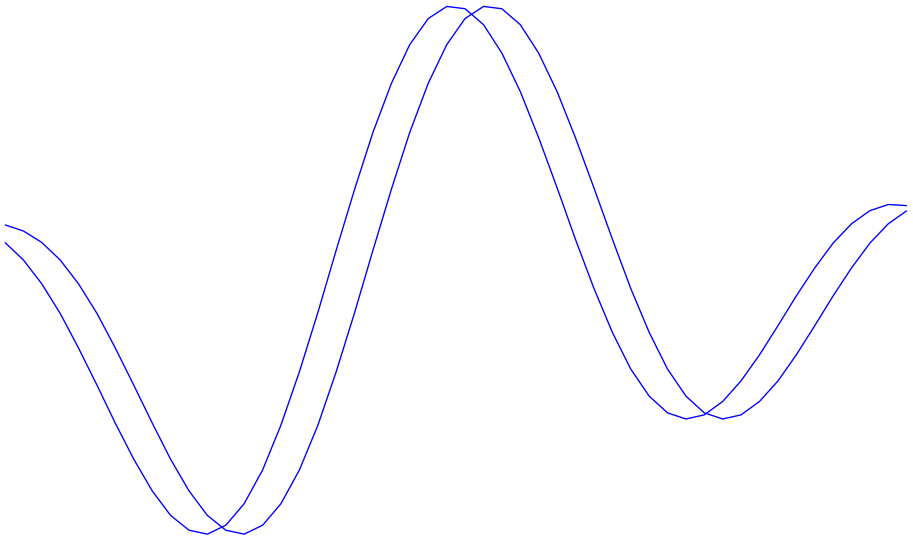
Linköping Studies in Science and Technology
Thesis No. 1252

Contributions to Frequency Offset and Time Delay Estimation

Mattias Olsson



Linköping University
INSTITUTE OF TECHNOLOGY
LiU-Tek-Lic-2006:33

Department of Electrical Engineering
Linköping University, SE-581 83 Linköping, Sweden

Linköping, May 2006

Contributions to Frequency Offset and Time Delay Estimation

Department of Electrical Engineering
Linköpings universitet, SE-581 83 Linköping, Sweden

ISBN 91-85523-63-1 ISSN 0280-7971

Printed in Sweden by LiU-tryck, Linköping 2006

To boldly go

Abstract

The demand for efficient and reliable high rate communication is ever increasing. In this thesis we look at two different problems in such systems, and their possible solutions.

In recent years orthogonal frequency division multiplexing (OFDM) has gone from a promising data transmission technique to become a mainstream technique used in several current and future standards. The main attractive property of OFDM is that it is inherently resilient to multipath reflections because of its long symbol time. However, this comes at the cost of a relatively high sensitivity to carrier frequency offsets (CFOs).

In this thesis we present a technique for CFO estimation in OFDM systems that is based on locating the spectral minimas within so-called null or virtual subcarriers embedded in the spectrum. The spectral minimas are found iteratively over a number of symbols and is therefore mainly useful for frequency offset tracking or in systems where an estimate is not immediately required, such as in TV or radio broadcasting systems. However, complexity wise the estimator is relatively easy to implement and it does not need any extra redundancy beside a nonmodulated subcarrier. The estimator performance is studied both in a channel with additive white Gaussian noise and in a frequency selective channel environment.

A goal for many years has been to be able to implement as much as possible of a radio system in the digital domain, the ultimate goal being so called software defined radio (SDR). One important part of an SDR receiver is the high speed analog-to-digital converter (ADC) and one path to reach this goal is to use a number of parallel, time-interleaved, ADCs. Such ADCs are, however, sensitive to sampling instant offsets, DC offset and gain offset.

This thesis also discusses iterative time-delay estimators (TDEs) utilizing adjustable fractional-delay filters. The TDEs could for example be used to estimate and calibrate the relative delay between the ADCs comprising the time interleaved ADC. TDEs using a direct correlator and an average

squared difference function are compared. Furthermore, an analysis of the effects of the batch length dependence is presented.

Acknowledgements

I am very thankful for being given the opportunity to pursue my doctoral studies. Being a PhD student sure has its ups and downs, but at the end of the day it feels good to have something to strive for, or as Karin Boye wrote:

> Nog finns det mål och mening i vår färd –
> men det är vägen, som är mödan värd.

I would like to thank my supervisor Håkan Johansson for always taking the time to discuss research issues. My thanks also go to all my collegues at the Electronics Systems Division for providing a friendly environment to work in.

I have come to realize that work can only be one part of a happy life. The dance and joy in the company of my good friends in *Folkmusik i Linköping* have become a way for me to recharge my batteries.

Furthermore I would like to thank my family. You have always supported me throughout life and school. You mean a lot to me and I love you.

Finally I would like to thank Kristin, the love of my life. It feels great to have you by my side.

Brokind, April 2006
Mattias Olsson

Abbreviations

Name	Meaning
ADC	Analog-to-Digital Converter
AWGN	Additive White Gaussian Noise
CFO	Carrier Frequency Offset
CIR	Channel Impulse Response
CP	Cyclic Prefix
DAC	Digital-to-Analog Converter
DFT	Discrete Fourier Transform
FD	Fractional Delay
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
ICI	Inter-Carrier Interference
IDFT	Inverse Discrete Fourier Transform
IFFT	Inverse Fast Fourier Transform
IIR	Infinite Impulse Response
IQ	In-phase and Quadrature
ISI	Inter-Symbol Interference
LS	Least Squares
ML	Maximum Likelihood
NR	Newton-Raphson

Name	Meaning
OFDM	Orthogonal Frequency Division Multiplexing
RF	Radio Frequency
RGN	Recursive Gauss-Newton
SDR	Software Defined Radio
SNR	Signal to Noise Ratio
TDE	Time Delay Estimation/Estimator
WLAN	Wireless Local Area Network

Contents

1 Introduction **1**
- 1.1 Applications . . . 1
  - 1.1.1 Multicarrier Modulation . . . 1
  - 1.1.2 Time-Interleaved Analog to Digital Converters . . . 3
- 1.2 Publications . . . 4
- 1.3 Outline of the Thesis . . . 6

2 Introduction to OFDM CFO Estimation **7**
- 2.1 Introduction . . . 7
- 2.2 System Model . . . 7
- 2.3 Previous Work . . . 12
  - 2.3.1 Maximum Likelihood Estimation . . . 13
  - 2.3.2 Time Domain Estimators . . . 14
  - 2.3.3 Frequency Domain Estimators . . . 16

3 CFO Estimation Using Null Subcarriers **19**
- 3.1 Introduction . . . 19
- 3.2 CFO Estimation Using Null Subcarriers . . . 20
  - 3.2.1 Proposed Estimator . . . 23
- 3.3 Complexity . . . 26
- 3.4 Simulations . . . 27
  - 3.4.1 Convergence . . . 27
  - 3.4.2 Performance . . . 28
  - 3.4.3 Multipath Environment . . . 30
- 3.5 Conclusions . . . 31

4 Introduction to Time Delay Estimation **33**
- 4.1 Introduction . . . 33
- 4.2 Time Delay Estimation . . . 33

4.2.1 Cost Functions 34
4.2.2 Interpolation 36
4.2.3 Maximization/Minimization Methods 37
4.2.4 Fundamental Performance Limits 37

5 Time Delay Estimation Using Adjustable FD Filters **39**
5.1 Introduction 39
5.2 Locating the Minimum/Maximum 40
5.2.1 Steepest Descent 40
5.2.2 Newton-Raphson 41
5.2.3 Recursive Gauss-Newton 43
5.3 Interpolation Methods 43
5.3.1 First Order Linear Interpolation 44
5.3.2 Interpolation Using FIR FD Filters 45
5.3.3 Interpolation Using All-pass IIR FD Filters 49
5.4 Error Analysis 61
5.4.1 Estimator Offset 61
5.4.2 Estimator Variance 65
5.5 Complexity 72
5.6 Simulations 74
5.6.1 Estimator Offset 74
5.6.2 Estimator Variance 79
5.7 Conclusions 82
5.7.1 Future Work 83

6 Summary **85**
6.1 Future Work 86

References **87**

Chapter 1

Introduction

The demand for efficient and reliable high rate communication is ever increasing. In this thesis we look at two different problems in such systems, and propose possible solutions.

The first part is about carrier frequency offset (CFO) estimation in multicarrier modulation methods like for example Orthogonal Frequency Division Multiplexing (OFDM). The idea is to introduce null subcarriers and estimate the offset by locating the center of the subcarrier.

The second, and larger, part is about subsample time delay estimation. The proposed technique is iterative and is based on adjustable fractional delay filters.

1.1 Applications

The research presented in this thesis has several applications, two of which will be described briefly here. The first is carrier frequency offset estimation in multicarrier systems and the second is delay estimation in time-interleaved analog-to-digital converters (ADCs).

1.1.1 Multicarrier Modulation

In recent years multicarrier modulation and especially OFDM has received much attention for its resilience to multipath fading. This resilience is achieved by dividing the available bandwidth into densely packed, parallel, sub bands with lower data rates. Lower data rate means longer symbols and if the symbol length is long compared to the length of the multipath channel the symbol is virtually unaffected by the channel. However, one

downside of OFDM is its sensitivity to carrier frequency offset between the transmitter and receiver.

The basic principle of an OFDM symbol is illustrated in Fig. 1.1. The side lobes of each sub band are zero at the peak of the main lobes of the other sub bands. It is easy to see that the sub bands would interfere with each other in the receiver if a carrier frequency offset were present. Thus, there is a need for a way to estimate the carrier frequency offset (CFO) in the reciever and then try to compensate it.

During the years quite many frequency estimation methods have been proposed. In this thesis we propose another method that works iteratively in the frequency domain and which uses embedded, nonmodulated, sub bands in the OFDM symbol to estimate the CFO. Its main advantage is that it does not rely on the cyclic prefix (CP) or any other added redundancy, beside the null subcarrier.

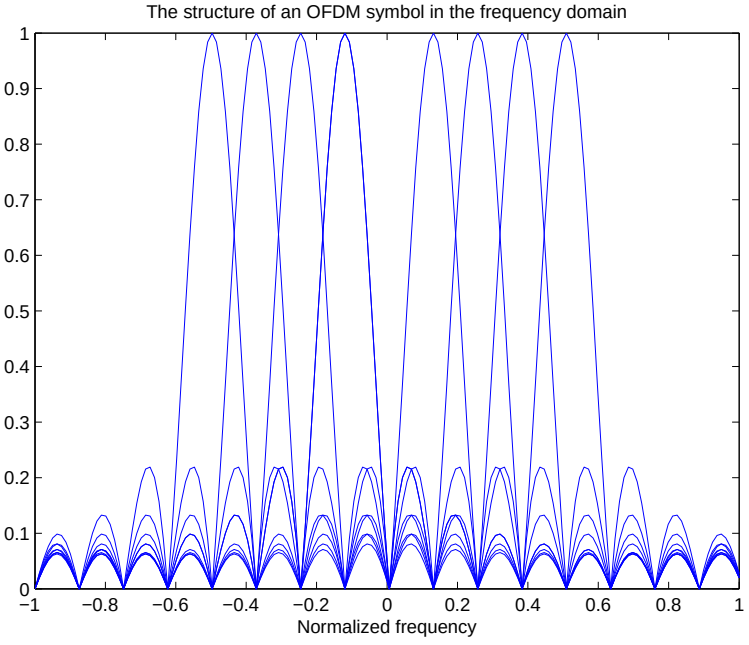


Figure 1.1. *Illustration of how the subcarriers can fit together without disturbing each other in the frequency domain.*

1.1.2 Time-Interleaved Analog to Digital Converters

High-speed analog-to-digital converters (ADCs) are needed in many applications, e.g. in video cameras, radio transmitters/receivers, and medical applications. One way to achieve this is to interleave N ADCs in time and theoretically get a N-fold speedup. However, in reality timing offsets, DC offsets, unequal gain, etc, limit the performance.

An illustration of the basic principle of a two-fold time-interleaved ADC can be seen in Fig. 1.2. If the output sample period is T, the sample period of the individual ADCs is equal to $2T$ and the desired delay between the sampling instants is equal to T. Any time difference between the sampling instants will introduce undesired frequencies in the digital signal. This difference can be modeled as the unknown delays τ_1 and τ_2. However, without loss of generality there is only a need to estimate the relative time delay $\tau_1 - \tau_2$ and not the absolute delays. Calibration of the time delay can either be done online or offline, depending on the application. In a time-interleaved ADC the delays can be compensated as described in for example [10].

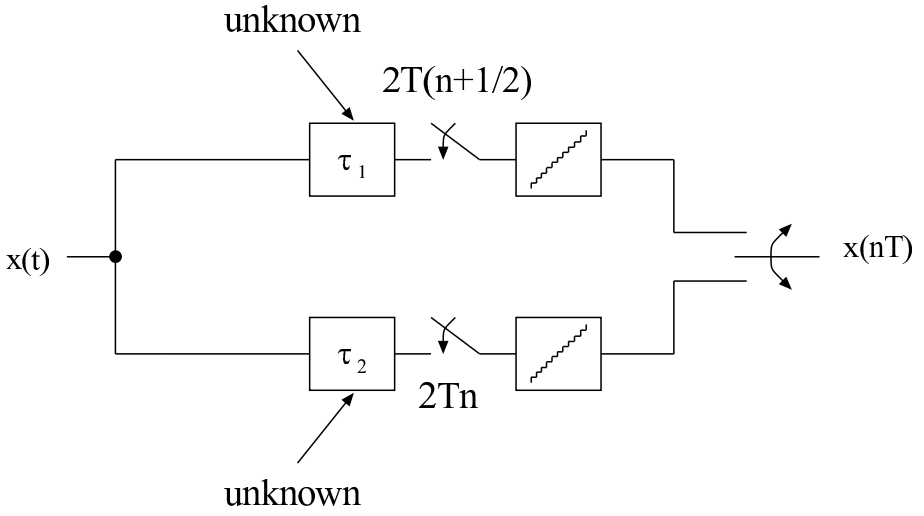


Figure 1.2. *An illustration of a time-interleaved ADC with unknown delays τ_1 and τ_2.*

Other applications where time delay estimation is an essential part are for example radar and sonar.

1.2 Publications

The main contributions to this thesis are summarized in conjunction with the publications below.

OFDM Carrier Frequency Offset Estimation

> M. Olsson and H. Johansson, "Blind OFDM carrier frequency offset estimation by locating null subcarriers", *Proc. of 9th Int. OFDM-workshop*, Dresden, Germany, Sept. 2004

In this paper we present an OFDM CFO estimation algorithm that works by locating the spectral minima within a null subcarrier. The spectrum is contracted and the minimum is found through an exhaustive search. The resolution is limited by the number of points used.

> M. Olsson and H. Johansson, "OFDM carrier frequency offset estimation using null subcarriers", *Proc. of 10th Int. OFDM-workshop*, Hamburg, Germany, Sept. 2005

In this paper the method in the previous paper is run in an iterative mode using Newton-Raphson's technique and the resolution is therefore increased.

> M. Olsson and H. Johansson, "An overview of OFDM synchronization techniques", *Proc. National Conf. Radio Science*, RVK'05, Linköping, Sweden, 2005

This OFDM overview paper was a contribution at the RVK'05 conference, Linköping.

Time-delay Estimation

> M. Olsson, H. Johansson and, P. Löwenborg, "Time-delay estimation using Farrow-based fractional-delay FIR filters: Filter approximation vs. estimation errors", *to appear in Proc. XIV European Signal Processing Conf*, EUSIPCO'06, Florence, Italy, Sept. 2006

In this paper we present a CFO estimator using Farrow FIR fractional delay filters, analytical derivatives and the Newton-Raphson technique. An analysis of estimator offsets and how to optimize the fractional delay filters are also provided.

> M. Olsson, H. Johansson and, P. Löwenborg, "Delay estimation using adjustable fractional delay allpass filters", *to appear in Proc. Nordic Signal Processing Symp.*, NORSIG'06, Reykjavik, Iceland, June 2006

In this paper we present a CFO estimator using all-pass IIR fractional delay filters, analytical derivatives and the Newton-Raphson technique. An analysis of estimator variance caused by batch truncation is also provided.

Scaling and Noise in Multistage Interpolators/Decimators

> M. Olsson, P. Löwenborg and, H. Johansson, "Scaling of Multistage Interpolators", *Proc. XII European Signal Processing Conf.*, EUSIPCO'04, Vienna, Austria, Sept 2004

The work in this paper is outside the scope of this thesis. In the paper we present a method to compute the scaling coefficients needed in a fixed point implementation of multistage interpolators to reduce the risk of overflow. The method is based on polyphase expansion and multirate identities.

> M. Olsson, P. Löwenborg and, H. Johansson, "Scaling and Roundoff Noise in Multistage Interpolators and Decimators", *Proc. Fourth Int. Workshop Spectral Methods Multirate Signal Processing*, SMMSP'04, Vienna, Austria, Sept 2004

The work in this paper is outside the scope of this thesis. This paper is an extension of the paper above and it also covers decimation and roundoff noise.

1.3 Outline of the Thesis

The thesis is basically divided into two parts, one about OFDM CFO estimation and one about delay estimation. First, we begin with an introduction to OFDM CFO estimation, followed by a chapter about a proposed CFO estimator. The next part begins with an introduction to subsample delay estimation, followed by a chapter about a number of proposed delay estimators and an analysis of the estimator errors. Finally we summarize the thesis.

Chapter 2

Introduction to OFDM CFO Estimation

2.1 Introduction

The demand for high data rate radio tranciever services is ever increasing. In the single-carrier modulation case higher data rate means shorter symbol times, which might lead to higher risk for intersymbol interference (ISI) when the delay spread in the channel becomes large compared to the symbol time.

Multicarrier modulation techniques such as orthogonal frequency division multiplexing (OFDM) reduce this problem. OFDM splits a high rate single carrier system into a number of parallel carriers with lower data rate, providing a better resistance to multipath fading. However, among other things, this comes at the cost of a higher sensitivity to a carrier frequency offset (CFO) between the transmitter and receiver.

The outline of this chapter is as follows. We begin in the next section by describing the system model used, followed by a more detailed overview of some of the more common CFO estimation algorithms in the literature, to illustrate some of the basic concepts.

2.2 System Model

The basic idea of OFDM is based on the observation that overlapping subcarriers can be placed closely together without interfering with each other if the side lobes of the surrounding subcarriers are located in between the

other subcarriers. This is illustrated in Fig. 2.1.

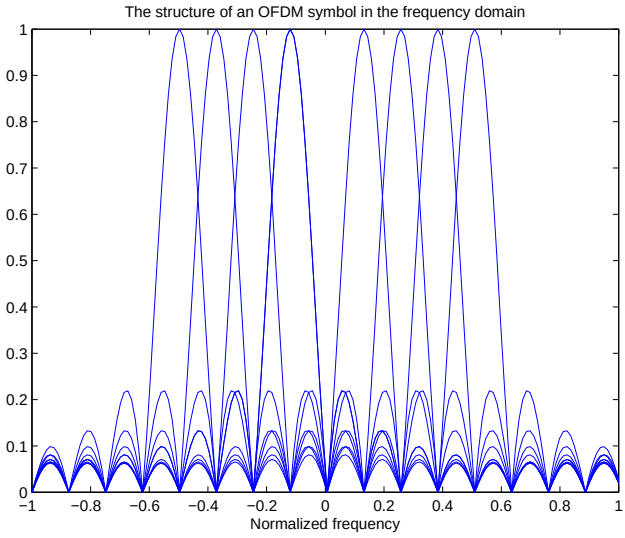


Figure 2.1. *An illustration of how the subcarriers can fit together without disturbing each other in the frequency domain.*

An OFDM symbol is normally created in the frequency domain. The data to be transmitted is mapped onto complex-valued numbers, representing certain phases and amplitudes. In Fig. 2.2 an illustration of an OFDM symbol in the frequency and time domain can be seen. The outer subcarriers are unused to allow a low pass filter with a wider transition band after the digital-to-analog converter (DAC). The central subcarrier is normally not used either since it corresponds to DC in the baseband. In the next chapter we will introduce more unused or virtual subcarriers in the frequency domain and use them to estimate the CFO.

The OFDM symbol in the frequency domain, represented as complex-valued numbers, is transformed into the time domain using the inverse discrete Fourier transform (IDFT). Assume that the total number of subcarriers is N, including the unused subcarriers, and that $X(k)$ contains the modulated complex data. The data is transformed into the time domain by calculating the IDFT as

$$x_N(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk} \tag{2.1}$$

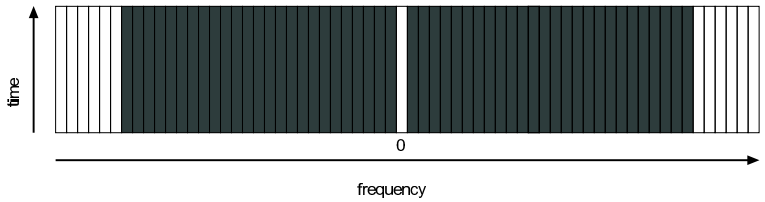


Figure 2.2. *An OFDM symbol in the frequency domain.*

where $W_N = e^{-j2\pi/N}$ and $n = 0, ..., N-1$. In reality, the IDFT is usually implemented using the Inverse Fast Fourier Transform (IFFT) algorithm.

To form a complete OFDM symbol, a Cyclic Prefix (CP) is then added in the time domain by copying the last N_{CP} samples and inserting them in front of the symbol, making the symbol $N + N_{\text{CP}}$ long, see Fig. 2.3. The complete OFDM symbol, including the CP, can be written mathematically as

$$s(n) = x_N(n) \;\; n = -N_{\text{CP}}, \ldots, N-2, N-1 \tag{2.2}$$

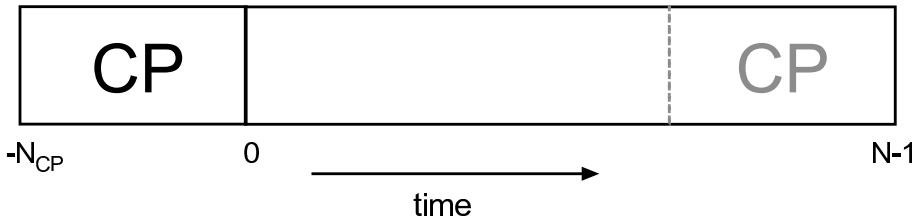


Figure 2.3. *An OFDM symbol in the time-domain with a cyclic prefix.*

The CP works both as a guard interval to prevent Inter-Symbol Interference (ISI) and as a way to ensure that the subcarriers remain orthogonal in a situation where we have a multipath channel or a timing offset. It is well known that the FFT requires cyclic convolution for the time and frequency domain convolution-multiplication relation to be valid. The extension of the symbol with the cyclic prefix therefore reduces the equalization to complex multiplications in the frequency domain. A downside of the CP is that since it contains redundant data, the CP decreases the efficiency of the transmission. Multicarrier systems without a cyclic prefix have been proposed, see for example [26]. In such systems, however, the problems

with ISI and channel equalization have to be dealt with by using a more complex equalizer.

The complete OFDM symbol consists of complex-valued numbers and before it can be transmitted through the air it is sent through an IQ-modulator (In-phase/Quadrature). An IQ-modulator transforms the complex signal into a real sum of two modulated and 90-degree phaseshifted sinusoids. The signal also has to be moved up in frequency to the correct frequency band. The details are beyond the scope of this thesis. For most purposes it is sufficient to use the complex baseband model in all calculations regarding OFDM and ignore the RF (Radio Frequency) up conversion.

The power spectral density of the transmitted OFDM signal is illustrated in Fig. 2.4. Note the null subcarrier in the middle.

