both qualitatively and quantitatively.



(A) Single modification for n_i small



(B) Double modification for n_i small



(C) Single modification for n_i large

Figure 4.4: Consequences of a cepstral modification in the time domain. Blue: Time domain signal x for the original cepstrum. Red: Time domain signal $\tilde{x}$ of the altered cepstrum. For Subfig. 4.4(a), $n_1 = 6$ and $\triangle c = 0.2$ For Subfig. 4.4(b), $n_1 = 6$ and $n_2 = 3$. For Subfig. 4.4(c), $n_i = 40$ and $\triangle c = 1$.

4.2.1 Principle

The definition of the cepstrum as given in Chapter 2 was motivated by the idea to find a space where a convolution is transformed into a summation. Therefore, by definition of the cepstrum, we have

$$y = h * s \quad \Leftrightarrow \quad c_y = c_h + c_s \tag{4.41}$$

As we assumed the channel h as known, its cepstrum c_h is known as well. Hence, after the cepstrum c_y has been computed from the observation y, cepstrum of the signal s can be obtained by subtraction.

$$c_s \quad = \quad c_y - c_h \tag{4.42}$$

Then, by taking the inverse cepstral transformation $\mathcal{D}^{-1}$, we obtain the signal s. The deconvolution process consists thus of the following three steps.

1. Calculate the cepstra $c_y = \mathcal{D}[y]$ and $c_h = \mathcal{D}[h]$.
2. Subtract them: $c_{\hat{s}} = c_y - c_h$.
3. Take the inverse cepstral transformation to achieve the estimate $\hat{s}$: $\hat{s} = \mathcal{D}^{-1}[c_{\hat{s}}]$

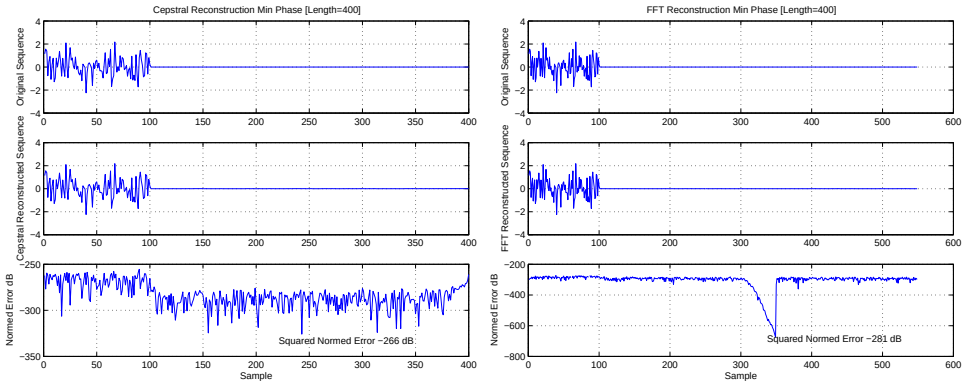
Equivalence

In Section 4.1, we noted that in practice, the operator $\mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}$ has to be used instead of $\mathcal{D}$. Because of the uniqueness property (4.4), however, the use of $\mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}$ and $\mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}^{-1}}$ instead of $\mathcal{D}$ and $\mathcal{D}^{-1}$ will not affect the deconvolution concept.

4.2.2 Examples

The goal of this subsection is to illustrate the above explained idea of solving the deconvolution problem by use of the cepstrum. Furthermore, the results are compared to the results obtained by filtering the observed signal y with the inverse of h defined as

$$h^{-1} \quad = \quad \mathcal{IDFT}\left[\frac{1}{H(e^{i\Omega})}\right] \tag{4.43}$$



(A) Cepstral deconvolution (B) Inverse filtering

Figure 4.5: One-channel deconvolution. A comparison. Setup: WGN of length $N_s = 100$ been convoluted with a minimum phase channel of length $N_h = 50$ having its zeros uniformly distributed between the disc $0.2 < |z| < 0.8$. The two methods yield both very low NSE $e_{2\mathrm{N}}$.

where

$$H(e^{i\Omega}) \quad = \quad \mathcal{DTFT}\ [h[n]] \tag{4.44}$$

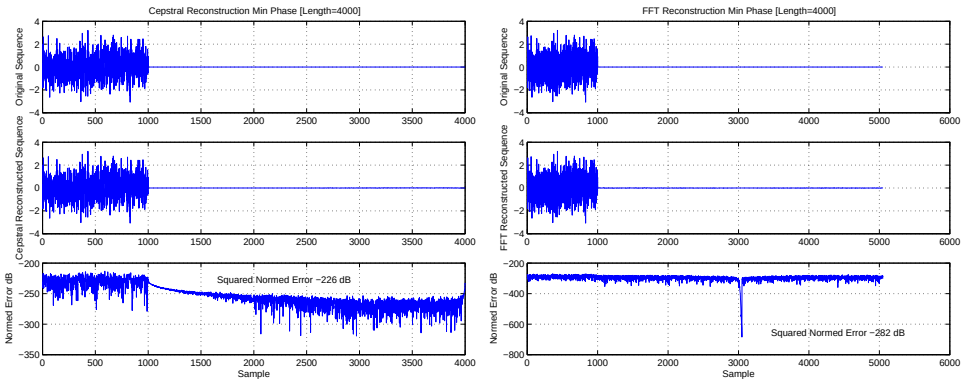
The cepstral method operates block-wise on the signal y causing therefore a processing delay equal to the block size. The delay of the inverse filtering method, however, depends on the DFT length used and the presence of a–causal terms in the inverse h^{-1}. The curves in the Figures 4.5 to 4.10 have been corrected for the time shift to make a simple comparison between the the two methods possible.

4.2.3 Noisy case

In the previous section, we have seen that the cepstral deconvolution method works fine for mixed phase signals. However, no noise was considered.
In this section, the deconvolution scheme operating in noisy conditions is analysed both qualitatively and quantitatively.

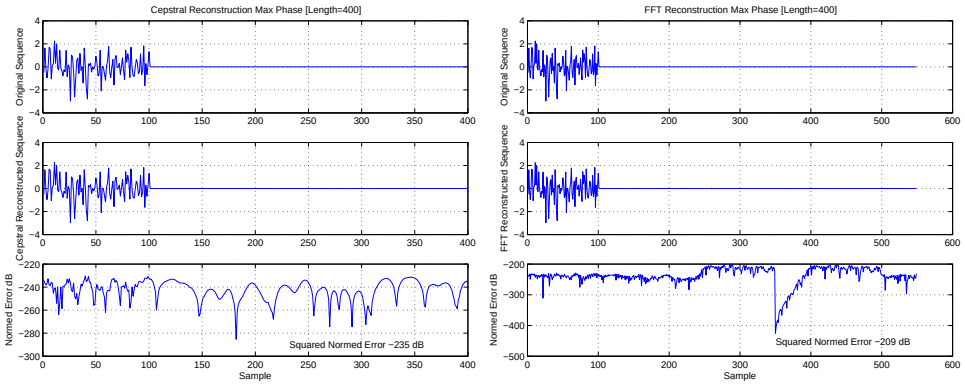
Qualitative Analysis

To analyse systematically the influence of noise on the deconvolution scheme, we used the following experimental setting:



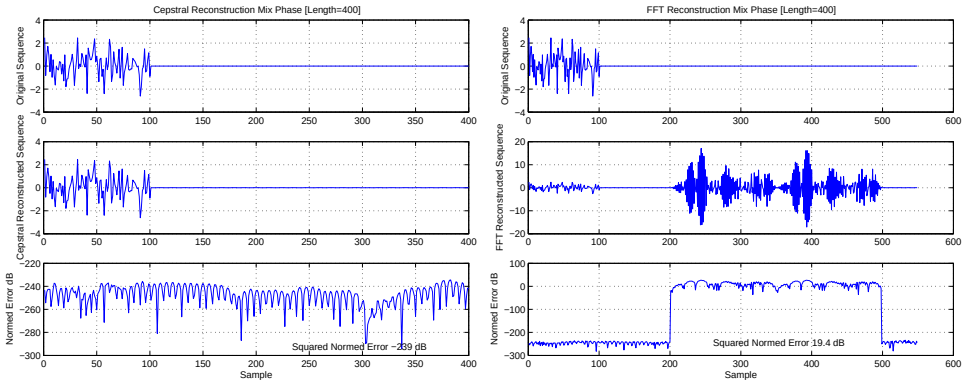
(A) Cepstral deconvolution (B) Inverse filtering

Figure 4.6: One-channel deconvolution. A comparison. Setup: Like in Figure 4.5, but $N_s = 1000$. The two methods yield both very low NSE e_{2N}.



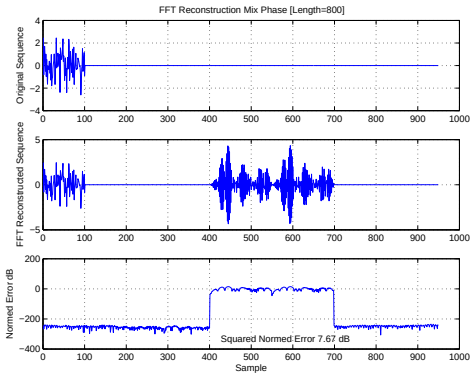
(A) Cepstral deconvolution (B) Inverse filtering

Figure 4.7: One-channel deconvolution. A comparison. Setup: Like in 4.5, but for a maximum phase channel having its zeros uniformly distributed in $1.2 < |z|1.8$. The two methods yield both very low NSE e_{2N}.

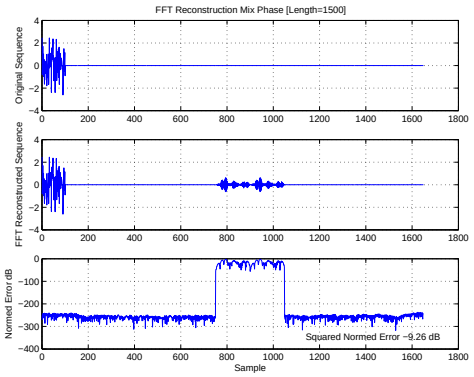


(A) Cepstral deconvolution (B) Inverse filtering

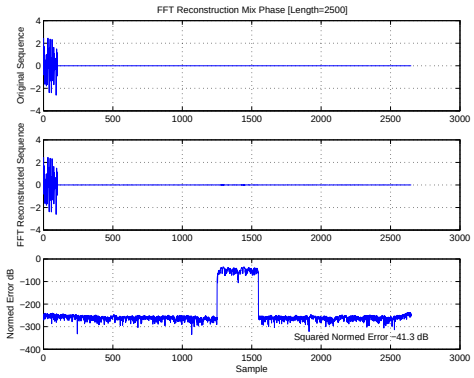
Figure 4.8: One-channel deconvolution. A comparison. Setup: Figure 4.5, but h has its zeros very close to the unit circle. $0.99 < |z| < 1.1$. There are differences in the deconvolution results. As we forced the zeros to be located close to the unit circle, the inverse filter h^{-1} decays very slowly. Further, we chose the length N of the DFT to be $N = 400$. Because of the low decay rate, however, an considerable error is caused if the infinite impulse response h^{-1} is truncated at $N = 400$. This error becomes manifest in the reconstructed sequence.



(A) DFT length $N = 800$

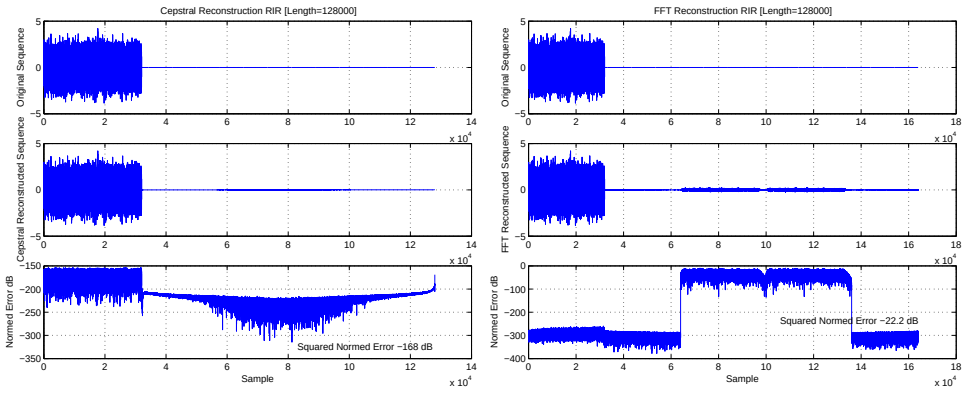


(B) DFT length $N = 1500$



(C) DFT length $N = 2500$

Figure 4.9: Inverse filtering. Effect of increasing the DFT length. Setup as in Figure 4.8. Increasing the DFT-length improves the deconvolution process by suppressing the truncation error. In turn, it causes the processing delay to increase.



(A) Cepstral deconvolution (B) Inverse filtering

Figure 4.10: One-channel deconvolution. A comparison under realistic conditions. Setup: Simulated RIR of length $N_h = 4000$ has been convoluted with WGN of length $N_s = 32000$. For the used DFT length N, the inverse method does not provide a result as good as the cepstral method.

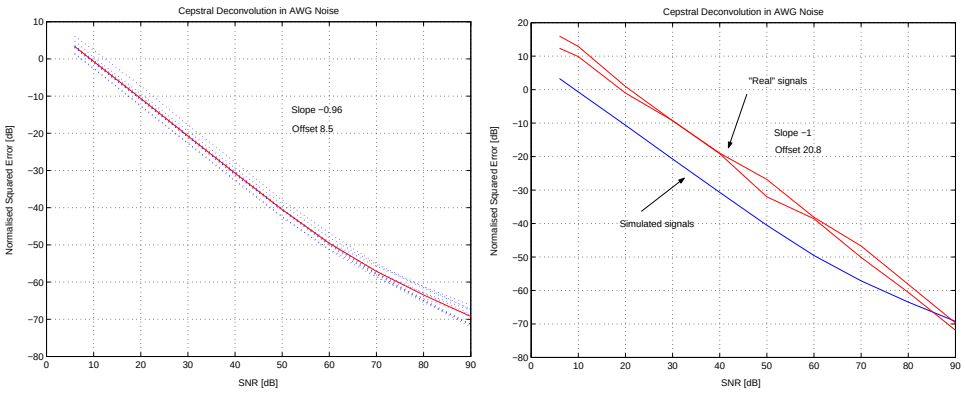
30 Gaussian white noise sequences of length $N_s = 32000$ were convoluted with a simulated Room Impulse Response (RIR) of length $N_h = 1000$ for different noise levels. The lowest Signal to Noise Ratio (SNR) was set to SNR=6. For each signal and SNR, the Normalised Squared Error (NSE) $e_{2\mathrm{N}}$ has been calculated. To eliminate fluctuations, the average over the 30 signals was used to measure the ability of the cepstral method to invert the simulated channel. To yield a more significant measure for the cepstral deconvolution scheme, the experiment was done for 10 different channels. The error $e_{2\mathrm{N}}$ in dB is defined as

$$e_{2\mathrm{N}} = 10\log_{10}\left[\frac{||s-\hat{s}||^2}{||s||^2}\right] \tag{4.45}$$

and the SNR in dB as

$$\mathrm{SNR} = 10\log_{10}\left[\frac{||s||^2}{||w||^2}\right] \tag{4.46}$$

Figure 4.11(a) shows that the error $e_{2\mathrm{N}}$ made in the deconvolution process depends linearly on the SNR. From Figure 4.11(b), it is seen that the results obtained with simulated signals correspond to the results with real world RIRs h and real world speech signals s.



(A) Mean behaviour, simulated signals (B) Real world signals

Figure 4.11: Deconvolution in presence of AWGN. Deconvolution ability of the cepstrum is measured by the NSE e_{2N}. Subfig. 4.11(a) shows the average deconvolution ability of the cepstrum in case of simulated signals (red). Subfig. 4.11(b) states that it behaves similarly in real world conditions. Signal s is a 30s long speech signal that has been sampled at 8kHz. It has been convolved with two 8200 samples long RIR to provide the measurements y_1 and y_2.

Quantitative Analysis

The qualitative analysis has shown that the deconvolution ability of the cepstrum depends linearly on the SNR, if expressed in dB. A quantitative below will find the same result. Further, it will help to understand how the cepstral deconvolution method relates to the inverse filtering approach.
Let $\tilde{y}$ be given by

$$\tilde{y} = h * s + w \tag{4.47}$$

Respectively in the frequency domain

$$\tilde{Y}(e^{i\Omega}) = H(e^{i\Omega})S(e^{i\Omega}) + W(e^{i\Omega}) \tag{4.48}$$

$$= H(e^{i\Omega})S(e^{i\Omega})\left(1 + \frac{W(e^{i\Omega})}{H(e^{i\Omega})S(e^{i\Omega})}\right) \tag{4.49}$$

The cepstrum $c_{\tilde{y}}$ of $\tilde{y}$ is therefore

$$c_{\tilde{y}}[n] = c_s[n] + c_h[n] + \mathcal{IDTFT}\left[\log\left[1 + \frac{W}{HS}\right]\right] \tag{4.50}$$

$$\tag{4.51}$$

and thus

$$c_{\hat{s}} = c_s[n] + \mathcal{IDTFT}\left[\left(1 + \frac{W(e^{i\Omega})}{S(e^{i\Omega})}\frac{1}{H(e^{i\Omega})}\right)\right] \tag{4.52}$$

where $c_{\hat{s}} = c_{\tilde{y}} - c_h$.
Going back into the frequency domain, we obtain

$$\hat{S}(e^{i\Omega}) = S(e^{i\Omega}) + \frac{W(e^{i\Omega})}{H(e^{i\Omega})} \tag{4.53}$$

where $\hat{S}$ is the DTFT of the estimate $\hat{s}$.
This equation shows how the cepstral deconvolution scheme relates to the inverse filtering approach. They are the same. To subtract from the cepstrum of the observed signal $\tilde{y}$ the cepstrum of the channel h and to go back into the time domain, is equivalent to filtering the observed signal $\tilde{y}$ with the inverse channel h^{-1}.

