University of Oslo
Department of Informatics

Signal processing using the Teager Energy Operator and other nonlinear operators

Cand. Scient Thesis

Eivind Kvedalen

May 2003

Preface

This report is written for the Candidatus Scientiarum (Cand. Scient) degree at the University of Oslo, Norway, Department of Informatics.

In late 1999, I was searching for a suitable project for my Cand. Scient thesis, and during meetings with Professor Sverre Holm at the Department of Informatics, we found two research areas that we thought would be possible candidates for a project.

One of the topics was about optimum combination of spectrograms, and its application to speech analysis. The other topic was inspired by an article presented at NORSIG'98 about the Teager Energy Operator. In January-March 2000, I wrote an essay for the compulsory introductory course for Cand. Scient students, IN-IFHS, which "scratched the surface" of both these topics, and concluded that I would like to find out more about the Teager Energy Operator. The result of this process is this thesis.

I would like to thank my teaching supervisor Professor Sverre Holm, for his help and useful comments during the work with this thesis.

Contents

List of Figures

Problem definition

Many signal processing problems can be described by linear models, and in many cases be realized as analog or digital linear, time-invariant filters; finite impulse response (FIR) or infinite impulse response filters (IIR).

The filter acts on the input, creating the output as a function of the input. We can thus regard the filtering as an operator on the signal, transforming a given input signal to an output signal.

Recently, a new non-linear operator called the Teager Energy Operator, (or the Teager-Kaiser Energy Operator, dependent on the sources), has been introduced and investigated.

This operator is interesting because it has a small time window, making it ideal for local (time) analysis of signals, and because the basic definition of the operator is very simple. The operator thus proves to be very powerful in certain situations. Furthermore, the operator is very easy to implement efficiently, as we shall see later.

This thesis aims to explore the nature of the Teager Energy Operator and related operators, when the operator is used to analyse and demodulate AM-FM signals, and to highlight the research that has been done already. The analysis of AM-FM signals using the Teager Energy Operator is probably the field where most research regarding the operator has been done so far.

The motivation for this thesis is thus not to solve some specific problem using the Teager Energy Operator, but to look at both the theoretical and practical aspects of it, with emphasis on its performance in noisy environments.

Chapter 1

Teager's Energy Operator and related work in the literature

In this chapter, the research already done on the Teager Energy Operator will be reviewed. All references here are made with the complete title and authors, as these usually tell what the papers are all about, and we shall also see that many of the papers are written by distinct groups of people. A short summary of the important papers is given.

It must be mentioned that the text below does unfortunately not cover all work done on the Teager Energy Operator; it is certainly possible to collect the most important work, as the knowledge about the Teager Energy Operator is not widespread as of today. I have managed to get hold of the most important papers and applications, though.

1.1 A short definition of the Teager Energy Operator

Before we start on the history of the Teager Energy Operator, it makes sense to briefly describe the basic operator here. We will describe the operator in detail later, but the text below requires some knowledge of its definition.

The Teager Energy Operator is defined as

$$\Psi(x(t)) = \dot{x}^2(t) - x(t)\ddot{x}(t) \tag{1.1}$$

in the continuous case (where $\dot{x}$ means the first derivative of x, and $\ddot{x}$ means the second derivative), and as

$$\Psi[x[n]] = x^2[n] - x[n-1]x[n+1] \tag{1.2}$$

in the discrete case.

1.2 Teager's experiments

In 1983, "A Phenomenological Model for Vowel Production in the Vocal Tract" by Herbert M. Teager and Shushan M. Teager [1], was published. This article argued that the current speech model (at that time), basically modelled using linear filter theory, was inaccurate, and that the production of speech involved nonlinear processes not described

by the current model. The authors pointed out several aspects that are not explained good enough by the linear theory.

"Evidence for Nonlinear Sound Production Mechanisms in the Vocal Tract" [2] from 1990, by Teager & Teager described the nonlinearities further, and in this article a plot was shown of the "energy creating the sound", but the algorithm to calculate this "energy" was not given.

1.3 Kaiser's initial work

The method to compute the energy of the generator of a sound caught J. F. Kaiser's interest, and in 1990, "On a Simple Algorithm to Calculate the 'energy' of a Signal" [3] was published. The algorithm presented in this paper was derived by Kaiser alone; Teager would never reveal to any other exactly how he had calculated the energy in the first place [4, 5]. It was this algorithm that Kaiser named "Teager's Energy Algorithm", and this is what we shall study in this thesis. This article defined the discrete version of Teager's Energy Operator. The work presented in the paper, was later extended to cover continuous time signals in [6].

1.4 Applications are developed, and the theory is extended

During the last twelve years of research, many applications have been developed using Teager's Energy Operator. One of the first applications of the operator, was presented in "Speech Nonlinearities, Modulations, and Energy Operators" [7] by Petros Maragos, Thomas F. Quatieri and Kaiser. In this paper, we see the first traces of the AM-FM demodulation techniques that have been thoroughly treated in other papers by Maragos et.al.

Many of the published papers on Teager's Energy Operator have been presented at The International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Kaiser's first article was presented at ICASSP-90, and two more followed in 1992, and at least seven more in 1993. One of the papers from ICASSP-91 was "A New Class of Nonlinear Filters for Image Enhancement" [8] by S. K. Mitra, H. Li, I.S. Lin, T. H. Yu. This paper generalized the operator to two dimensions, to make it suitable as an image filter. These operators are also treated in several other papers by Stefan Thurnhofer and Mitra.