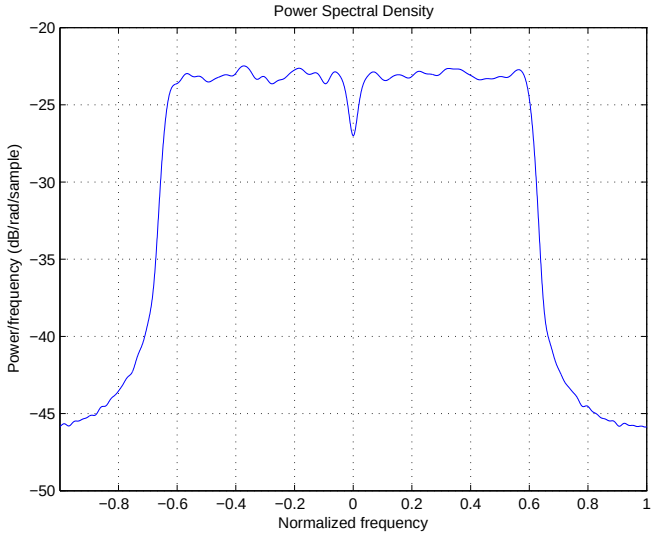


Figure 2.4. *The mean power spectral density of 30 OFDM symbols.*

In this thesis we have chosen to model the multipath channel as a fixed, complex, FIR filter. In reality the channel is not static and a statistical model might, depending on the situation, have to be employed. However, it is common to assume that the channel is at least constant for one OFDM symbol.

In Fig. 2.1 it was seen that the subcarriers are densely packed and it

is easily understood that an offset in carrier frequency between the transmitter and receiver would destroy the orthogonality of the subcarriers and cause inter-carrier interference (ICI). It is therefore essential for an OFDM receiver that the CFO is estimated and compensated. The offset compensation can be done either in the time domain before the FFT or by directly adjusting the carrier frequency oscillator. For a more thorough explanation of the effects of a CFO see for example [22] or [23].

Mathematically the CFO can be seen as a multiplication of each sample $s(n)$ by $e^{j2\pi\epsilon n/N}$, where ϵ is the normalized CFO and N is the number of subcarriers. The recieved samples can now be modeled as

$$r(n) = h * s(n-\theta)e^{j2\pi\epsilon n/N} + e(n) \tag{2.3}$$

where $h(n)$ is the Channel Impulse Response (CIR), θ is the unknown timing, ϵ is an unknown normalized CFO and $e(n)$ is additive noise. We assume that the noise is Gaussian and white. If a channel with only one path and additive noise, a so-called Additive White Gaussian Noise (AWGN) channel, is assumed, we let the CIR be equal to $h(n) = 1$. In the rest of this section we assume such an AWGN channel.

In the receiver the corresponding inversed transmitter operations are performed. First, remove the CP by letting

$$r_N(n) = \begin{cases} r(n) & \text{for } n = 0, 1, ..., N-1 \\ 0 & \text{otherwise.} \end{cases}$$

Here we have assumed no additive noise for clarity. The received samples are then transformed into the frequency domain using the Discrete Fourier Transform (DFT) by calculating

$$X(k) = \sum_{n=0}^{N-1} r_N(n) W_N^{nk}. \tag{2.4}$$

$X(k)$ is then used to demodulate the data.

In Fig. 2.5 an overview of a simple OFDM system can be seen. A CFO estimator and a corrector are usually applied before the FFT in the receiver. Note that the model described in this section is simplified and we have only considered one OFDM symbol at a time. When the channel is multipath with a long impulse response it can cause inter-symbol interferrence (ISI), which is not considered in this model.

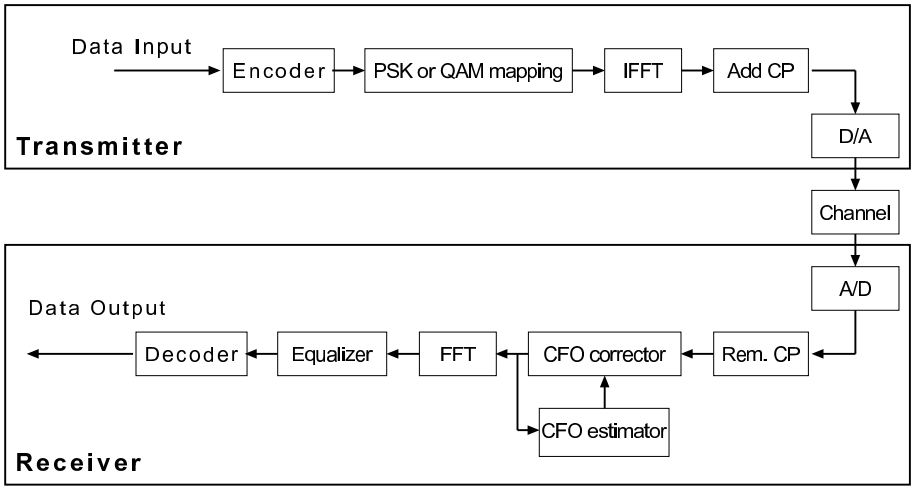


Figure 2.5. *The basic structure of an OFDM system.*

2.3 Previous Work

A number of CFO estimation algorithms have been presented in the literature. Some of them are quite simple, while some of them are more computationally demanding. In 1994 Moose [18] proposed a frequency domain ML CFO estimator that uses two repeated, identical, symbols. This is in practice a form of training symbol and hence lowers the capacity of the communication scheme.

In 1997 van de Beek et al. [29] proposed a blind maximum likelihood (ML) estimation algorithm that uses the redundancy introduced in the cyclic prefix to estimate the CFO. However, the algorithm is derived for an AWGN channel. In a multipath environment the cyclic prefix is more or less destroyed which reduces the performance of the algorithm. Still, it is one of the most widely used CFO estimation algorithms.

In 2001 Choi et al. [5] proposed an ML estimation algorithm that assumed that the OFDM symbol is a Gaussian distributed signal, which is asymptotically true for circularly modulated OFDM symbols. However, it also assumes perfect second order knowledge of the channel statistics. In 2001 Chen and Wang [4] also presented a blind CFO estimation algorithm based on two time oversampling.

In 1998 and 2000 Liu and Tureli [13], [27], presented algorithms that use virtual or nonused subcarriers and techniques similar to the spectral

analysis techniques used in algorithms such as MUSIC and ESPRIT. However, it requires multiple OFDM symbols to achieve good performance and it uses singular value decomposition (SVD) which is computationally demanding.

In the next chapter we will present a technique for CFO estimation in OFDM systems that is also based on locating the spectral minimas within so-called null or virtual subcarriers embedded in the spectrum. The spectral minimas are found iteratively over a number of symbols and is therefore mainly useful for frequency offset tracking or in systems where an estimate is not immediately required, such as in TV or radio broadcasting systems. However, complexity wise the estimator is relatively easy to implement and it does not need any extra redundancy beside a nonmodulated subcarrier.

We will continue with a short recap of maximum likelihood estimation (MLE). After that we will describe some of the algorithms introduced above in more detail, as an introduction to some of the techniques used in the area of CFO estimation and as a reference when we in the next chapter present our proposed CFO estimator.

2.3.1 Maximum Likelihood Estimation

The principle of maximum likelihood was introduced by Fisher (1912) and is a method for parameter estimation. The idea behind it is to make the parameters θ *as likely as possible* by maximizing the joint probability function $f_y(\theta; y_1, y_2, \ldots, y_N)$ when the observed N values are given by the vector $\mathbf{y}$. The maximum likelihood estimator is then given by [14]

$$\hat{\theta}_{ML}(\mathbf{y}) = \arg\max_{\theta} f_y(\theta; \mathbf{y}). \tag{2.5}$$

According to [1] ML-estimators are usually consistent and often result in an estimate with a smaller variance compared to other non-biased estimators. It is not certain that an ML estimator is non-biased, but it can usually be corrected to become non-biased. With a non-biased estimator the estimate becomes equal to the true value after an infinite number of samples have been observed, while a biased estimator still contains an offset.

Let P be the mean-square error matrix defined as

$$P = [\hat{\theta}(\mathbf{y}) - \theta_0][\hat{\theta}(\mathbf{y}) - \theta_0]^T \tag{2.6}$$

where θ_0 is the correct parameter vector. Now, it is interesting to note that there exists a theoretical lower bound on the mean-square error matrix P

that can be obtained with any non-biased estimator. This lower bound is called the *Cramér-Rao inequality* (C-R) and can be written as

$$E[P] \geq M^{-1} \tag{2.7}$$

where

$$M = E\Big[\frac{d}{d\theta}\log f_y(\theta;\mathbf{y})\Big]\Big[\frac{d}{d\theta}\log f_y(\theta;\mathbf{y})\Big]^T\Big|_{\theta=\theta_0} \tag{2.8}$$

is called the *Fisher information matrix*. If the estimator is biased the C-R inequality might or might not hold.

2.3.2 Time Domain Estimators

We will now look at two time domain algorithms. The first one is the most common and well known algorithm, which is the maximum-likelihood estimator using the CP.

ML CFO and Timing Estimation with an AWGN channel

An ML timing and CFO estimator is derived in [29] that utilizes the redundancy introduced by the CP. The log-likelihood function can, under the assumption that the received samples $r(n)$ are Gaussian, be written as

$$\Lambda(\theta,\epsilon) = |\gamma(\theta)|\cos(2\pi\epsilon + \angle\gamma(\theta)) - \rho\Phi(\theta) \tag{2.9}$$

where

$$\gamma(m) = \sum_{n=m}^{m+N_g-1} r(n)r^*(n+N), \tag{2.10}$$

is the complex correlation between N_g samples N samples apart and

$$\Phi(m) = \frac{1}{2}\sum_{n=m}^{m+N_g-1} |r(n)|^2 + |r(n+N)|^2 \tag{2.11}$$

and

$$\rho = \frac{\sigma_s^2}{\sigma_s^2+\sigma_n^2} = \frac{\text{SNR}}{\text{SNR}+1}. \tag{2.12}$$

The timing instant Θ that maximizes (2.9) can be found to be

$$\hat{\Theta}_{ML} = \arg\max_{\Theta}\left\{|\gamma(\Theta)| - \rho\Phi(\Theta)\right\}. \tag{2.13}$$

When the timing is found the normalized CFO estimate can be calculated as

$$\hat{\epsilon}_{ML} = -\frac{1}{2\pi}\angle\gamma(\hat{\Theta}_{ML}) \tag{2.14}$$

Since the CFO estimator depends on the angle of (2.10) it will be periodic and therefore the upper limit on the CFO that can be estimated is

$$|\Delta f| \leq \frac{1}{NT_s} = \Delta f_{\max} \tag{2.15}$$

where N is the delay between the correlated samples and T_s is the sampling period. N is usually equal to the symbol length without the CP. If the frequency offset is greater than $\Delta f_{\max}$ the resulting estimate will be unable to detect the part of the CFO that consists of an integer number times the distance between the carriers.

One downside with algorithms that use the CP is that one of the reasons for having a CP in the first place is to protect the symbol from Inter-Symbol Interference (ISI) caused by a channel with multiple paths. Such algorithms might hence perform badly in such an environment, seen for example in [21].

ML Estimation in Rayleigh Fading Channels

Rayleigh fading is not taken into account in the algorithm above. In a practical situation there might be so much fading that the performance becomes poor. In [5] an ML estimator is derived that also depends on the autocorrelation of the received signal and thus indirectly on the autocorrelation of the transmitted signal and the channel.

Assume that M consecutive samples $r(n)$ are observed, where $k \leq n \leq k + M - 1$ and

$$r(n) = s(n)e^{j2\pi\epsilon/N} + w(n), \tag{2.16}$$

where $s(n)$ are the samples at the receiver without the fractional frequency offset ϵ.

Define R_{ss} as

$$R_{ss} \equiv R_s + \sigma^2 I \tag{2.17}$$

where R_s is the autocorrelation of $s(n)$ and σ^2 is the variance of the added white noise.

Let $\tilde{a}_{m,n}$ be the elements of R_{ss}^{-1}. Under these assumptions the log-likelihood cost function can be shown to be [5]

$$\Lambda(\epsilon) = \sum_{m=1}^{M}\sum_{n=1}^{M} r^*(m)r(n)\tilde{a}_{m,n}e^{j2\pi(m-n)\epsilon/N}. \tag{2.18}$$

Finding the CFO that maximizes (2.18) is rather difficult and therefore a simpler estimator is proposed in [5], namely

$$\hat{\epsilon}_N = \frac{1}{2\pi}(\pi \text{sign}(\theta_N) - \theta_N). \tag{2.19}$$

where θ_N is given by $\theta_N = \angle \sum_{n=1}^{2L} r^*(n+N)r(n)\tilde{a}_{n+N,n}$.

The proposed estimator above requires knowledge of the channel statistics, i.e. the noise power, Doppler spread, delay spread and multipath intensity profile, which makes it difficult to use as an initial CFO estimator.

2.3.3 Frequency Domain Estimators

We will look at a CFO estimation algorithm that is working in the frequency domain.

With Known or Unknown Training Symbols

In [18] an ML estimator in the frequency domain is presented that relies on the transmission of two identical symbols. First, compute the DFT of the two symbols $m = 1$ and $m = 2$ as

$$R_{m,k} = \frac{1}{N}\sum_{n=0}^{N-1} r_m(n)e^{-j\frac{2\pi nk}{N}} \tag{2.20}$$

where k is the subchannel number.

The CFO estimate can then be computed in closed form as

$$\hat{\epsilon} = \frac{1}{2\pi}\arctan\Big(\frac{\sum_{k=0}^{N-1}\Im[R_{2,k}R_{1,k}^*]}{\sum_{k=0}^{N-1}\Re[R_{2,k}R_{1,k}^*]}\Big), \tag{2.21}$$

The estimator above is used in [32] to calculate a coarse estimate using the short training symbols in the IEEE802.11a preamble. The fine estimate is performed after the demodulation by correlating the demodulated output and the expected output, finding the maximum and shifting the position accordingly.

A downside with the algorithm is that it needs two identical symbols, which in practice means that a training symbol has to be used. Any such training symbol lowers the capacity, but depending on the specific requirements this may or may not be acceptable.

Chapter 3

CFO Estimation Using Null Subcarriers

3.1 Introduction

This chapter presents a novel algorithm for estimating the carrier frequency offset (CFO) in an orthogonal frequency division multiplexing (OFDM) receiver. The algorithm is based on locating the spectral minimas within so called null or virtual subcarriers embedded in the spectrum.

We first proposed in [19] to do this by scaling an FFT so that its output was the frequency components centralized around the null subcarrier and then by locating the center of the null subcarrier by finding the spectral minima using an exhaustive search. The performance of the algorithm is, however, limited by the selected resolution of the frequency axis, i.e. the width of the FFT used. In this chapter we present an expansion of that algoritm that does not have this limitation. The expanded algorithm has previously been described in [20].

A somewhat similar algorithm to the one presented in this chapter was presented a few years ago in [13] and it also uses null subcarriers to estimate the CFO. However, their approach is different and from the paper it is not clear how the estimate is to be found in practice and at what complexity.

The outline of this chapter is as follows. We begin with a summary of the algorithm in [19], followed by a presentation of the expanded algorithm proposed in [20]. Finally, we perform an evaluation of the algorithm using simulations, followed by conclusions.

3.2 CFO Estimation Using Null Subcarriers

As stated earlier, in an OFDM system it is common that unused subcarriers are embedded in the spectrum, either as pilot subcarriers or as a null subcarriers. Such an OFDM symbol is illustrated in Fig. 3.1.

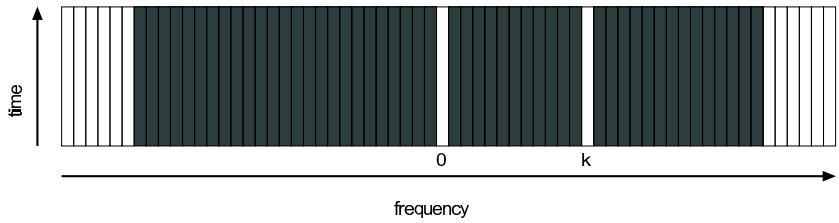


Figure 3.1. *An OFDM symbol with a null subcarrier at position k.*

The idea behind the CFO estimation algorithm, which we first presented in [19], is to locate the center of such a null subcarrier. This can be done in several ways. In the original paper it was done by contracting the window of a DFT around each of the null subcarriers and estimating the CFO by locating the minimum of the subcarrier spectrum, e.g. as an exhaustive search or using interval halving.

The window can be contracted by writing a modified DFT equation as

$$X_\delta(k) = \sum_{n=0}^{N-1} x_N(n) e^{-j\frac{2\pi\delta}{N}nk} \tag{3.1}$$

where $\delta = (0, 1]$ is the normalized window width. In this way a high resolution spectrum of the frequencies surrounding the central subcarrier is acquired. An example of a contracted spectrum around the central subcarrier can be seen in Fig. 3.2. In this example a small CFO has been added to the symbol and therefore the minimum is shifted somewhat to the left.

Note that the central subcarrier is used for convenience in this example. In practice this subcarrier will represent the DC level in an OFDM system and hence it cannot be used in a transmission system. However, the spectrum can easily be shifted in the frequency domain by multiplying the samples by $e^{j\frac{2\pi}{N}kn}$ in the time domain to shift the subcarrier $-k$ to the center.

To be able to compute the contracted spectrum with an FFT, the multipliers in the FFT must be general or at least able to switch between a limited

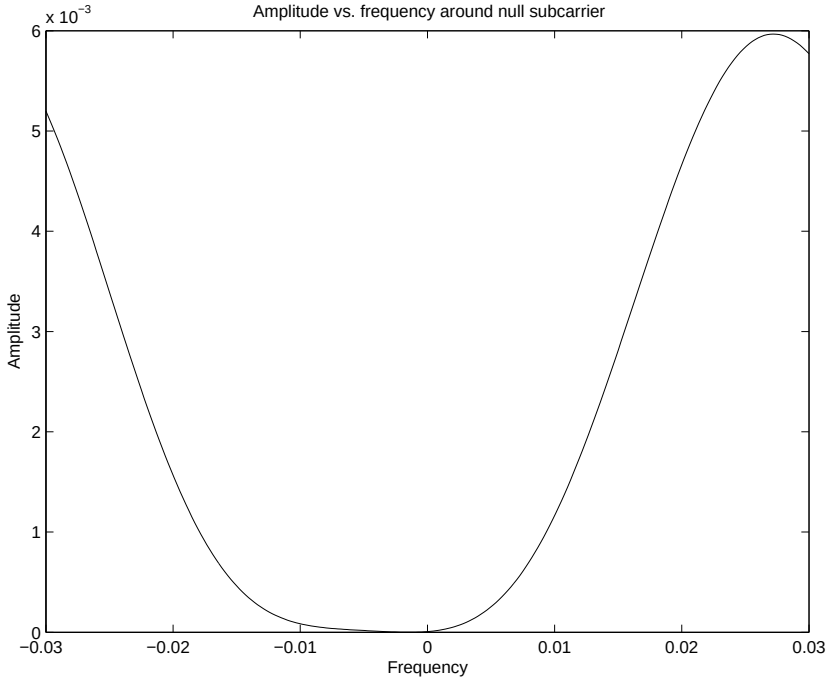


Figure 3.2. *The spectrum of an OFDM symbol around a null subcarrier.*

number of factor sets. Different window widths can be used, depending on the amount of CFO, to maintain the smallest possible quantization of the frequency axis.

Noise and different modulated data will affect the contracted spectrum as can be seen in the upper part of Fig. 3.3. We assume that the noise is white, or at least have a constant spectral density on a narrow band level, and that the distribution of the modulated data is rectangular. The effect of the noise can be decreased by calculating the average of the absolute square of the spectrum. In the lower part of Fig. 3.3 the resulting averaged spectrum amplitude can be seen.

The CFO estimate is then found by locaing the center of the null subcarrier. The most straightforward method to locate the center is to find the minimum, e.g. by an exhaustive search or by using interval halving. The resolution of the DFT can be increased by calculating the spectrum for more frequencies. This corresponds to a non-square DFT matrix and is mathematically equivalent to embedding zeros at the end of the batch

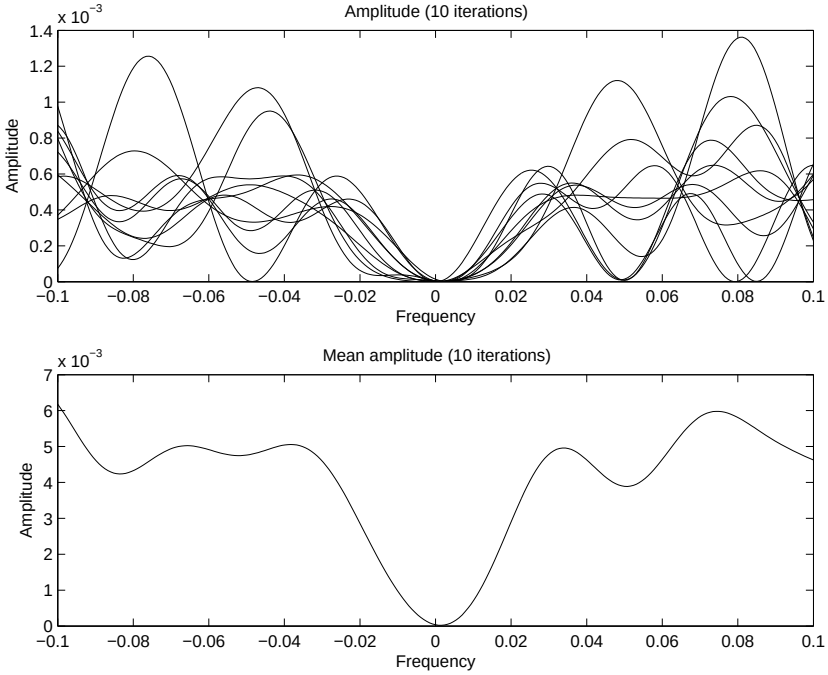


Figure 3.3. *The spectrum averaged over 10 symbols.*

before it is transformed using an DFT.

The resolution of the estimator depends on the width δ and the number of points L calculated in the FFT. The frequency axis is quantized into steps that are

$$\frac{B\delta}{L} \text{ Hz} \tag{3.2}$$

apart, where B is the available bandwidth. In Fig. 3.4 the Mean Squared Error (MSE) for the estimator can be seen for $\delta = 0.02$ and $\delta = 0.01$. If δ is made smaller the error floor is lowered, but the maximal CFO that can be estimated is also lowered, however there is no point in using a higher resolution than the actual noise. This is illustrated in Fig. 3.4, where an obvious error floor is visible. The iterative estimator presented in the next section can easily lower this floor by using more iterations.

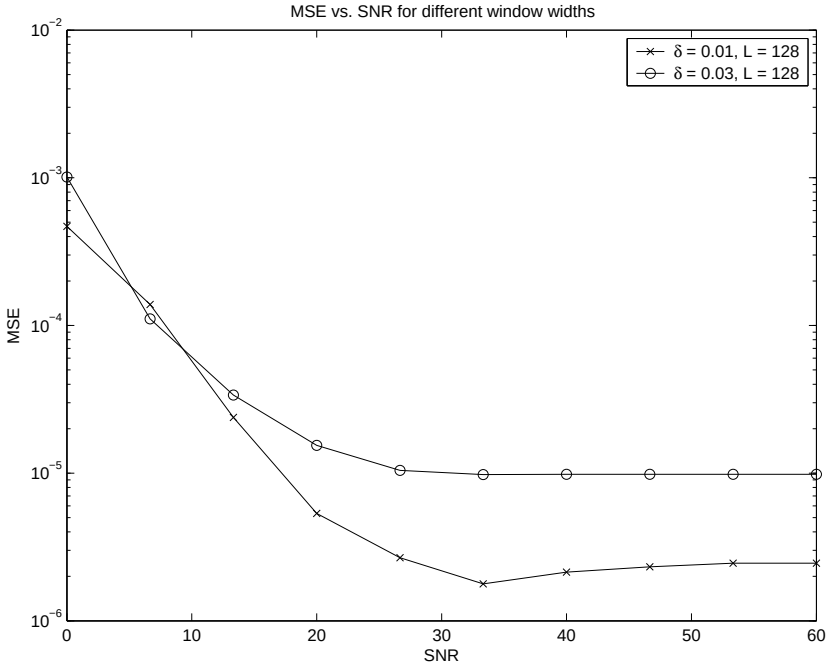


Figure 3.4. *The MSE for different window widths δ using the CFO estimation method based on a scaled FFT.*

3.2.1 Proposed Estimator

We will now see how the estimate can be found iteratively and we propose two variants, **Method A** and **Method B**.