It follows that the spectral power density $S_{e_{2\mathrm{A}}}(e^{i\Omega})$ of the Absolute Squared Error (ASE) $e_{2\mathrm{A}}$ is given by

$$S_{e_{2\mathrm{A}}}(e^{i\Omega}) = S_w(e^{i\Omega})\left|\frac{1}{H(e^{i\Omega})}\right|^2 \tag{4.54}$$

$$= (\mathrm{SNR}(e^{i\Omega}))^{-1} S_s(e^{i\Omega})\left|\frac{1}{H(e^{i\Omega})}\right|^2 \tag{4.55}$$

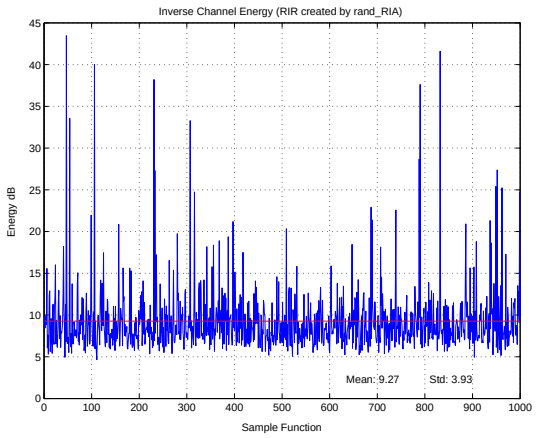


Figure 4.12: Inverse channel energy of RIR sequences. The offset in the linear relation between the error $e_{2\mathrm{N}}$ and the SNR, as defined in (4.46), is given by the inverse channel energy.

In our experiment, we used white Gaussian noise for w and s. The ASE $e_{2\mathrm{A}}$ is found by integration

$$e_{2\mathrm{A}} \quad = \quad \mathrm{SNR}^{-1}\sigma_s^2 \frac{1}{2\pi}\int_{-\pi}^{\pi}\left|\frac{1}{H(e^{i\Omega})}\right|^2 \mathrm{d}\Omega \tag{4.56}$$

and the NSE $e_{2\mathrm{N}}$ is therefore

$$e_{2\mathrm{N}} \quad = \quad -\,\mathrm{SNR}|_{\mathrm{dB}} + \frac{1}{2\pi}\int_{-\pi}^{\pi}\left|\frac{1}{H(e^{i\Omega})}\right|^2 \mathrm{d}\Omega\Bigg|_{\mathrm{dB}} \tag{4.57}$$

if expressed in dB. Equation (4.57) expresses in formula what is depicted in Figure 4.11(a) and 4.11(b):

- Both in the qualitative analysis and the quantitative analysis, we have found a linear relation between the error $e_{2\mathrm{N}}$ and the SNR.
- The offset is given by the energy of the inverse channel. See Figure 4.12.

Note that if we defined the SNR in the frequency domain as

$$\mathrm{SNR}(e^{i\Omega}) \quad = \quad \frac{H(e^{i\Omega})S(e^{i\Omega})}{W(e^{i\Omega})} \tag{4.58}$$

we would have obtained the following result for the NSE in dB:

$$e_{2\mathrm{N}} \quad = \quad -\,\mathrm{SNR}|_{\mathrm{dB}} \tag{4.59}$$

This result is easier to interpret because it shows that the ability of the cepstral deconvolution scheme to invert the channel is directly related to the SNR of the observed signal $\tilde{y}$.

4.2.4 Uncertainty on the channel

In the previous section, we have assumed noise that is additive to the convoluted signal. Here, we like to analyse how the deconvolution scheme works if the channel h is noisy, e.g. because it has been estimated. For this convolutive noise, we can write

$$\tilde{y} = (h + w) * s \tag{4.60}$$

which is the same as

$$\tilde{y} = h * s + w * s, \tag{4.61}$$

respectively

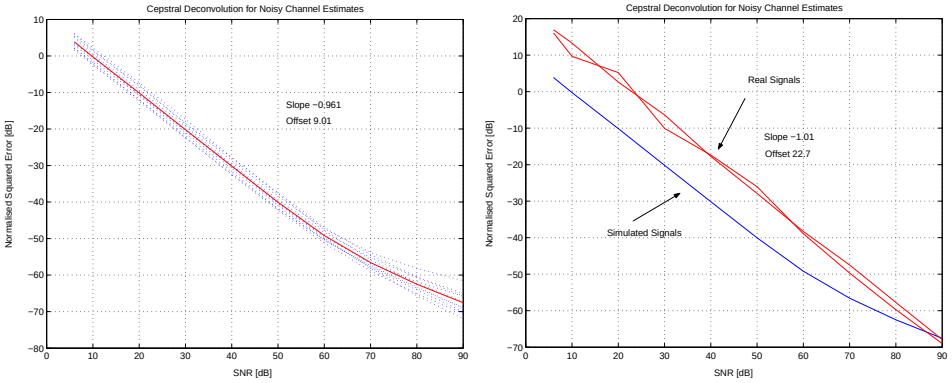
$$y = h * s + \tilde{w} \tag{4.62}$$

which resembles in form (4.47). The noise $\tilde{w}$, however, is signal-dependent. Figure 4.13 shows the results of a systematic analysis of the influence of convolutive noise on the deconvolution ability of the cepstrum. The setup of this analysis is the same as in Subsection 4.2.3.

4.2.5 Summary

The main points of this section are listed below.

- The cepstral deconvolution method is in principle the same as the inverse filtering approach. The cepstral method, however, is less sensitive to zeros that are close to the unit circle.
- Aliasing effects do not disturb the deconvolution process.
- Cepstral computing has no special noise-suppressing properties.



(A) Mean behaviour, simulated signals (B) Real world signals

Figure 4.13: Deconvolution in convolutive noise. Setup is the same as in Subsection 4.2.3. Similarly, a linear dependency of the NSE on the SNR is observed, if the latter is expressed in dB. Furthermore, the results obtained from simulated signals match the results found for real world signals.

4.3 Minimum maximum phase separation

In this section we will show how the cepstrum can be used to extract both minimum and maximum phase part of a signal of finite extent. The idea is to use the basic property of the cepstrum that the minimum phase part accounts for cepstral coefficients at positive sample indices while the maximum phase part accounts for cepstral coefficients at negative sample indices.

4.3.1 Principle

As we assumed that x is of finite extent, it follows from 2.32, that the z-transform of x is given by

$$X(z) \quad = \quad A \prod_{k=1}^{M_i} (1 - a_k z^{-1}) \prod_{k=1}^{M_o} (1 - b_k z) \tag{4.63}$$

A sequence with all poles and zeros inside the unit circle is said to be minimum phase while a sequence with all poles and zeros outside the unit circle

is said to be maximum phase. Therefore:

$$X(z) = A \underbrace{\prod_{k=1}^{M_i}(1-a_k z^{-1})}_{\text{Minimum Phase}} \underbrace{\prod_{k=1}^{M_o}(1-b_k z)}_{\text{Maximum Phase}} \; satisfy \tag{4.64}$$

We know from Chapter 2 that the cepstrum of a signal x with a rational z-transform is given by

$$c_x[n] = \begin{cases} \log(A) & n=0 \\ -\frac{A[n]}{n} & n>0 \\ \frac{B[-n]}{n} & n<0 \end{cases} \tag{4.65}$$

where

$$A[n] = \sum_{k=1}^{k=M_i} \frac{a_k^n}{n} \tag{4.66}$$

$$B[n] = \sum_{k=1}^{k=M_o} \frac{b_k^n}{n} \tag{4.67}$$

because of the finite extent assumption on x.
Obviously, for $n>0$, the cepstrum only depends on the minimum phase part of the signal while, for $n<0$, it depends on the maximum phase part. From that property of the cepstrum, we deduce the following minimum-maximum phase separation scheme:

1. Calculate the cepstrum c_x.

$$c_x = \mathcal{D}[x]$$

2. Separate c_x into two parts:

$$c_x^{\min}[n] = \begin{cases} c_x[n] & n \geq 0 \\ 0 & n<0 \end{cases}$$

And

$$c_x^{\max}[n] = \begin{cases} c_x[n] & n<0 \\ 0 & n>0 \end{cases}$$

3. Transform $c_x^{\min}$ and $c_x^{\max}$ back into the time domain.

$$\begin{aligned} x^{\min} &= \mathcal{D}^{-1}\left[c_x^{\min}\right] \\ x^{\max} &= \mathcal{D}^{-1}\left[c_x^{\max}\right] \end{aligned}$$

$x^{\min}$ is minimum phase only and $x^{\max}$ is maximum phase only. They further satisfy

$$x = x^{\min} * x^{\max}$$

Equivalence

In Section 4.1, we noted that in practice, the operator $\mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}$ has to be used instead of $\mathcal{D}$. Here, we like to comment on the consequence that this constraint has on step 3.
In step 3, the condition is imposed that the convolution of $x^{\min}$ with $x^{\max}$ must yield x. By construction of $x^{\min}$ and $x^{\max}$, this is verified if the z-cepstrum is used in the calculation. For the DFT-cepstrum, however, this is not true in general. This can be seen by the following calculation.
The sequences $x_{\min}^{\mathrm{NDFT}}$ and $x_{\max}^{\mathrm{NDFT}}$ that are obtained by means of a N-DFT have to verify

$$x_{\min}^{\mathrm{NDFT}} * x_{\max}^{\mathrm{NDFT}} \stackrel{!}{=} x \tag{4.68}$$

or, writing the convolution in the cepstral domain, it has to be verified that

$$\begin{aligned} \mathcal{D}_{\mathcal{N}'}^{\mathcal{DFT}}\left[\mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}^{-1}}\left[c_{\min}^{\mathrm{NDFT}}\right]\right] + \mathcal{D}_{\mathcal{N}'}^{\mathcal{DFT}}\left[\mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}^{-1}}\left[c_{\max}^{\mathrm{NDFT}}\right]\right] \\ \stackrel{!}{=} \mathcal{D}_{\mathcal{N}'}^{\mathcal{DFT}}[x]. \end{aligned} \tag{4.69}$$

where at least, $N' = 2N - 1$ because $x_{\min}^{\mathrm{NDFT}}$ and $x_{\max}^{\mathrm{NDFT}}$ are of length N. Hence, as unless there is no aliasing

$$\mathcal{D}_{\mathcal{N}'}^{\mathcal{DFT}^{-1}} \circ \mathcal{D}_{\mathcal{N}}^{\mathcal{DFT}}\,[x] \neq x, \tag{4.70}$$

we conclude that the discussed condition is violated when aliasing arises in the calculation of the cepstrum. This issue will below be of further consideration.

4.3.2 Qualitative analysis

In this subsection, we like to illustrate how the described separation scheme works in practice. For the minimum-maximum phase separation scheme to be considered as successful, two requirements have to be satisfied. First, the convolution $\hat{x}$ of $x^{\min}$ with $x^{\max}$ must equal x. Secondly, $x^{\min}$ must be minimum phase, $x^{\max}$ maximum phase, respectively.
The first requirement is quantified by the NSE $e_{2\mathrm{N}}$, as defined in (4.45), i.e.

$$e_{2\mathrm{N}} = 10\log_{10}\left[\frac{||x-\hat{x}||^2}{||x||^2}\right] \tag{4.71}$$

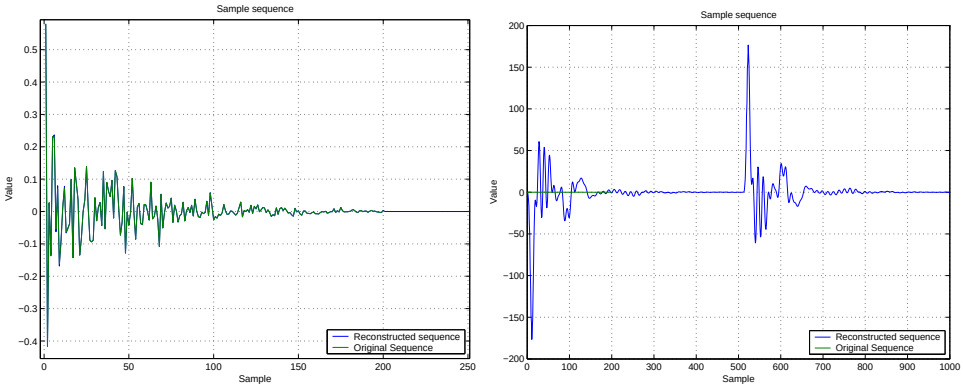
To quantify the second requirement, we remark that $x^{\min}$ is supposed to have only zeros inside the unit circle. Its stable inverse is thus expected to be a causal sequence. Hence, the L_2-norm of the anti-causal part of the inverse is considered as a measure for the "minimum-phaseness" of $x^{\min}$. Similarly, the L_2-norm of the causal part of the inverse of $x^{\max}$ is calculated to measure the "maximum-phaseness" of $x^{\max}$.

Example

In this example, we will focus on the first measure. Figure 4.14 shows the influence of the DFT length N on the ability to reconstruct the original sequence from $x^{\min}$ and $x^{\max}$. Obviously, reconstruction works fine if the DFT length is chosen long enough. However, this is not the case for shorter DFT lengths. With reference to Subsection 4.3.1, we put the blame on aliasing. Below, the special form of $\hat{x}$ in Figure 4.14 is discussed. See Figure 4.15(a) for a plot of the separation accuracy $e_{2\mathrm{N}}$ in function of the DFT length.

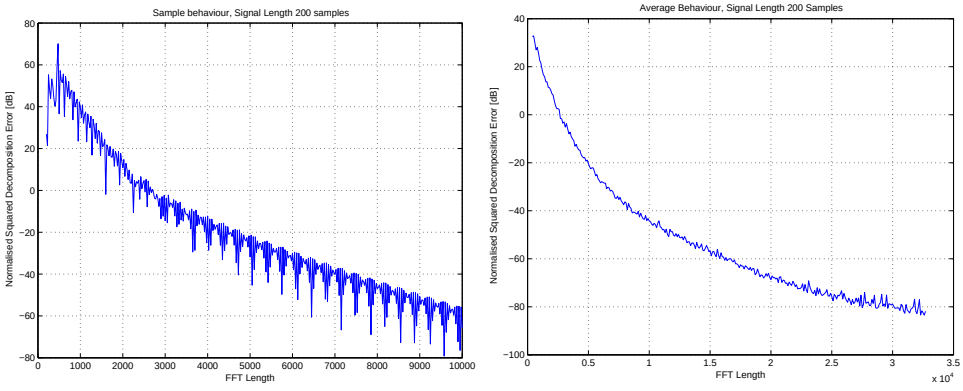
In more generality

For the particular sample sequence of the previous section, we could increase the DFT length to improve the separation. Here, we shall see that this is true in the average, but also, that there are sequences that cannot be separated with a reasonable long DFT length.
100 simulated RIRs x of length $N_x = 200$ have been separated as described above for multiple DFT lengths N, $N \in \{400, , \ldots, 32700\}$ with a spacing of $\triangle N = 100$. Then the mean over the 100 RIRs has been taken.
Figure 4.15(b) shows the mean decomposition error $< e_{2\mathrm{N}} >$ in function



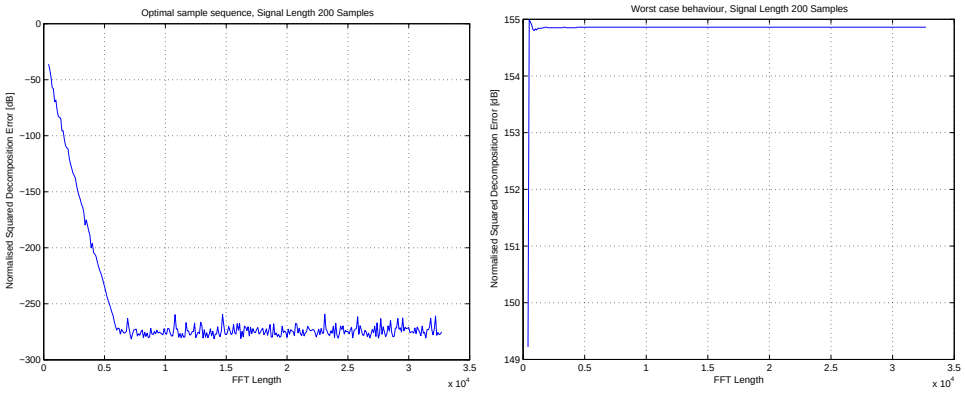
(A) DFT length $N = 4000$ (B) DFT length $N = 512$

Figure 4.14: Reassembling x from extracted minimum, respectively maximum phase signals. If N is too small, reconstruction is not possible. $\hat{x}$ corresponds neither in scaling nor in form to x.



(A) Sample sequence of Figure 4.14 (B) Average behaviour

Figure 4.15: Reconstruction error in function of the DFT length N. The NSE e_{2N} decreases linearly with the used DFT length. This behaviour is found for the sample sequence, and also in mean.