1.5 ICASSP-93

In 1993, Maragos, Kaiser and Quatieri published "On Amplitude and Frequency Demodulation Using Energy Operators" [9]. This was the first paper which treated the Teager Energy Operator extensively with regard to its demodulation properties. This is one of the fundamental articles on the operator, where AM-FM demodulation is extensively treated, error bounds are calculated, and examples are shown for a variety of signal classes. Much of the theoretical work on demodulation in this thesis is based on this paper, and similar publications by Maragos et.al.

"Theorems for Discrete Filtered Modulated Signals" by Alan C. Bovik, Joseph P. Havlicek and Mita D. Desai [10] derives limits on AM-FM demodulation mechanisms related to the Teager Energy Operator, and can be viewed as a parallel to [9]. In the paper, a filtered version of the input signal was used to derive the limits.

A new application for the operator was presented in "Detection of transient signals using the Energy Operator" by R.B. Dunn, Quatieri and Kaiser [11]. In this paper, the background noise is assumed to be slowly varying AM-FM signals, in addition to white noise, and a detector using the Teager Energy Operator is developed to detect transients in the noise.

Three more applications were presented at ICASSP-93, and they were all on the use of the Teager Energy Operator on speech. The algorithms derived by Maragos, Kaiser, and Quatieri in a tech report at Harvard Robotics Laboratory (a revision of this was published later in 1993, which we will describe below) was used in "Stop Classification using DESA-1 High Resolution Formant Tracking" by J. T. Foote. D. J. Mashao, and H. F. Silverman [12] to do stop classification (estimation of endpoints) of speech.

"Finding Speech Formants and Modulations via Energy Separation: With Applications to a Vocoder" by Helen M. Hanson, Maragos, and Alexandros Potamianos [13] presents an iterative algorithm to estimate the formants of speech signals. Its application to an AM-FM vocoder is also discussed.

The last application at ICASSP-93 using the Teager Energy Operator was endpoint detection of utterances in speech signals, which is described in "Endpoint Detection of Isolated Utterances Based on a Modified Teager Energy Measurement" by G. S Ying, C. D Mitchell, and L. H. Jamieson [14]. In this paper, the ideas from Teager and Kaiser are used, and a new algorithm for the estimate of the energy in a speech frame is derived. This estimate is different from the usual RMS estimate, in that it also accounts for the frequencies of the input signal.

Kaiser derived the fundamental and simple properties of Teager's Energy Operator in "Some Useful Properties of Teager's Energy Operators", which was also published on ICASSP-93.

1.6 More theoretical work is done

In "Energy Separation in Signal Modulations with Application to Speech Analysis" by Maragos, Quatieri, and Kaiser [15], the energy separation algorithms DESA-1a, DESA-1 and DESA-2 are derived. These were derived as early as in 1991, but were then only published in a tech report (mentioned above). In this paper, the formulas are derived for the continuous case first, and then adapted to the discrete case, which results in multiple versions, as there are several ways to create discrete versions of continuous differential operator. The paper derives (error-)estimates for several classes of signals, and a few examples of its estimation capabilities are shown.

As the Teager Energy Operator uses only a few samples of the input signal to estimate the energy, it should come as no surprise that the operator is highly sensitive to noise. To remedy this situation, "AM-FM Energy Detection and Separation in Noise Using Multiband Energy Operators" by Bovik, Maragos and Quatieri [16], presents an algorithm using filterbanks to suppress noise in the signal. This paper also contains some of the first work on the statistics of the Teager Energy Operator.

Although Teager's Energy Operator is an *energy* measure, it is possible to construct signals which yield negative values. A few special cases are considered in [16], but "Conditions for Positivity of an Energy Operator" by Bovik and Maragos gives us general conditions for positivity.

By 1994, the demodulation properties of the Teager Energy Operator were well known, but this is by far not the only way to estimate instantaneous frequency and amplitude. The best known traditional method is probably the Hilbert transform (the analytic signal), and in "A comparison of the energy operator and the Hilbert transform approach to signal and speech demodulation" by Potamianos and Maragos the performance of the two methods are compared. Potamianos and Maragos published "Speech Formant Frequency and Bandwidth Tracking Using Multiband Energy Demodulation" in 1995. This article describes a new frame-based system for formant tracking and bandwidth estimation of speech signals.

Wei Lin, Chris Hamilton, and Prabhakar Chitrapu extended the Teager Energy Operator further in "A Generalization to the Teager-Kaiser Energy Function & Application to Resolving Two Closely-Spaced Tones". In this paper, a simple extension is done to the operator, which makes it suitable for resolving of two closely-spaced tones. Examples including noise are shown. The same generalization is later used in [17].

In 1996, L. B. Fertig and J. H. McClellan wrote "Instantaneous Frequency Estimation Using Linear Prediction with comparisons to the DESAs". This paper simplifies the modified covariance and Prony's method to simple formulas, which are compared to the DESAs derived earlier.