The Fourier Transform (FT) of $x_N(n)$ is calculated as

$$X(e^{j\omega T}) = \sum_{n=-\infty}^{\infty} x_N(n)e^{-j\omega Tn} = \sum_{n=0}^{N-1} x_N(n)e^{-j\omega Tn}. \tag{3.3}$$

Now, by calculating the FT for the discrete frequencies

$$\omega = [0+\epsilon,\ \omega_0+\epsilon, ...,\ (N-1)\omega_0+\epsilon] \tag{3.4}$$

where $\omega_0 = \frac{2\pi}{NT}$ and ϵ is a normalized subcarrier offset, we see that we can write the shifted Discrete Fourier Transform (DFT) as

$$X_N(k+\epsilon) = X(e^{j(k+\epsilon)\omega_0 T}) = \sum_{n=0}^{N-1} x_N(n)W_N^{n(k+\epsilon)}. \tag{3.5}$$

Using (3.5) we can find the frequency content of $x_N(n)$ at the fractional subcarrier distance ϵ from subcarrier k. An estimate of the CFO, $\hat{\epsilon}$, is then found (**Method A**) by minimizing the absolute square of (3.5),

$$\hat{\epsilon} = \min_{-1<\epsilon<1} |X_N(k+\epsilon)|^2, \tag{3.6}$$

This minimum can be found in an iterative manner by using the well-known numerical minimization method Newton-Raphson. The basic method finds, in the one-dimensional case, the closest zero crossing of a function. To use it for minimization we want to find the zero crossing of the derivative. This way, instead of having to compute the spectrum for a limited number of frequencies as in [19], only the first and second derivatives have to be computed in order to find the estimate. The complexity is lowered (fewer DFT sums to compute) and the accuracy increased (no frequency quantization).

If we let

$$F(k+\epsilon) = |X_N(k+\epsilon)|^2, \tag{3.7}$$

the spectral minima can be found using the iterative estimator written as

$$\hat{\epsilon}_{p+1} = \hat{\epsilon}_p - \frac{F'(k+\hat{\epsilon}_p)}{F''(k+\hat{\epsilon}_p)}. \tag{3.8}$$

An illustration of the minimization problem (3.6) can be seen in Fig. 3.5. If the function $F(k+\epsilon)$ has an exactly quadratic shape the minimum is found using only one iteration, however, in reality this is not the case and a number of iterations are needed.

To compute the iterative step in (3.8) we need to compute the first and second derivatives of $F(k+\epsilon)$ with respect to ϵ. The derivatives can be found to be

$$\begin{aligned} F'(k+\epsilon) =& X'X^* + (X^*)'X \\ =& X'X^* + (X')^*X \end{aligned} \tag{3.9}$$

and

$$\begin{aligned} F''(k+\epsilon) =& X''X^* + 2X'(X^*)' + (X^*)''X \\ =& X''X^* + 2X'(X')^* + (X'')^*X. \end{aligned} \tag{3.10}$$

Note that the derivatives are still real, although they contain complex components. Now, if we calculate the first and second derivative of (3.5) with

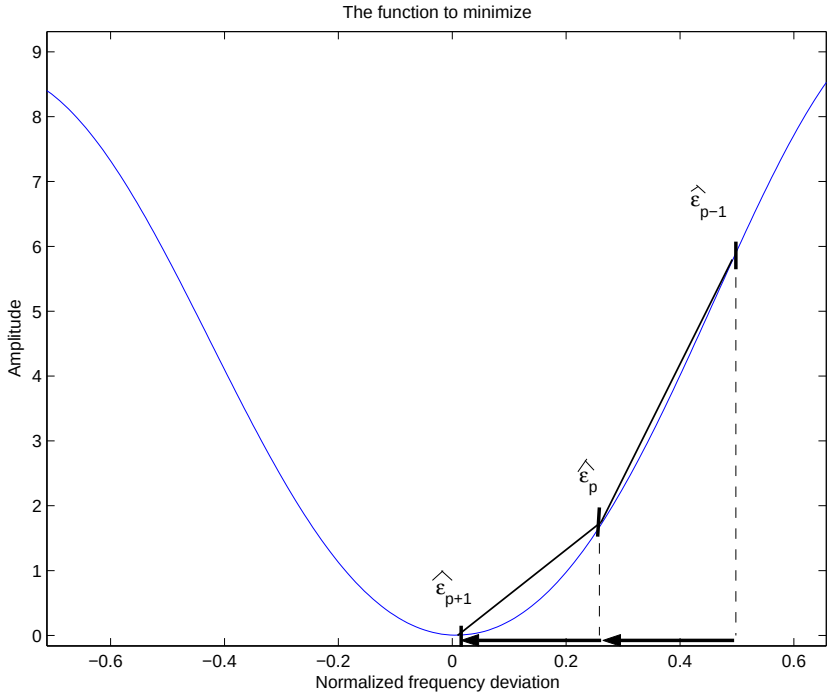


Figure 3.5. *Finding the minima using an iterative method.*

respect to ϵ we get

$$X'(k+\epsilon) = -j\frac{2\pi}{N}\sum_{n=1}^{N-1} nx_N(n)W^{n(k+\epsilon)} \tag{3.11}$$

and

$$X''(k+\epsilon) = -\left(\frac{2\pi}{N}\right)^2 \sum_{n=1}^{N-1} n^2 x_N(n)W^{n(k+\epsilon)}. \tag{3.12}$$

As we can see, the only differences between (3.5), (3.11) and (3.12) are the constants in front of the sums and the integers n and n^2. These numbers can be precalculated, but they still require two real multiplications.

To combat noise, averaging of $F(k+\epsilon)$, $F'(k+\epsilon)$ and $F''(k+\epsilon)$ is introduced before the iterative step is calculated. We found, as will be seen later, that an average over a small number of instances will reduce the Mean-Square Error (MSE) significantly.

An alternative (**Method B**) to minimizing $F(k+\epsilon)$ is to introduce another null subcarrier at $-k$ and minimize

$$\begin{aligned} G(k+\epsilon) = &[X(k+\epsilon) + jX(-k+\epsilon)] \\ &[X^*(k+\epsilon) - jX^*(-k+\epsilon)] \end{aligned} \tag{3.13}$$

instead, which can be rewritten as

$$\begin{aligned} G(k+\epsilon) = &|X(k+\epsilon)|^2 + |X(-k+\epsilon)|^2 + \\ &jX(-k+\epsilon)X^*(k+\epsilon) - jX(k+\epsilon)X^*(-k+\epsilon) \end{aligned} \tag{3.14}$$

Ideally, if $X(k)$ and $X(-k)$ were equal the last two terms would disappear. In reality, they are not completely equal, but, as we will later see from simulations, the difference is sufficiently small to not affect the performance significantly for small SNR. For higher SNR **Method B** reaches an error floor and for an SNR larger than a certain value **Method A** performs better. The main advantage of **Method B**, though, is that $X(k+\epsilon)+jX(-k+\epsilon)$ can be rewritten as

$$X(k+\epsilon) + jX(-k+\epsilon) = C\sum_{n=0}^{N-1} x(n)\cos(\frac{2\pi}{N}kn - \frac{\pi}{4})W_N^{\epsilon n} \tag{3.15}$$

where $C = \sqrt{2}(1-j)$ is a constant. The complexity is reduced since half of the multiplications in the sum are now a real value times a complex value, which is easier to perform. The sequence $\cos(\frac{2\pi}{N}kn - \frac{\pi}{4})$ for $n = 0, 1, ..., N-1$ can be precalculated. At runtime this sequence is modulated by multiplication with $W_N^{\epsilon n}$. The derivatives of $G(k+\epsilon)$ can be found in a way similar to how it was done for **Method A**. As we will see later the cost is a somewhat lower performance and the need to use another null subcarrier.

3.3 Complexity

In the original algorithm the complexity comes from an FFT calculation, an absolute value computation, a center finding operation, and an averaging function [19]. If we let L denote the number of points in the frequency domain, the total number of operations that we have to perform is of order $4\frac{L}{N} + 4 + 4\frac{L}{N}\log_2 L$ real multiplications, $2\frac{L}{N} + 2\frac{L}{N}\log_2 L + 2$ real additions, and 1 comparison per sample. As will be seen from the simulations, L need to be of the order of $4N$ for the resolution to be acceptable.

The number of operations needed for the first algorithm (**Method A**) proposed here can also be derived. Let P denote the number of iterations and M the number of symbols to average $F(k+\delta)$, $F'(k+\delta)$, and $F''(k+\delta)$ over. If we assume that each complex multiplication requires four real multiplications[1] and two real additions, the number of real multiplications and additions needed per sample is approximately $\frac{8 \cdot P \cdot M \cdot N}{M \cdot N} = 8P$. In addition we have to compute W_N^{nk} and $W_N^{\delta n}$. The first can be precalculated and the latter would be needed anyway to correct the frequency offset. We will later see from the simulations what parameters are required to get similar performance.

The second algorithm proposed (**Method B**) reduces W_N^{nk} to $W_N^{nk} + jW_N^{-kn}$, see (3.15), which, beside a complex constant C, is real and thus lowers the number of additions and multiplications needed. As we will see later, this is at the cost of a somewhat lower performance.

3.4 Simulations

To evaluate the two variants of the algorithm a number of simulations were performed. The OFDM symbols were generated using the same parameters as are used in the IEEE802.11a/g standard for wireless LAN, $N = 64, N_{\text{CP}} = 16$, and the constellation used was Quadrature Phase Shift Keying (QPSK). 16-QAM was also tested, but we observed no significant change in performance. We have assumed an Additive White Gaussian Noise (AWGN) channel for most of the simulations, except when it is explicitly stated otherwise. In the plots the MSE has been normalized with respect to the squared subcarrier width $\left(\frac{2\pi}{N}\right)^2$.

3.4.1 Convergence

The Newton-Raphson algorithm converges towards the first local minimum it finds. The rate of convergence depends on a number of parameters, like for example closeness to the minimum. The most important component is the direction towards the minimum, which corresponds to the sign of the first derivative. For a quadratic function the optimal step is the first derivative divided by the second derivative, however, if the second derivative is small or even negative, the direction of the step might become incorrect, causing the algorithm to diverge. In practice, to avoid this we modify

[1] This can be reduced to three multiplications and five additions, seen from $(a+jb)(c+jd) = (ac-bc) + j[(a-b)(c-d) - (ac-bd)]$.

(3.8) by taking the absolute value of $F''(k+\epsilon)$ and by adding a small value Δ.

In Fig. 3.6 an example of the convergence of the iterative algorithm can be seen. The starting points were randomly chosen using an equal distribution, the added noise was Gaussian and the SNR was 10 dB.

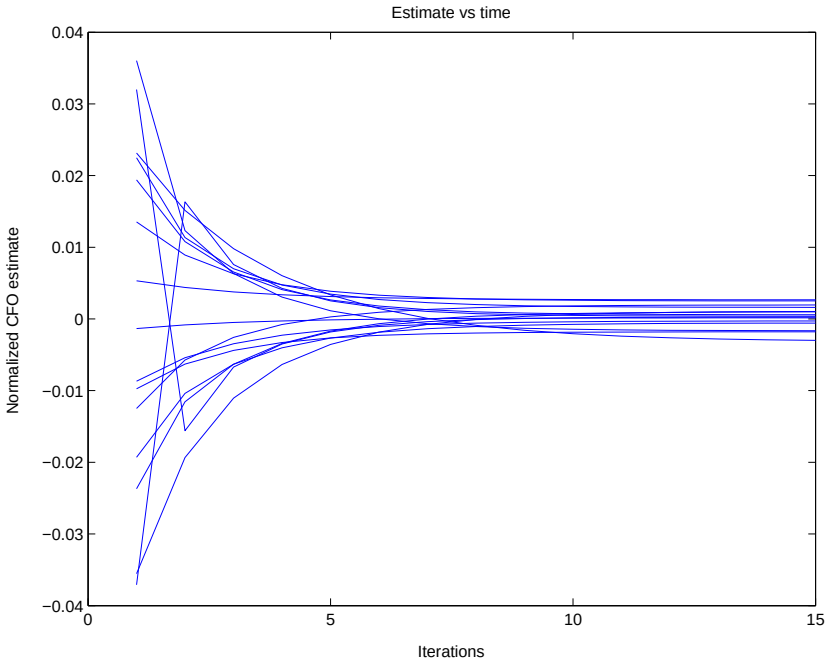


Figure 3.6. *Convergence*

3.4.2 Performance

The following normalized MSE definition has been used

$$\text{MSE} = \frac{\frac{1}{M_c}\sum_{k=1}^{M_c}\left(\epsilon - \hat{\epsilon}(k)\right)^2}{\left(\frac{2\pi}{N}\right)^2} \tag{3.16}$$

where M_c is the number of Monte-Carlo simulations, ϵ is the true CFO, $\hat{\epsilon}$ is the estimate and N is the number of subcarriers.

In Fig. 3.7 we have compared the MSE of the original exhaustive search algorithm and the iterative algorithm. The performance is similar up to

about 11 dB. The error floor for the original algorithm depends on the window width and the number of points, in this case 0.02 and $L = 256$ points.

$F(k + \epsilon)$ was averaged 5 times in the frequency domain before the step length was calculated. To illustrate the effect of the number of iterations the algorithm was simulated with 3, 4, and 10 iterations. It can be seen that for 3 iterations the performance of the two algorithms are very similar, but the number of multiplications per sample for the original algorithm can be estimated to 148 and to 24 for the proposed algorithm.

To avoid running the algorithm for too many iterations we can let it finish as soon as the step size is smaller than a constant or when the maximum number of iterations have been used.

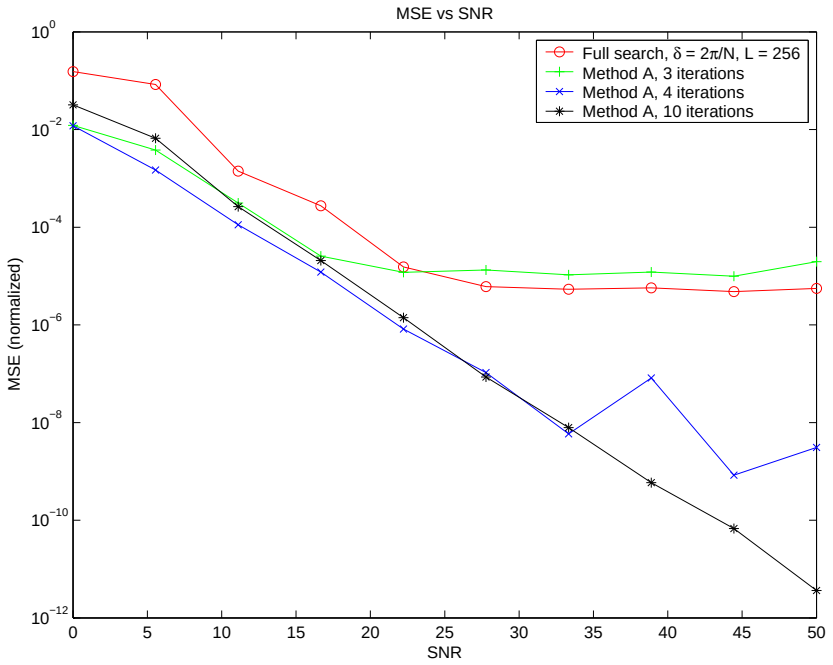


Figure 3.7. *The MSE vs SNR for an AWGN channel.*

In Fig. 3.8 the normalized MSE for **Method A** and **Method B** can be seen. 15 iterations were used. The MSE for the two algorithms are similar up to approximately 20 dB. Increasing the number of iterations for **Method B** does not affect the error floor.

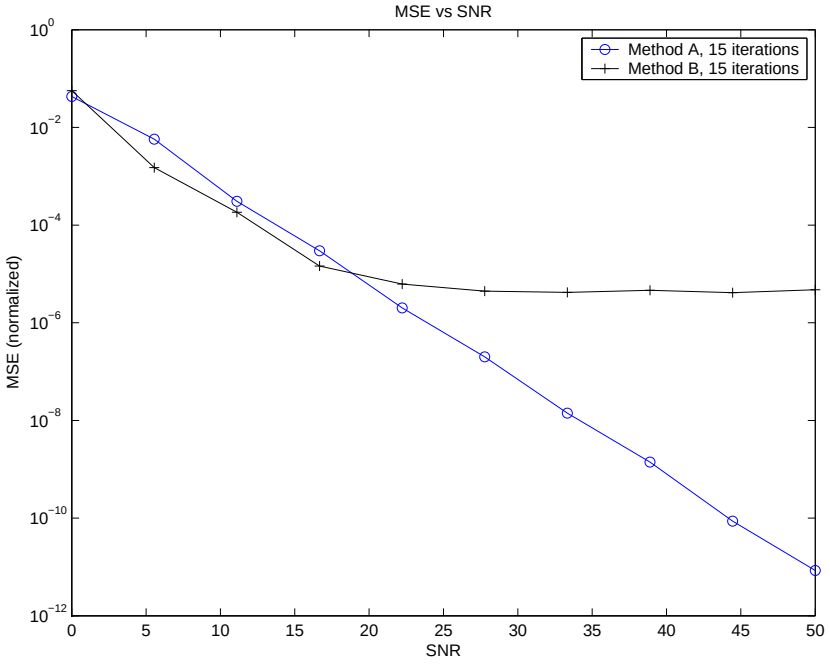


Figure 3.8. *The MSE for Method A and Method B.*

3.4.3 Multipath Environment

In a frequency selective multipath environment the amplitude and phase for different frequencies are distorted. To evaluate the performance of **Method A** in such an environment, we used the channel model number two and three described in the HiperLan/2 standard [17]. Channel two is a Rayleigh fading channel with length 16, which is equal to the length of the CP. A comparison between **Method A** and the algorithm presented in [29] is shown in Fig. 3.9. In the AWGN case the CP-based estimator has a better performance, but with a Rayleigh-fading channel the CP is destroyed and the performance goes down. In this simulation the length of the CP and the channel is equal.

For channel three, a Rayleigh fading channel with length 21, the performance goes down, see Fig. 3.10. This is becuase the channel is longer than the CP and inter-symbol interference occurs. To limit the degradation somewhat, the null subcarrier can be moved around randomly.

For the simulations we assumed that the channel was quasi-static,

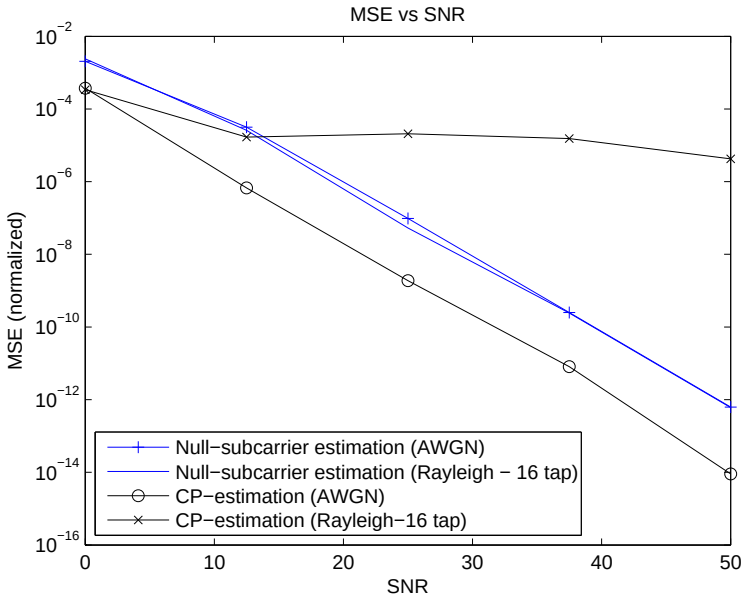


Figure 3.9. *Performance comparison with AWGN and a Rayleigh fading channel of length 16 samples for Method A and the CP-based estimator from [29].*

meaning that it did not change within the symbols.

3.5 Conclusions

We have proposed a CFO estimation technique that uses null subcarriers to find an estimate. The technique is based on the technique that we presented in [19]. The original algorithm estimated the CFO by finding the spectral minimum within an unused null subcarrier using a scaled FFT, however, the achievable resolution depended on the number of points in the FFT.

Here we have instead used an iterative algorithm. From simulations it has been seen that the error floor for one of the proposed algorithms is lowered and depends mainly on the number of iterations used. Complexity wise the new algorithm requires less operations and thus is more efficient.

In simulations it was also seen that **Method A** is resilient to multipath channels and that the performance is proportional to the number of itera-

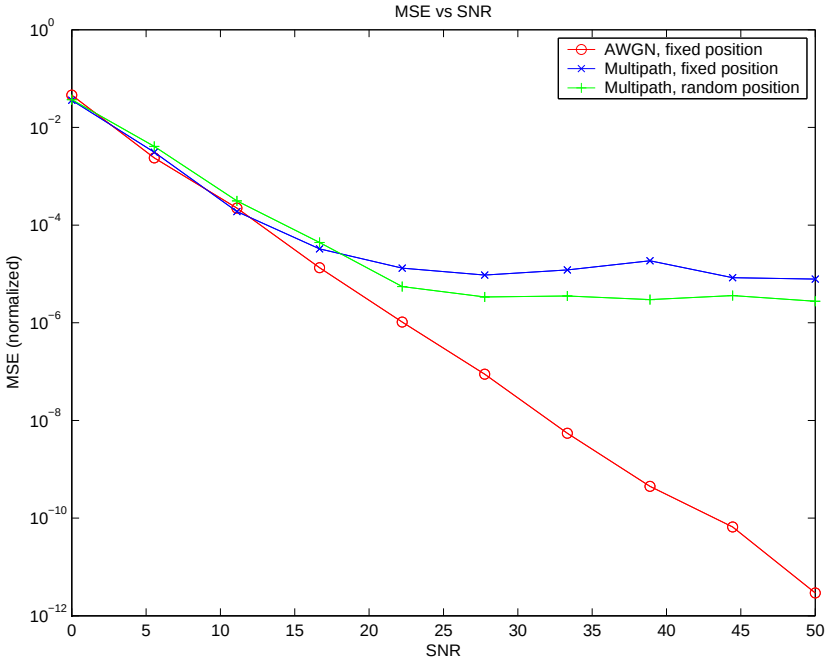


Figure 3.10. *Plot showing the performance with and without random null subcarrier position when the channel is a Rayleigh fading channel of length 21 samples.*

tions used. The algorithm is also unaffected by timing errors as long as the samples are taken within the undisturbed part of the CP.

The other proposed algorithm, Method B, has similar performance as Method A up to an SNR equal to 20 dB, but at a lower complexity. However, at a higher SNR the performance of Method B is worse.

Chapter 4

Introduction to Time Delay Estimation

4.1 Introduction

The need for time delay estimation (TDE) arise in many different fields, including biomedicine, communications, geophysics, radar, and ultrasonics.