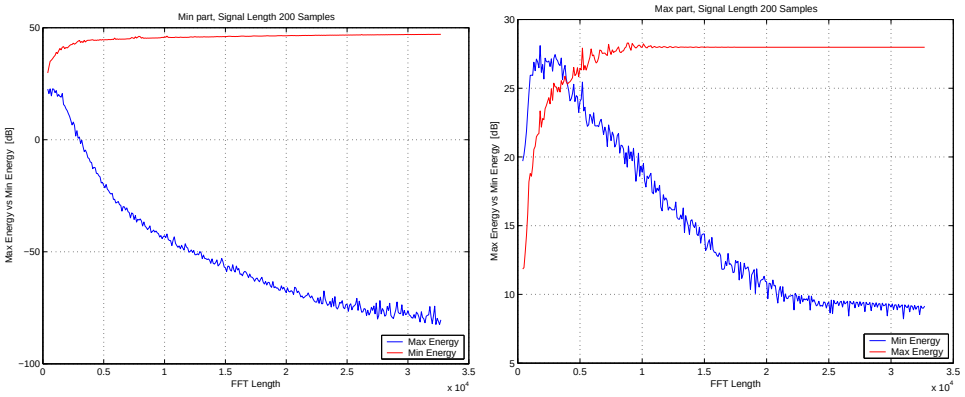
(A) Optimal case (B) Worst case

Figure 4.16: Reconstruction variance. While the NSE $e_{2\mathrm{N}}$ decays fast for one sequence, for the other it does not seem to be affected by an increase of the DFT length. Comparing to the mean behaviour of Figure 4.15(b), we observe a significant variance of $e_{2\mathrm{N}}$.

of the DFT length N. We find in average the same behaviour as in Figure 4.15(a). Furthermore, in the experiment, it has been observed that the wrongly reconstructed sequences $\hat{x}$ often looked like the curve in Figure 4.14(b): There are replicates at the index that is half the DFT length and the sequence is heavily scaled.

A comparison of the curves in Figure 4.16(a) and 4.16(b) shows that the variance of $e_{2\mathrm{N}}$ is high. In Figure 4.16(b) the sequence could not be separated at all, while in Figure 4.16(a), the error curve saturates fast at a small separation error $e_{2\mathrm{N}}$.

Now, we address the second requirement for the minimum-maximum separation to be a success, "minimum-phaseness" of $x^{\min}$ and "maximum-phaseness" of $x^{\max}$. From Figure 4.17(a), we see that in average the energy contained in the anti-causal part of the inverse of $x^{\min}$ decreases as the DFT length N increases while the energy in the causal part remains about constant. Thus, we can say that increasing the DFT length N improves strongly "minimum-phaseness" of $x^{\min}$. Similarly for the maximum phase signal $x^{\max}$ in Figure 4.17(b) whereas we note here that the signal does not become purely maximum phase.



(A) "Minimum phaseness" (B) "Maximum phaseness"

Figure 4.17: Second criterion for the minimum maximum separation scheme. We see that in average the signal $x_{\min}$ is really minimum phase for high DFT length as the energy contained in the anti-causal part of its inverse becomes very small whereas $x_{\max}$ does not become fully maximum phase.

4.3.3 Understanding the separation process

We have seen that in average the minimum-maximum phase separation scheme described in 4.3.1 works fine. However, we observed a high variance. This section will explain why this big differences are possible and particularly why the wrongly reconstructed sequences have that eye-catching form of the curve in Figure 4.14(b) or 4.18 .

For this purpose, we have analysed a mixed phase FIR sequence x of length $N_x = 100$ with known zeros that could not be separated with a DFT length of $N = 256$. Figure 4.18 shows the poor reconstruction result $\hat{x}$. To verify in experiment the suggestion of Subsection 4.3.1 that aliasing is responsible for the shown reconstruction, we calculated from the known zeros of x the z-cepstrum c_x and compared it to the DFT-cepstrum. Figure 4.3.3 shows the Error (E) $e_{\mathrm{A}}[n] = c_x^{\mathrm{DFT}}[n] - c_x[n]$. As noticed in Paragraph 4.1.3, the error is dominant at low sample indices. Note that errors at negative indices are circularly shifted to the end of the sequence. The error at $n = -1$ is thus found at the index $n = N - 1$. Recall further from Paragraph 4.1.3 that if we add $\triangle c$ to the cepstrum c_x at the cepstral index $n = n_i$, in the time domain, we cause the signal x to be convoluted with a weighted Dirac

train of spacing n_i. Considering only the dominant values of the error $e_{\mathrm{A}}[n]$, we are able to explain the curve in Figure 4.18:

- The error $e_{\mathrm{A}}[0]$ causes the scaling of the reconstructed signal $\hat{x}$.
- The errors $e_{\mathrm{A}}[1]$ to approximately $e_{\mathrm{A}}[10]$ cause the altered form of $\hat{x}$.
- The errors $e_{\mathrm{A}}[255]$ to approximately $e_{\mathrm{A}}[250]$ cause the replication of the re-scaled and form-altered sequence.

To underline that it is aliasing that causes the observed deterioration of x, we show in Figure 4.20 the cepstrum of the minimum phase sequence $x_{\min}^{\mathrm{DFT}}$. The cepstrum has been calculated with a big DFT length N and is here assumed to be equal to the z-cepstrum of $x_{\min}^{\mathrm{DFT}}$. From Subsection 4.1.3 we know that $x_{\min}^{\mathrm{DFT}}$ is given by the minimum phase sequence $x^{\min}$ convoluted with several Dirac trains of spacing n_i. Moreover, in Chapter 2 it has been shown that the cepstrum of a Dirac train of spacing n_i is as well spaced by n_i. Hence, we expect the cepstrum of $x_{\min}^{\mathrm{DFT}}$ to have dominant peaks at multiples of $n = 256$ because of the significant error at $n = -1$. From Figure 4.20, it is seen that the cepstrum of $x_{\min}^{\mathrm{DFT}}$ matches our expectations.

Refer to Appendix A.2 for a proof by explicit calculus that aliasing is the main culprit of deterioration.

4.3.4 Summary

The main points of this section are listed below.

- The rate of successful minimum-maximum phase separation shows a high variance.
- The separation improves with increasing DFT length.
- A signal with zeros near the unit circle will be difficult to be separated in maximum and minimum phase parts because aliasing will cause the DFT-cepstrum to be significantly different from the z-cepstrum.

4.4 Conclusion & Outlook

In this chapter, we have dealt with the practical realisation of the most important points of Chapter 2. We have looked at how the cepstrum calculated

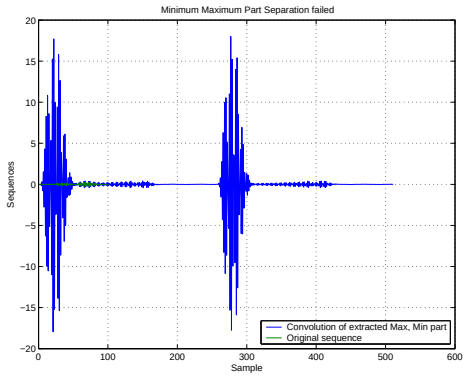


Figure 4.18: Reassembling x from extracted minimum, respectively maximum phase signals. A DFT length of $N = 256$ is not able to correctly separate the signal x in maximum and minimum phase parts. The reconstructed sequence $\hat{x}$ shows a replication effect at $n = 256$ and noteworthy scaling.

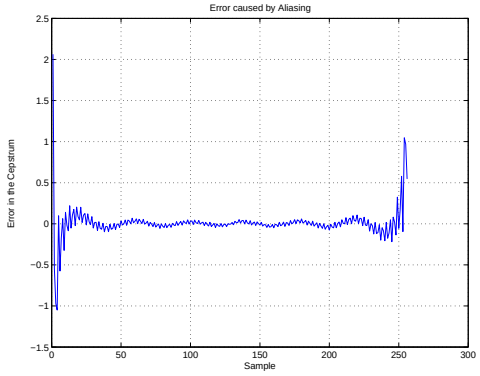


Figure 4.19: Absolute deviation of the DFT-cepstrum from the z-cepstrum. The error in the DFT-cepstrum has the highest values around the indices $n = 0$ and $n = 255$. Furthermore, the error decays fast around these local maxima. As errors at n_i in the cepstrum cause the original sequence to be convolvoluted with Dirac trains of spacing n_i, we obtain for time domain signal $\hat{x}$ the signal shown in Figure 4.18.

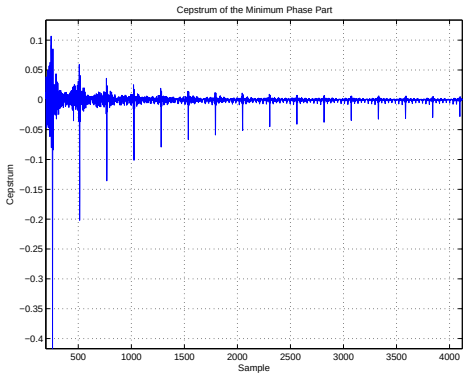


Figure 4.20: Part of the z-cepstrum of $x_{\min}^{\mathrm{DFT}}$. It has peaks that are spaced by $n = 256$. As the cepstrum of a Dirac train of spacing n_i is again spaced by n_i, this curve means that $x_{\min}^{\mathrm{DFT}}$ has been convoluted with a Dirac train of spacing $n = 256$. This underlines the explanation that aliasing is responsible for the poor reconstruction result $\hat{x}$.

by the DFT, i.e. the DFT-cepstrum, differs from the cepstrum calculated by the DTFT, i.e. the z-cepstrum. The result is that they do not correspond in general. Nevertheless, thanks to the uniqueness property of the cepstrum, cepstral deconvolution as considered in the chapter works fine. However, it limits the ability of the cepstrum to isolate correctly the minimum and maximum phase parts of a signal.

Several further investigations present themselves. First, it can be tried to keep aliasing at bay using for example exponential weighting techniques. Then, the minimum-maximum phase separation ability of the cepstrum can be used to do minimum phase-allpass separation. Further, one should take advantage of the quasi-time domain structure of the cepstrum. Cepstral prediction for example, applied together with the recursion formulas presented in the appendix, could be promising. Especially, when on-line applications are targeted.

Chapter 5

Signal Reconstruction from Fourier Phase

In dereverberation problems, the convolutive channel is in ordinary not known. In one way or the other, the channel has to be estimated during the deconvolution process. In Chapter 3, conditions were presented when the Fourier phase of a signal already contains all the information about the signal. This is helpful in the two-channel dereverberation problem. In this chapter, iterative algorithms that reconstruct a signal from its Fourier phase are evaluated in convergence speed, robustness and reconstruction accuracy.

5.1 Algorithms

In this section, various iterative algorithms are discussed that reconstruct a mixed phase signal from the Fourier phase alone. The theory behind these algorithms is explained in Chapter 3. Note that there is also a class of algorithms that starts from the closed form solution of Section 3.3. Refer to [1]. First, the basic algorithm is explained, then some derivates.

5.1.1 Basis

This algorithm was first published in [2]. In accordance with Section 3.2, we make the following assumptions on the signal x:

- It is of finite length N_x.

- It has no zeros on the unit circle or in reciprocal pairs.

Then, the following algorithm reconstructs the signal x from its Fourier phase θ_x except for a real scaling factor. The N-DFT of the unknown signal x is denoted by $X[k]$. The estimate of x by $\hat{x}$.

1. From the known phase θ_x and an initial guess $|\hat{X}_0|$ of the unknown magnitude $|X[k]|$, we form the first estimate $\hat{X}_1[k]$ of $X[k]$:

$$\hat{X}_1[k] = |\hat{X}_0[k]|e^{i\theta_x[k]}$$

   Its IDFT provides the first estimate $\hat{x}_1$ of x.

2. The original signal x is by assumption N_x samples long. Thus only the first N_x samples of $\hat{x}_1$ are retained:

$$\hat{\hat{x}}_1[n] = \begin{cases} \hat{x}_1[n] & 0 \leq n \leq N-1 \\ 0 & N \leq n \leq M-1 \end{cases}$$

3. The magnitude of the DFT of $\hat{\hat{x}}$ is then considered as a new estimate of $|X[k]|$, and the new estimate of $X[k]$ is formed by

$$\hat{X}_2[k] = |\hat{\hat{X}}_1[k]|e^{i\theta_x[k]}$$

   From that, a new estimate $\hat{x}_2$ is obtained by IDFT.

Repetitive application of Steps 2 and 3 defines the iteration, i.e. time-limiting and phase-substitution, that can be represented as follows:

$$\hat{x}_{p+1} = T\hat{x}_p \tag{5.1}$$

Where T is a nonlinear operator that corresponds to the combination of time-limiting and phase-substitution of step 2 and 3.

The idea behind the algorithm is to held the phase constant and to iterate till the phase of the estimate equals the known phase. From Section 3.2, we know that the estimate corresponds then to the true signal. Based on the theory of nonexpansive mappings, it is shown in [12] that the above algorithm converges.

5.1.2 Derivates

Relaxation techniques

Convergence speed of the above algorithm can be improved if relaxations techniques are employed. The basic idea is to modify the the above iteration

as follows:

$$\hat{x}_{p+1} = \hat{x}_p + \alpha_p(T\hat{x}_p - \hat{x}_p) \tag{5.2}$$

α is referred to as relaxation parameter and may be allowed to vary as a function of p. Details on this algorithm are found in [13].

Cepstral domain

In Section 2.2, it has been noticed that the Fourier phase of a signal can be found by taking the Fourier transform of its angle cepstrum. Recall that the angle cepstrum is given by the difference of the cepstral coefficients.

$$c_{\angle x} = \frac{c_x[n] - c_x[-n]}{2} \tag{5.3}$$

Therefore, it is possible to carry out the basic algorithm in the cepstral domain. Refer to [3] for a detailed discussion of that variant. Here, we shall only mention that energy-limiting the cepstrum is reported to increase convergence speed.

5.1.3 Performance measure

As the goal is to reconstruct a mixed phase time domain signal from its Fourier phase only, a first performance measure for the described algorithms is their capability to truly reconstruct the signal. The Normalised Squared Error $e_{2\mathrm{N}}$ is used. It is defined in (4.45) by

$$e_{2\mathrm{N}} = 10\log_{10}\left[\frac{||x - \hat{x}||^2}{||x||^2}\right] \tag{5.4}$$

Similarly, the Absolute Squared Error (ASE) $e_{2\mathrm{A}}$ quantifies if the cepstrum of the reconstructed signal corresponds to the true cepstrum. This ASE is defined by

$$e_{2\mathrm{A}} = 10\log_{10}\left[||c_x - c_{\hat{x}}||^2\right] \tag{5.5}$$

Then, the number of iterations till convergence is used to measure convergence speed.

5.1.4 Summary

The main points of this section are listed below.

- The iterative algorithms are all based on a fixed point algorithm as published first in [2]. This algorithm is proved to converge.
- Relaxation techniques can be used to improve the rate of convergence.
- The algorithm can be implemented in the cepstral domain which offers additional acceleration possibilities.

5.2 Examples

On the basis of the following examples, we try characterise the algorithms described in Section 5.1. For that purpose, we use the performance measures of Subsection 5.1.3.
First, we will illustrate the convergence behaviour of the different algorithms on the basis of short sequences. Then, we will show that the algorithms work also for longer sequences.

5.2.1 Convergence criterion

The convergence criterion is twofold. In [2] it is shown that the reconstruction error $e_{2\mathrm{A}}$ of the basic algorithm is nonincreasing. Thus, we decide upon the progress E_p of an iteration if the algorithm is to be stopped at iteration p. Progress E_p is defined as

$$E_p \quad = \quad ||\hat{x}_{p+1} - \hat{x}_p||^2 \tag{5.6}$$

Furthermore, we have the Fourier phase at hand to measure convergence. The ASE $e_{2\mathrm{A}}$ of the angle cepstrum of the estimate and the original signal is used. As it has been noticed in [6] that this error is not nonincreasing during the iteration process, we use it as an a-posteriori convergence measure. The threshold for E_p has been set to $E_{\overline{p}} = 10^{-15}$ for the basic algorithm and its relaxation variant, and for the cepstral domain algorithm to $E_{\overline{p}} = 10^{-8}$.

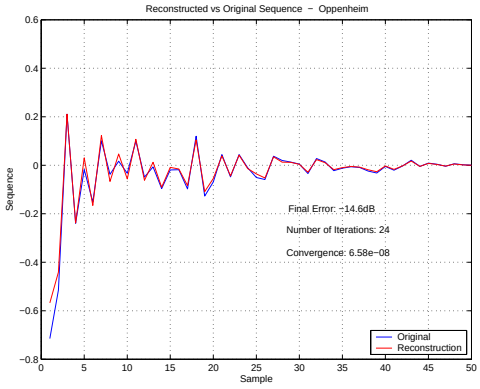
5.2.2 Short sequences

First example

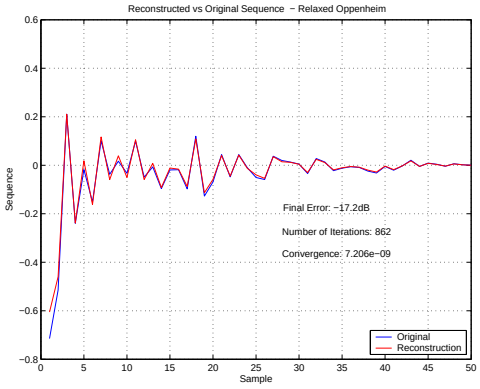
In the first experiment, we try to reconstruct a mixed phase signal x of length $N_x = 50$. Figures 5.1(a) to 5.1(c) show the reconstruction result after convergence of the algorithm. It can be seen that the relaxation variant yields a slightly better result than the basic algorithm. The algorithm operating in the cepstral domain, however, reconstructs the original sequence only roughly in form. The results differs considerably from the original.
In order to achieve the better result, the relaxation algorithm needs abundantly more iterations than the basic algorithm. A comparison of Figures 5.2(a) and 5.2(b) shows that the relaxation algorithm was able to improve the quality of reconstruction late in the iteration process.
Figure 5.2(c) shows that the error $e_{2\mathrm{A}}$ is not nonincreasing in case of the cepstral domain algorithm.