An alternative approach to the formant tracking problem is used in "Nonlinear Modeling and Processing of Speech Based on Sums of AM-FM Formant Models" by Shan Lu and Peter C. Doerschuk [18]. In this paper, a statistical model is used, where the problem is modelled as an extended Kalman filter.

As written above, there are several methods available for frequency and amplitude estimation, and "On the Analytic Signal, the Teager-Kaiser Energy Algorithm, and Other Methods for Defining Amplitude and Frequency" by David Vakman [19] gives some reasonable physical conditions that estimators should adhere to. Mandelstram's and Shekel's methods for amplitude and frequency estimation are described.

A new application of the Teager Energy Operator is presented in "Watershed segmentation of an image enhanced by Teager Energy Driven Diffusion" by D. De Vlesschauwer, F. Alaya Cheikh, R. Hamila, and M. Gabbouj. Here, the two-dimensional version of the Teager Energy Operator is used to preserve edges in images when they are scaled, and the results presented suggests that this method is better than the traditional linear method.

As the definition of the Teager Energy Operator is so simple, a variety of discrete generalizations have been made. In "A Generalization of the Teager Algorithm", Michael Moore, Sanjit Mitra, and Reinhard Bernstein [20], the discrete version is extended, and a link to Volterra systems is established.

In "Time-Frequency Signal Analysis Using Teager Energy" by Ridha Hamila, Markku Renfors, Moncef Gabbouj, and Jaakko Astola [21] and "Teager Energy and the Ambiguity Function" [22], the connection between the Teager Energy Operator and the Ambiguity function is developed, and a new formant tracking algorithm is presented.

In "The Teager Correlation Function" by T. K. Bysted, Hamila, Gabbouj and

Renfors, the complete relationship between the Teager Energy Operator and the Ambiguity function is established. A new correlation function is introduced and compared to the conventional correlation function. Two applications are also presented; The Teager Energy as an error norm, and time and frequency difference of arrival (TDOA and FDOA) estimation.

1.7 Summary

As we have seen above, the Teager Energy Operator has had many interesting applications in various fields of signal processing, despite its simple definition. The most successful application so far has been demodulation of AM-FM signals, where we have seen particularly many applications in speech processing.

Chapter 2

Definitions

This chapter introduces the basic concepts that will be used throughout the thesis. All properties related to the Teager Operator are gathered in chapter 3. This chapter also contains some ideas about amplitude, frequency and phase for signals in general, and the estimation of these values. We end this chapter by summarizing the results in this thesis.

2.1 Notation and basic definitions

Operations in continuous time and discrete time are treated differently, and we thus introduce the following notation for this: Continuous time functions are written as $x(t)$, while discrete ones (sequences) are written as $x[n]$, or x_n depending on the complexity of the expression. This should also be clear from the context in most cases.

$\dot{x}$ is used to designate dx/dt and $\approx dx/dn$. Unless otherwise stated, $\approx dx/dn$, means the approximations $x[n] - x[n-1]$ and $x[n+1] - x[n]$, as one cannot argue that one of them is more correct than the other. These expressions are called the backward and forward differences, respectively. Similarly, $\ddot{x}$ is used for the second derivative, which in the discrete case is $x[n-1] - 2x[n] + x[n+1]$, unless stated otherwise.

2.1.1 RMS – Root Mean Square

The RMS value for a sequence x is defined as

$$RMS\{x\} = \sqrt{\frac{1}{N}\sum_{n=1}^{N} x[n]^2} \tag{2.1}$$

and the RMSE – Root Mean Square Error is defined as

$$RMSE\{x, y\} = \sqrt{\frac{1}{N}\sum_{n=1}^{N} (x[n] - y[n])^2} \tag{2.2}$$

2.1.2 SNR – Signal-to-noise ratio

The signal to noise ratio is defined as

$$SNR\{x, n\} = 20 \log_{10} \frac{\text{Var}\{x\}}{\text{Var}\{n\}} \tag{2.3}$$

where, x is the noise-free signal, n is the noise, and $\text{Var}\{\cdot\}$ is the usual variance operator.

2.1.3 The median filter

The median filter is a nonlinear filter

$$y = F\{x, L\}$$

taking the sequence x and a length L as parameters. L is the number of consecutive samples used to compute an output value. For each n, the output given as $y[n]$ is the median value of the sequence $x[n-L/2]$ to $x[n+L/2-1]$ for L even and $x[n-(L-1)/2]$ to $x[n+(L-1)/2]$ for L odd.

This filter is often used in post-processing of data, where we want to remove a few spurious values, spikes, from the output data.

2.2 Amplitude, phase and frequency

In "On the Analytic Signal, the Teager-Kaiser Energy Algorithm and Other Methods for defining Amplitude and Frequency" [19] by David Vakman, the general concepts of instantaneous frequency and amplitude are discussed. This article explains the physical constraints between amplitude, phase and frequency (APF) of a signal, and shows how the Analytic Signal procedure (AS) is used to estimate these quantities. We will re-visit this procedure when the various estimators are introduced.

2.2.1 Physical conditions for the APF

We are here only concerned with real-valued signal, which we in general can write as

$$\begin{aligned} s_r(t) &= a(t)\cos(\phi(t)) \\ &= a(t)\cos(\omega_0 t + \Phi(t)) \end{aligned} \tag{2.4}$$

We here want to estimate $a(t)$, $\phi(t)$ and $\omega(t) = d\phi/dt$. However, 2.4 is *one* equation, which has two unknowns (a and ϕ) that we want to estimate. This means that we must have some other equation linking these two quantities together for unambiguous determination.