Two (or more) discrete-time signals, originally coming from one source $x_a(t)$, might experience different delays. We model this as

$$x(n) = x_a(nT) + e_1(n) \tag{4.1}$$
$$v(n) = x_a((n - D_0)T) + e_2(n) \tag{4.2}$$

where D_0 is the unknown delay difference between the signals, T is the sampling period and $e(n)$ is additive noise. This model is shown in Fig. 4.1. It is assumed that $e_1(n)$ and $e_2(n)$ are uncorrelated with each other and with $x_a(t)$. Furthermore, we assume that the delay $D_0 = \lfloor D_0 \rfloor + d_0$ consists of an integer delay $\lfloor D_0 \rfloor$ and a subsample delay d_0. In this thesis we will only focus on the subsample delay and we assume that the integer sample delay has already been taken care of in a proper manner using a coarse estimator.

4.2 Time Delay Estimation

In the literature a number of approaches to TDE have been used. First, we will look at two common cost functions, followed by interpolation methods and ways to find the estimates numerically.

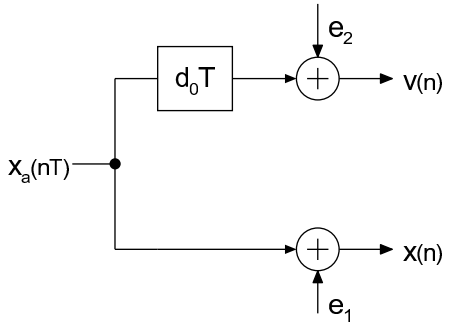


Figure 4.1. *Two channels with an unknown delay d_0 and additive noise.*

4.2.1 Cost Functions

We will focus on the two most common time domain cost functions, first the one that in the literature is called a Direct Correlator (DC) and then the Averaged Squared Difference Function (ASDF). In the literature frequency domain cost functions have also been proposed, for example the Generalized Cross-Correlator (GCC) [11] and Generalized Least Square (GLS) [3], but they are beyond the scope of this thesis.

Direct Correlator

Let

$$y(n,d) = x(n-d) \tag{4.3}$$

be $x(n)$ ideally delayed by the subsample delay d. In reality it is impossible to calculate $y(n,d)$ exactly, but we will later look at different ways to approximate the delay. Now, an estimate of the time delay can be found by maximizing the so called direct correlation (DC) between $y(n,d)$ and $v(n)$,

$$\widehat{d}_{\text{DC}} = \max_d F_{\text{DC}}(d) \tag{4.4}$$

where

$$F_{\text{DC}}(d) = E\{y(n,d)v(n)\} = R_{vd}(d). \tag{4.5}$$

and $R_{vd}(d)$ is the correlation between $v(n)$ and $y(n,d)$ [8].

In [11] it is shown that the direct correlator TDE is the maximum likelihood estimator (MLE) under certain conditions. The signal must however be prefiltered using signal dependent filters, see [11].

Average Squared Difference Function

Another time delay estimate is found by minimizing the averaged squared difference function (ASDF), defined as

$$\widehat{d}_{\text{ASDF}} = \min_d F_{\text{ASDF}}(d) \tag{4.6}$$

where

$$F_{\text{ASDF}}(d) = E\left\{(y(n,d) - v(n))^2\right\} = \sigma_y^2 + \sigma_v^2 - 2R_{vd}(d) \tag{4.7}$$

and σ_y^2 and σ_v^2 are the variance of $y(n,d)$ and $v(n)$ [8].

From these equations it seems like the two cost functions, and hence the estimators, are identical except for the constant noise variances. However, this is not true in practice, as can be seen in for example [8] and in the simulations in Section 5.6.

Practical Realization

In a practical realization we are of course restricted to finite batch lengths. The expectations are approximated as mean values. The DC TDE estimator for a finite batch length N can hence be written as

$$\widehat{d}_{\text{DC}} = \max_d F_{\text{DC}}(d) \tag{4.8}$$

where the cost function is calculated as

$$F_{\text{DC}}(d) = \frac{1}{N}\sum_{n=n_0}^{n_0+N-1} y(n,d)v(n) \tag{4.9}$$

where n_0 is an arbitrary index number.

The ASDF TDE for a finite batch length N can similarly be written as

$$\widehat{d}_{\text{ASDF}} = \min_d F_{\text{ASDF}}(d) \tag{4.10}$$

where the cost function is calculated as

$$F_{\text{ASDF}}(d) = \frac{1}{N}\sum_{n=n_0}^{n_0+N-1} (y(n,d) - v(n))^2 \tag{4.11}$$

and n_0 is an arbitrary index number.

4.2.2 Interpolation

To be able to minimize/maximize the cost function $F_{\text{ASDF}}(d)$ or $F_{\text{DC}}(d)$ we must somehow approximate the delayed samples of $x(n)$, i.e. $y(n,d) = x(n-d)$. This situation is illustrated in Fig. 4.2.

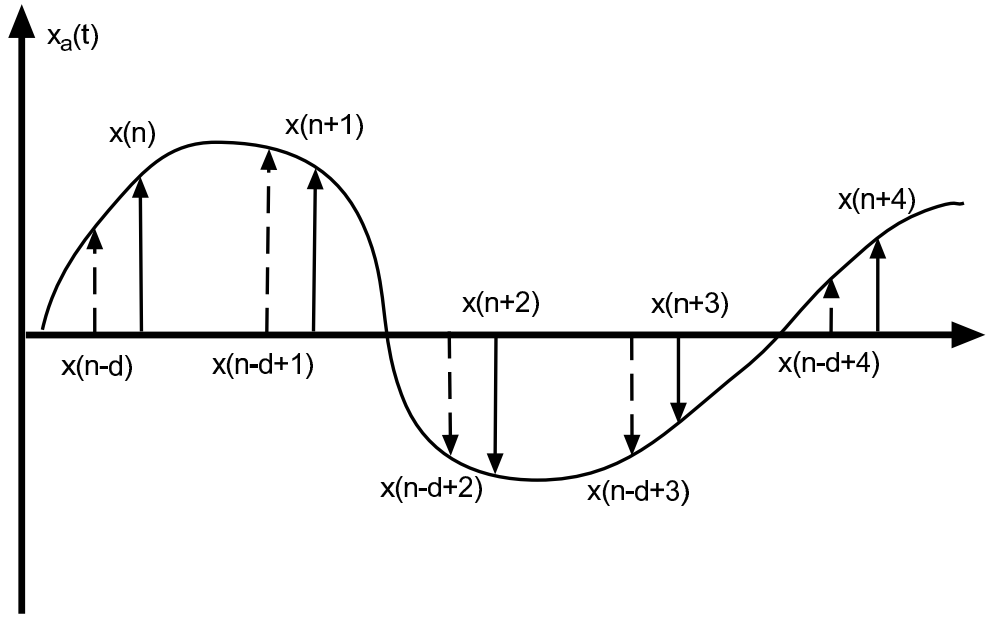


Figure 4.2. *Illustration of interpolated sample values that are delayed by d samples.*

There are several possible ways to find interpolated sample values $y(n,d)$. One of the first that comes to mind is to use some kind of linear interpolation. However, as we will see in the next chapter, this is far too restricted. Instead of linear interpolation we could for example use higher order interpolators such as splines, but it would still be quite difficult to get control over the error and at the same time limit the estimator complexity.

Another way to solve the interpolation problem is to note that a filter with a linear phase response would delay all frequencies equally. The ideal transfer function of such a filter with a constant phase delay d is therefore equal to

$$H_d(e^{j\omega}) = e^{-j\omega d}. \tag{4.12}$$

In recent years methods have been proposed [9], [30], [31], to design adjustable filters which approximate the linear phase response of (4.12), so

called adjustable fractional delay (FD) filters. With the computing power of today it is possible to use optimization tools to design the FD filters with arbitrarily good performance at the cost of implementation complexity. In the next chapter we will discuss how to design such FD filters.

4.2.3 Maximization/Minimization Methods

To find the time delay estimate, the cost functions must either be maximized or minimized. There are a number of common approaches to this problem. The most straightforward solution would of course be to use a full search. This method works, but the convergence would be slow. Other methods include interval halving and multiresolution techniques.

In the next chapter we will look at better methods such as steepest descent, recursive Gauss-Newton and the well known Newton-Raphson method.

4.2.4 Fundamental Performance Limits

As it was stated earlier, it has been found that the DC TDE is the maximum likelihood estimator (MLE) when the signals are prefiltered, however in reality the DC estimator has several problems, as we will see later. For example, it is much more sensitive to the truncation of the sample batches.

In [2] it is shown that the mean squared error for a bias-free TDE is bounded by the Cramér-Rao lower bound (CRLB)

$$\sigma^2_{CRLB} = \frac{3}{8\pi^2}\frac{1+2SNR}{SNR^2}\frac{1}{B^3 T_{\text{obs}}} \tag{4.13}$$

where T_{obs} is the observation time and the SNR is constant for the signal with bandwidth B. A problem with this bound is that it is only valid for bias-free estimators and a biased estimator might actually have a smaller mean squared error. However, it is possible to calculate the CRLB for a biased estimator, but then it is not guaranteed to be valid for other biased estimators.

Chapter 5

Time Delay Estimation Using Adjustable FD Filters

5.1 Introduction

In this chapter we will discuss time delay estimation using adjustable fractional delay (FD) filters and investigate the effects of magnitude and phase delay errors so that we know how to best design these filters.

In 1999 Dooley introduced a technique [6] utilizing Farrow-based digital FIR fractional delay filters [7] for time delay estimation (TDE). The use of FD filters has two major advantages over other delay estimation techniques working in the digital domain. First, it is eminently suitable to handle delays that are fractions of the sampling interval. Second, it can handle general band limited signals. This is in contrast to techniques that assume a known input signal, like a sinusoidal signal [16]. However, no analysis of the filter approximation errors versus estimation errors was provided in [6]. Such an analysis will be provided in this chapter. Furthermore, a new estimator using all-pass IIR fractional delay filters is introduced.

As stated earlier in Chapter 4, the two most common cost functions used for TDE are the Direct Correlator and the Average Squared Difference Function. To find the extreme value of these cost functions, and thus a delay estimate, two techniques are used, Newton-Raphson (NR) and Recursive Gauss-Newton (RGN). The convergence of Recursive Gauss-Newton is slower, but depending on the application this may or may not be of importance. NR needs both the first and second derivative of the cost function, whereas RGN needs the first derivative. Since the delay of adjustable FD filters is governed by only one parameter their analytical derivatives

can be derived. Thereby, the problems associated with the use of numerical derivatives are avoided.

The outline of this chapter is as follows. First we will look at methods to find the extremum of the cost functions, followed by interpolation techniques. We then analyze the estimators from an offset and variance point of view. To verify the results we perform simulations and finally we draw some conclusions.

5.2 Locating the Minimum/Maximum

The time delay estimate is found by locating the extremum of the cost functions $F_{\text{DC}}(d)$ or $F_{\text{ASDF}}(d)$, introduced in (4.9) and (4.11) in the previous chapter and repeated here for convenience:

$$F_{\text{DC}}(d) = \frac{1}{N} \sum_{n=n_0}^{n_0+N-1} y(n,d)v(n) \tag{5.1}$$

$$F_{\text{ASDF}}(d) = \frac{1}{N} \sum_{n=n_0}^{n_0+N-1} \left(y(n,d) - v(n)\right)^2 \tag{5.2}$$

Simple ways to find the extrema are for example to use full search or interval halving, but these methods suffer from slow convergence. More efficient ways, with approximately quadratic convergence, are Steepest Descent (SD), Newton-Raphson (NR), and Recursive Gauss-Newton (RGN).

5.2.1 Steepest Descent

The method of steepest descent (SD), which is also called the gradient descent, minimizes a function by moving iteratively in the direction of the downhill gradient.

In the one-dimensional case we are interested in, the iterative update equation is

$$\widehat{d}_{n+1} = \widehat{d}_n - \mu F'(\widehat{d}_n) \tag{5.3}$$

where μ is the step size. The problem is: how to select the step size without getting slow convergence or even divergence? Instead, we can use other methods which attempt to find the optimum step length.

5.2.2 Newton-Raphson

To find the minimum or maximum of a function $F_{\text{ASDF}}(d)$ or $F_{\text{DC}}(d)$ with respect to d, the well-known Newton Raphson (NR) algorithm can be used. The algorithm is iterative and tends towards the closest zero of a function, in this case the derivative of $F(d)$.

To find the closest extremum in the one-dimensional case the iterative NR update equation is

$$\widehat{d}_{n+1} = \widehat{d}_n - \frac{F'(\widehat{d}_n)}{F''(\widehat{d}_n)}. \tag{5.4}$$

If the starting point is close enough to the extremum, the estimate $\widehat{d}_n$ will converge towards it.

Since the cost functions are almost quadratic with respect to d, the NR-based algorithm will converge to the extremum after a small number of iterations. For a perfectly quadratic function the ideal step length is equal to $1/F''(\widehat{d}_n)$ and only one iteration is needed, however, in a real situation a few more, typically three or four, iterations might be needed to get a good estimate.

The derivatives needed for (5.4) can be calculated analytically, for the DC approach, as

$$F'_{\text{DC}}(d) = \frac{1}{N} \sum_{n=n_0}^{n_0+N-1} v(n) y'(n,d) \tag{5.5}$$

and

$$F''_{\text{DC}}(d) = \frac{1}{N} \sum_{n=n_0}^{n_0+N-1} v(n) y''(n,d) \tag{5.6}$$

and, for the ASDF approach, as

$$F'_{\text{ASDF}}(d) = \frac{2}{N} \sum_{n=n_0}^{n_0+N-1} (y(n,d) - v(n)) y'(n,d) \tag{5.7}$$

and

$$F''_{\text{ASDF}}(d) = \frac{2}{N} \sum_{n=n_0}^{n_0+N-1} y'(n,d)^2 + [y(n,d) - v(n)]\, y''(n,d). \tag{5.8}$$

The principle of the iterative estimator is depicted in Fig. 5.1. The computation of $F'(d)$ and $F''(d)$ changes depending on the cost function used. In Fig. 5.2 and 5.3 straightforward realizations of the derivatives of the cost functions (5.5)–(5.8) are seen. In Section 5.3 we will see how we can calculate $y'(n,d)$ and $y''(n,d)$, depending on the type of interpolation filter used.

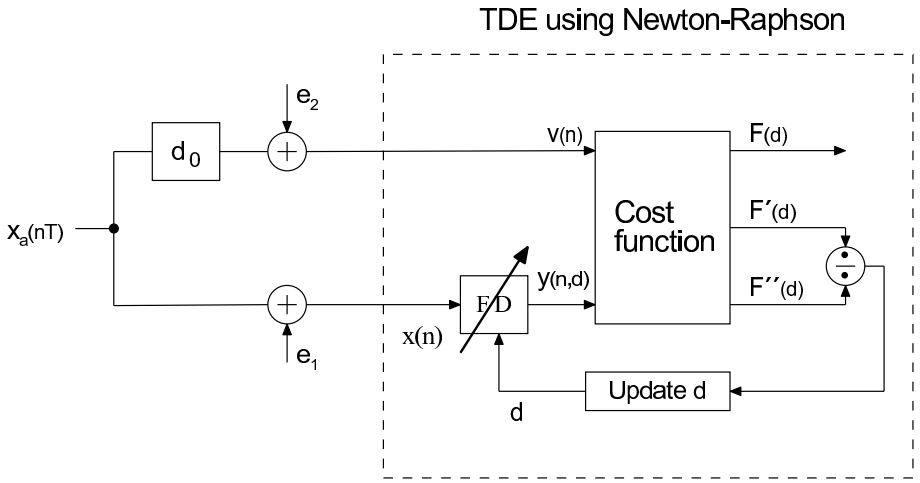


Figure 5.1. *TDE using Newton-Raphson.*

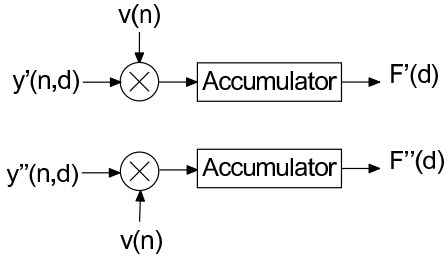


Figure 5.2. *Realization of the derivatives of the direct correlator (DC) cost function.*

As we will see later, the second derivative $F''(d)$ is fairly constant. To reduce the computational complexity we might use this fact to calculate the second derivative once and then use that value in the subsequent iterations. Another way is to recursively approximate the second derivative using the first order derivative.

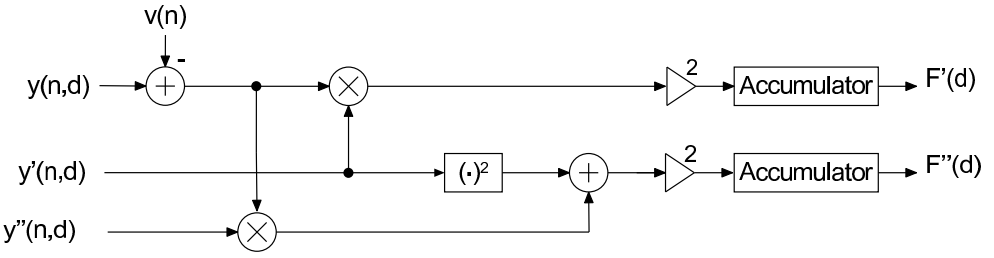


Figure 5.3. *Realization of the derivatives of the average squared difference function (ASDF) cost function.*

5.2.3 Recursive Gauss-Newton

Another way to find the minimum is to use the so-called recursive Gauss-Newton (RGN) technique, which was proposed for ASDF TDE in [24]. Compared to the Newton-Raphson algorithm the RGN uses an approximation of the second derivative.

For the ASDF cost function the approximation of the second derivative F''_{ASDF} is calculated as

$$R_{n+1} = \lambda R_n + \frac{1}{N} \sum_{n=n_0}^{n_0+N-1} y'(n,d)^2 \tag{5.9}$$

where λ is a forgetting factor. If λ is close to 0, only a few of the previous iterations are "remembered". The iterative estimate is now calculated as

$$\widehat{d}_{n+1} = \widehat{d}_n - \frac{F'_{\text{ASDF}}(\widehat{d}_n)}{R_n}. \tag{5.10}$$

Compared to the NR algorithm the RGN has a somewhat slower convergence, but in some applications this might not be very important.

5.3 Interpolation Methods

To be able to perform TDE using the principles presented in the previous section, approximations of intermittent sample values are needed, illustrated earlier in Fig. 4.2. The first method that probably comes to mind is linear interpolation. As we will see linear interpolation however has severe limitations.

Better solutions are to use adjustable fractional delay (FD) filters. Such a filter approximates an ideal delay $y(n,d) = x(n-d)$. The frequency function of such an ideal delay can be written as

$$H_d(e^{j\omega}) = e^{-j\omega d} \tag{5.11}$$

where d is the delay. However, an ideal interpolator is impossible to implement in practice and therefore we define the non-ideal FD filter as

$$H(e^{j\omega}, d) = A(\omega, d)e^{-j\omega(d+\tilde{d}(\omega,d))} \tag{5.12}$$

where $A(\omega, d) = 1 + \delta(\omega, d)$ is the filter magnitude, which should nominally be 1, and $\tilde{d}(\omega, d)$ is the phase delay error. We will later discuss the offset and variance of the TDE with respect to these errors. The idea to use FD filters in TDE has previously been described in for example [6]. However, the effects of the nonidealities have not been analyzed before to our knowledge.

We will now continue by looking at the first-order interpolation filter, followed by the general FIR and IIR FD interpolation filters.

5.3.1 First Order Linear Interpolation

The simplest way to interpolate samples is to use linear interpolation. The interpolated value of $x(n)$ that has been delayed $0 < d < 1$ can be written as

$$y(n,d) = \frac{x(n-1) - x(n)}{1}d + x(n) = (1-d)x(n) + dx(n-1). \tag{5.13}$$

It is easy to realize that the linear interpolator can be seen as a first order FIR filter with the transfer function

$$H(z,d) = (1-d) + dz^{-1} \tag{5.14}$$

and the respective frequency function,

$$H(e^{j\omega}, d) = (1-d) + de^{-j\omega}. \tag{5.15}$$

The magnitude error can then be calculated as

$$\delta(\omega, d) = \sqrt{1 + 2d\,(1-d)\,[\cos\omega - 1]} - 1 \tag{5.16}$$

and the phase delay error as

$$\tilde{d}(\omega, d) = -\frac{1}{\omega}\arg\{H(d, e^{j\omega})\} = \tag{5.17}$$

$$= \begin{cases} \frac{1}{\omega}\arctan\left(\frac{d\sin(\omega)}{1-d+d\cos(\omega)}\right) - d & \text{when } \cos(\omega) >= 1 - \frac{1}{d} \\ \frac{1}{\omega}\left(\arctan\left(\frac{d\sin(\omega)}{1-d+d\cos(\omega)}\right) + \pi\right) - d & \text{when } \cos(\omega) < 1 - \frac{1}{d}. \end{cases} \tag{5.18}$$

In Fig. 5.4 the phase delay and magnitude error for the first order interpolator can be seen. The delay error grows quickly with ω and therefore the first order interpolator is not a good choice.

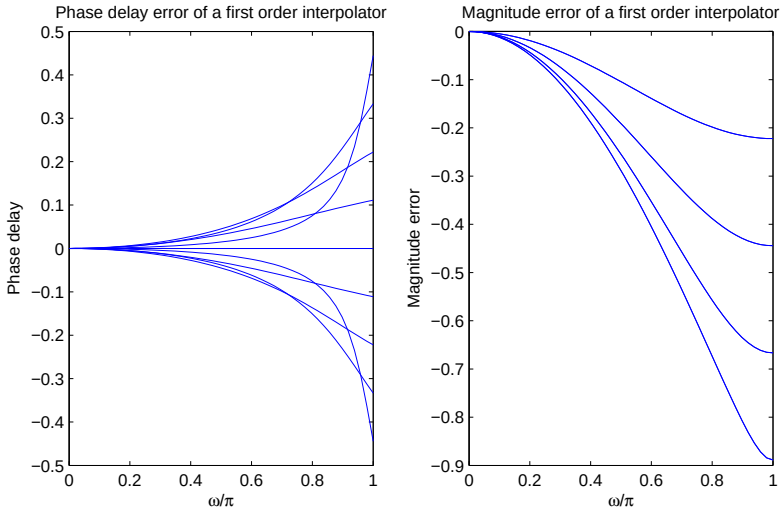


Figure 5.4. *The phase delay error and magnitude error of the first order interpolator vs. frequency.*

5.3.2 Interpolation Using FIR FD Filters

Higher order FIR filters can be designed with approximately linear phase, and hence an approximately equal delay for all frequencies. However, it is usually too computationally costly to redesign the filters for each desired delay. A solution to this is to use a structure proposed by Farrow in 1988 [7]. The basic idea is to approximate each coefficient in an FIR filter as a

polynomial. The impulse response can be written as

$$h(n,d) = \sum_{p=0}^{P} g_p(n) d^p, \quad n = 0, \ldots, M \tag{5.19}$$

where P is the order of the polynomial and $g_p(n)$ is the coefficient for the p-th order term for the n-th value in the impulse response $h(n,d)$.

The z-transform of (5.19) can be written as

$$\begin{aligned} H(z,d) &= \sum_{n=0}^{M} h(n,d) z^{-n} = \sum_{n=0}^{M} \left[\sum_{p=0}^{P} g_p(n) d^p \right] z^{-n} = \\ &= \sum_{p=0}^{P} d^p \left[\sum_{n=0}^{M} g_p(n) z^{-n} \right] = \sum_{p=0}^{P} d^p G_p(z) \end{aligned} \tag{5.20}$$

where $G_p(z) = \sum_{n=0}^{M} g_p(n) z^{-n}$ are identified as FIR subfilters. The corresponding structure of this filter is the so-called Farrow structure shown in Fig. 5.5.