Second example

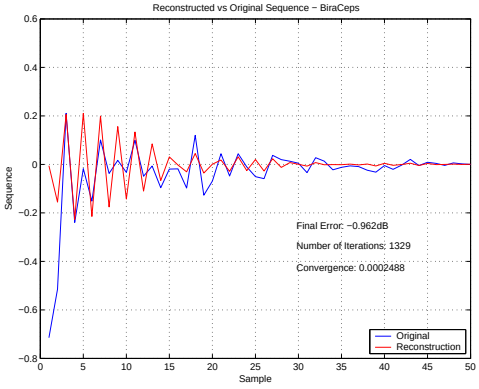
The experimental setting is similar to the previous one. The signal x is again mixed phase and of length $N_x = 60$. However, it will illuminate some other properties of the algorithms at stake. First, the basic algorithm did not converge in a reasonable amount of time. Secondly, the algorithm working in the cepstral domain yields a better result than the relaxation variant. The result is better both in the number of required iterations and in the reconstruction error, see Figure 5.3. What concerns the speed of convergence, in case of the cepstral domain algorithm, the error decays fast and steadily in the beginning, saturating later on. For the relaxation variant, the decay is not as fast and steady, but there is also no saturation, see Figure 5.4.
It is instructive to compare Figure 5.5 and to Figure 5.3. The first figure shows the cepstrum of the reconstructed signals. Although the difference between reconstruction and original is easily seen in the time domain, in the cepstral domain, the differences are not that visible in case of the cepstral domain algorithm.



(A) Basic algorithm

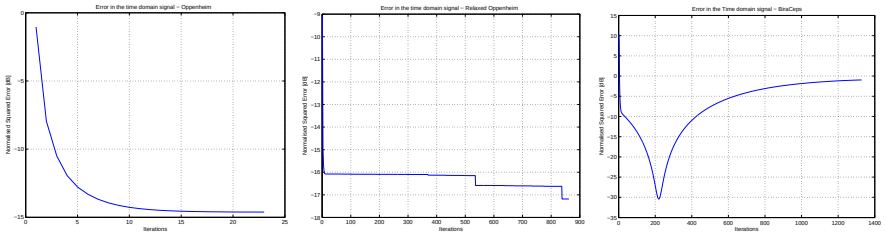


(B) Relaxation variant



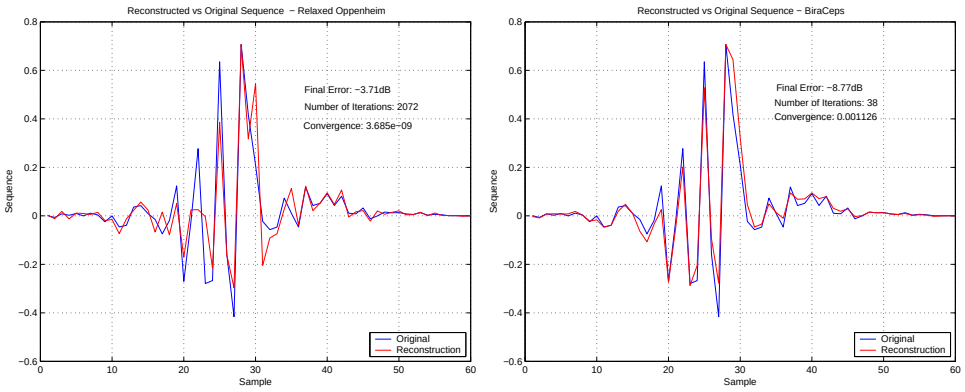
(C) Cepstral algorithm

Figure 5.1: Reconstruction from Fourier phase information only. The signal x is mixed phase and of length $N_x = 50$. Blue curve: original x, red curve: reconstruction $\hat{x}$. Note that the indicated error refers to $e_{2\mathrm{N}}$ and that "Convergence" quantifies a-posteriori convergence by calculating the AE $e_{1\mathrm{A}}$ of the angle cepstrum of $\hat{x}$. From Subfig. 5.1(c) it is seen that the cepstral algorithm reconstructs poorly in this example. Refer to Subfig. 5.2(c). The quality of reconstruction in Subfig. 5.1(b) and 5.1(a) is the same.



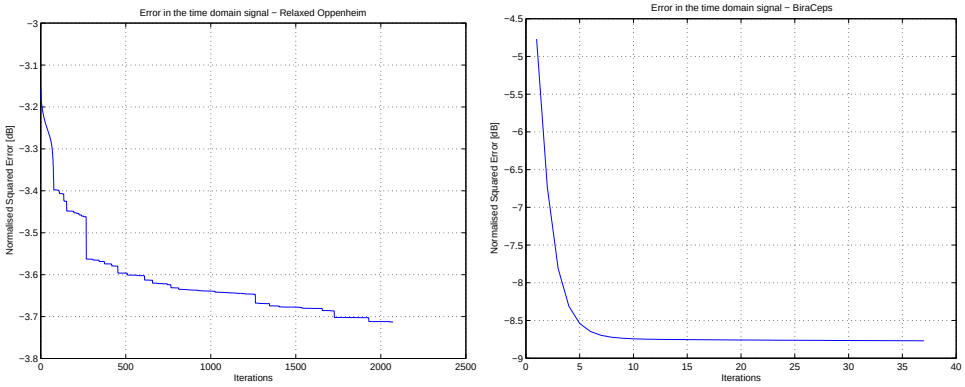
(A) Basic algorithm (B) Relaxation variant (C) Cepstral algorithm

Figure 5.2: Convergence behaviour. A comparison. The NSE e_{2N} is given in function of the number of iterations. Consider Subfig. 5.2(b) and note that the relaxation algorithm has the ability to improve reconstruction late in the iteration process. Subfig. 5.2(c) shows that the NSE made by the cepstral algorithm as used in this example is not nonincreasing.



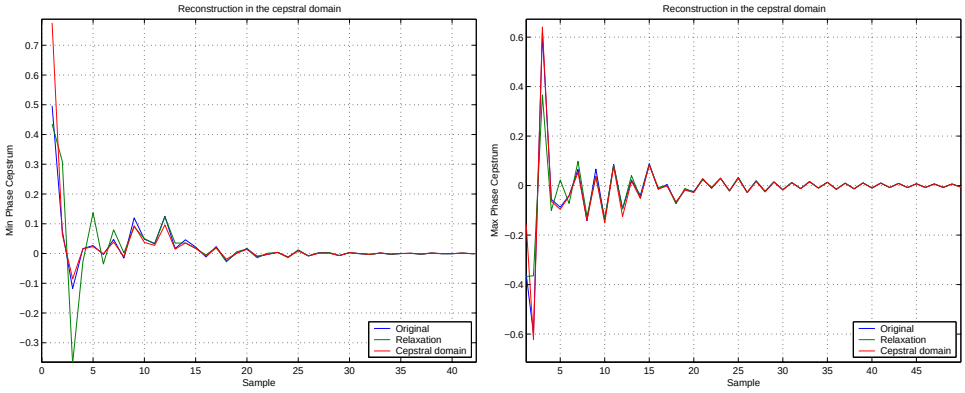
(A) Relaxation variant (B) Cepstral algorithm

Figure 5.3: Signal reconstruction from Fourier phase only. Signal x is of length $N_x = 1000$. The cepstral algorithm needs only 38 iterations to reconstruct the signal to a accuracy of $e_{2N} = -9dB$. The relaxation algorithm, however, needs more iterations for a worse result.



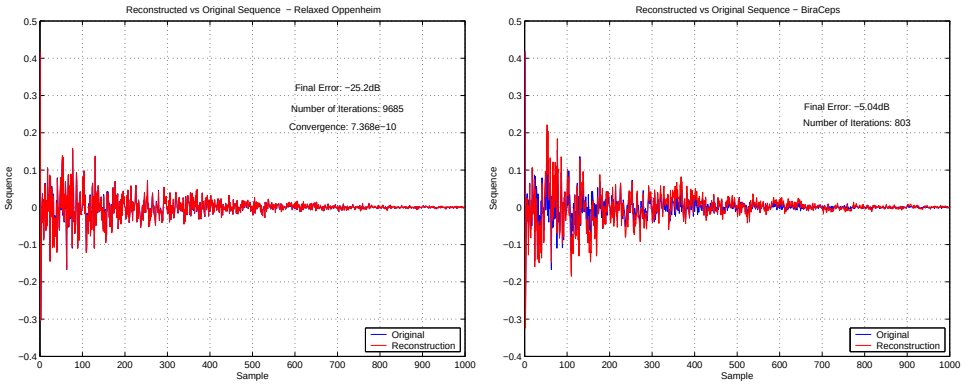
(A) Relaxation variant (B) Cepstral algorithm

Figure 5.4: Rate of convergence of the iterative reconstruction algorithms. Convergence in case of the relaxation variant is "bumpier" and slower than the cepstral algorithm. However, it does not saturate as the cepstral algorithm.



(A) Minimum phase cepstrum (B) Maximum phase cepstrum

Figure 5.5: Reconstruction from the cepstral point of view. Blue: original cepstrum c_x. Green: cepstrum of the reconstructed sequence $c_{\hat{x}}$, relaxation algorithm. Red: cepstrum of the reconstructed sequence $c_{\hat{x}}$, cepstral algorithm. The reconstruction error is both in case of the relaxation algorithm and in case of the cepstral algorithm biggest for low indices.



(A) Relaxation variant (B) Cepstral algorithm

Figure 5.6: Reconstruction from Fourier Phase information. Signal x is of length $N_x = 1000$ and mixed phase. At the cost of a larger number of iterations, the relaxation variant achieves a better reconstruction result. For the NSE e_{2N}, it is $e_{2N} = -5$ dB versus $e_{2N} = -25$ dB.

5.2.3 Long sequences

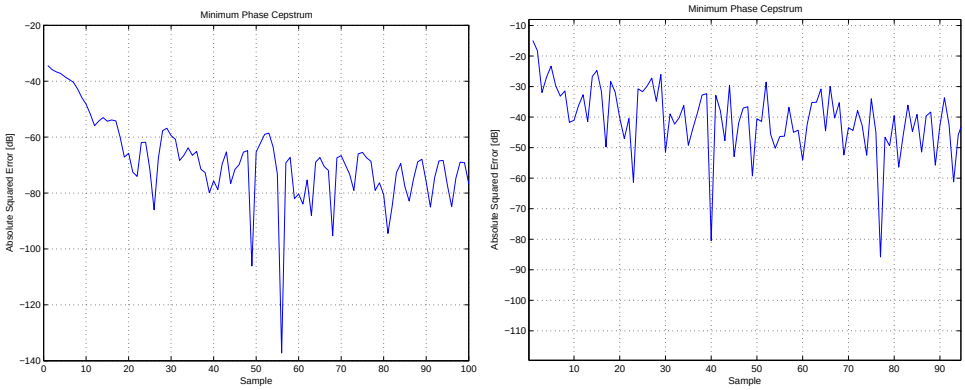
In this subsection, two examples show the capability of the algorithms to reconstruct longer mixed phase sequences from Fourier phase information only. Because of the slow convergence rate of the basic reconstruction algorithm, only its relaxation variant and the cepstral domain algorithm will be considered.

Figure 5.6 shows the reconstruction results of the cepstral domain algorithm and the relaxation variant in case of a mixed phase signal of length $N_x = 1000$. The relaxation variant achieves the better result at the expense, however, of the number of needed iterations. See Figure 5.7 for the reconstruction of the cepstrum. In Figure 5.8 and 5.9, the reconstruction results of mixed phase signals x of length $N_x = 2000$ are shown.

5.2.4 Summary

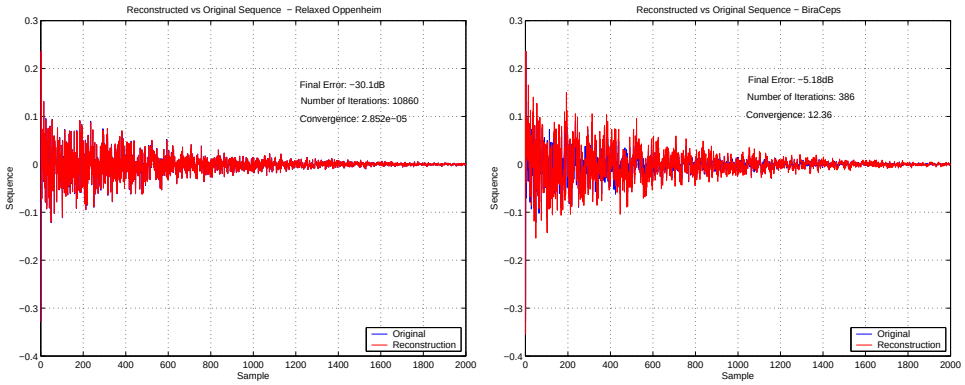
The main points of this section are listed below.

- The basic fixed point algorithm has the lowest rate of convergence while the cepstral domain algorithm might be the fastest of the three



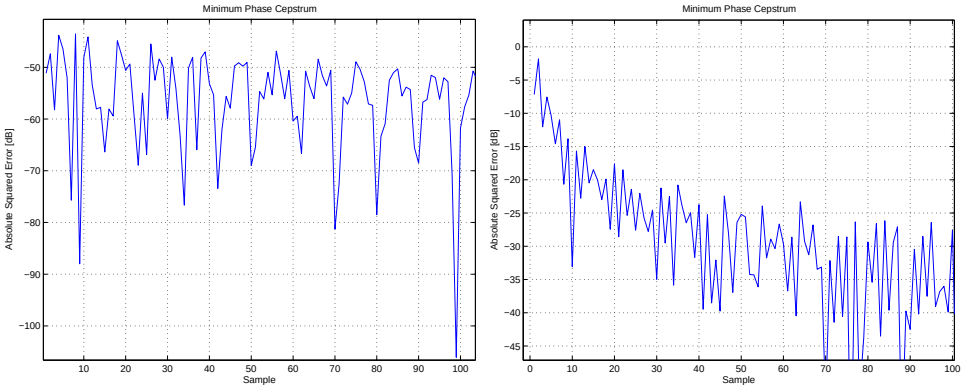
(A) Relaxation variant (B) Cepstral algorithm

Figure 5.7: Reconstruction from the cepstral point of view. In average, ASE e_{2A} equals $e_{2A} = -70$dB in case of the relaxation algorithm, while for the cepstral algorithm it is $e_{2A} = -40$dB. Note that the error e_{2A} is largest at low sample indices. Compare to Figure 5.6 to see how the error in the cepstral domain translates to the time domain.



(A) Relaxation variant (B) Cepstral algorithm

Figure 5.8: Reconstruction from Fourier Phase information. Signal x is of length $N_x = 1000$ and mixed phase. Reconstruction results correspond to the results of Figure 5.6.



(A) Relaxation variant (B) Cepstral algorithm

Figure 5.9: Reconstruction from the cepstral point of view. Signal x is of length $N_x = 2000$ and mixed phase Left: The better reconstruction result of the relaxation variant is also in the cepstrum observable.

addressed algorithms. However, the error turned out to be not always nonincreasing.

- Convergence in case of the cepstral algorithm is smooth while the relaxation variant shows a bumpier behaviour.

- Evaluating the reconstruction process from the cepstral point of view, cepstral coefficients at low indices are observed to be the last to converge.

5.3 Further insights

In this section, we will address some specific points in the behaviour of the reconstruction algorithms introduced in Section 5.1.
First, we will look at the dependency of the iterative reconstruction algorithms on their initialisation. Then, we will address the problem of aliasing in the cepstral domain algorithm. Finally, the robustness of the reconstruction algorithms is discussed.

5.3.1 Initial conditions

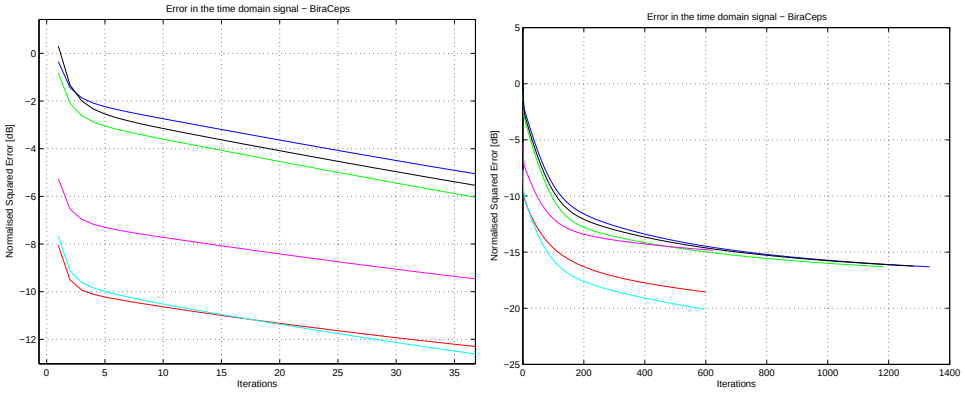
In Section 5.2, we have seen that the rate of convergence of the cepstral algorithm is fast at the beginning. With increased number of iterations, however, we have observed the convergence curve to flatten out. Here, we will investigate how much the reconstruction ability of the cepstral domain algorithm depends on its initialisation.
For that purpose, we have run the reconstruction algorithm several times for the same sequence varying the initialisation of the iterative algorithm. Figure 5.10 graphically presents the effect of the initialisation on the accuracy of reconstruction. We observe that the NSE $e_{2\mathrm{N}}$, as defined in Subsection 5.1.3 , varies in value up to $\triangle e_x = 5\mathrm{dB}$ (33%). Furthermore, we can see that the number of iterations till convergence of the algorithm varies by a factor of 2. In addition to these observations, the detail-plot of the error for the first iterations, as shown in Figure 5.10(a), suggests the importance of the initialisation for the iterative cepstral domain algorithm.