If we write the signal $s(t)$ as the sum of a real and imaginary signal

$$\begin{aligned} s(t) &= s_r(t) + j s_j(t) \\ &= a(t) e^{j\phi(t)} \end{aligned} \tag{2.5}$$

we can express the APFs as (t omitted for clarity)

$$a = \sqrt{s_r^2 + s_j^2} = |s| \tag{2.6}$$

$$\phi = \arctan(s_j/s_r) = Arg(s) \tag{2.7}$$

$$\omega = \frac{\dot{s}_j s_r - s_j \dot{s}_r}{s_j^2 + s_r^2} = Im(\dot{s}/s) \tag{2.8}$$

If $s_j(t)$ happens to be the Hilbert transform, for some signal $x(t)$ defined as

$$H\{x(t)\} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(s)}{t-s} ds \tag{2.9}$$

then $s(t)$ is what we call the Analytic Signal (AS), which was introduced by Gabor in 1946 [23]. Figure 2.1 shows a hypothetical device to measure the APF, using some arbitrary operator to generate the imaginary part of the signal.

Now, there is an infinite number of possible operators H that can compute the imaginary signal, but it is shown in [19] that only the Hilbert transform satisfies the reasonable physical conditions

1. Amplitude continuity and differentiability.

   This means that the operator H must be continuous and differentiable.

2. Phase independence of Scaling and Homogeneity.

   This means that if a signal $s(t)$ is replaced by a signal $c \times s(t)$, where c is a constant, then the phase and frequency of s remains the same.

3. Harmonic correspondence.

   This means that the amplitude and frequency of a simple sinusoidal signal should retains its values, that is,

   $$H\{a\cos(\omega t + \phi)\} = a\sin(\omega t + \phi) \tag{2.10}$$

   for any a, ω and ϕ.

Any other definition of the operator H will violate at least one of the conditions above. This puts the AS procedure in a special position among the APF estimators. We will later see that many APF algorithms have been suggested, among them the Energy Separating Algorithms (ESAs) based on the Teager Energy Operator, which we will describe in detail in this thesis.

One final thing should be noted about the AS procedure. For decades, other methods of estimating the instantaneous frequency have been suggested, and one must wonder why we want other methods, if the AS method with the Hilbert transform operator is the *only* one that gives us the correct answer. The answer to this is that the other methods that have been developed, are *local*, meaning that they need only a few samples (short time windows) of the input signal to estimate the instantaneous frequency. On the contrary, the AS procedure is a *global* method, which actually requires the complete signal for the Hilbert transform to get the best performance (although a relatively short time window may be enough for this method, too).

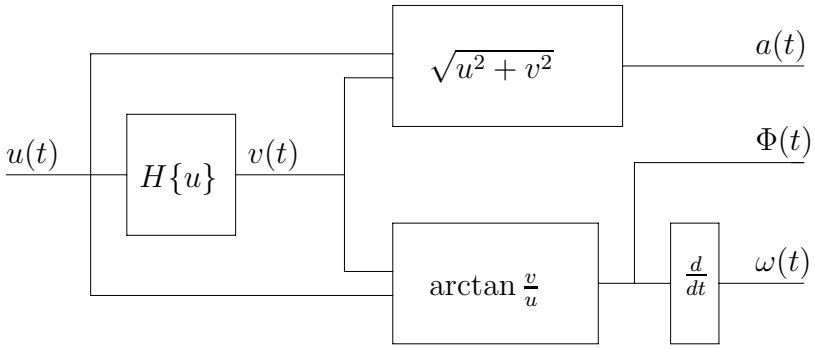

Figure 2.1: Hypothetical APF measuring device using the operator H to generate the imaginary part of the signal.

2.3 Common signals of interest

In this section, we define signals that we often use in the thesis. In chapter 5, we will see how some the parameters of the signals below can be estimated, but we define them here because they are needed in the succeeding chapters as well.

2.3.1 AM signals

An amplitude modulated (AM) signal is the combination of two signals, where one signal is the "carrier", which is a single frequency sinusoidal signal, and the other is the information we want to transmit; the baseband signal. We can model the amplitude modulated signal like this:

$$a(t) = A[1 + \kappa m(t)] \tag{2.11}$$

$$s_{AM}(t) = a(t)\cos(\omega_c t) \tag{2.12}$$

where $A > 0$ is the signal amplitude, ω_c is the carrier frequency (in radians/second), $-1 \leq m(t) \leq 1$ is the baseband signal, and $0 < \kappa \leq 1$ is the *modulation factor*. These AM signals are called AM signals *with carrier*, abbreviated as AM/WC, as we require $a(t)$ to be positive.

2.3.2 FM and PM signals

Just as for AM signals, an FM signal is the combination of two signals, where one is the single frequency sinusoidal signal, the carrier, and the other is the baseband signal. The difference between the AM and FM signal, is that we in FM signals use the baseband signal to change the frequency of the carrier signal.