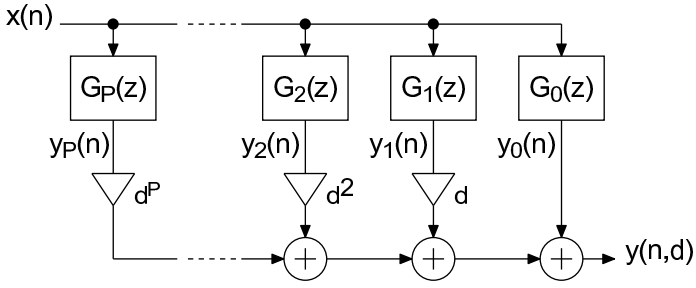


Figure 5.5. *The Farrow FIR FD structure.*

In this thesis we will restrict the subfilters to be linear-phase FIR filters of even order, say M, and with symmetric (anti-symmetric) impulse responses $g_p(n)$ for p even (p odd), i.e. $g_p(n) = g_p(M-n)[g_p(n) = -g_p(M-n)]$ for p even (p odd). The reason for using linear-phase filters is that they are easier to implement, requiring fewer multiplications, than completely general filters. Note that it is not necessary that the filter orders of the individual subfilters are equal. Furthermore, when the filters are even order linear-phase filters, the inherent delay of $H(z,d)$ is an integer, $M/2$. By inherent delay we mean the delay of the filter when $d = 0$. This is suitable for the time delay estimation problem.

From (5.20) and Fig. 5.5 it is seen that the filter output $y(n,d)$ from the FD filter can be written as

$$y(n,d) = \sum_{p=0}^{P} d^p y_p(n) \tag{5.21}$$

where

$$y_p(n) = g_p(n) * x(n). \tag{5.22}$$

Now, the derivatives of $y(n,d)$ with respect to d can be calculated analytically as

$$y'(n,d) = \sum_{p=1}^{P} p d^{p-1} y_p(n), \quad P > 0 \tag{5.23}$$

and

$$y''(n,d) = \sum_{p=2}^{P} k(p-1) d^{p-2} y_p(n), \quad P > 1. \tag{5.24}$$

If $P = 1$ the second derivative (5.24) will be equal to zero. These derivatives can then be used in (5.5)–(5.8) to calculate the next iterative step in the estimator. In Fig. 5.6 a straightforward implementation of the derivatives (5.23) and (5.24) can be seen.

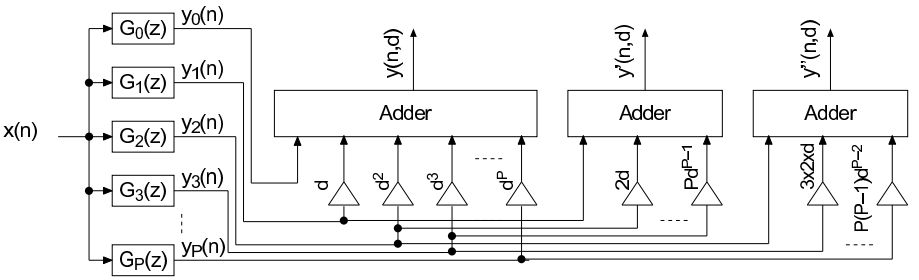


Figure 5.6. *A straightforward realization of the derivatives of the FIR FD filter.*

The magnitude of the FD filter can of course be calculated as $\left|H(d, e^{j\omega})\right|$, but if the subfilters $G_p(z)$ have been designed to have linear phase the mag-

nitude of the FD filter can also be be written as [28]

$$|H(d,e^{j\omega})| = \left| \sum_{p=1}^{\lfloor P/2 \rfloor} 2pd^{2p-1}G_{(2p)R}(\omega) + \right. \\ \left. + j \sum_{p=1}^{\lceil P/2 \rceil} (2p-1)d^{2p-2}G_{(2p-1)R}(\omega) \right| \tag{5.25}$$

where $G_{(2p)R}(\omega)$ and $G_{(2p-1)R}(\omega)$ are the zero-phase frequency response of $G_p(z)$. This expression will be useful later when we want to compute the analytical derivative of the filter magnitude $A(\omega, d) = 1 + \delta(\omega, d)$.

The phase delay is then computed from the definition of the phase delay as

$$\tau(\omega, d) = -\frac{1}{\omega} \arg\{H(d, e^{j\omega})\}. \tag{5.26}$$

If the subfilters have even order the inherent phase delay of $H(z, d)$, i.e. when $d = 0$, is equal to $M/2$. Using the expression above we can define the phase delay error for an FIR FD filter as

$$\tilde{d}(\omega, d) = \tau(\omega, d) - (M/2 + d). \tag{5.27}$$

Designing the FIR FD Filters

A number of techniques to design the FIR subfilters $G_p(z)$ exist, see for example [12], [30], and [9]. In this thesis we will use an optimization approach similar to the one presented in [9]. The goal is to reduce the estimator offset errors.

First we need to find a filter to use as an initial solution to the optimization problem. In the case of an FIR FD filter this can be done by designing the subfilters separately to approximate differentiators. For more details regarding the initial filters see [9].

As we will later see in (5.59) in Section 5.4.1, the main cause of estimator offset is the phase delay error $\tilde{d}(\omega, d)$, at least as long as the derivative of the filter magnitude is small. Hence, if we limit the phase delay error and the magnitude error (and indirectly the derivative of the magnitude) to be smaller than a constant C, the estimator offset will be small. The minimax problem can then be formulated as

$$\text{minimize } \epsilon \text{ subject to } \left|\tilde{d}(\omega, d)\right| < \epsilon \text{ and } |\delta(w, d)| < C \tag{5.28}$$

over a range of ω and d. The `fminimax`-routine in MATLAB efficiently implements a sequential quadratic programming method that is capable of solving the minimax constraint problem in (5.28). However, the solution is not guaranteed to be the global minimum as the problem is nonlinear.

As we will also see in Section 5.4.1, when the derivative of the magnitude error is nonzero the magnitude error $\delta(\omega, d)$ and the frequency ω will cause an additional estimator offset $d_{\text{err}}(\omega, d)$. So, instead of choosing the magnitude limit C "blindly" we can minimize the estimator offset directly by minimizing the sum of $\tilde{d}(\omega, d)$ and $d_{\text{err}}(\omega, d)$.

Formulated as a minimax problem this new problem can be written as

$$\text{minimize } \epsilon \text{ subject to } |\tilde{d}(\omega, d) + d_{\text{err}}(\omega, d)| < \epsilon \tag{5.29}$$

which is solved over a range of d and ω.

As an example, two FIR FD filters were optimized, one using (5.28) and the other using (5.29). The initial filter was found using the algorithm described in [9], which does not require that the filter orders of the subfilters are equal. The orders of the respective subfilters are $[14\ \ 10\ \ 12\ \ 8\ \ 10\ \ 6\ \ 6]$. The magnitude offset limitation was $C = 0.01$.

In Fig.5.7 and 5.8 the resulting magnitude $A(\omega, d)$ and phase delay error $\tilde{d}(\omega, d)$ and their derivatives for the two filters can be seen. The main difference between the two filters is the magnitude. We will later see in simulations how the estimator performs with these filters.

5.3.3 Interpolation Using All-pass IIR FD Filters

All-pass filters are filters with constant magnitude, which means that no frequency is attenuated in the filter. However, even though the magnitude is constant the filter has a non constant phase delay, meaning that not all frequencies are delayed equally.

The transfer function of a general all-pass filter with order M can be written as

$$H_A(z) = \frac{z^{-M} A(z^{-1})}{A(z)} \tag{5.30}$$

where

$$A(z) = 1 + \sum_{m=1}^{M} a_m z^{-m}. \tag{5.31}$$

By selecting a_m appropriately, the all-pass filter can be given an approximately linear phase, i.e., a constant phase delay. Now, if we want the filter

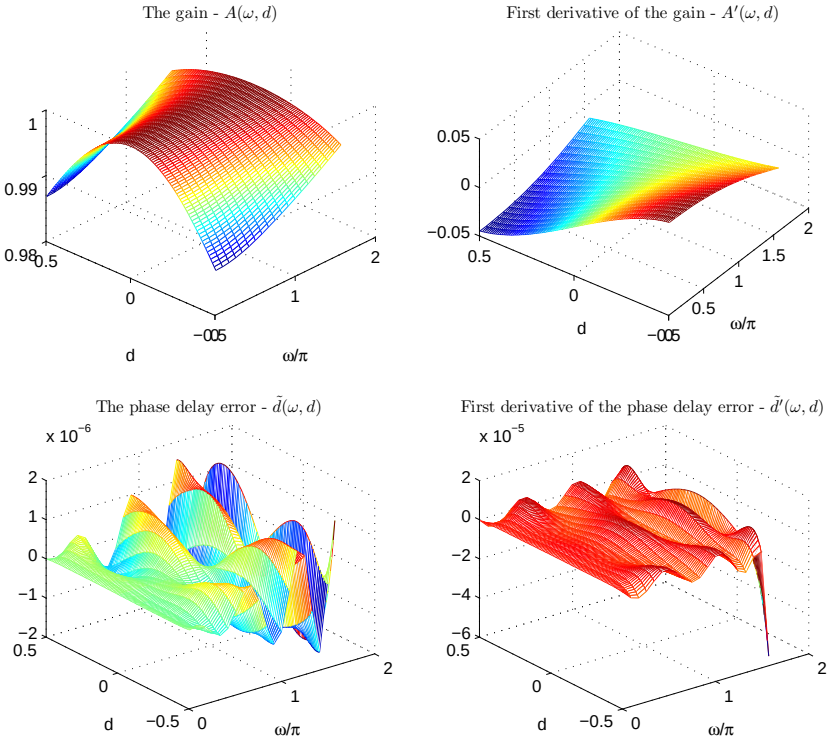


Figure 5.7. *Magnitude and phase delay error of an example filter optimized with respect to* $\tilde{d}(\omega, d)$.

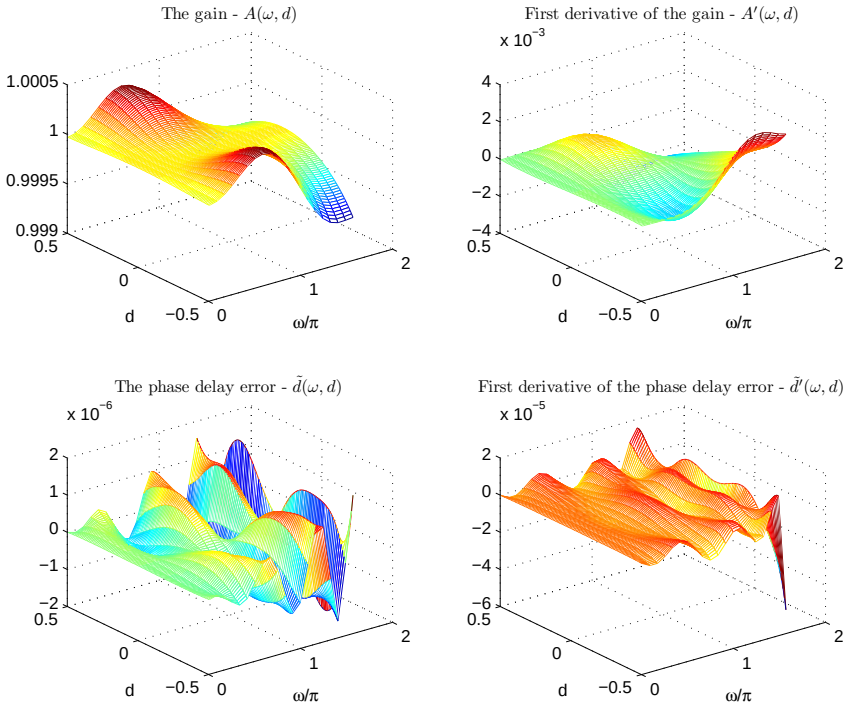


Figure 5.8. *Magnitude and phase delay error of an example filter optimized with respect to* $\tilde{d}(\omega, d) + d_{err}(\omega, d)$.

to be adjustable over a wide range of delays we can interpolate the filter coefficients using a degree P polynomial [15],

$$a_m(d) = \sum_{p=0}^{P} C_{pm} d^p. \tag{5.32}$$

Using (5.32) we can rewrite (5.31) as

$$A(z,d) = \sum_{m=1}^{M} \left(\sum_{p=0}^{P} C_{pm} d^p \right) z^{-m}. \tag{5.33}$$

The phase delay of the filter for a certain delay d can be written as

$$\tau_A(\omega, d) = M - \frac{2}{\omega} \arctan \left(\frac{\sum_{m=1}^{M} a_m(d) \sin(m\omega)}{1 + \sum_{m=1}^{M} a_m(d) \cos(m\omega)} \right) \tag{5.34}$$

where $a_m(d)$ is defined in (5.32). The inherent integer delay of the all-pass filter is equal to M, which means that when $d = 0$ the phase delay is equal to M. In Fig. 5.9 a realization of the IIR FD filter can be seen.

The difference equation corresponding to the all-pass FD filter (5.30) is equal to

$$y(n,d) = x(n-M) + \sum_{p=0}^{P} \sum_{m=1}^{M} d^p C_{pm} \left[x(n+m-M) - y(n-m,d) \right]. \tag{5.35}$$

The derivatives of (5.35) with respect to d are then calculated as

$$\begin{aligned} y'(n,d) = \sum_{p=1}^{P} \sum_{m=1}^{M} d^p \Big[& C_{pm} \frac{p}{d} (x(n+m-M) - \\ & y(n-m,d)) - C_{pm} y'(n-m,d) \Big] \end{aligned} \tag{5.36}$$

and

$$\begin{aligned} y''(n,d) = \sum_{p=2}^{P} \sum_{m=1}^{M} d^p \Big[& C_{pm} \frac{p}{d^2} (x(n+m-M) - \\ & y(n-m,d)) - C_{pm} y'(n-m,d) \Big]. \end{aligned} \tag{5.37}$$

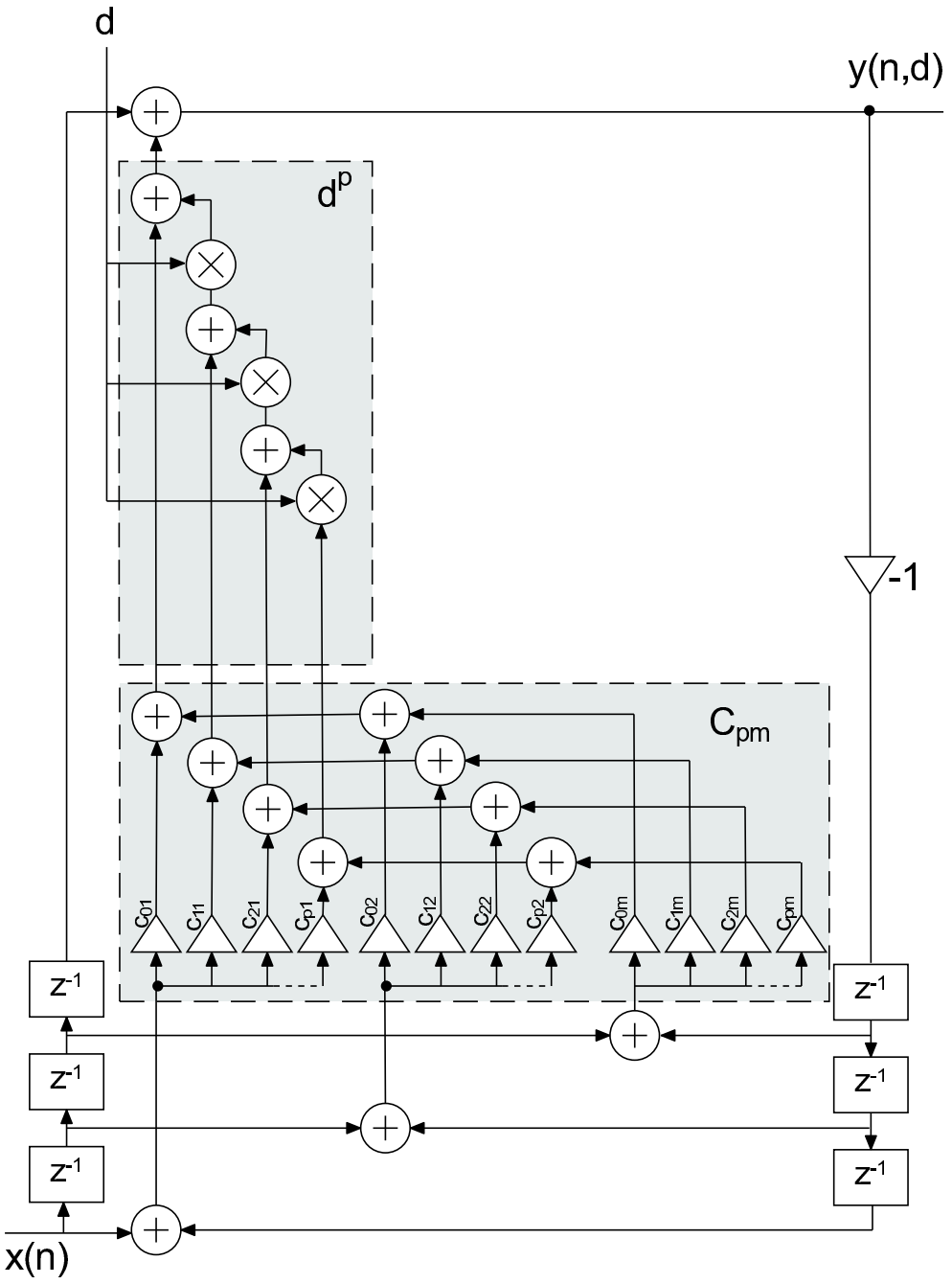


Figure 5.9. *A direct-form realization of an IIR FD filter.*

An all-pass FD filter has an initial transient response when a signal is applied to it that is, in theory, infinite in length. However, in practice the transient response decays rather quickly when the filter is stable. The rate of decay is direcly connected to the maximum pole radius of the filter and a larger radius means a longer transient response.

To illustrate this, the transient response of two FD filters with different maximal pole radii can be seen in Fig. 5.10. The first filter has a maximum pole radius of 0.98 and a maximum phase delay error of $3.12 \cdot 10^{-6}$. The second filter has a maximum pole radius of 0.87 and a maximum phase delay error of $3.31 \cdot 10^{-6}$. The filter order was $M = 8$ and $P = 5$. As it is seen, the worst case impulse response becomes longer when the pole radius is larger and more samples must be discarded. A smaller maximum pole radius gives a somewhat larger maximum phase delay error, though. Note that the transient responses have been generated by applying a sinusoid to the filter and by subtracting the theoretic output from the filter output, leaving only the ripple caused by the feedback in the filter.

In Fig. 5.11 a straightforward realization of the derivatives of the IIR all-pass FD filter can be seen. The derivatives can then be used in either an ASDF or a DC estimator. Because of the feedback inherent in the IIR filter the computation of the derivatives of an all-pass IIR FD filter is more complex than the derivatives of an Farrow FIR FD filter. However, if the derivatives are implemented in hardware the different building blocks can be shared and interleaved in time to lower the hardware cost.

Designing the IIR FD Filters

As when we designed FIR FD filters we will use a minimax optimization approach to design the IIR FD filters and hence a good initial FD filter is needed.

In 1971 Thiran presented a method to design recursive all-pole digital filters with maximally flat group delay [25]. If the poles instead are used in an all-pass filter we get twice the group delay, meaning that the delay-value in the equations can be halved. The resulting closed form filter coefficient expression for an Mth-order all-pass filter with a constant delay $D = d+M$ is

$$a_m = (-1)^m \begin{pmatrix} M \\ m \end{pmatrix} \prod_{n=0}^{M} \frac{D - M + n}{D - M + m + n} \text{ for } m = 0, 1, 2, ..., M. \quad (5.38)$$

Since the expression is developed with a maximally flat group delay in

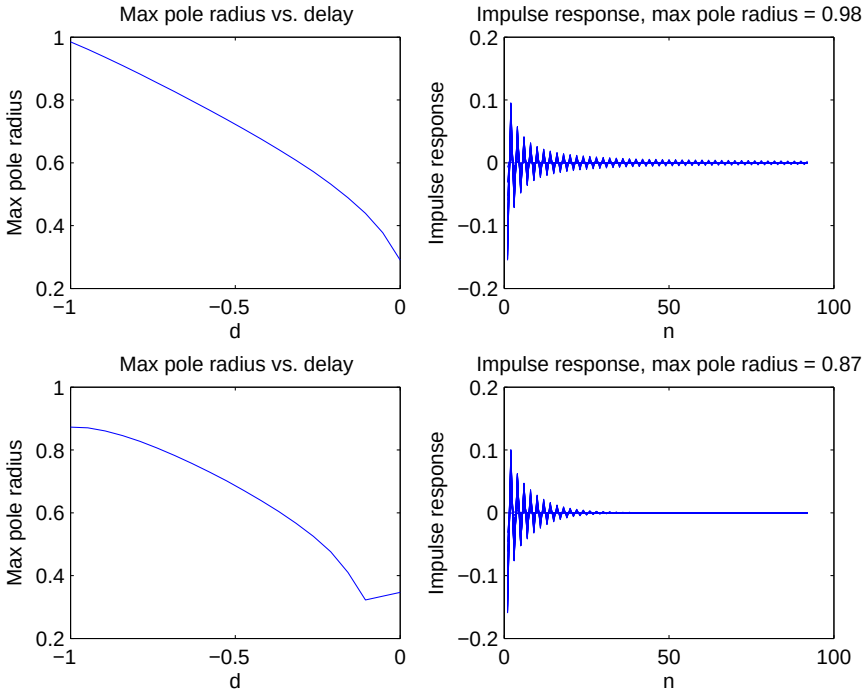


Figure 5.10. *Transient responses and maximum pole radius for two different filters.*

mind, the phase delay for low frequencies is excellent, but for higher frequencies the phase delay deviates more and more and the necessary filter order has to be increased rapidly. These problems are similar to the limitations of the Lagrange-interpolator method that can be used to design FIR FD filters.

An example of a filter designed using Thiran's method can be seen in Fig. 5.12. The phase delay error is small for low frequencies, but increases rapidly with frequency.