5.3.2 Aliasing

The cepstral domain algorithm reconstructs the signal from its Fourier phase information in the cepstral space. That is, the angle cepstrum is used instead of the Fourier phase. The angle cepstrum can be calculated from the Fourier phase, see (2.25), or directly from the time domain signal. In latter case, first the cepstrum of the signal is calculated, and then the angle cepstrum, see (2.26).
With reference to Section 4.1, we will discuss here the influence of aliasing on the reconstruction process when the angle cepstrum is obtained from the cepstrum of the signal.
In Section 4.1, is was noticed that the DFT-cepstrum does not correspond to the z-cepstrum, but that it is unique and invertible as it is the z-cepstrum. Stating the reconstruction problem as follows: Given $c_{\angle x}^{\mathrm{NDFT}}$, find $\hat{x}$, such that

$$c_{\angle \hat{x}}^{\mathrm{NDFT}} \quad = \quad c_{\angle x}^{\mathrm{NDFT}}$$

we will not observe any aliasing problems because of the uniqueness of the DFT-cepstrum.



(A) Convergence - In detail (B) Convergence

Figure 5.10: Dependency of the cepstral domain algorithm on its initialisation. The number of iterations needed till convergence and reconstruction accuracy e_{2N} depends strongly on the initialisation of the iterative cepstral domain algorithm. The detail-plot in Subfig. 5.10(a) shows that after a sharp decrease in the beginning, the error curves begin to flatten out. Note that both the absolute value of decrease during the period of sharp fall and also the period of fall itself is the same for all curves. This underlines the strong dependency on the initial initialisation.

5.3.3 Robustness

In Section 5.2, we presented successful reconstructions for both longer and shorter sequences. Here, we like to analyse how robust the reconstruction algorithms are, i.e. how often the algorithms succeed in their reconstruction task. We limit ourselves to the cepstral domain algorithm.
Figure 5.11 depicts the results of the following experiment. For a given signal length, here $N_x = 60$ and $N_x = 1000$, 50 sequences were reconstructed from their Fourier phase. In case of the shorter sequences, we see that the algorithm is almost always able to find the correct sequence. In average, the error e_{2N} equals $< e_{2N} = -46\text{dB} >$. But, there are big variations. These variations are less pronounced in case of $N_x = 1000$. In average, however, reconstruction is also less accurate. From Figure 5.12 we learn that not the number of iterations is critical to a successful reconstruction.
Based on these observations, we note the following:

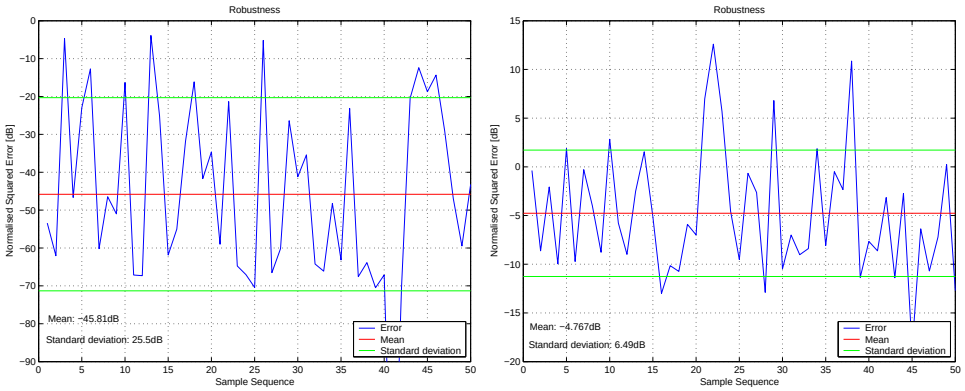
- There are signals that cannot be correctly recovered from there Fourier phase even if they are of moderate length. This can be explained by the fact that the algorithm is not able to recover zeros in reciprocal pairs. If the constellation of a signal is such that there occur just too many zeros in reciprocal pairs, the reconstruction result will be poor independent of the actual length of the signal.

- For longer sequences, reconstruction is less accurate in average. But there are also less variations. Again, this is explained by the reciprocal-pairs condition. For longer sequences, it is not sensible to consider the zeros as isolated entities. They are more likely thought to be smeared over a given region. Therefore, a reciprocal-pair situation will happen more often in average, i.e. for almost every signal at stake. This causes the reconstruction to be less accurate in mean. It accounts also for the smaller differences between the various reconstruction attempts.

- Increasing the number of iterations will thus not produce a better reconstruction result.

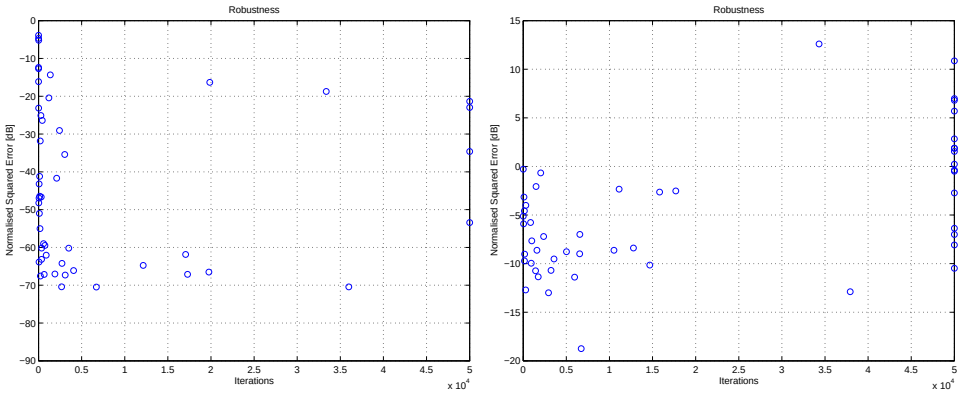
5.3.4 Summary

The main points of this section are listed below.



(A) Signals of length $N_x = 60$ (B) Signals of length $N_x = 1000$

Figure 5.11: Robustness of the cepstral domain reconstruction algorithm. Blue: NSE $e_{2\mathrm{N}}$ for the tested 50 sequences. Red: mean. Green: standard deviation. We observe that with increasing signal length, reconstruction becomes less accurate. The big standard deviation in case of $N_x = 60$, however, indicates that a bad signal constellation can make the reconstruction difficult.



(A) Signals of length $N_x = 60$ (B) Signals of length $N_x = 1000$

Figure 5.12: Convergence time and robusteness for the cepstral domain algorithm.

- The iterative cepstral domain algorithm shows a strong dependency on its initialisation values.
- At the level of the angle cepstrum, aliasing does not effect the reconstruction process.
- The iterative algorithms are not very robust. Further, they are not expected to yield especially good results for longer sequences.

5.4 Conclusion & Outlook

In this chapter, we have discussed three iterative reconstruction algorithms that are based on the theory of Chapter 3. We have seen that they differ in convergence speed and behaviour. Considering the reconstruction success from the cepstral point of view, we noted as well that cepstral coefficients at low indices are the last to converge.
The uniqueness property of the cepstrum allows to bypass aliasing problems when calculating the phase from the angle cepstrum. A robustness analysis, however, has shown that the algorithms are not very reliable. Further, their ability to reconstruct long sequences accurately is limited. Additionally, at least for the cepstral domain algorithm, reconstruction results were seen to depend on the initialisation values of the algorithm.

Further investigations could be directed towards iterative algorithms that are derived from the closed form solution of the reconstruction problem.

Part III

Cepstral Dereverberation – In Use

Chapter 6

Blind Deconvolution

This chapter is dedicated to the study of the two-channel blind deconvolution problem. The discussed algorithm uses explictly concepts that were at issue in the previous chapters. Particularly, it works in the cepstral domain in order to have direct access to the minimum and maximum phase parts of a signal. It is further based on reconstruction algorithms that find from the Fourier phase information alone the original signal.
First, a proper formulation of the two-channel blind deconvolution problem is given, then the performance of the used algorithm is discussed.

6.1 Background

In this section, we will first properly define the blind deconvolution problem. Then, a solution as proposed in [4] will be discussed. The section closes with a discussion of the limitations of the proposed solution as determined by the used tools.

6.1.1 Problem formulation

We consider the following problem which we shall call "two-channel blind deconvolution problem" or "blind deconvolution problem". Given the two measurements y_1 and y_2:

$$y_1 = h_1 * s \tag{6.1}$$

$$y_2 = h_2 * s \tag{6.2}$$

find the signal s.
For a unique solution to exist, the following assumptions are made on the channels h_1 and h_2, see [4].

A1 h_1 and h_2 are finite extent sequences whose z-transforms have no common zeros.

A2 There are no zero-pole cancellations between s and h_i, $i = 1, 2$.

A3 The channels h_i, $i = 1, 2$, have no zeros on the unit circle.

Assumption A1 is explained by the fact, that if both h_1 and h_2 have the same zero, having only access to the measurements y_i, $i = 1, 2$, it is impossible to determine whether the latter zero belongs to the channels h_i, $i = 1, 2$, or to the signal s. Assumption A2 is motivated by the fact that if zero-pole cancellations take place, the cancelled terms are not visible in the measurement. Similarly, Assumption A3 excludes the possibility that there are frequency components of s that cannot be measured.

6.1.2 Principle

The blind deconvolution problem is solved using the techniques discussed in the previous chapters. First these tools are recalled, then the concepts leading to a solution for the blind deconvolution problem are explained. The solution that is presented below was first published in [4].

Tools

The solution procedure needs the following tools:

- A tool for transforming a convolution into a summation and vice versa.
  See Section 4.2.

- A tool for extracting minimum phase and maximum phase part of a signal.
  See Section 4.3.

- A tool for reconstructing a mixed phase FIR signal from its phase.
  See Chapter 5.

Tool 1 means basically transforming a sequence into the cepstral domain. Tool 2 uses the fact that for positive indices, the cepstrum is only function of the minimum phase part of the FIR signal, while for negative indices, only of the maximum phase part. Tool 3 stands for the iterative reconstruction algorithms of Chapter 5.

Concept

The solution procedure consists of the following six steps.

Step 1 Apply tool 1.

The cepstra of the two measurements y_i, $i = 1, 2$ are calculated. Then (6.1) and (6.2) can be written in form of a summation. See Figure 6.1.

$$c_{y_1} = c_{h_1} + c_s \tag{6.3}$$
$$c_{y_2} = c_{h_2} + c_s \tag{6.4}$$

Step 2 Apply tool 2.

As the cepstral coefficients for $n > 0$, i.e. the cepstrum of the minimum phase part, are independent from the ones for $n < 0$, i.e. the cepstrum of the maximum phase part, each of the above equations can be split into two. See Figure 6.2.

$$c_{yi}^{\min} = c_{hi}^{\min} + c_s^{\min} \tag{6.5}$$
$$c_{yi}^{\max} = c_{hi}^{\max} + c_s^{\max} \tag{6.6}$$
$$\text{with } i = 1, 2$$

Step 3 Focus on h_i and eliminate c_s.

From the previous equations, we see that the cepstrum c_s can be eliminated if the two minimum phase cepstra, maximum phase cepstra respectively, are subtracted from each other. See Figure 6.3.

$$d_{\min} = c_{y_1}^{\min} - c_{y_2}^{\min} \tag{6.7}$$
$$= c_{h_1}^{\min} - c_{h_2}^{\min} \tag{6.8}$$

and

$$d_{\max} = c_{y2}^{\max} - c_{y1}^{\max} \tag{6.9}$$
$$= c_{h2}^{\max} - c_{h1}^{\max} \tag{6.10}$$

Step 4 Apply tool 3.

Imagine an artificial signal, denoted by $h_{\min}$ whose cepstrum is given by $c_{h1}^{\min}$ for $n > 0$ and $c_{h2}^{\min}$ for $n < 0$. Then $d_{\min}$ equals the difference between $c_{h1}^{\min}$ and $c_{h2}^{\min}$. In Chapter 2 it is shown that $d_{\min}$ is related to the angle cepstrum $c_{\angle_{h_{\min}}}$. From the difference $d_{\min}$ is is possible to calculate the Fourier phase of $h_{\min}$ up to a linear phase shift. Hence, by use of tool 3 the artificial signal $h_{\min}$ can be constructed up to a time shift n_0. Similar for the cepstral difference $d_{\max}$ which provides the signal $h_{\max}$. See Figure 6.4.

Step 5 Apply tool 2.

The minimum phase part of $h_{\min}$ is given by $h_1^{\min}$, the maximum phase part by $h_2^{\min}$. Similarly, $h_2^{\max}$ forms the minimum phase part of $h_{\max}$ and $h_1^{\max}$ its maximum phase part. From these components, the channels h_1 and h_2 can be built by minimum-maximum phase extraction. See Figure 6.5.

Step 6 Apply tool 1.

The unknown channels h_1 and h_2 have been identified in the previous step. Therefore, the signal s is found by deconvolution, i.e. subtraction in the cepstral domain. Building the average and going back into the time domain achieves the blind deconvolution. See Figure 6.6.

$$c_s = c_{y1} - c_{h1} \tag{6.11}$$

$$c_s = c_{y2} - c_{h2} \tag{6.12}$$

6.1.3 Theoretical limitations

The previous subsection explained the concept that solves the blind deconvolution problem. Further, it presented the needed tools. From the detailed analysis of these tools in Sections 4.2 and 4.3, respectively Chapter 5, we know about the conditions under which the tools work correctly. Zeros that are too close to the unit circle limit the use of tool 1 and 2. Zeros on the unit circle or in reciprocal pairs tool 3. Note that these conditions go hand in hand with the Assumptions A1-A2 of Subsection 6.1.1. The requirement that there are no zeros on the unit circle corresponds to Assumption A3. From the description of step 4, it can be seen that Assumption A1 corre-

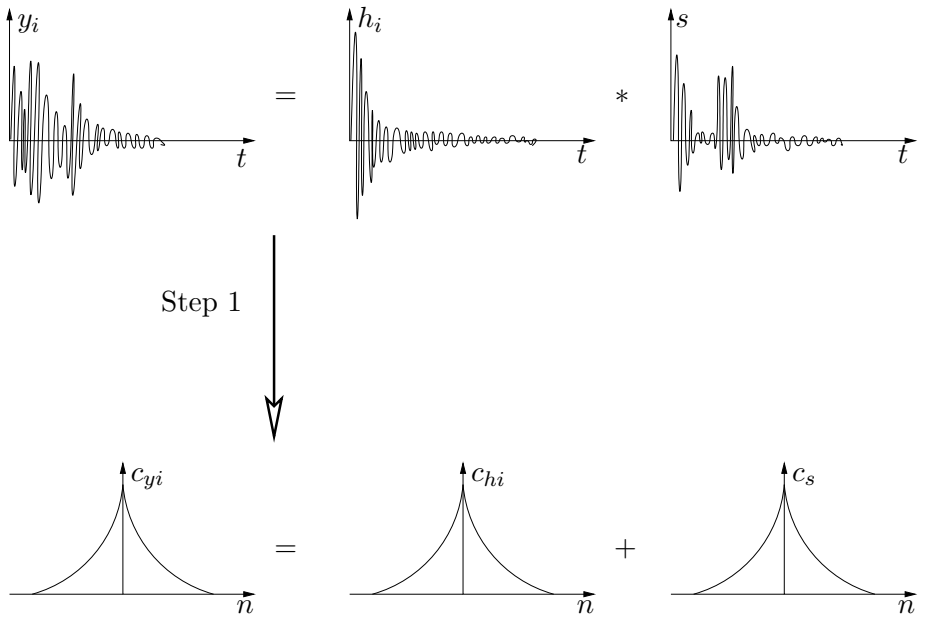


Figure 6.1: Step 1. Calculate the cepstrum of the measured signals.