A frequency modulated signal (FM signal) can be modelled like this:

$$\phi(t) = \omega_c t + \omega_m \int_0^t m(\tau)d\tau + \theta \tag{2.13}$$

$$s_{FM}(t) = A\cos(\phi(t)) \tag{2.14}$$

where $A > 0$ is the signal amplitude, ω_c is the carrier frequency, ω_m is the maximum frequency deviance, $-1 \leq m(t) \leq 1$ is the baseband signal, and θ is the initial phase.

The *instantaneous frequency* is defined as the derivative of $\phi(t)$:

$$\omega_i(t) = \frac{d\phi(t)}{dt} = \omega_c + \omega_m m(t) \tag{2.15}$$

This means that ω_m is the maximum deviance of ω_i from ω_c.

A phase modulated signal (PM signal) is a signal of the form

$$\phi(t) = \omega_c t + \omega_m m(t)dt \tag{2.16}$$

$$s_{PM}(t) = A\cos(\phi(t)) \tag{2.17}$$

where $A > 0$ is the signal amplitude, ω_c is the carrier frequency, ω_m is the maximum deviance from ω_c and $-1 \leq m(t) \leq 1$ is the baseband signal.

Comparing this with the equations for FM signals, we find that there is a close relationship between PM and FM signals: We can regard an FM signal as a PM signal if we set $m_{PM}(t) = \int_0^t m_{FM}(\tau)d\tau$. Using differentiation, we can also create an FM signal from a PM signal.

2.3.3 AM-FM signals

The AM-FM signal is the combination of both the AM and FM signals above, and we can model it like this:

$$a(t) = A[1 + \kappa m_{am}(t)] \tag{2.18}$$

$$\phi(t) = \omega_c t + \omega_m \int_0^t m_{fm}(\tau)d\tau \tag{2.19}$$

$$s_{AM-FM}(t) = a(t)\cos(\phi(t)) \tag{2.20}$$

where the constants are as described above, and $m_{am}(t)$ and $m_{fm}(t)$ are the two baseband signals of the AM and FM parts of the signal, respectively.

2.3.4 Discrete versions

We can get discrete versions of the AM, FM, and AM-FM signals above by sampling them. We can derive new expressions for these if we substitute t by nT and ω by Ω/T, where Ω is the digital frequency (in radians/sample), and T is the sampling period. The integrations are replaced by sums.

2.4 Summary of results

We summarize the results found during the work with this thesis in this section:

- The Laplace transform of the Teager Operator is found using the synthesis of basic (filter) blocks based on the work by Schetzen [24].

- The validity of the Teager Operator approximation is commented on, which suggests that the Operator approximation developed in [8] is only suited for signals with a non-zero mean. Image data is a common example of such signals, as those signals usually have a lower bound of zero (and are thus not zero-mean).
- An example of a formant tracking/detection routine is presented. Error estimates for the Teager Energy are computed for a bandpass filtered signal composed of two AM signals.
- A number of estimators are presented. These have been gathered from various related articles. Error estimates for the various estimators under different noise conditions are presented. This presentation includes most relevant short-time instantaneous frequency estimators found during the work with this thesis.

Chapter 3

The Teager Energy Operator and its properties

We begin this chapter by discussing signal energy in general. We then look at Kaiser's "alternative" definition, and how the basic Teager Energy Operator can be found by considering a second order differential equation, which describes the motion of an object suspended by a spring.

3.1 Signal energy

In electrical systems, the *instantaneous power* of a system can be described as [25]

$$p(t) = |v(t)|^2/R \qquad \text{or} \qquad p(t) = R \times i(t)^2 \tag{3.1}$$

where $R, v(t), i(t)$ is the resistance, voltage and the current of the system. If we normalize this by choosing $R = 1\Omega$, we see that the power is the square of the input signal, regardless of whether we choose to measure the voltage or the current. We can therefore express the instantaneous power as

$$p(t) = |s(t)|^2 \tag{3.2}$$

where $s(t)$ is either voltage or current.

The *energy* of the signal over some time $2T$ is

$$E = \int_{t=-T}^{T} |s(t)|^2 dt \tag{3.3}$$

If we want the *total energy*, we define this as

$$\lim_{T\to\infty} E = \int_{t=T}^{T} |s(t)|^2 dt \tag{3.4}$$

Note that this is the energy of the signal *over some time* T; it's not the *instantaneous energy* that we seek to define.

Another way to define energy, is to Fourier transform the signal, and use the absolute value squared of the different frequency bands as a measure for the energy level of the respective bands. In the discrete case, we can say that if

$$S[n] = DFT\{s\} \tag{3.5}$$

then $|S[n]|^2$ determines how much energy the signal contains at frequency band n. DFT is the Discrete Fourier Transform operator

$$S[n] = \frac{1}{N}\sum_{i=0}^{N-1} s[i]e^{-j\frac{2\pi ni}{N}} \tag{3.6}$$

This means that a signal of 10Hz with amplitude 1 and another signal of 1000 Hz and amplitude of 1 will contain the same energy (but in different bands) using this model.

By studying a second order differential equation (as the one below), we find that the energy to *generate* a simple sinusoidal signal varies as a function of both amplitude and frequency. In light of this, the above definition therefore seems a bit odd. This observation is what Kaiser used to derive the Teager Energy Operator.