Another method which gives an initial filter with lower filter order is the least-squares (LS) method described in [12]. We will briefly summarize the idea behind it. The idea is to minimize the weighted least-squares cost function over a frequency range $[0 \cdots \alpha\pi]$ defined as

$$E = \frac{1}{\pi} \int_0^{\alpha\pi} W(\omega) |\tilde{d}(\omega)|^2 d\omega \tag{5.39}$$

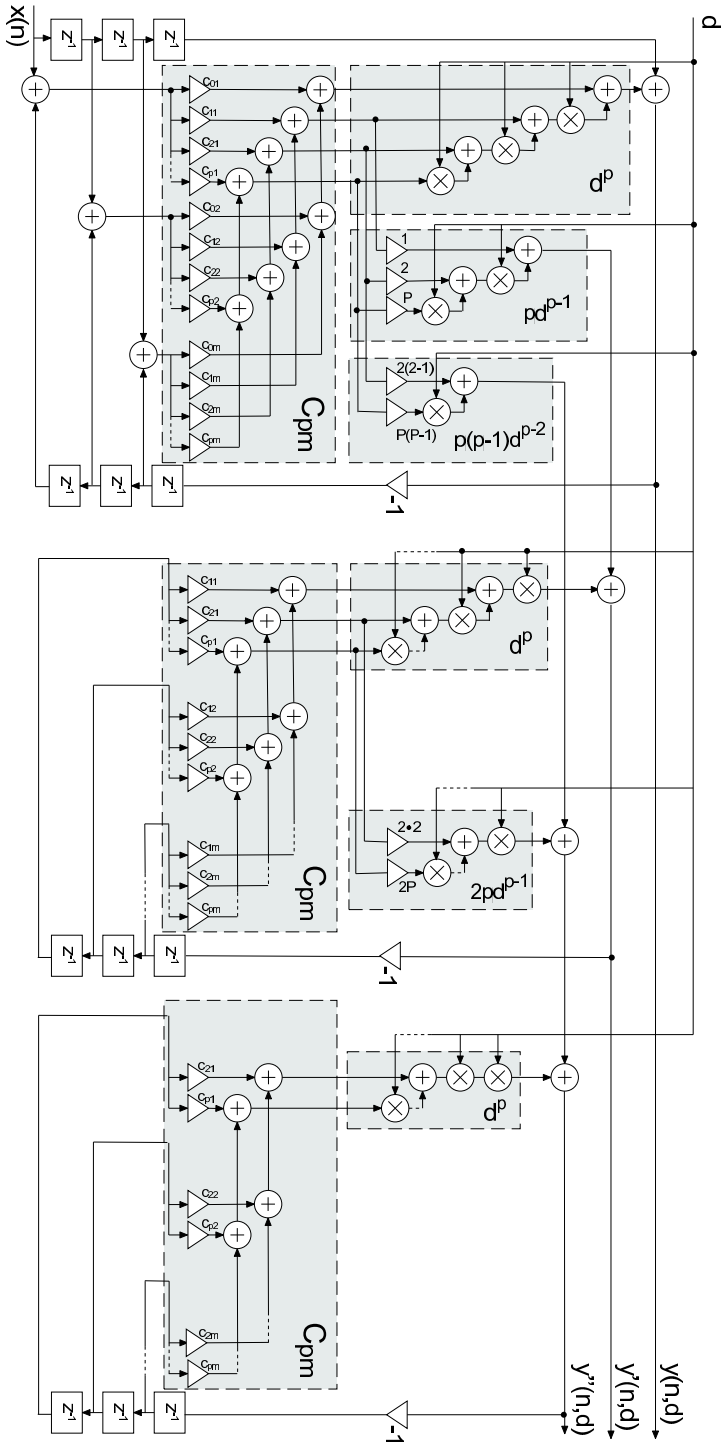


Figure 5.11. *A straightforward realization of the derivatives of an IIR FD filter.*

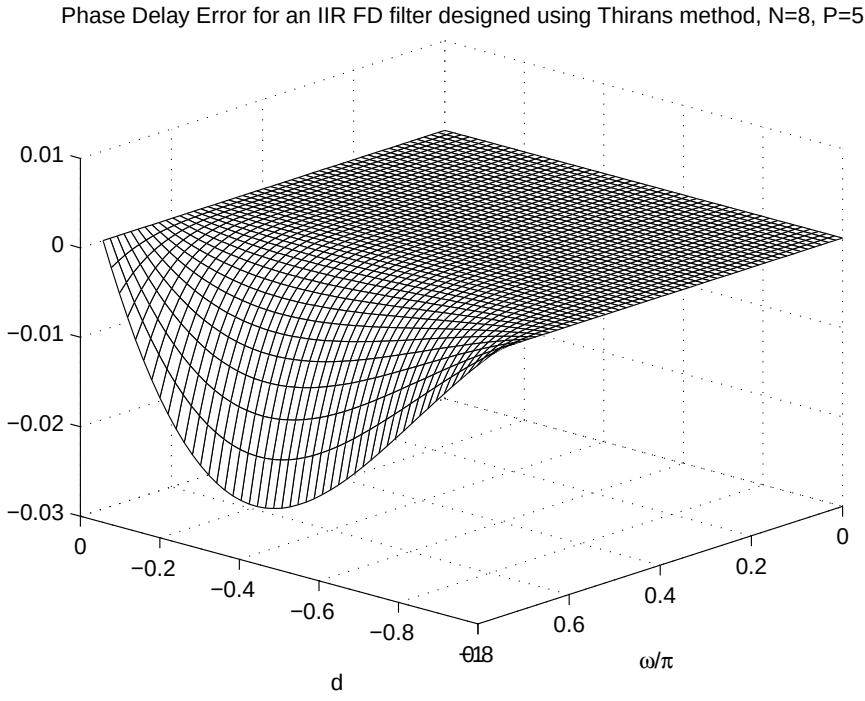


Figure 5.12. *The phase delay error of an IIR FD filter designed using Thiran's method vs. delay and frequency.* $N = 8$, $P = 5$.

where $W(\omega)$ is a weight function and

$$\begin{aligned}\tilde{d}(\omega) &= -\tau_{\text{id}}(\omega, d) + \tau_A(\omega, d) = \\ &= -(M + d) + (M - \frac{2}{\omega}\theta(\omega, d))\end{aligned} \tag{5.40}$$

is the difference between the ideal phase delay of an all-pass IIR FD filter with the inherent phase delay M and the phase delay for an all-pass filter defined in (5.30). By assuming constant coefficients and inserting (5.34) into (5.40) we get

$$\tilde{d}(\omega) = -d - \frac{2}{\omega}\theta(\omega) = \tag{5.41}$$

$$= -d - \frac{2}{\omega}\arctan\left\{\frac{\sum_{m=0}^{M} a_m \sin(m\omega)}{\sum_{m=0}^{M} a_m \cos(m\omega)}\right\}. \tag{5.42}$$

Using trigonometric identities this can be rewritten as

$$\tilde{d}(\omega) = -\frac{2}{\omega}\arctan\left\{\frac{\sum_{m=0}^{M} a_m \sin(-\frac{\omega d}{2} - m\omega)}{\sum_{m=0}^{M} a_m \cos(-\frac{\omega d}{d} - m\omega)}\right\}. \tag{5.43}$$

If we use the approximation $\arctan(x) \approx x$ and insert (5.43) into (5.39) we get

$$E_2 = \frac{1}{\pi}\int_0^{\alpha\pi} W(\omega)\left|\frac{\sum_{m=0}^{M} a_m \sin(-\frac{\omega d}{2} - m\omega)}{\sum_{m=0}^{M} a_m \cos(-\frac{\omega d}{d} - m\omega)}\right|^2 d\omega. \tag{5.44}$$

The filter coefficients are both in the numerator and denominator, which makes it difficult to find the optimum coefficients directly from E_2. One way to find suboptimal coefficients is to view the denominator as constant and neglect it. This will lead to an offset from the correct optimum, but it is noted in [12] that the offset is small and it is still good enough as an initial filter.

If the denominator is neglected and we let

$$\mathbf{s} = \left[\sin\left(-\frac{\omega d}{2}\right)\ \sin\left(-\frac{\omega d}{2} - \omega\right)\cdots\sin\left(-\frac{\omega d}{2} - M\omega\right)\ \right] \tag{5.45}$$

and $\mathbf{a} = [a_0\ \ a_1\ \ a_2 \cdots a_M]^T$ we can rewrite the cost function (5.44) using vector notation as

$$E_3 = \mathbf{a}\left[\frac{4}{\pi}\int_0^{\alpha\pi} W(\omega)\frac{\mathbf{s}^T\mathbf{s}}{\omega^2}d\omega\right]\mathbf{a}^T = \mathbf{a}\mathbf{S}\mathbf{a}^T. \tag{5.46}$$

We now let the weight function $W(\omega)$ be piecewise constant. If we let $W(\omega) = 1$ in the frequency range then the matrix elements of the matrix $\mathbf{S}$ are equal to

$$S_{k,l} = \frac{4}{\pi} \int_0^{\alpha\pi} \frac{1}{\omega^2} \left\{ \cos[(k-l)\omega] - \cos[(N-(k+l+d))\omega] \right\} d\omega$$
$$k, l = 1, 2, \ldots, P. \qquad (5.47)$$

The integral can not be solved analytically, we have to use either numerical integration or some other approximations. We can use the fact that

$$\int \frac{cos(ax)}{x^2} = -aD(ax) - \frac{\cos(ax)}{x} \qquad (5.48)$$

where

$$D(x) = \int_0^x \frac{\sin t}{t} dt = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{(2n+1)(2n+1)} \, t \geq 0. \qquad (5.49)$$

A good approximation of $D(x)$ is obtained using a rather small number of terms in the summation.

Now, let the first coefficient in the vector $\mathbf{a}$ be equal to 1 and define $\mathbf{a}_1 = [a_1 a_2 \ldots a_N]^T$ as the rest of the coefficients. The cost function E_3 can now be written as

$$\mathbf{a}\mathbf{S}\mathbf{a}^T = [1\mathbf{a}_1^T] \begin{bmatrix} s_0 & \mathbf{S}_1^T \\ \mathbf{s}_1 & \mathbf{S}_1 \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{a}_1 \end{bmatrix}$$
$$= \mathbf{a}_1^T \mathbf{S}_1 \mathbf{a}_1 + 2\mathbf{s}_1^T \mathbf{a}_1 + s_0. \qquad (5.50)$$

The minimum of this quadratic equation is equal to

$$\mathbf{a}_1 = -\mathbf{S}_1^{-1} \mathbf{s}_1. \qquad (5.51)$$

Now we have the initial filter coefficients for a constant delay d, but we also need to find the polynomials so that we can interpolate the filter coefficients for different delays without having to redesign the filter for every new d. This can for example be done using the well known least squares fitting.

Let N_d be the number of delay values that we have computed the initial filter for and let each row in the matrix

$$\mathbf{A} = \begin{bmatrix} 1 & a_{1,1} & a_{2,1} & \dots & a_{P,1} \\ 1 & a_{1,2} & a_{2,2} & \dots & a_{P,2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_{1,N_d} & a_{2,N_d} & \dots & a_{P,N_d} \end{bmatrix} \tag{5.52}$$

contain the filter coefficients for the different delays, found using the method above. Furthermore we let the matrix

$$\mathbf{B} = \begin{bmatrix} d_1^0 & d_1^1 & \dots & d_1^{P-1} & d_1^P \\ d_2^0 & d_2^1 & \dots & d_2^{P-1} & d_2^P \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{N_d}^0 & d_{N_d}^1 & \dots & d_{N_d}^{P-1} & d_{N_d}^P \end{bmatrix} \tag{5.53}$$

contain the different delays to the power of 0 to P. The least squares polynomial fit can then be calculated as

$$C = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}\mathbf{A}. \tag{5.54}$$

Now when we have the initial filter we can optimize this filter further using sequential quadratic programming. Let $\tilde{d} = (M+d) - \tau_A(\omega, d)$ be the phase delay error. The minimax solution is then obtained by minimizing

$$\text{minimize } \epsilon \text{ subject to } |\tilde{d}| < \epsilon \tag{5.55}$$

over a range of ω and d. To preserve the stability of the filter we must check that the poles remain inside the unit circle. As we saw before, the transient response depends on the maximum pole radius and by forcing the poles to have a distance margin from the unit circle we can reduce the transient response.

It is noted in [12] that the method described above is not guaranteed to converge if the order M of the initial filter is chosen too large and therefore an iterative method is proposed instead. First, design a filter with order M. Then, extend the order to $M+1$ by adding a zero coefficient and use this new filter as the starting point for the minimax optimization.

As an example an IIR FD filter with $M = 8$, $P = 5$ and $\omega_{\text{max}} = 0.75\pi$ was optimized. The initial FD filter had a maximum phase delay error of

$3.8726 \cdot 10^{-4}$ samples and can be seen in Fig. 5.13. The number of ripples is related to the order M. As a side note, an IIR FD filter designed with Thiran's method and comparable phase delay error requires a filter order $M = 29$.

After the initial FD filter has been optimized with respect to the phase delay error, the maximum delay error was $9.4238 \cdot 10^{-5}$. The resulting filter can be seen in Fig. 5.14. One of the features of minimax optimization is that the resulting FD filter becomes approximately equiripple, which is easily seen in the figure.

5.4 Error Analysis

The interpolation errors and noise will affect the performance of the estimator, we will now see how and to what amount. We have mainly considered narrow-band signals, but wideband signals are also covered briefly. In an application where the TDE is used for calibration we can usually select the signal, e.g. as a sinusoid. The estimator is, however, not restricted to single-frequency signals, as we will see in in Section 5.6.

We begin by looking at the estimator offset, which is mainly caused by the phase delay error of the filter, but also by the magnitude offset. After that we look at the estimator variance, which is mainly caused by additive noise, but also by batch truncation.

5.4.1 Estimator Offset

Narrow-band Signals

We consider two types of FD filter errors, magnitude errors and time delay errors. Let $\delta(d, \omega_0)$ denote the magnitude error and let $\tilde{d}(d, \omega_0)$ denote the delay error of the FD filter for a certain d and a certain ω_0.

Now, if we assume that $x(n)$ is sinusoidal, the output from the FD filter at a frequency ω_0 is

$$y(n, d) = (1 + \delta(\omega_0, d)) \sin(\omega_0(n - d - \tilde{d}(\omega_0, d))) \tag{5.56}$$

and the reference signal $v(n)$ is

$$v(n) = \sin(\omega_0(n - d_0)). \tag{5.57}$$

Henceforth, the dependence of δ and δ' on d and ω_0 will be omitted in the notation for the sake of simplicity. After inserting (5.56) and (5.57) into (5.7)

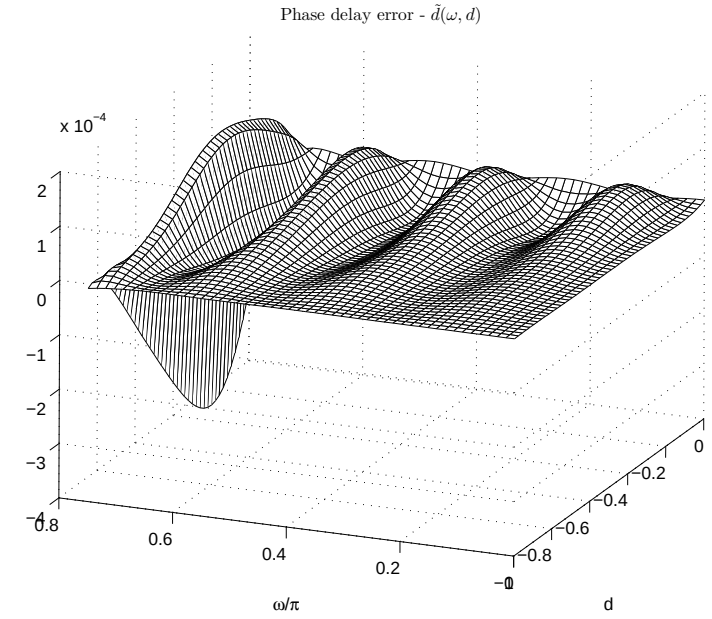


Figure 5.13. *The phase delay error for different d and ω, before minimax optimization.*

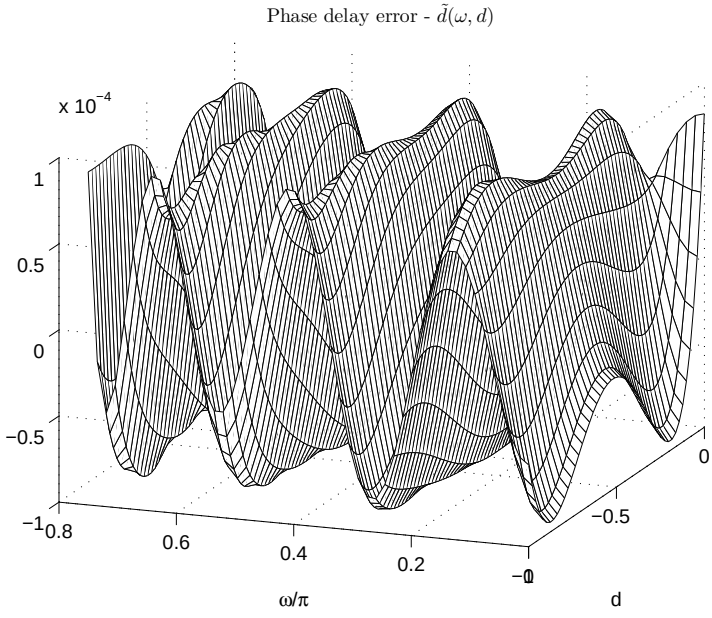


Figure 5.14. *The phase delay error for different d and ω, after minimax optimization.*

and some simplifications we arrive at

$$F'(\omega_0, d) = F'_0(\omega_0, d) + F'_N(\omega_0, d) \tag{5.58}$$

where

$$\begin{aligned} F'_0(\omega_0, d) = \delta'[(1+\delta) - \cos(\omega_0(d - d_0 + \tilde{d}))] + \\ + (1+\delta)\omega_0(1+\tilde{d}')\sin(\omega_0(d - d_0 + \tilde{d})) \end{aligned} \tag{5.59}$$

does not depend on the batch length N and

$$\begin{aligned} F'_N(\omega_0, d) = \frac{1}{N}\Big[&(1+\delta)\omega_0(1+\tilde{d}')\sin(\omega_0(-(d_0+d+\tilde{d}))- \\ &- (1+\delta)^2\omega_0(1+\tilde{d}')\sin(\omega_0(-(d+\tilde{d}))+ \\ &+ \delta'\cos(\omega_0(-(d_0+d+\tilde{d}))- \\ &-(1+\delta)\delta'\cos(\omega_0(-(d_0+\tilde{d}))\Big]\frac{\sin(N\omega_0)}{\sin(\omega_0)} \end{aligned} \tag{5.60}$$

contains the terms that do.

In Section 5.4.2 we will see that if the initial phase of the signals is unknown and random, F'_N is also random with zero mean. If we assume that N is so large that F'_N can be approximated as zero, it can be seen that, if δ' is zero, F'_0 will be zero when $d = d_0 - \tilde{d}$, irrespective of δ.

On the other hand, if δ' is small, but not zero, the effect of a constant magnitude error δ will affect the estimator. The effect of a delay error derivative $\tilde{d}'$ is normally negligible since it tends to be small compared to 1.

The iterative Newton-Raphson algorithm in (5.4) will, if the function is behaving well, converge towards the zero of F'_0. Unfortunately, it is impossible to directly from (5.59) analytically find the d that makes F'_0 zero. To find an approximation of the estimation error we do a first-order Taylor expansion of the sine and cosine in F'_0 and write it as

$$F'_0 \approx \delta'(\delta + \frac{\omega_0^2}{2}(d - d_0 + \tilde{d})^2) - (1+\delta)(1+\tilde{d}')\omega_0^2(d - d_0 + \tilde{d}). \tag{5.61}$$

If $1+\tilde{d}'$ and $1+\delta$ are approximated by 1, we can solve for $d_{\text{err}} \approx d - d_0$ and get

$$d_{\text{err}} \approx -\frac{1}{\delta'} + \frac{1}{\omega_0\delta'}\sqrt{\omega_0^2 - 2(\delta')^2\delta} - \tilde{d}. \tag{5.62}$$

This expression can be used to predict the final estimation error and to design FD filters which take the error caused by magnitude errors into account. The first part becomes small when ω_0^2 is large compared to $2(\delta')^2\delta$ and the error will then be dominated by $\tilde{d}$. For small ω_0, if δ or δ' are not zero, the error deviates more and more from $\tilde{d}$.

The derivative of δ for a Farrow FIR FD filter can be found analytically by noting that the magnitude can be rewritten as [28]

$$\begin{aligned} |H(d,e^{j\omega})| &= \left| \sum_{k=1}^{\lfloor L/2 \rfloor} d^{2k} G_{(2k)R}(\omega T) + \right. \\ &\quad \left. + j \sum_{k=1}^{\lceil L/2 \rceil} d^{2k-1} G_{(2k-1)R}(\omega) \right| \\ &= |A(d) + jB(d)| = 1 + \delta. \end{aligned} \tag{5.63}$$

Since the derivative of the magnitude is equal to the derivative of the magnitude error we can calculate δ' as

$$\delta' = 2\frac{A(d)A'(d) + B(d)B'(d)}{\sqrt{A(d)^2 + B(d)^2}} \tag{5.64}$$

where $A'(d)$ and $B'(d)$ can be calculated as

$$A'(d) = \sum_{k=1}^{\lfloor L/2 \rfloor} 2k d^{2k-1} G_{(2k)R}(\omega T) \tag{5.65}$$

and

$$B'(d) = \sum_{k=1}^{\lceil L/2 \rceil} (2k-1) d^{2k-2} G_{(2k-1)R}(\omega T). \tag{5.66}$$

Since the magnitude of an all-pass FD filter is constant, the derivative of the magnitude error $\delta' = 0$, which means that if an all-pass FD filter is used the offset error simply becomes $d_{\text{err}} = \tilde{d}$. To minimize the estimator offset, all we have to do is minimize to $\tilde{d}$, see Section 5.3.3.

Wideband Signals

Now we will discuss what happens when a wideband signal is applied to the estimator. An ideal bandlimited, but wideband, signal and its delayed

version are difficult to generate synthetically. One way to generate an approximation is to delay the signal with an FD filter with a much smaller error than the one used in the estimator. Another is to approximate a wideband signal using a sum of many sinusoids with random frequency and phase. We will use such a signal to analyze the wideband offset and later compare it in simulations.

As input signal we use a sum of K sinusoids at random frequencies, with random phases φ_k and amplitudes $A(\omega_k)$, according to

$$y(n,d) = \sum_{k=1}^{K} A(\omega_k) \sin(\omega_k(n - d - \tilde{d}_k(\omega_k, d)) + \varphi_k) \tag{5.67}$$

and

$$v(n) = \sum_{k=1}^{K} \sin(\omega_k(n - d_0) + \varphi_k). \tag{5.68}$$

To simplify the calculation we here assume that the derivatives of A and $\tilde{d}$ are so small that they can be neglected. The derivative of $y(n,d)$ can then be calculated as

$$y'(n,d) = -\sum_{k=0}^{K-1} A_k \omega_k \cos(\omega_k(n - d - \tilde{d}_k) + \varphi). \tag{5.69}$$

The static part of the first derivative of $F'(d)$ can then be written as

$$F_0'(d) = -\sum_{k=0}^{K-1} A_k \sin(\omega_k(d_0 - d - \tilde{d}_k)) \approx -\sum_{k=0}^{K-1} A_k \omega_k (d_0 - d - \tilde{d}_k). \tag{5.70}$$

To analytically find the value for d where $F_0'(d) = 0$ is difficult since $\tilde{d}_k$ can only be expressed as the arctangent of an order P polynomial. However, it is possible to calculate it numerically. It also turns out that the estimator offset can be approximated quite well as the mean of $\tilde{d}(\omega_k, d_0)$ over the frequencies ω_k.

5.4.2 Estimator Variance

There are several causes for estimator variance. First we have the additive noise, seen as e_1 and e_2 in Fig. 4.1. Another is the variance caused by the truncation of signal batch length. In a fixed point implementation we

will also have quantization noise, but it can be made arbitrarily small by increasing the wordlengths.

The following analysis has mainly been performed using a signal consisting of a single sinusoid, but we will also briefly look at wideband signals. This could be seen as a limitation of the results, but in many applications the signal can be freely selected and hence the results can be useful if a sinusoid is chosen.

In this thesis we have treated the frequency of single frequency signals as unknown, but fixed. If the frequency is allowed to vary randomly over a frequency interval it will, because of the nonidealities in the FD filters, introduce randomness in the estimate too. We will briefly discuss this in this section.

Additive Noise

Additive noise is of course the main source of estimator variance. We have not yet analyzed the relation between additive noise and estimator variance theoretically, but we will see the effects of additive noise in Section 5.6.