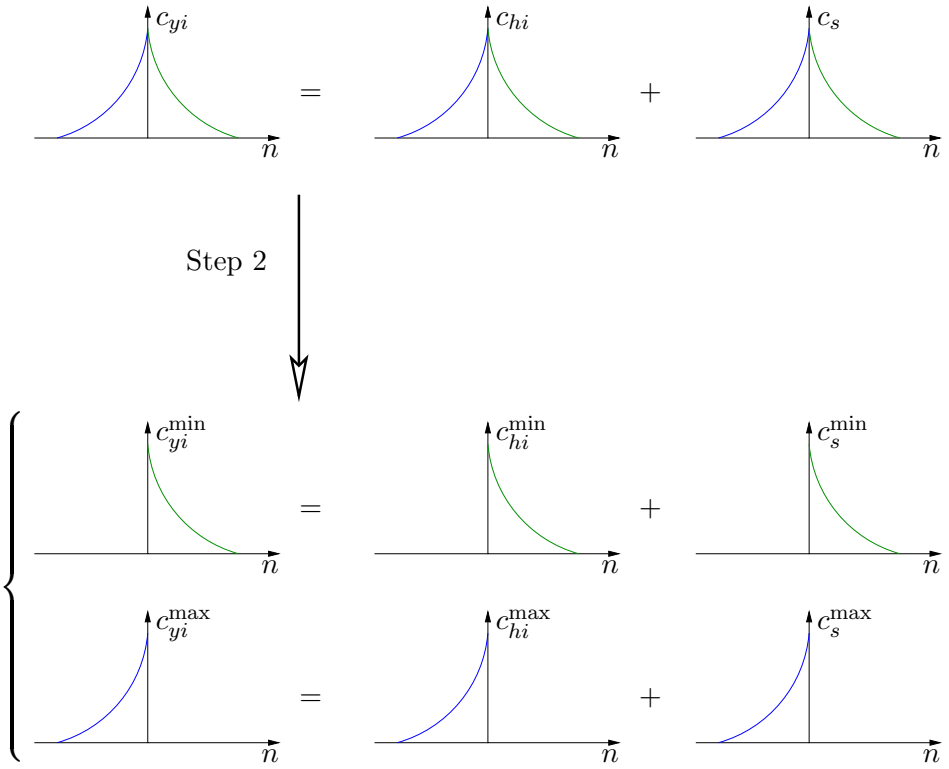


Figure 6.2: Step 2. Extract the minimum phase and the maximum phase parts.

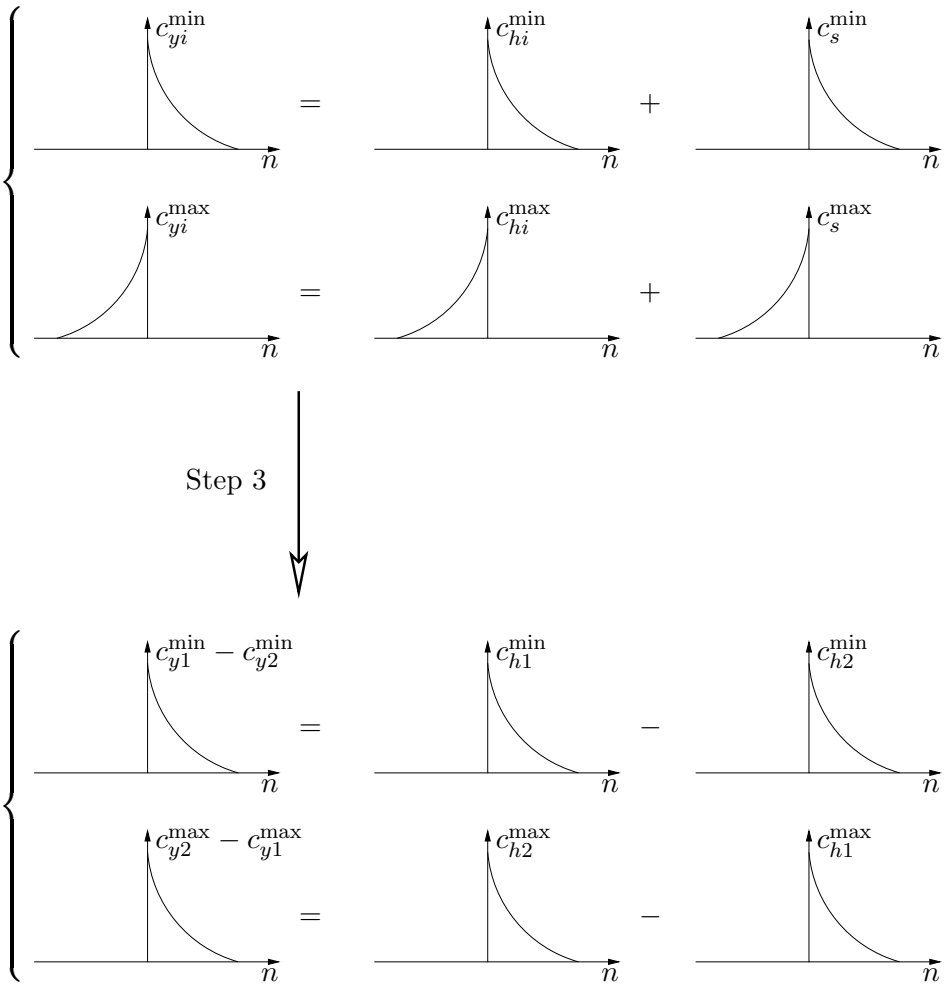


Figure 6.3: Step 3. Calculate the difference of the two minimum phase cepstra, maximum phase cepstra, respectively.

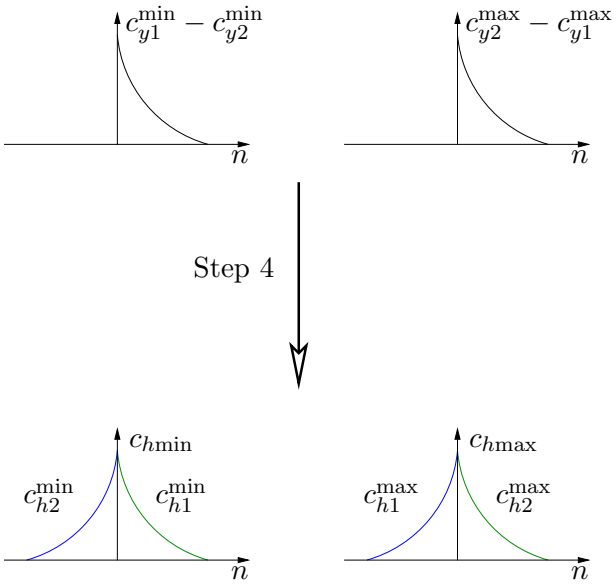


Figure 6.4: Step 4. Construct the artificial signals $h_{\min}$ and $h_{\max}$.

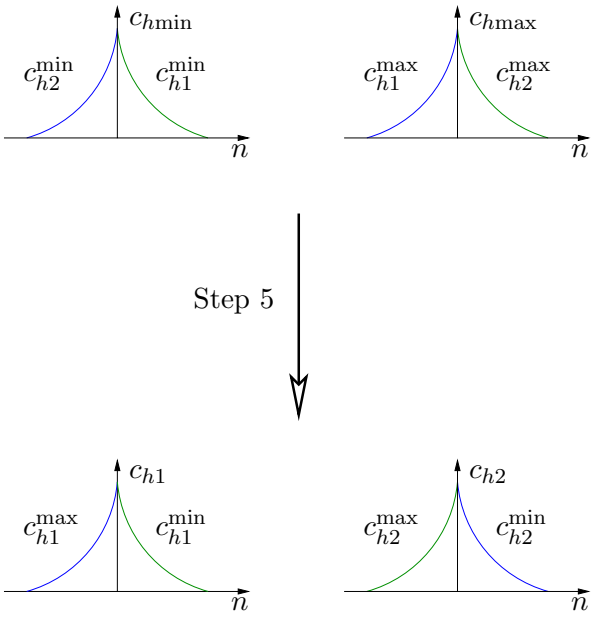


Figure 6.5: Step 5. Build the channels h_1 and h_2.

Step 6

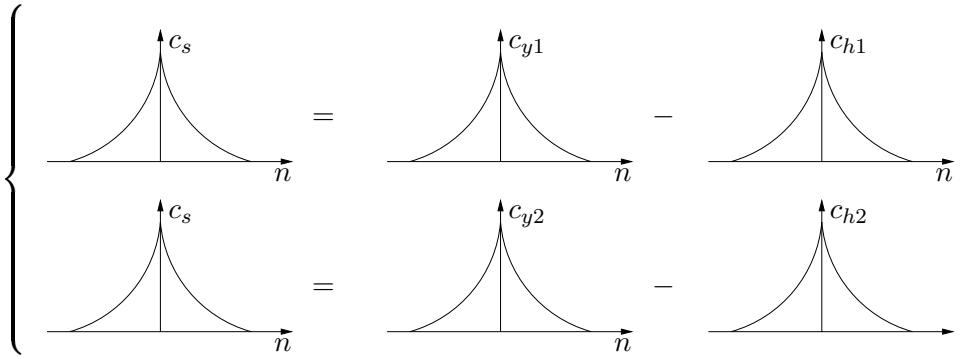


Figure 6.6: Step 6. Find s by deconvolution.

sponds to the requirement that there are no zeros in reciprocal pairs in the sequences $h_{\min}$ and $h_{\max}$.

6.1.4 Noisy variant

The problem formulation of Subsection 6.1.1 excluded noise in our measurements. Here, we like to mention how is dealt with the realistic case when noise is present.

Problem formulation

The blind deconvolution problem is then stated as follows: Given the two measurements $\tilde{y}_1$ and $\tilde{y}_2$:

$$\tilde{y}_1 = h_1 * s + w_1 \tag{6.13}$$

$$\tilde{y}_2 = h_2 * s + w2 \tag{6.14}$$

find the signal s, where w_1 and w_2 is Additiv White Gaussian Noise (AWGN).

Solution procedure

Statistical methods are introduced to deal with the AWGN. The basic idea is to calculate first the third-order cumulant of the noisy measurements $\tilde{y}_1$ and $\tilde{y}_2$ in order to suppress the AWGN. Then, to transform the third-order cumulants into the cepstral domain and to derive from these expressions the cepstra c_{y_1} and c_{y_2}. The remaining steps in the blind deconvolution

procedure are the same as in Section 6.1.2. Therefore, only the way the cepstra c_{y_1} and c_{y_2} are calculated is modified when AWGN is taken into account. Refer to [5], [15] and [8].

6.1.5 Summary

The main points of this section are listed below.

- The knowledge of how to transform a convolution into a summation, of how to extract minimum and maximum phase parts of a signal and of how to reconstruct a signal from its Fourier phase alone can be used to solve the stated blind deconvolution problem.
- Except for calculating the cepstrum with higher-order statistics, the solution procedure remains the same when AWGN is taken into account.
- The limitations posed by the used tools are in line with the conditions for a unique solution to exist for the blind deconvolution problem.

6.2 With a-priori knowledge

In this section, we will show how the blind deconvolution procedure of Section 6.1 works in practice. First, the parameters that are considered as known are detailed. Then, experimental results are shown. The presented results aim at showing that the blind deconvolution algorithm *can* work. See Section 6.3 for a robustness discussion.

6.2.1 Assumptions

Consider step 4 in the blind deconvolution procedure, where the time domain signals $h_{\min}$ and $h_{\max}$ are constructed from their angle cepstra. In Chapter 5, where we explained the used reconstruction techniques, we assumed the length of the signal as known. However, we do not know the length of $h_{\min}$ and $h_{\max}$ in step 4. When we apply the reconstruction techniques of Chapter 5 in the blind deconvolution problem, we have thus to estimate the lengths of the signals $h_{\min}$ and $h_{\max}$. Section 6.3 discusses that case, here, we assume the lengths as known.
Furthermore, recall that linear phase shift information is lost in the cepstral

domain. Thus, it is only possible to obtain the Fourier phase up to a linear phase shift from the angle cepstrum. Here, we consider that phase shift as known. In Section 6.3, we forgo this assumption.
Assumptions made in this section:

A1 The signal lengths of $h_{\min}$ and $h_{\max}$ are known.

A2 The time shifts n_0 of $h_{\min}$ and $h_{\max}$ are known.

6.2.2 Experiments

First, we will illustrate the blind deconvolution procedure outlined in Subsection 6.1.2 reconstructing a mixed phase sequence s of length $N_s = 30$. The channels h_1 and h_2 of length $N_{h_1} = N_{h_2} = 30$ have been simulated. Then, dereverberation examples of real speech are shown.

Mixed phase signal reconstruction

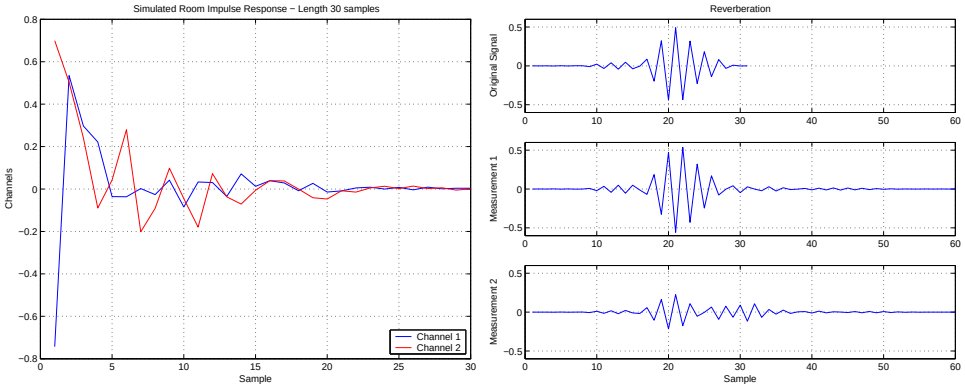
Figure 6.7(a) shows the channels h_1 and h_2, Figure 6.7(b) the original signal s and the two measurements y_1 and y_2. The channels h_1 and h_2, as well as the signal s are all mixed phase signals.

Step 1-3 Obtain the angle cepstra of $h_{\max}$ and $h_{\min}$.
The first part of the blind deconvolution algorithm aims at laying the foundations for the blind channel identification. For that, the angle cepstra of $h_{\max}$ and $h_{\min}$ are extracted from the measurements y_1 and y_2. Figure 6.8 shows the Absolute Squared Error (ASE) $e_{2\mathrm{A}}$ made in the extraction process where $e_{2\mathrm{A}}$ is defined in (5.5) as

$$e_{2\mathrm{A}} \quad = \quad 10\log_{10}\left[(c-\hat{c})^2\right]$$

Step 4 Reconstruct the sequences $h_{\min}$ and $h_{\max}$.
Here, the core part of the channel identification is done. From their Fourier phase information, the signals $h_{\min}$, respectively $h_{\max}$, are reconstructed. Figure 6.9 shows the constructed sequences $h_{\max}$ and $h_{\min}$. Recall that the reconstruction error $e_{2\mathrm{N}}$ is defined (4.43) as

$$e_{2\mathrm{N}} \quad = \quad 10\log_{10}\left[\frac{||x-\hat{x}||^2}{||x||^2}\right]$$



(A) Channels (B) Reverberation

Figure 6.7: Effect of reverberation. The channels h_1 and h_2 are simulated RIRs. They are mixed phase signals, and their length was chosen to be $N_{h_i} = 30$, $i = 1, 2$. Signal s is a mixed phase signal of length $N_s = 30$.

Step 5 Build the channels h_1 and h_2.

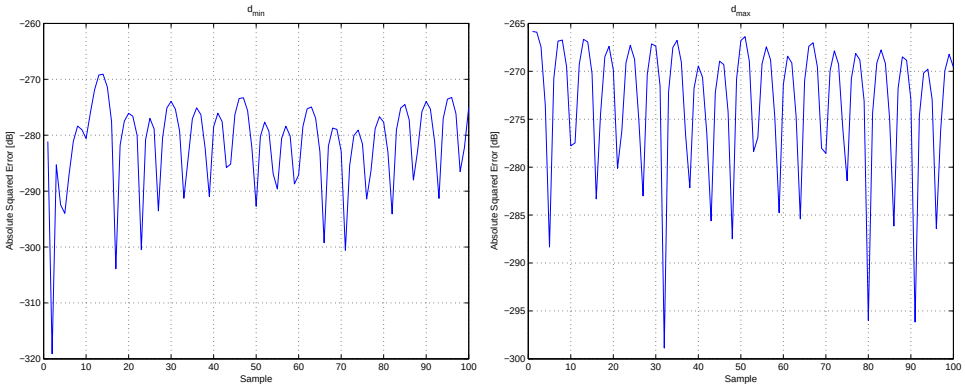
By use of minimum-maximum phase extraction techniques, the channel h_1 and h_2 can be built from $h_{\min}$ and $h_{\max}$. Figure 6.10 shows the reconstruction result.

Step 6 Determine the original signal s.

From the measurements y_i, $i = 1, 2$ and the reconstructed channels h_i, $i = 1, 2$, the signal s is obtained by deconvolution. Figure 6.2.2 shows the reconstructed mixed phase signal $\hat{s}$.

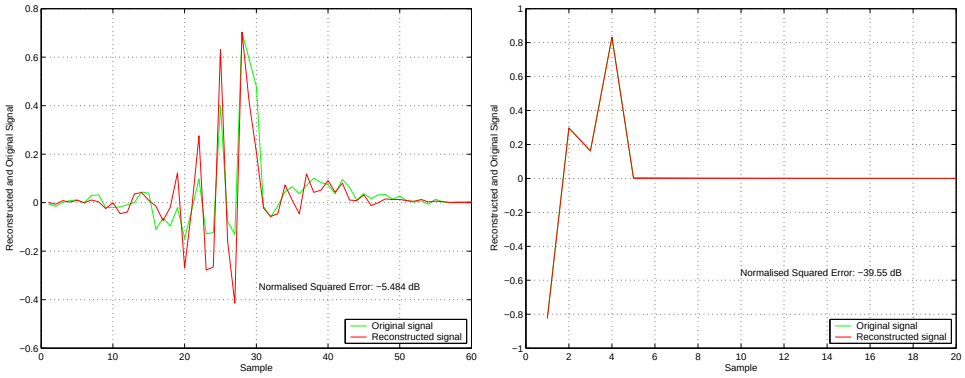
Real speech

This example illustrates dereverberation of real speech. The channels h_1 and h_2 are shown in Figure 6.12. Figure 6.13 shows the original speech signal and the reverberated measurements. Figure 6.14 the reconstruction result. It seen that the reconstruction results are similar to the results when for the same channels a much short speech signal was used, see Figure 6.10. It is possible to deduce that channel estimation is a critical part in the blind deconvolution algorithm.