3.2 Teager's Energy Operator

There are apparently more than one way to derive the basic definition of the Teager Operator. In fact, in the first articles by H. M. Teager [2, 1], no definition was given; only plots from the operator were shown. The first article by Kaiser [3] also shows traces of this, and suggests that Kaiser did not know much about Teager's expression used in the H. M. Teager's articles.

In "On a Simple Algorithm to Calculate the 'energy' of a Signal" by Kaiser [3], only the discrete definition is investigated and explained. This article looks at the process that *generated* the signal, and shows how we can express the energy from the signal as a simple and elegant (discrete) function. The Teager Operator has since been defined for continuous signals, both real and complex ones.

Kaiser used the following differential equation as a starting point for the operator:

$$\frac{d^2x}{dt^2} + \frac{k}{m}x = 0 \tag{3.7}$$

This second order differential equation describes an object with mass m suspended by a string with constant k. We can regard this as a *simple* (but incomplete) model of a mechanical-acoustical system, where the object may oscillate, thus creating pressure waves in the surrounding medium. None of the medium's characteristics are included in the model, though.

The solution to equation 3.7 is a periodic oscillation given by $x(t) = A\cos(\omega t + \phi)$ where $x(t)$ is the position of the object at time t, A is the amplitude of the oscillation, $\omega = \sqrt{k/m}$ is the frequency of the oscillation, and ϕ is the initial phase. If $\phi \neq 0$, we have that the object is not initially in equilibrium.

The total energy of the object is in Newtonian physics given as the sum of the potential energy of the spring and the kinetic energy of the object, given by

$$E = \frac{1}{2}kx^2 + \frac{1}{2}mv^2 \tag{3.8}$$

By substituting $v = dx/dt$, and $x = A\cos(\omega t + \phi)$, we get

$$E = \frac{1}{2}m\omega^2 A^2 \tag{3.9}$$

From this, we immediately see that the energy of the object is proportional to *both* A and ω. Note that the energy E is implicitly a function of time.

3.2.1 The continuous Teager Energy Operator

Historically, the discrete version of the Teager Energy Operator was defined first. However, it is natural to begin with the continuous operator here. In continuous time, we define the Teager Energy Operator to be

$$\Psi(x(t)) = (\dot{x}^2(t)) - x(t)\ddot{x}(t) \tag{3.10}$$

Inserting $x(t) = A\cos(\omega t)$ into the equation above yields:

$$\begin{aligned}\Psi(x(t)) &= (-A\omega\sin(\omega t))^2 - A\cos(\omega t)(-\omega^2 A\cos(\omega t)) \\ &= A^2\omega^2(\sin^2(\omega t) + \cos^2(\omega t)) \\ &= A^2\omega^2\end{aligned} \tag{3.11}$$

which is the amplitude and frequency product squared.

3.2.2 The discrete Operator

We now look at a digital signal x_n given by

$$x_n = Acos(\Omega n + \phi) \tag{3.12}$$

where Ω is the digital frequency $\Omega = 2\pi f/F_s$, where ϕ is an arbitrary phase, f is the analog frequency, and F_s is the sampling frequency.

The equation above has three parameters, which means that we *in principle* should be able to set up three different instances of this formula, and solve it to determine the three unknowns. To do this, we choose x_{n-1}, x_n, and x_{n+1} as unknowns:

We then get

$$x_n = A\cos(\Omega n + \phi) \tag{3.13}$$

$$x_{n-1} = A\cos(\Omega(n-1) + \phi) \tag{3.14}$$

$$x_{n+1} = A\cos(\Omega(n+1) + \phi) \tag{3.15}$$

By using the trigonometric identities

$$\cos(\alpha+\beta)\cos(\alpha-\beta) = \frac{1}{2}[\cos(2\alpha) + \cos(2\beta)] \tag{3.16}$$

$$\cos(2\alpha) = 2\cos^2(\alpha) - 1 = 1 - 2\sin^2(\alpha) \tag{3.17}$$

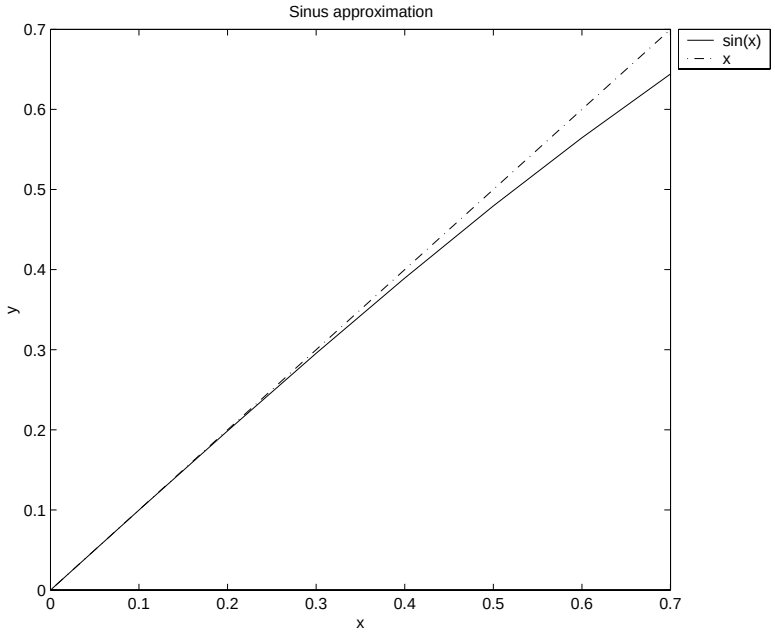

Figure 3.1: Linear approximation of the sine function in the interval $[0, \pi/4]$