Limited Batch Length Effects for Sinusoids

In simulations it is seen that, when the noise becomes small, the DC algorithm reaches an error floor. This was noted in [8] and it was explained that it was due to the truncation of the estimation window. We will now see why there is a difference between the DC and ASDF based estimators.

For this analysis we have assumed a sinusoidal input with unknown frequency and a random initial phase φ. Let

$$v(n) = \sin(\omega(n - d_0) + \varphi) \tag{5.71}$$

be the reference signal with an unknown delay and let

$$y(n, d) = A(\omega, d)\sin(\omega(n - d - \tilde{d}(\omega, d)) + \varphi) \tag{5.72}$$

be the output from the fractional delay filter, where $A(\omega, d)$ is the relative magnitude of $y(n, d)$ compared to $v(n)$ and $\tilde{d}(\omega, d)$ is the phase delay error of the FD filter.

Note that the relative magnitude $A(\omega, d)$ can be caused both by the filter, as is the case when an FIR FD filter is used as the interpolator, and/or by mismatch in the system. In the case when an IIR FD filter is used we assume that the relative magnitude is constant for all frequencies and delays.

We have also assumed the we have no additive noise since this analysis will mainly consider a situation where a finite batch length is the main source of estimator variance.

We begin by studying the ASDF estimator. If we insert (5.71) and (5.72) into (5.7) and assume that $\tilde{d}'$ and $\tilde{d}''$ are negligible we get

$$F'_{\text{ASDF}}(d) = F'_0(d) + F'_N(d) \tag{5.73}$$

where

$$F'_0(d) = A\omega \sin\left(\omega\left(d_0 - d - \tilde{d}\right)\right) \tag{5.74}$$

and

$$\begin{aligned} F'_N(d) = \frac{1}{N}\sum_{n=0}^{N-1} A\omega \Big[& A\sin(2\omega(n + d + \tilde{d}) + 2\varphi) - \\ & - \sin(\omega(2n + d_0 + d + \tilde{d}) + 2\varphi)\Big] \end{aligned} \tag{5.75}$$

and

$$F''(d) = F'_0(d) + F'_N(d) \tag{5.76}$$

where

$$F''_0(d) = A\omega^2 \cos(\omega(d_0 - d - \tilde{d})) \tag{5.77}$$

and

$$\begin{aligned} F''_N(d) = \frac{1}{N}\sum_{n=0}^{N-1} A\omega^2 \Big[& 2\cos^2(\omega(n + d + \tilde{d}) + 2\varphi) - \\ & - 2A\sin^2(2\omega(n + d + \tilde{d}) + 2\varphi) - \\ & - \cos(\omega(2n + d_0 + d + \tilde{d}) + 2\varphi)\Big]. \end{aligned} \tag{5.78}$$

F'_N and F''_N depend on the batch size N and tend toward 0 when N becomes large. However, since F'_N and F''_N also depend on the random unknown initial phase φ, they will contribute to the variance of the final estimate.

In Fig. 5.15 an example of F'_N for different N can be seen. When N becomes large F'_N decays to zero.

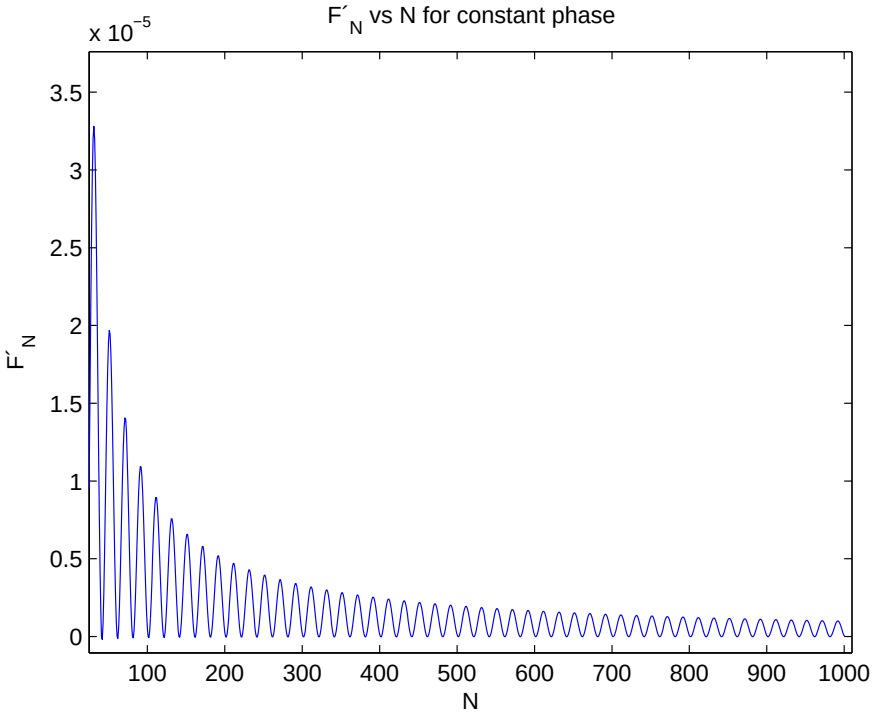


Figure 5.15. *F'_N for different N and constant phase.*

Using trigonometric identities we can rewrite (5.75) as

$$F'_N(d) = \frac{1}{N} A\omega \frac{\sin(N\omega)}{\sin(\omega)} \left[A \sin\left(2\omega\left(d + \tilde{d} + \frac{N-1}{2}\right) + 2\varphi\right) - \right. \\ \left. - \sin\left(2\omega\left(\frac{d_0 + d + \tilde{d}}{2} + \frac{N-1}{2}\right) + 2\varphi\right)\right]. \tag{5.79}$$

We will now analyze F'_N to see how it affect the variance of the final estimate. If we calculate the mean of F'_N and F''_N for a random phase φ we get

$$E\{F'_N(d)\} = \int_0^{2\pi} F'_N(d) d\varphi = 0. \tag{5.80}$$

The variance can then be calculated as

$$\text{var}\{F'_N\} = E\{(F'_N - E\{F'_N\})^2\} = \int_0^{2\pi} (F'_N(d))^2 d\varphi = \tag{5.81}$$

$$= \left(\frac{A\omega \sin(N\omega)}{N \sin(\omega)}\right)^2 \left(\frac{A^2+1}{2} - A\cos(\omega(d + \tilde{d} - d_0))\right). \tag{5.82}$$

When $d + \tilde{d}$ is close to d_0 the variance of F'_N can be approximated as

$$\text{var}\{F'_N\} \approx \left(\frac{A\omega \sin(N\omega)}{N \sin(\omega)}\right)^2 \frac{(A-1)^2}{2} \tag{5.83}$$

which is very small when A is close to 1. However, if A differs from 1, the variance increases.

The variance for the first derivative of the DC cost function can analogously be calculated then we get

$$\text{var}\{F'_N\} = \left(\frac{A\omega \sin(N\omega)}{N \sin(\omega)}\right)^2 \frac{1}{2}. \tag{5.84}$$

From this we see that the truncation induced variance for the DC estimator is relatively insensitive to small magnitude errors, but larger compared to the ASDF estimator.

Now we will see how the variance of F'_N affects the the variance of the estimator. Using the notation introduced in (5.75) and (5.78) the iterative Newton-Raphson update equation (5.4) can be written as

$$\widehat{d}_{n+1} = \widehat{d}_n - \frac{F'_0 + F'_N}{F''_0 + F''_N}. \tag{5.85}$$

By using Taylor series expansion the second part of (5.85) can then be written as

$$\frac{F_0' + F_N'}{F_0'' + F_N''} \approx \frac{F_0'}{F_0''} + \frac{1}{F_0''}F_N' - \frac{F_0'}{(F_0'')^2}F_N'' - \frac{1}{(F_0'')^2}F_N'F_N''. \tag{5.86}$$

The first term, $\frac{F_0'}{F_0''}$, is the desired iterative step and it can be approximated as

$$\begin{aligned}\frac{F_0'(d_n)}{F_0''(d_n)} &= -\frac{\sin(\omega(d_0 - d_n - \tilde{d}))}{\omega\cos(\omega(d_0 - d_n - \tilde{d}))} \approx \\ &\approx d_n - (d_0 - \tilde{d})\end{aligned} \tag{5.87}$$

which means that the iterative step ideally is the difference between the current estimate d_n and the unknown delay d_0 minus the delay error $\tilde{d}$. However, the remaining terms in (5.86) will contribute to the variance of the estimate and will limit the obtainable accuracy of the estimator.

As we will later confirm in simulations, the main variance contributor in (5.86) is the term $\frac{1}{F_0''}F_N'$, i.e. a good approximation of the estimator variance induced by the finite batch length can be calculated using the second term of (5.86) as

$$\text{var}\{\widehat{d}\} \approx \frac{1}{(F_0'')^2}\text{var}\{F_N'\}. \tag{5.88}$$

In Fig. 5.16 the theoretical variance for the DC and ASDF estimator for different N, a constant phase and frequency $\omega = 0.05\pi$ and $A = 1.001$ is plotted. The variance for the ASDF estimator is much lower when A is close to 1. If $A = 1$ the ASDF estimator would practically be unaffected by N.

It is also seen that the variance sometimes drops significantly. That happens at certain N's where the sums in F_N' and F_N'' adds up to zero. However, since the frequency ω is generally not known exactly we cannot always choose N that results in such a small variance.

Wideband Signals

For a wideband signal the limited batch length effect will also affect the performance. We will later see in simulations that the ASDF estimator performs better than the DC estimator for a wideband signal.

The performance is limited by the phase delay ripple of $\tilde{d}(\omega)$. An example of phase delay ripple can be seen in Fig. 5.17, where the phase delay

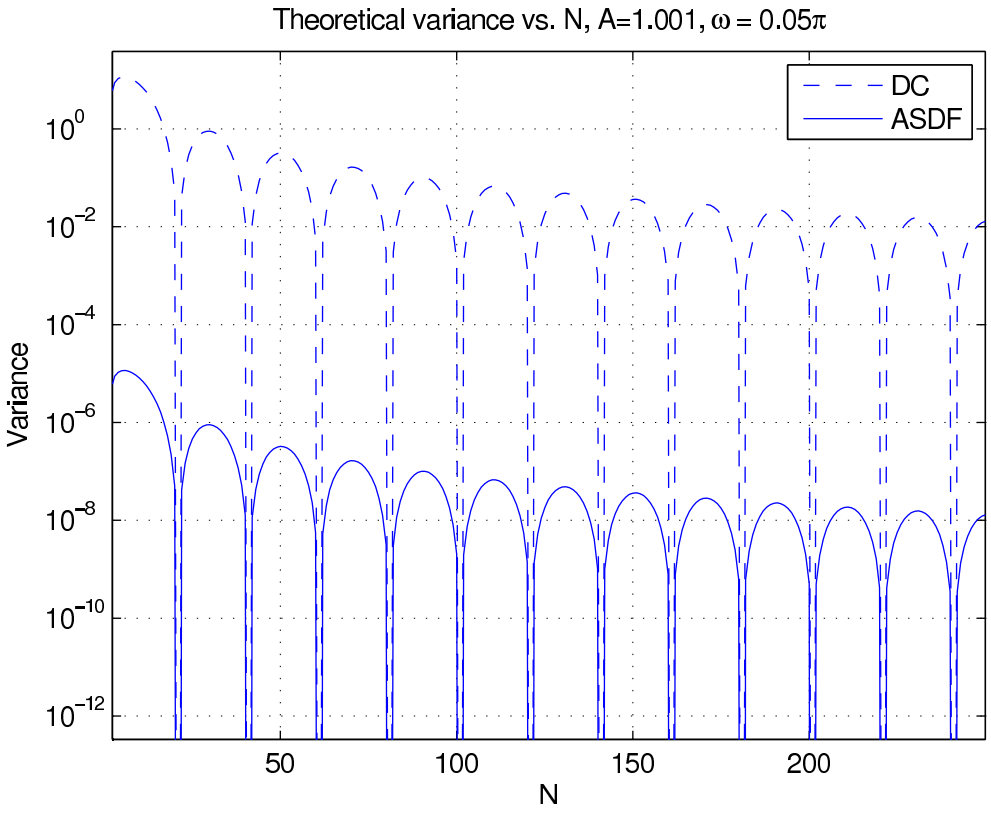


Figure 5.16. *Example showing the theoretical variance for DC and ASDF estimators.*

of a FIR FD filter has been plotted. Assume that the frequency of the input signals are random. Since the estimator offset is so closely related to the phase delay error the estimate will also vary, as seen in Fig. 5.17, and hence increase the estimator variance. It is hard to compute a closed form expression for the variance caused by the phase delay ripple.

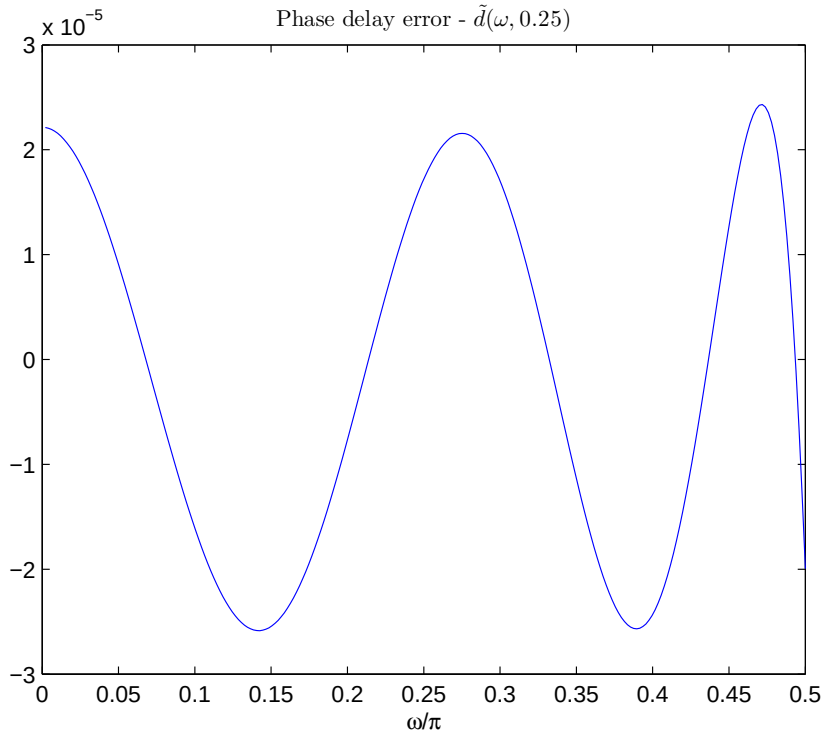


Figure 5.17. *Example of the phase delay error $\tilde{d}(\omega, d)$ for d=0.25.*

5.5 Complexity

Let M_k denote the filter order of the FIR subfilter $G_k(z)$. In Table 5.1 the number of multiplications, additions and delays needed per iteration to calculate $y(n, d)$, $y'(n, d)$, and $y''(n, d)$ for the FIR and IIR FD filters, i.e. Newton-Raphson is assumed. When the DC (ASDF) cost function is used an additional complexity of 2 multiplications and 2 additions (3 multiplica-

tions and 5 additions) per sample is added to the expressions in the table. Hence, the difference in complexity is small between the DC and ASDF cost function.

	FIR	All-pass
Mults/sample	$\frac{1}{2}\sum_{k=0}^{P} M_k + \lfloor \frac{P}{2} \rfloor + 3(P-1)$	$3P(M+3)-4$
Adds/sample	$\sum_{k=0}^{P} M_k + 3(P-1)$	$3P(M+1)+M-4$
Delay	$\sum_{k=0}^{P} M_k + \lfloor \frac{P}{2} \rfloor$	$4M$

Table 5.1. *The number of operations needed per sample for an FIR FD filter and an IIR FD filter and Newton Raphson.*

If Recursive Gauss-Newton is used instead of Newton-Raphson we do not need to calculate the second derivative, although the number of required iterations increase. In Table 5.2 the complexity for the compuation of $y(n,d)$ and $y'(n,d)$ is seen for this case. The additional cost for RGN is 2 multiplications and 3 additions per sample. The complexity per sample is obviously lower than for Newton-Raphson, however more iterations are needed.

	FIR	All-pass
Mults/sample	$\frac{1}{2}\sum_{k=0}^{P} M_k + \lfloor \frac{P}{2} \rfloor + 2P - 1$	$2P(M+2)+M-2$
Adds/sample	$\sum_{k=0}^{P} M_k + 2P - 1$	$2M(P+1)+P-1$
Delay	$\sum_{k=0}^{P} M_k$	$3M$

Table 5.2. *The number of operations needed per sample with an FIR FD filter and an IIR FD filter and Recursive Gauss-Newton.*

To investigate the computational requirements for the different estimators two filters were designed, using the previously described methods, with approximate equal phase delay error, one FIR and one IIR all-pass filter. In Table 5.3 the resulting filter orders for different estimator errors are seen. Even though that required filter order is lower for the IIR based estimator the required number of operations per sample is still higher compared to the FIR based estimator. We will later see that the estimator using the IIR FD filters has other advantages though, which might make the estimator useful in certain applications.

	Max est. error	M	P	Iter	Mult	Add	Delay
FIR NR	$2.48 \cdot 10^{-4}$	[10 6 8 4 4]	5	5	27	45	10
FIR RGN	$1.10 \cdot 10^{-4}$	[10 6 8 4 4]	5	10	25	39	10
All-Pass NR	$1.84 \cdot 10^{-4}$	6	5	5	131	107	24
All-Pass RGN	$2.00 \cdot 10^{-4}$	6	5	10	84	74	18

Table 5.3. *The computational complexity for the different estimators. Operations per sample.*

5.6 Simulations

To verify the results a number of simulations have been performed. We begin by looking at the estimator offset caused by phase delay and magnitude errors in the FD filters, followed by the variance of the estimate.

5.6.1 Estimator Offset

We will verify that the main source of estimator offset is caused by the phase delay errors in the FD filter used. First we look at the offset when we have a sinusoidal input and then a wideband input.

Sinusoidal Input

To verify the estimator offset analysis, a number of simulations were performed. We have assumed that the frequency ω is unknown, but static.

To verify that the expression for the expected error is correct we simulated the estimator using Newton-Raphson for a number of frequencies ω_0 and delays d_0. In Fig. 5.18 and 5.19 the estimated and simulated errors are seen for the two example FIR FD filters designed in Section 5.3.2. The batch length was $N = 2000$. The upper left plots show the phase delay errors of the FD FIR filters. The upper right plot shows the predicted estimator error, which is the sum of the phase delay error $\tilde{d}(\omega, d)$ and the offset error $d_{\text{err}}(\omega, d)$ caused by magnitude errors. The lower left and right plots show the difference between the predicted estimator offset and the actual estimator offset with and without $d_{\text{err}}(\omega, d)$.

An alternative to Newton-Raphson is to use Recursive-Gauss-Newton. The disadvantage, though, is that RGN usually needs more iterations to

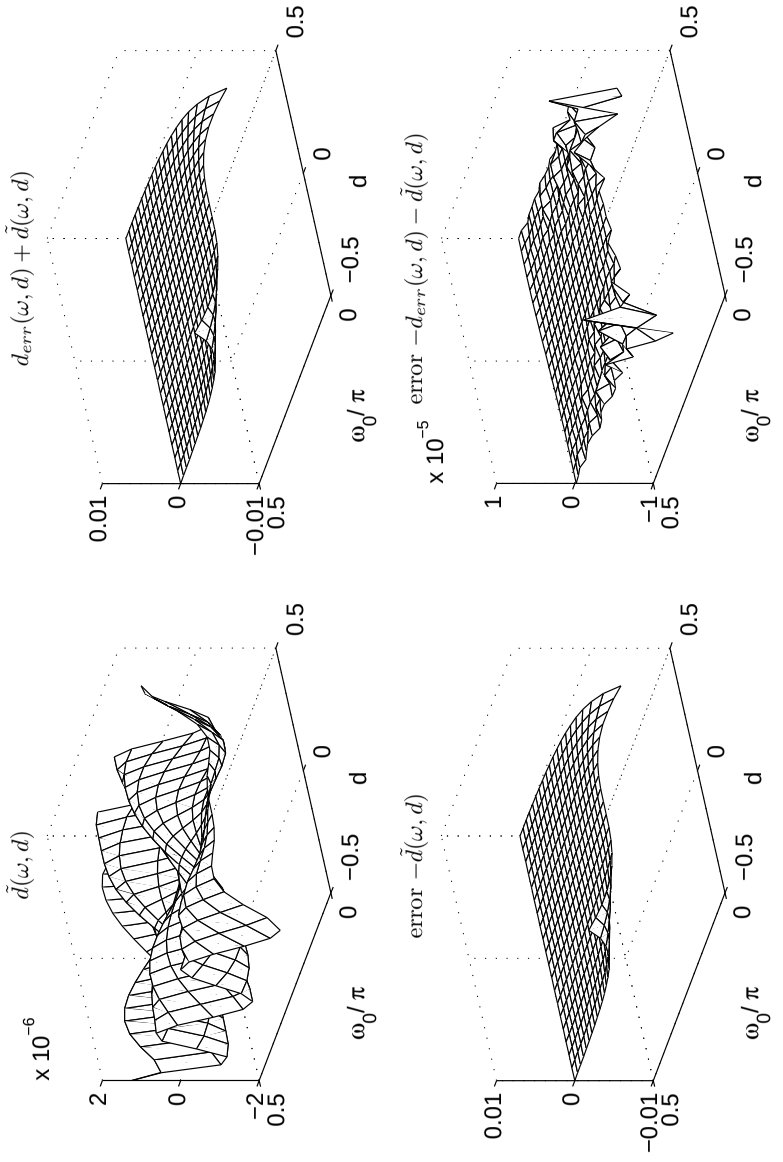


Figure 5.18. *The error estimate for the filter optimized with respect to* $\tilde{d}(\omega, d)$ *and* $||H(\omega, d)| - 1| < C$.

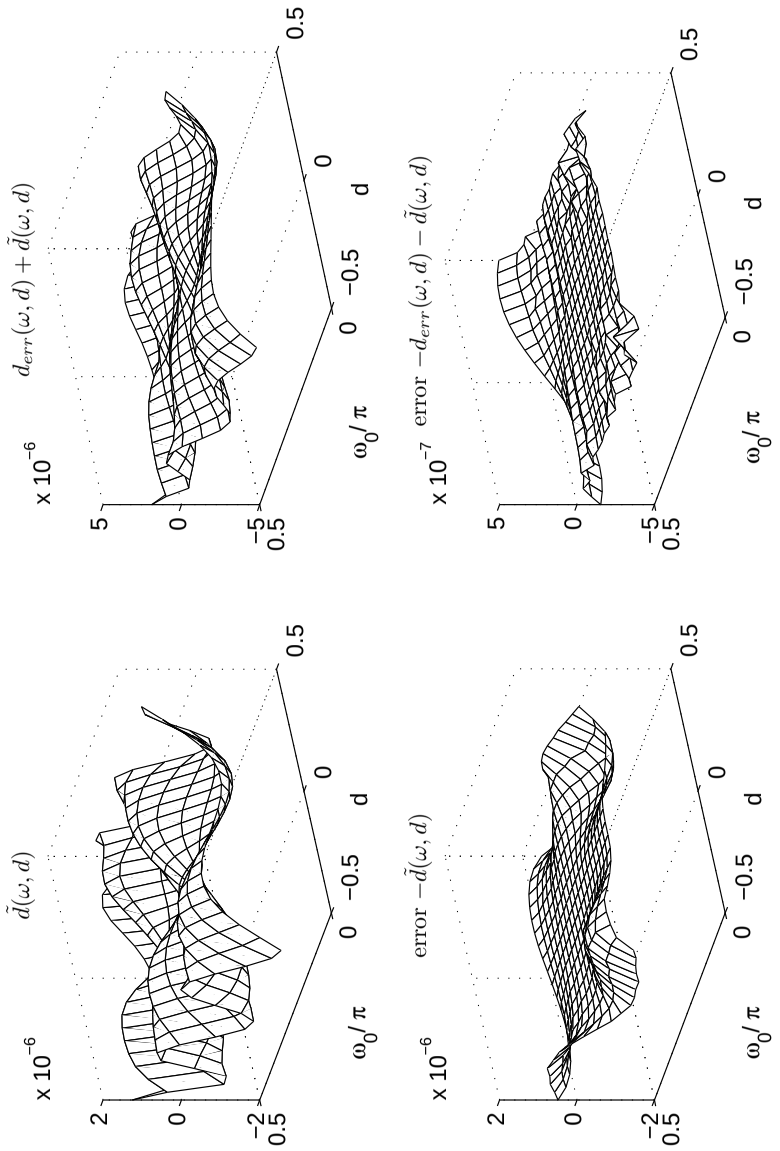


Figure 5.19. *The error estimate for the filter optimized with respect to* $\tilde{d}(\omega, d) + d_{err}(\omega, d)$.

converge. In Fig. 5.20 an example of this can be seen, where RGN needs 8 iterations, while NR only needs 3 or 4. However, depending on the application, this difference might not be important. The use of RGN in TDE was first seen in [24].