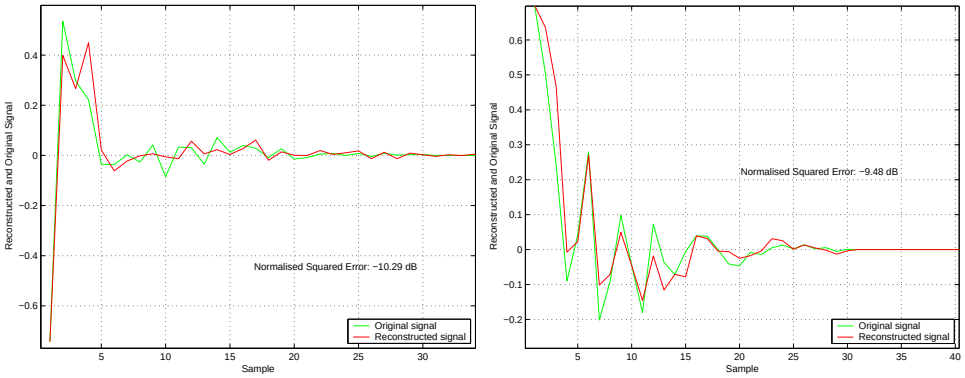
(A) Error in the angle cepstrum of $h_{\min}$ (B) Error in the angle cepstrum of $h_{\max}$

Figure 6.8: Accuracy in the minimum-maximum phase extraction process. The plotted error is the ASE $e_{1\mathrm{A}}$ of the extracted angle cepstrum. Hence, in that example, the extraction process is quite precise.



(A) Artificially built channel $h_{\min}$ (B) Artificially built channel $h_{\max}$

Figure 6.9: Reconstruction result of $h_{\min}$ and $h_{\max}$. Blue: original. Red: reconstructed. The depicted error is the NSE $e_{2\mathrm{N}}$. It is seen that the signal $h_{\max}$ is much simpler than its antagonist $h_{\min}$. Consistently, reconstruction is done much better for the former signal. The structure of the two signals $h_{\min}$ and $h_{\max}$ suggests that the minimum phase part dominates the maximum phase part for both the channel h_1 and h_2.



(A) Channel h_1 (B) Channel h_2

Figure 6.10: Blindly identified channels h_1 and h_2. The error refers to the NSE e_{2N} of the estimated channels.

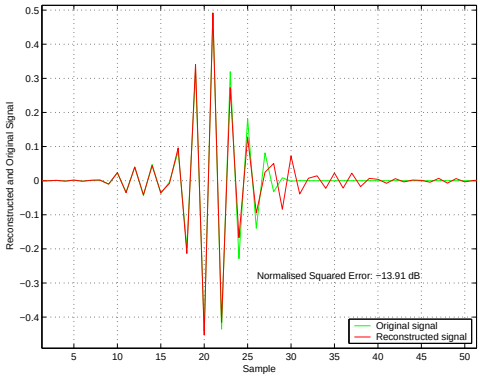
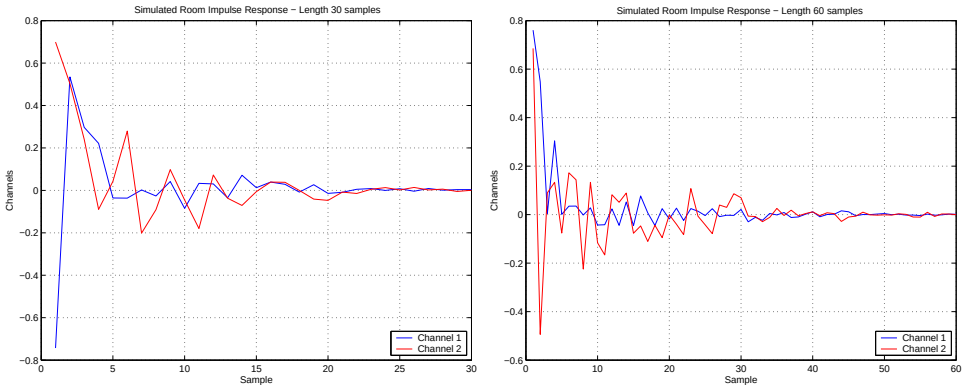


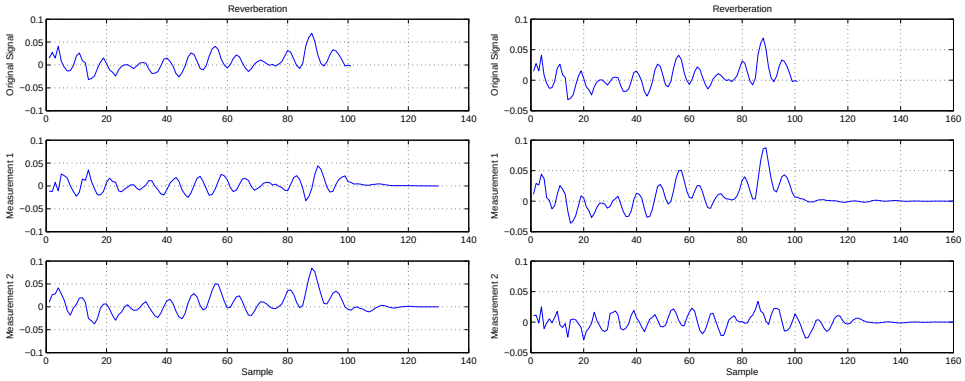
Figure 6.11: Blind deconvolution result. Green : Original sequence s. Red : Reconstruction $\hat{s}$. The error refers to the NSE $e_{2\mathrm{N}}$.



(A) Channels of length $N_{h_i} = 30$

(B) Channels of length $N_{h_i} = 60$

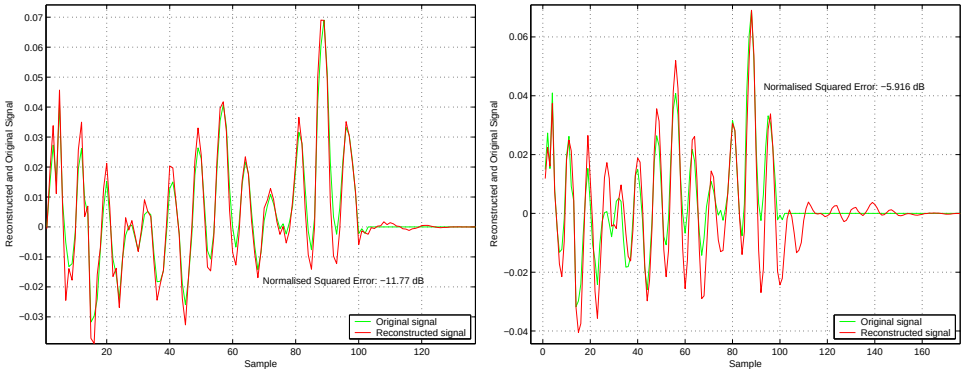
Figure 6.12: Room impulse responses. Figures 6.12 and 6.14 show how the blind deconvolution scheme works for the two cases, $N_{h_i} = 30$, respectively $N_{h_i} = 60$.



(A) Reverberation caused by the channels of length $N_{h_i} = 30$

(B) Reverberation caused by the channels of length $N_{h_i} = 60$

Figure 6.13: Original signal and the reverberated measurements.



(A) Reconstruction of speech dereverberated by channels of length $N_{h_i} = 30$

(B) Reconstruction of speech dereverberated by channels of length $N_{h_i} = 60$

Figure 6.14: Original signal s and its blind reconstruction $\hat{s}$. Note that reconstruction is not very accurate at the tail of the signal. Hoewever, even after doubling the channel length, the signal is well reconstructed. Increasing the signal complexity has not changed the reconstruction results.

6.2.3 Summary

The main points of this section are listed below.

- The blind deconvolution algorithm works for the presented examples.
- Even for not perfect identification of the channels, the algorithm achieves successful deconvolution results.
- Channel complexity is a dominating factor that limits the deconvolution ability of the algorithm.

6.3 Totally blind

Here, we relax the assumptions of the previous section. That is, we consider the case when do not have any a-priori knowledge about the reverberation channels. Only the reverberated measurements are available.
First, we will show how the constraints of Section 6.2 are resolved. Afterwards, we will compare the reconstruction results obtained under these

conditions to the results from the last section. Then, we will comment on the performance of the algorithm.

6.3.1 Resolving constraints

In Section 6.2, we made the following assumptions at the identification of the signals $h_{\min}$ and $h_{\max}$, respectively the channels h_1 and h_2:

A1 The signal lengths of $h_{\min}$ and $h_{\max}$ are known.

A2 Linear phase components of the Fourier phase of $h_{\min}$ and $h_{\max}$ are known.

In [6] and [4] it is proposed to iterate the two quantities till the reconstruction algorithm converges. This has been tested at the following examples where the same channels and speech signals were used as in Section 6.2 to make an easy comparison possible.

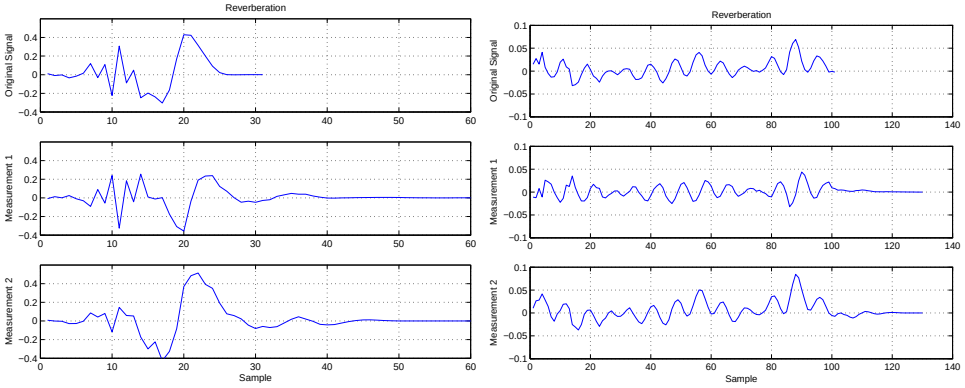
6.3.2 Experiments

In accordance with Section 6.2 we take for the channels h_1 and h_2 the signals shown in Figure 6.7(a). The source signal s was first chosen to be a $N_s = 30$ samples long mixed phase signal, then real speech. Figure 6.15 shows the orignal signal s and the reverberated measurements y_1 and y_2. Figure 6.16 shows the reconstruction result $\hat{s}$. We note that for these examples, the results obtained by deconvolution under relaxed conditions are comparable to the results of Section 6.2.

6.3.3 Practical limitations

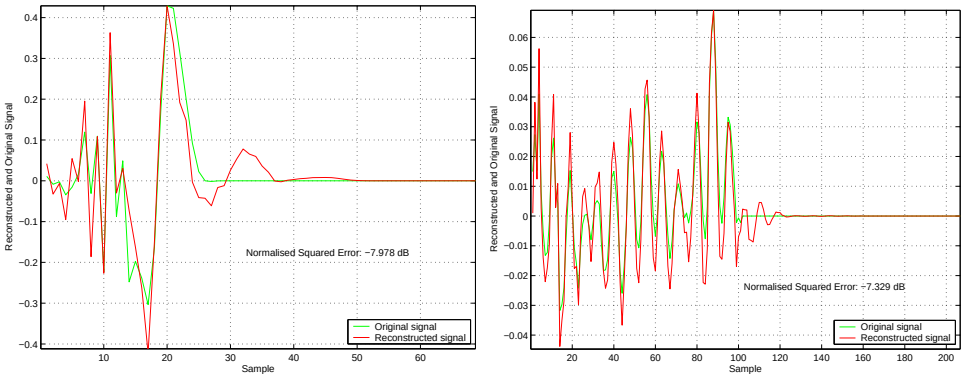
The solution procedure of Subsection 6.1.2 uses three different tools to achieve blind deconvolution. Therefore, their individual limitations, mixed together, will account for the overall performance limits of the blind deconvolution algorithm. Refer to the results established in the previous chapters for a detailed discussion of the single tools. Here, we will discuss their interaction.

Besides the limitations introduced by the mentioned tools, we shall comment on the need to to estimate the lengths of the signals $h_{\min}$ and $h_{\max}$, as well as their Fourier phase shifts. As the proposed iteration causes an



(A) Simulated signal (B) Real speech

Figure 6.15: Reverberated measurements and the original signals.



(A) Simulated signal (B) Real speech

Figure 6.16: Blind deconvolution results. In comparison with the previous section where we have assumed some a-priori knowledge we see that not knowing the time shift and the signal length has no influence on reconstruction accuracy in this example.

important increase in processing time, we consider it as an excessively practical limitation. The individual limitations are driven by the positions of the zeros of the involved signals. Zeros too close to the unit circle cause aliasing to take effect when the DFT-cepstrum is calculated, zeros in reciprocal pairs will cause the reconstruction algorithms of Chapter 5 to work poorly.
We noted in Chapter 5, that aliasing does not affect reconstruction of a signal from its Fourier phase when it occurs at the level of the angle cepstrum. From Chapter 4, however, we know that aliasing can flaw the minimum-maximum phase separation. In that case, the inputs $h_{\min}$ and $h_{\max}$ to the reconstruction algorithms are falsified. Indeed, they will be of the eye-catching form of the signals depicted in Chapter 4. Hence, even if these signals can be correctly constructed from their phase, the blind deconvolution result would be wrong. Assuming now that the inputs $h_{\min}$ and $h_{\max}$ are correct, blind deconvolution results will nevertheless be poor if the structure of these signals is such that there are a lot of reciprocal zeros present. Hence, the limitations of the used tools manifest themselves additively in the blind deconvolution procedure.

6.3.4 Summary

The main points of this section are listed below.

- The blind deconvolution algorithm works for the presented examples.
- Not knowing signal lengths and time delays of $h_{\min}$ and $h_{\max}$ increases processing time, but it does not conceptionally limit blind deconvolution.
- Each of the individual limitations of the used tools fully act on the deconvolution procedure.

6.4 Conclusion & Outlook

In this chapter, we applied the gathered tools of the previous chapters to the two-channel deconvolution problem. We have seen that, if the conditions that the tools impose are met, blind deconvolution as presented in this chapter works. Although processing time might be long. In turn, we saw that in case they are not met, deconvolution results become poor. Note that

especially for long signals, the imposed conditions are likely to be violated. First, aliasing will cause wrong channels to be reconstructed. Secondly, reconstruction will not be precise.

As the imposed conditions go hand in hand with the conditions for a unique solution of the deconvolution problem, we conclude that the conditions are not too restrictive, but that their violation has a impact too big on the deconvolution result to be tolerated. In other words, if the conditions for a unique solution are not met, we do not obtain a solution that is close to the original but a false one.

Further investigations should therefore aim at finding more robust methods. More sophisticated ideas like cepstral prediction could be used together with iterative reconstruction algorithms that are based on the closed form solution of Chapter 3.

Conclusion

This document aims at evaluating the performance of a particular algorithm that solves the two-channel blind deconvolution problem in the cepstral domain. Viewing the algorithm as a scheme where a few self-contained tools are applied severals times in succession, performance of the algorithm is ascribed to the individual characteristics of the used tools and to their interaction. Hence, after identification of this self-contained tools, we have analysed each them separately. This approach should make a clear understanding of the addressed algorithm possible and, additionally, reveal useful tools for other applications.

The algorithm uses the following tools: Cepstral transformation, minimum-maximum phase signal extraction and reconstruction of mixed phase signals from their Fourier phase information only. When literature has not given the conditions when these tools fail to work properly, we have localised them. Indeed, we have shown that in digital signal processing, aliasing flaws the cepstral transformation and the minimum-maximum phase signal extraction process. Moreover, we have found that the conditions that cause the three tools not to work as desired, i.e. if the involved signals have zeros too close to the unit circle or if they appear in reciprocal pairs, are linked to the conditions for the blind deconvolution problem to have a unique solution. Therefore, we judge the constraints on the use of these tools not to be too restrictive. Experimental results and analytical analysis, however, shows that a violation of the conditions has often a impact too big on the tools to work correctly. Furthermore, we have pointed out that each of their individual limitations enters fully into overall performance of the addressed cepstral dereverberation algorithm.

As the used tools are easily brought out of balance by violation of one of the constraints, the use of the particular blind deconvolution algorithm is limited to situation where the conditions are fully verified. In this cases, the

algorithm achieves good results. If the deconvolution problem involves long sequences, however, the condition not to have zeros near the unit circle or in reciprocal terms is likely to be violated. That the algorithm fully reflects the sensity of the used tools to the required conditions is its major drawback.

The discussed blind deconvolution algorithm does not rely on any fancy method to do dereverberation, it merely takes advantage of the definition of the cepstrum and a classical result for signal reconstruction from Fourier phase information only. Deconvolution results would be much improved if an appropriate way of dealing with the influence of aliasing in the algorithm were found. A more robust variant of the algorithm that reconstructs a signal from Fourier phase information only would also be convenient. Conceptually, the cepstrum should be viewed as a basic tool which is particular apt to deconvolution because in the cepstral domain mixed phase signals appear demixed and convolution passes into a summation. Viewing thus the cepstrum not as a high-end tool but as a basic instrument, some more sophisticated concepts such prediction theory could additionally be applied.

Bibliography

[1] J. M. Rouvaen A. Djebbari. An algorithm for signal reconstruction from its phase based upon least square criteria. *Signal Processing*, 79, 67-72, 1999.