we get

$$x_{n-1}x_{n+1} = A^2\cos^2(\Omega + \phi) - A^2\sin^2(\Omega) \tag{3.18}$$

From this, we see that the first term of the sum is x_n^2; we can then write just

$$A^2\sin^2(\Omega) = x_n^2 - x_{n-1}x_{n+1} \tag{3.19}$$

If we restrict Ω to be positive and less than $\pi/2$, the solution above is exact and unique. In practice, this is when Ω is less than one quarter of the sampling frequency.

The formula above can be approximated further if we notice that $\sin(\Omega) \approx \Omega$, for small Ωs. The approximation error is less than 11% if $\Omega < \pi/4$. From figure 3.1, we see that the largest deviation from $sin(x)$ is at $x = \pi/4$. The ratio between the two functions is $\frac{\pi/4}{\sqrt{2}/2} \approx 1.11$ at that point, which explains the 11% approximation error.

We thus end up with the following basic definition of the discrete Teager Energy Operator:

$$\Psi[x_n] = x_n^2 - x_{n-1}x_{n+1} \tag{3.20}$$

So, for a given signal $x[n]$, we can estimate the instantaneous energy of the signal by substituting for x in the formula above.

3.2.3 Energies of well-known signals

In this section we present energy plots of some well-known signals.

Sinusoidal signal

The sinusoidal signal with constant amplitude and phase is modelled as $s(t) = A\cos(\omega t)$, and we calculate the energy of this to be

$$\Psi(A\cos(\omega t)) = A^2\omega^2 \tag{3.21}$$

which we of course knew would be constant.

Exponential signal

The exponentially decaying signal modelled as $s(t) = e^{-\alpha t}$ has zero Teager Energy:

$$\Psi(e^{-\alpha t}) = (\alpha e^{-\alpha t})(\alpha e^{-\alpha t}) - e^{-\alpha t}(\alpha^2 e^{-\alpha t}) = 0 \tag{3.22}$$

AM signals

We model the AM signal as described in section 2.3.1. Although this is a simple analytic signal, calculating the Teager Energy is rather cumbersome if done directly. Later in this chapter we shall derive a few properties about the Teager Energy Operator that will help us in these calculations.

The Teager Energy of an AM signal is

$$\begin{aligned}\Psi(s_{AM}(t)) &= \dot{a}^2(t)\cos^2(\omega_c t) + a^2(t)\omega_c^2 - a(t)\cos^2(\omega_c t)\ddot{a}(t) \\ &= a^2(t)\omega_c^2 + \cos^2(\omega_c t)\Psi(a(t))\end{aligned} \tag{3.23}$$

We notice that the Teager Energy of an AM signal is composed by a term similar to the energy of a sinusoidal signal, and an oscillation scaled by the Teager Energy of the amplitude signal. Figure 3.2 shows a sample AM signal (a) and its Teager Energy (b). Notice the similarity between the envelope of the AM signal and the output of the Teager Energy Operator.

Exponentially damped sinusoidal signal

An exponentially damped sinusoidal signal is a special case of an AM signal, where the carrier is the sinusoidal signal, and the baseband is the exponential signal. The Teager Energies of these signals are

$$\begin{aligned}\Psi(e^{-\alpha t}A\cos(\omega t)) &= e^{-2\alpha t}\Psi(\cos(\omega t)) + \cos^2(\omega t)\Psi(e^{-\alpha t}) \\ &= e^{-2\alpha t}A^2\omega^2 + \cos^2(\omega t)\times 0 \\ &= e^{-2\alpha t}A^2\omega^2\end{aligned} \tag{3.24}$$

which are damped versions of the Teager Energy of simple sinusoidal signals. An example of such a signal is shown in figure 3.3. The envelope of the AM signal is also in this case similar to the output of the Teager Energy Operator. This suggest that the operator can be used to track the envelope of an AM signal. We shall later in chapter 5 see that this indeed is true.

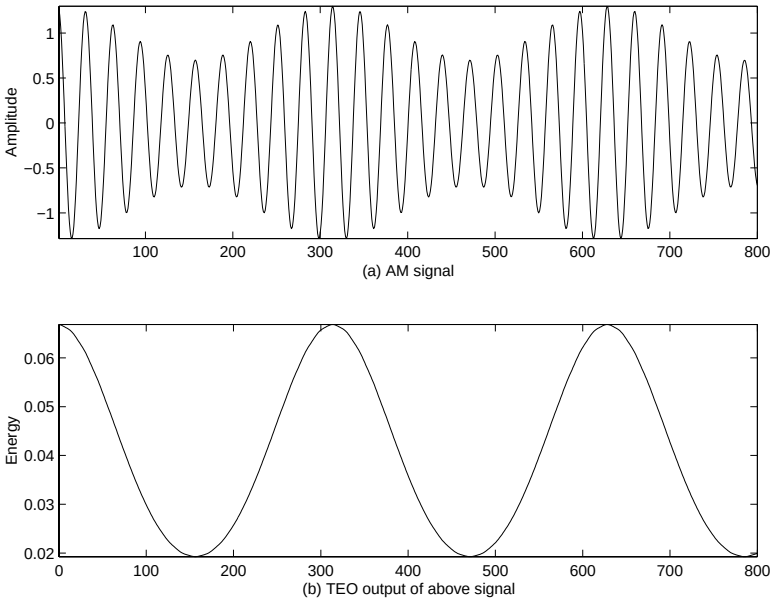

Figure 3.2: Teager Energy of an AM signal.