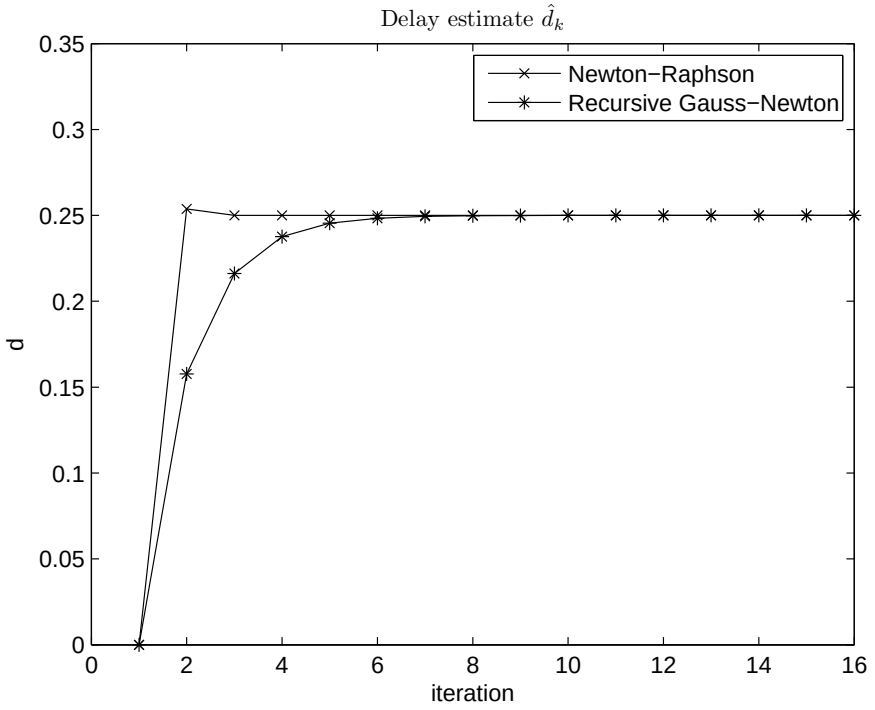


Figure 5.20. *Illustrating the convergence time for NR and RGN.* $d_0 = 0.25$, $\omega_0 = 0.25\pi$, *no noise.*

To further compare NR and RGN we ran the IIR all-pass ASDF estimator using RGN for 5 and 10 iterations and compared the estimate with the expected estimate $\tilde{d}(\omega, d)$. Furthermore we ran the IIR all-pass estimator using NR for 5 iterations. The result can be seen in Fig. 5.21. Five iterations are clearly too few when RGN is used, but for 10 iterations the estimators are comparable and the maximum difference between the estimators is $2.7 \cdot 10^{-4}$ samples.

Since the derivative of the magnitude is equal to zero for an IIR all-pass filter, the predicted estimator error is equal to the phase delay error $\tilde{d}(\omega, d)$ of the FD filter. In Fig. 5.22 it can be seen that the predicted and simulated

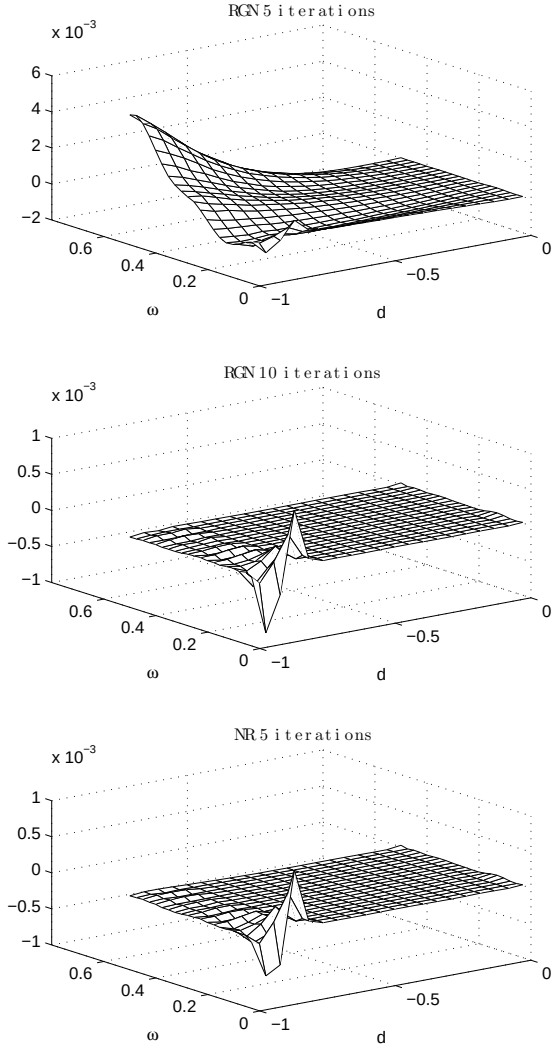


Figure 5.21. *The difference between the estimate and the expected estimate $\tilde{d}(\omega, d)$ for the estimator using RGN and NR, with different number of iterations.*

error are virtually identical. The maximum value of the the difference between the two is equal to $9.5 \cdot 10^{-10}$. The difference is most likely caused by the truncation of the batch length.

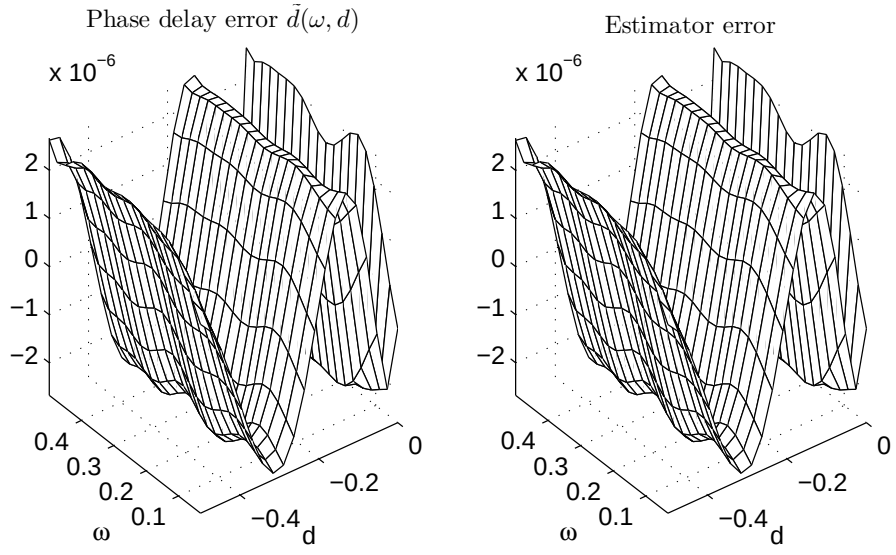


Figure 5.22. *The phase delay error and estimator error for an IIR FD filter.*

5.6.2 Estimator Variance

There are several sources of variance in the estimate. First we have additive noise, which we will assume is Gaussian and white. Another source of estimator variance is the limited batch length. In (5.82) and (5.84) approximations of the variance caused by the batch length truncation for the ASDF and DC estimator were derived and we will verify the expressions in simulations.

The differentiation naturally amplifies high frequency signals and attenuates low frequencies. In Fig. 5.23 it can be seen how the noise is affected by the first and second order derivation. As it will be seen later this is not catastrophic, but it can be good to keep this in mind.

We will first study the case when we have a sinusoidal input, followed by an examination of what occurs when we have a wideband signal.

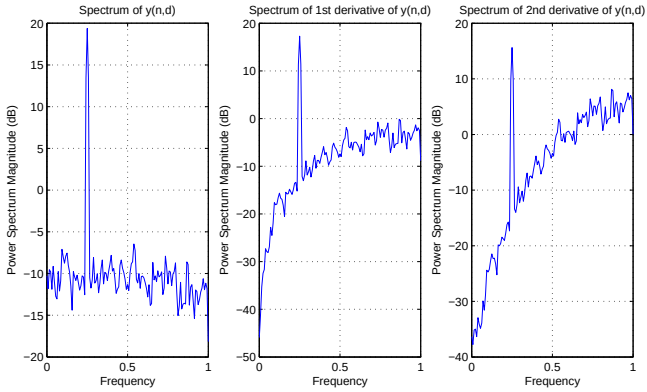


Figure 5.23. *The spectrum of* $y(n,d)$*,* $y'(n,d)$*, and* $y''(n,d)$ *for the Farrow-based FD.* $\omega_0 = 0.25\pi$*,* $d_0 = 0.25$*,* $SNR = 10dB$

Sinusoidal Input

Sinusoidal inputs were generated with constant frequency and random phase. In Fig. 5.24, the estimator variance for different SNR and relative magnitude A is plotted. The IIR all-pass FD filter based estimator was used. The predicted variance caused by batch length truncation, calculated using (5.82) and (5.84) and shown as horizontal lines in Fig. 5.24, corresponds well to the variance floors experienced in the simulation. If the relative magnitude A is close to 1 the variance for the ASDF estimator is much lower than the DC estimator.

Wideband Input

To verify that the estimator works for both a wide-band signal a signal was generated by adding a number of sinusoids with random frequency and random phase. A comparison of the resulting variance can be seen in Fig. 5.25 for both the ASDF and DC estimators. As can be seen, the ASDF method still has a much lower variance for wideband signals.

In Fig. 5.25 the estimator variance for single frequency signals are shown too as a comparison. Compared to Fig. 5.24 where we have a relative magnitude A that differs from 1, the variance caused by the batch length truncation for the ASDF estimator is nonexistent, as predicted in (5.82).

The reason for the variance floor in the ASDF based estimator when a

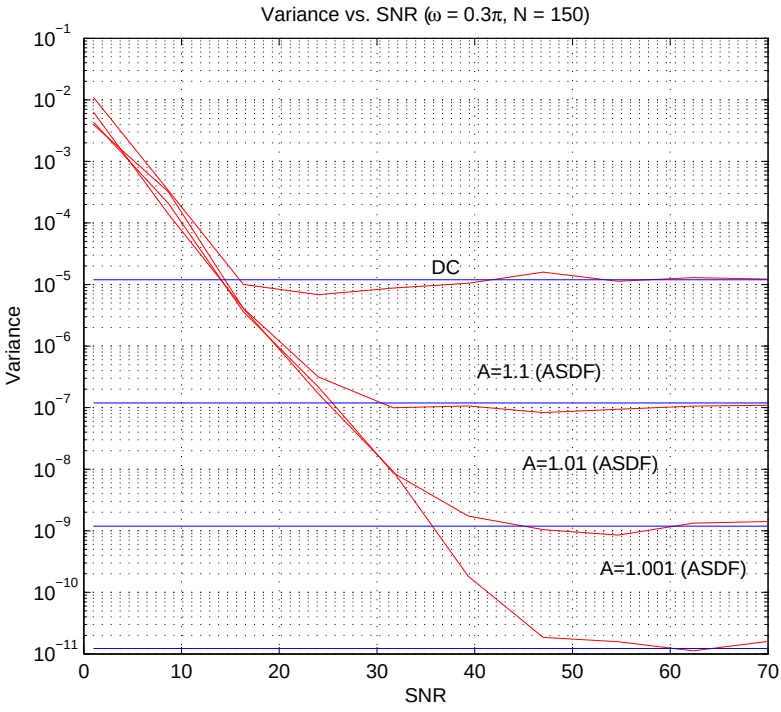


Figure 5.24. *The IIR all-pass FD estimator variance and the predicted variance for a sinusoidal input with different relative magnitude A.*

wideband signal is applied is the phase delay error ripple seen in Fig. 5.17. The exact relationship needs more investigation.

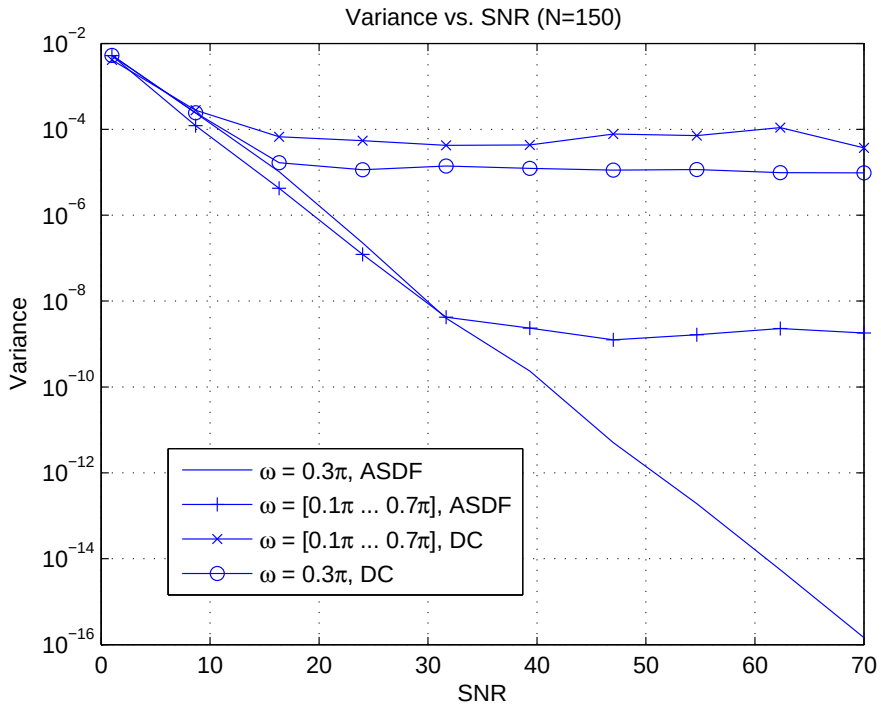


Figure 5.25. *The variance for a wideband input.*

5.7 Conclusions

We have presented novel methods to estimate the delay error between two sets of samples using an adjustable FD filter. One advantage is that the signals do not have to be known, only band limited. The method could be used for example in time-interleaved analog-to-digital converters and could easily be extended to more sets of samples.

The effects of magnitude and delay error have been studied theoretically and verified in simulations. An expression of the expected estimator offset was derived and it was seen that the main contributor is the phase delay error of the FD filter, but an additional offset is caused by the first

derivative of the magnitude error. The derived expression can be used to optimize the FD filters to be used in the estimator. Since the magnitude error is zero in an IIR all-pass FD filter it is only necessary to optimize such a filter with respect to the phase delay error.

Furthermore, the direct correlator (DC) and average squared difference (ASDF) cost function were compared from an estimator variance point of view. It was seen that the variance caused by sample batch length truncation is much smaller for the ASDF estimator than for the DC estimator. If a constant magnitude error is present, both estimators will have a variance floor, but the estimator based on an IIR all-pass FD filter has an advantage because it is easier to control the magnitude error of such a filter.

It was also seen that if the number of iterations needed to find an estimate is not critical we can use recursive Gauss-Newton instead of Newton-Raphson to find the maximum/minimum for the cost functions.

5.7.1 Future Work

There are a couple of issues that we would like to investigate further. First, we would like to extend the estimator to magnitude and DC offset estimation. In this thesis we have only briefly covered magnitude offset and it needs more investigation. Furthermore, we would like to explain the estimator variance when we have a wideband signal instead of a single frequency signal.

Initial results show that when gain estimation is introduced in the ASDF estimator the variance caused by batch length truncation is reduced significantly, especially in the estimator using an all-pass FD filter since all the estimated gain is valid for all frequencies.

Chapter 6

Summary

In this thesis we have studied basically two problems, namely frequency and delay offsets, and proposed a number of possible estimation methods.

The CFO estimation method presented in Chapter 3 uses a null subcarrier to estimate the CFO iteratively. The estimator has some similarities with the method presented in [13], although it is carried out differently. One of the benefits with the proposed CFO estimator is that no extra redundancy has to be added, besides the nonmodulated subcarrier. Furthermore, it does not rely on the cyclic prefix, which is often destroyed in a multipath channel.

We have proposed an iterative time delay estimator using adjustable fractional delay filters. The estimator works by minimizing a cost function by finding the closest zero of the first derivative of the respective cost function.

We have shown that the average squared difference function performs better than the direct correlator when it comes to estimator variance caused by observation window truncation.

Furthermore, we have shown that TDE based on IIR all-pass FD filters have some attractive properties. First, the expected error is equal to the phase delay error, making it easy to design. A disadvantage, though, is that the number of operations per sample is higher compared to an estimator based on an FIR FD filter, even when the lower filter order is considered.

Finally we saw that if the number of iterations needed to acquire an estimate is not critical, the recursive Gauss-Newton technique can be used to somewhat reduce the number of operations needed.

6.1 Future Work

A main line of future work will be in the area of time delay estimation. There are several tracks that remain to be investigated. First, we would like to generalize the estimator to more parameters, for example DC offset and gain. We would also like to implement the estimator in hardware and verify the predicted performance in practice, preferably running in an on-line fashion.

References

[1] G. Blom. *Probability theory and statistical theory with applications*. Studentlitteratur, 1989.

[2] G. C. Carter. Coherence and time delay estimation. *Proc. of the IEEE*, 75(2):236–255, Feb. 1987.

[3] Y. T. Chan, R. V. Richard, and J. B. Plant. The least squares estimation of time delay and its use in signal detection. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-26:217–222, June 1978.

[4] B. Chen and H. Wang. Blind OFDM carrier frequency offset estimation via oversampling. In *Conf. Rec. of the Thirty-Fifth Asilomar Conf. on Signals, Systems and Computers*, volume 2, pages 1465–1469, Nov. 2001.

[5] Y.-S. Choi, Peter J. Voltz, and F. A. Cassara. ML estimation of carrier frequency offset for multicarrier signals in rayleigh fading channels. *IEEE Trans. Vehicular Technology*, 50(2):644–655, Mar. 2001.

[6] S. R. Dooley and A. K. Nandi. On explicit time delay estimation using the Farrow structure. *Signal Processing*, 72:53–57, 1999.

[7] C. W. Farrow. A continously variable digital delay element. In *Proc. IEEE Int. Symp. Circuits & Syst.*, volume 3, pages 2641–2645, June 7–9 1988.

[8] G. Jacovitti and G. Scarano. Discrete time techniques for time delay estimation. *IEEE Trans. on Signal Processing*, 41(2), Feb. 1993.

[9] H. Johansson and P. Löwenborg. On the design of adjustable fractional delay FIR filters. *IEEE Trans. on Circuits and Syst. II: Analog and Digital Signal Processing*, 50(4):164–169, Apr. 2003.

[10] H. Johansson and P. Löwenborg. Reconstruction of periodically nonuniformly sampled bandlimited signals using time-varying FIR filters. In *Proc. Fourth Int. Workshop Spectral Methods Multirate Signal Processing*, Vienna, Austria, Sept. 11–12 2004.

[11] C. H. Knapp and G. C. Carter. The generalized correlation method for estimation of time delay. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-24, Aug. 1976.

[12] T. I. Laakso, V. Välimäki, M. Karjalainen, and U. T. Laine. Splitting the unit delay: Tools for fractional delay filter design. *IEEE Signal Processing Magazine*, Jan. 1996.

[13] H. Liu and U. Tureli. A high-efficiency carrier estimator for OFDM communications. *IEEE Commun. Lett.*, 2(4), Apr. 1998.

[14] L. Ljung. *System Identification - Theory for the user*. Prentice Hall, 1999.

[15] M. Makundi, T. I. Laakso, and V. Välimäki. Efficient tunable IIR and allpass filter structures. *IEEE Electronic Letters*, 37(6), Mar. 2001.

[16] D. L. Maskell and G. S. Woods. The discrete-time quadrature subsample estimation of delay. *IEEE Trans. Instrum. Meas.*, 51(1):133–137, Feb. 2002.

[17] J. Medbo and P. Schramm. *Channel Models for HIPERLAN2*, 1998. ETSI/BRAN document no. 3ERI085B.

[18] P. Moose. A technique for orthogonal frequency division multiplexing frequency offset correction. *IEEE Trans. Commun.*, 42, Oct. 1994.

[19] M. Olsson and H. Johansson. Blind OFDM carrier frequency offset estimation by locating null subcarriers. In *Proc. of 9th Int. OFDM-Workshop*, Sept. 2004.

[20] M. Olsson and H. Johansson. OFDM carrier frequency offset estimation using null subcarriers. In *Proc. of 10th Int. OFDM-Workshop*, 2005.

[21] M. Olsson and H. Johansson. An overview of OFDM synchronization techniques. In *Proc. National Conf. Radio Science, RVK '05*, 2005.

[22] T. Pollet. BER sensitivity of OFDM systems to carrier frequency offset and Wiener phase noise. *IEEE Trans. on Comm.*, 34(2):191–193, 1995.

[23] K. Sathananthan and C. Tellambura. Probability of error calculation of OFDM systems with frequency offset. *IEEE Transactions on communications*, 49(11), Nov. 2001.

[24] J. O. Smith and B. Friedlander. Adaptive interpolated time-delay estimation. *IEEE Trans. Aerospace, Electronic Syst.*, AES-21(2):180–199, Mar. 1985.

[25] J-P Thiran. Recursive digital filters with maximally flat group delay. *IEEE Trans. on Circuit Theory*, CT-18(6), Nov. 1971.

[26] M. Toeltsch and A. F. Molisch. Efficient OFDM transmission without cyclic prefix over frequency-selective channels. In *PIMRC2000*, Sept. 2000.

[27] U. Tureli, H. Liu, and M. D. Zoltowski. OFDM blind carrier offset estimation: ESPRIT. *IEEE Trans. Commun*, 48(9), Sept. 2000.

[28] P. P. Vaidyanathan. *Multirate Systems and Filter Banks*. Prentice Hall, 1993.

[29] J. van de Beek, M. Sandell, and P. O. Börjesson. ML estimation of time and frequency offset in OFDM systems. *IEEE Trans. Signal Processing*, 45(7):1800–1805, July 1997.

[30] J. Vesma and T. Saramäki. Optimization and efficient implementation of FIR filters with adjustable fractional delay. In *Proc. IEEE Int. Symp. Circuits Syst.*, volume IV, pages 2256–2259, June 9–12 1997.

[31] J. Yli-Kaakinen and T. Saramäki. An algorithm for the optimization of adjustable fractional-delay all-pass filters. In *Proc. of the 2004 Int. Symp. on Circ. and Syst.*, volume 3, pages 153–156, May 2004.

[32] C. J. You and H. Horng. Optimum frame and frequency synchronization for OFDM systems. In *Int. Conf. on Consumer Electronics*, pages 226–227. IEEE, 2001.