[2] M. H. Hayes A. V. Oppenheim. Signal Reconstruction from Phase or Magnitude. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-28, No. 6, 1980.

[3] A. P. Petropulu C. L. Nikias. Signal Reconstruction from the Phase of the Bispectrum. *IEEE Transactions on Signal Processing*, 40, No. 3, 1992.

[4] A. P. Petropulu C. L. Nikias. Blind Deconvolution using signal reconstruction from partial higher order cepstral information. *IEEE Transactions on Signal Processing*, 41, No. 6, 1993.

[5] D. H. Brooks C. L. Nikias. The cross-bispectrum: Properties and applications for signal reconstruction and system identification.

[6] S. Subramaniam C. Wendt, A. P. Petropulu. Cepstrum-Based Deconvolution for Speech Dereverberation. *IEEE Transactions on Speech and Audio Processing*, 4, No. 5, 1996.

[7] A. K. Nabelek D. Mason. Effect of noise and reverberation on binaural and monaural word identification by subject with various audiograms. *Journal of Speech and Hearing Research*, 24:375-383, 1981.

[8] S. Haykin. *Advances in Spectrum Analysis and Array Processing, Vol. 1.* Prentice Hall, 1991.

[9] S. Haykin. *Adaptive Filter Theory.* Prentice Hall, 1996.

[10] S. Haykin. *Unsupervised Adaptive Filtering*. John Wiley & Sons, 2000.

[11] S. T.Neely J. B. Allen. Invertibility of a room impulse respons. *Journal of the Acoustical Society of America*, 66 (1), 1997.

[12] T. F. Quatieri V. T. Tom J. H. McClellan, M. H. Hayes. Convergence of Iterative Nonexpansive Signal Reconstruction Algorithms. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-29, No. 5, 1981.

[13] A. V. Oppenheim J. S. Lim, M. H. Hayes. Iterative procedures for signal reconstruction from fourier transform phase. *Optical Engineering*, 21, No. 1, 1982.

[14] A. C. Neumann L. Eisenberg. Evaluation of a dereverberation technique. *Journal of Communication Disorders*, 24, 211-221, 1991.

[15] C. L. Nikias M. R. Raghuveer. Bispectrum Estimation: A Digital Signal Processing Framework. *Proceedings of the IEEE*.

[16] S. K. Mitra. *Digital Signal Processing - A Computer Based Approach*. McGraw-Hill, 2001.

[17] D. G. Childers R. C. Kemerait, D. P. Skinner. The Cepstrum: A Guide to Processing. In *Proceedings of the IEEE*, October 1977.

[18] A. V. Oppenheim R. W. Schafer. Homomorphic Analysis of Speech. *IEEE Transactions on Audio and Electroacoustics*, AU-16, No. 2, 1968.

[19] L. R. Rabiner R. W. Schafer. *Digital Processing of Speech Signals*. Prentice Hall, 1978.

[20] H. Reinhard. *Elément de mathématiques du signal*. Dunod, 1997.

[21] R. W. Schafer. *Speech Analysis*. IEEE Press, 1979.

[22] R. W. Schafer A. V. Oppenheim T. G. Stockham, JR. Nonlinear Filtering of Multiplied and Convolved Signals. *IEEE Transactions on Audio and Electroacoustics*, AU-16, No. 3, 1968.

[23] W. A. Yost. *Fundamentals of hearing–An Introduction*. Academic Press, 1995.

Appendix A

The Cepstrum

All additional information with regards the cepstrum is centralised in this chapter.

A.1 Chapter 2

Here, we like to present furher identities that the cepstrum, as defined in Chapter 2, fullfills.

A.1.1 Alternative definition

In 2.16, the definition of the cepstrum c_x of x is given. Here we like to give an alternative expression.
Integrating 2.16 by parts and using the fact that X is a conjugate symmetric, periodic function in Ω, we obtain for $n \neq 0$:

$$c_x[n] = \frac{-1}{2\pi i n} \oint_C z \frac{1}{X(z)} \frac{dX(z)}{dz} z^{n-1} dz \tag{A.1}$$

$$= \frac{-1}{n} \mathcal{Z}^{-1} \left[z \frac{1}{S(z)} \frac{dS(z)}{dz} \right] \tag{A.2}$$

and for $n = 0$

$$c_x[0] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log \left[X(e^{i\Omega}) \right] \tag{A.3}$$

.

A.1.2 Implicit relation

From the above equation or from the following calculation, an implicit relation between c_x and x is found. In case of purely minimum phase sequences, respectively maximum phase sequences, this relation defines recursively the cepstrum. From the definition of $\hat{X}$, i.e. $\hat{X} = \log[X]$, we obtain:

$$\frac{d\hat{X}(z)}{dz} = \frac{1}{X(z)}\frac{dX(z)}{dz} \tag{A.4}$$

$$X(z)\frac{d\hat{X}(z)}{dz} = \frac{dX(z)}{dz} \tag{A.5}$$

$$-zX(z)\frac{d\hat{X}(z)}{dz} = -z\frac{dX(z)}{dz} \tag{A.6}$$

Taking the inverse Z-transform, we get

$$(nc_x[n]) * x[n] = nx[n] \tag{A.7}$$

which represents a implicit relation between $x[n]$ and $c_x[n]$.

Minimum phase

With the convention of Chapter 2, we know that for x minimum phase, i.e. $x = x^{\min}$, both $x^{\min}[n]$ and $c_x^{\min}[n]$ zero for $n < 0$. In that case we can explicitly solve for $c_x^{\min}[n]$ in equation A.7

$$c_x^{\min}[n] = \begin{cases} \log[x[0] & n = 0 \\ \frac{x[n]}{x[0]} - \sum_{k=1}^{n-1} \frac{k}{n} c_x^{\min}[k] \frac{x[n-k]}{0} & n > 0 \end{cases} \tag{A.8}$$

The result shows that the transformation $\mathcal{D}[\,]$ is a realisable transformation, e.g. the value of the cepstrum at n depends only on the inputs till $n-1$.

Maximum phase

Similar to the above case, there is an implicit relation for $c_x^{\max}$

$$c_x^{\max}[n] = \begin{cases} \log[x[0] & n = 0 \\ \frac{x[n]}{x[0]} - \sum_{k=n+1}^{-1} \frac{k}{n} c_x[k] \frac{x[n-k]}{x0} & n < 0 \end{cases} \tag{A.9}$$

A.2 Chapter 4

Explicit calculus shows the influence of aliasing in the test-deconvolution result $\hat{x}$

$$\hat{x} = x^{\min} * x^{\max} \tag{A.10}$$

of Subsection 4.3.

The goal of this section is to find an analytical expression for the reconstructed sequence $\hat{x}$. The minimum maximum phase separation scheme consists of the following steps:

1. Calculate the cepstrum c_x^{NDFT} of the mixed phase sequence x.
2. Extract the minimum and maximum phase cepstra $c_{\min}^{\mathrm{NDFT}}$ and $c_{\max}^{\mathrm{NDFT}}$.
3. Transform them separately back into the time domain to obtain $x_{\max}^{\mathrm{NDFT}}$ and $x_{\min}^{\mathrm{NDFT}}$.

We will now show analytically what happens when $x_{\max}^{\mathrm{NDFT}}$ is convolved with $x_{\min}^{\mathrm{NDFT}}$.

The following convention is used:

$$\begin{aligned} \mathcal{DFT}[x] &= \sum_{n=0}^{N-1} x[n] W_N^{kn} \\ \mathcal{IDFT}[X] &= \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-kn} \end{aligned}$$

Where

$$W_N = e^{-i\frac{2\pi}{N}}$$

And c_x^{NDFT} given by

$$c_x^{\mathrm{NDFT}} = \sum_m c_x[n + mN] \tag{A.11}$$

Minimum phase

Extraction

$$c_{\min}^{\mathrm{NDFT}} = w[n] c_x^{\mathrm{NDFT}}[n] \tag{A.12}$$

Where

$$w[n] = \begin{cases} 1 & 0 \le n \le \lceil N/2 \rceil - 1 \\ 0 & \text{else} \end{cases} \tag{A.13}$$

Inverse

First the N-DFT of $c_{\min}^{\mathrm{NDFT}}$ is taken:

$$\hat{X}_{\min}^{\mathrm{NDFT}}[k] = \sum_{n=0}^{N-1} c_{\min}^{\mathrm{NDFT}}[n] W_N^{kn} \tag{A.14}$$

$$= \sum_{n=0}^{\lceil N/2 \rceil - 1} c_x^{\mathrm{NDFT}}[n] W_N^{kn} \tag{A.15}$$

$$= \sum_{n=0}^{\lceil N/2 \rceil - 1} \sum_{m} c_x[n + mN] W_N^{kn} \tag{A.16}$$

$$= \underbrace{\sum_{n=0}^{\lceil N/2 \rceil - 1} c_x[n] W_N^{kn}}_{\hat{X}_{\min}} + \underbrace{\sum_{n=0}^{\lceil N/2 \rceil - 1} \sum_{m \neq 0} c_x[n + mN] W_N^{kn}}_{\hat{X}_{\min}^{\mathrm{res}}} \tag{A.17}$$

Then the exponential:

$$X_{\min}^{\mathrm{NDFT}}[k] = \exp\left[\hat{X}_{\min} + \hat{X}_{\min}^{\mathrm{res}}\right] \tag{A.18}$$

$$= \exp\left[\hat{X}_{\min}\right] \exp\left[\hat{X}_{\min}^{\mathrm{res}}\right] \tag{A.19}$$

$$= X_{\min}[k] \exp\left[\hat{X}_{\min}^{res}\right] \tag{A.20}$$

And then the IDFT. Whereas, first the maximum phase part is addressed.

Maximum phase

Extraction

$$c_{\max}^{\mathrm{NDFT}} = w[n] c_x^{\mathrm{NDFT}}[n] \tag{A.21}$$

where

$$w[n] = \begin{cases} 0 & 0 \leq n \leq \lceil N/2 \rceil - 1 \\ 1 & \lceil N/2 \rceil \leq n \leq N - 1 \end{cases} \tag{A.22}$$

Inverse

First the DFT of c_{max} is taken.

$$\hat{X}^{\mathrm{DFT}}_{\mathrm{max}}[k] = \sum_{n=0}^{N-1} c^{\mathrm{DFT}}_{\mathrm{max}}[n] W_N^{kn} \tag{A.23}$$

$$= \sum_{\lceil N/2 \rceil}^{N-1} c^{\mathrm{DFT}}_{x}[n] W_N^{kn} \tag{A.24}$$

$$= \sum_{\lceil N/2 \rceil}^{N-1} \sum_{m} c_x[n+mN] W_N^{kn} \tag{A.25}$$

$$= \sum_{\lceil N/2 \rceil}^{N-1} c_x[n-N] W_N^{kn} + \sum_{\lceil N/2 \rceil}^{N-1} \sum_{m \neq -1} c_x[n+mN] W_N^{kn} \tag{A.26}$$

With the following change of variable

$$n \leftarrow n - N \tag{A.27}$$

$$m \leftarrow m + 1 \tag{A.28}$$

We can write the previous equation as follows:

$$\hat{X}^{\mathrm{NDFT}}_{\mathrm{max}}[k] = \underbrace{\sum_{-\lfloor N/2 \rfloor}^{-1} c_x[n] W_N^{kn}}_{\hat{X}_{\mathrm{max}}} + \underbrace{\sum_{-\lfloor N/2 \rfloor}^{-1} \sum_{m \neq 0} c_x[n+mN] W_N^{kn}}_{\hat{X}^{\mathrm{res}}_{\mathrm{max}}} \tag{A.29}$$

Then the exponential as in case of the minimum phase signal is applied.

$$X^{\mathrm{NDFT}}_{\mathrm{max}}[k] = X_{\mathrm{max}}[k] \exp\left[\hat{X}^{\mathrm{res}}_{\mathrm{max}}\right] \tag{A.30}$$

Followed by the IDFT. However, we will calculate the linear convolution of $x^{\mathrm{NDFT}}_{\mathrm{min}}$ with $x^{\mathrm{NDFT}}_{\mathrm{max}}$ by multiplication in the frequency domain followed by the IDFT.

Linear convolution

First, we multiply $X^{\mathrm{NDFT}}_{\mathrm{max}}$ with $X^{\mathrm{NDFT}}_{\mathrm{min}}$ in the frequency domain.

$$X^{\mathrm{NDFT}}[k] = \underbrace{X_{\mathrm{max}}[k] X_{\mathrm{min}}[k]}_{X[k]} \exp\left[\hat{X}^{\mathrm{res}}_{\mathrm{max}} + \hat{X}^{\mathrm{res}}_{\mathrm{max}}\right] \tag{A.31}$$

Then we write

$$\hat{X}^{\text{res}}_{\max} + \hat{X}^{\text{res}}_{\max} = \sum_{m\neq 0} \sum_{-\lfloor N/2\rfloor}^{\lfloor N/2\rfloor -1} c[n+mN] W_N^{kn} \tag{A.32}$$

$$= \sum_{-\lfloor N/2\rfloor}^{\lfloor N/2\rfloor -1} \sum_{m\neq 0} c[n+mN] W_N^{kn} \tag{A.33}$$

$$= \sum_{-\lfloor N/2\rfloor}^{\lfloor N/2\rfloor -1} \triangle c[n] W_N^{kn} \tag{A.34}$$

introducing $\triangle c[n]$

$$\triangle c[n] = \sum_{m\neq 0} c[n+mN]. \tag{A.35}$$

Therefore, we obtain

$$X^{\text{NDFT}}[k] = X[k] \exp\left[\sum_{-\lfloor N/2\rfloor}^{\lfloor N/2\rfloor -1} \triangle c[n] W_N^{kn}\right] \tag{A.36}$$

Next, we calculate the IDFTs of the two multiplicands in the last equation. The first one is obtained by definition of $X[k]$.

$$\mathcal{IDFT}\left[X[k]\right] = x[n] \tag{A.37}$$

The second one by power series expansion of the exp-function.

$$\exp[x] = 1 + \sum_{n>0} \frac{x^n}{n!} \tag{A.38}$$

and by writing the exponential of a sum as the product of exponentials:

$$\exp\left[\sum_{-\lfloor N/2\rfloor}^{\lfloor N/2\rfloor -1} \triangle c[n] W_N^{kn}\right] = \prod_{-\lfloor N/2\rfloor}^{\lfloor N/2\rfloor -1} \exp\left[\triangle c[n] W_N^{kn}\right] \tag{A.39}$$

Hence, we only have to calculate the IDFT of one of the summands:

$$\mathcal{IDFT}\left[\exp[\triangle c[n] W_N^{kn}]\right] = \mathcal{IDFT}\left[1 + \sum_m \frac{\triangle c[n]^m}{m!} W_N^{knm}\right] \tag{A.40}$$

By denoting this inverse by $e[p;n]$, we have for $0 \le p \le N-1$

$$e[p;n] = \begin{cases} \sum_{m>0} \frac{\triangle c[n]^m}{m!} \delta[p-mn] & p\neq 0 \quad n\neq 0 \\ 1 & p = 0 \quad n \neq 0 \\ e^{\triangle c[n]} \delta[p] & \forall p \quad n = 0 \end{cases} \tag{A.41}$$

Thus, we obtain for the convolution of $x_{\max}^{\mathrm{NDFT}}$ with $x_{\min}^{\mathrm{NDFT}}$

$$x^{\mathrm{DFT}}[p] \quad = \quad x[n] \coprod_{-\lfloor N/2 \rfloor}^{\lfloor N/2 \rfloor - 1} e[p;n] \tag{A.42}$$

where $\coprod$ means multiple convolutions.

Appendix B

Signal Reconstruction

Here, we like to proof the second statement of 3.1.

B.1 Chapter 3

For a rational z-transform as in 2.32, we analyse the contribution of a single element of the product to the phase of the Fourier transform.

$$\arg(1-bz) = \tan^{-1}\left[\frac{b_i\cos\Omega + b_r\sin\Omega}{b_r\cos\Omega - b_i\sin\Omega - 1}\right] \tag{B.1}$$

As

$$1-bz = 1-(b_r+ib_i)(\cos\Omega + i\sin\Omega) \tag{B.2}$$

$$= 1-b_r\cos\Omega + b_i\sin\Omega - i(b_i\cos\Omega + b_r\sin\Omega) \tag{B.3}$$

when z is evaluated on the unit circle. It follows that

$$\arg(1-\overline{b}z^{-1}) = \tan^{-1}\left[\frac{-(b_i\cos\Omega + b_r\sin\Omega)}{b_r\cos\Omega - b_i\sin\Omega - 1}\right] \tag{B.4}$$

$$\arg(1-bz^{-1}) = \tan^{-1}\left[\frac{b_i\cos\Omega - b_r\sin\Omega}{b_r\cos\Omega + b_i\sin\Omega - 1}\right] \tag{B.5}$$

$$\arg(1-\overline{b}z) = \tan^{-1}\left[\frac{-(b_i\cos\Omega - b_r\sin\Omega)}{b_r\cos\Omega + b_i\sin\Omega - 1}\right] \tag{B.6}$$

The above results can be summarized as follows:

$$\arg(1-bz) = -\arg(1-\overline{b}z^{-1}) \tag{B.7}$$

$$\arg(1-\overline{b}z) = -\arg(1-bz^{-1}) \tag{B.8}$$

That is, a zero at $z=z_0$ and a pole at $z=\frac{1}{\overline{z_0}}$ contribute the same amount of phase to the spectrum.