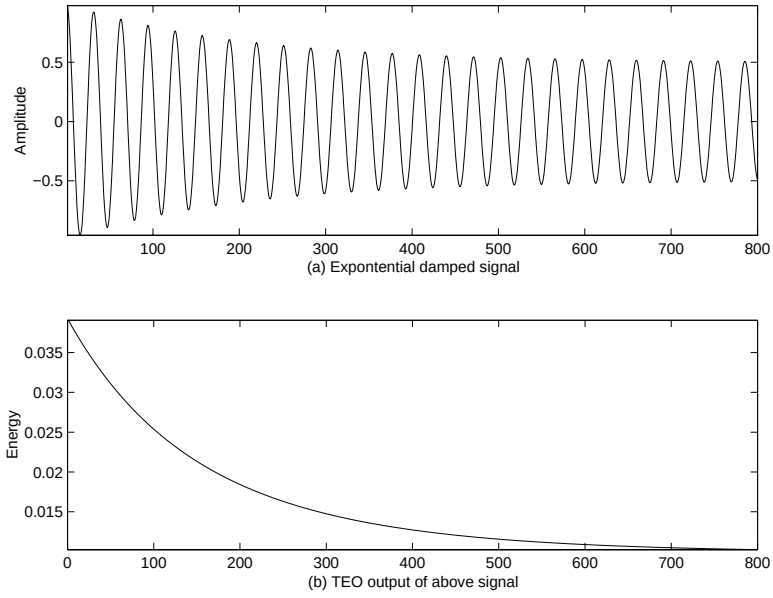

Figure 3.3: Teager Energy of an exponentially damped AM signal.

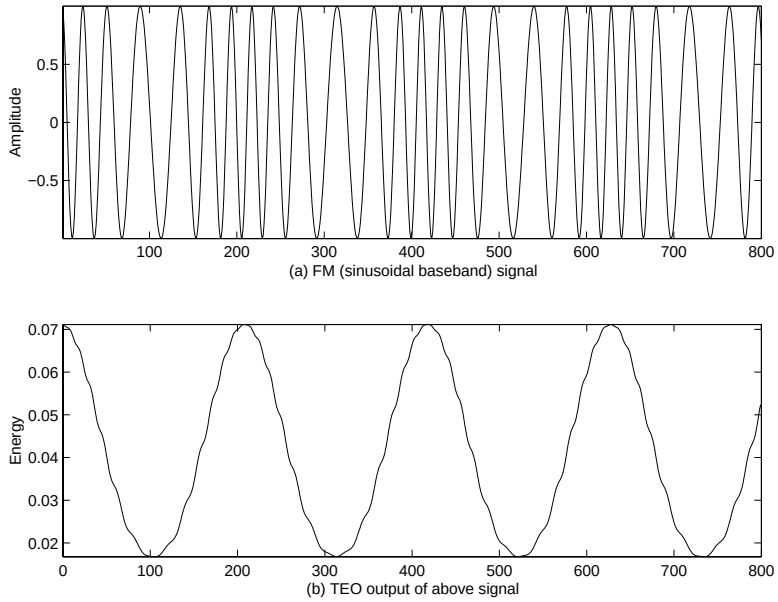

Figure 3.4: Teager Energy of a FM signal where the baseband signal is a sinusoid.

FM signals

The Teager Energy of an FM signal can be found to be

$$\Psi(s_{FM}(t)) = A^2\left(\dot{\phi}^2(t) + \ddot{\phi}(t)\frac{\sin(2\phi(t))}{2}\right) \tag{3.25}$$

In figure 3.4, a sample FM signal is shown. Here, the baseband signal is a pure sinusoidal signal. Notice that the output of the Teager Energy Operator in this case is a sinusoidal signal with the same frequency as the baseband signal (this is harder to see, but can be seen from the zero-crossings of the FM signal).

In figure 3.5, another FM signal is depicted. In this case, we have a chirp signal, where the baseband signal is a linear function, resulting in a quadratic frequency because of the integration. Notice that the output from the Teager Energy Operator also resembles a quadratic function. This suggests that the operator is able to track the (instantaneous) frequency of FM signals.

AM-FM signals

The most complex and general case we will show an example of, is the AM-FM signal. The Teager Energy of an AM-FM signal is

$$\Psi(s_{AM-FM}(t)) = [a(t)\phi(t)]^2 + \underbrace{\frac{1}{2}a^2(t)\ddot{\phi}(t)\sin(2\phi(t))}_{FM} + \underbrace{\cos^2(\phi(t))\Psi(a(t))}_{AM} \tag{3.26}$$

We notice the similarities to both the AM and FM signal energies, where similar terms from both signal types appear in the expression above.

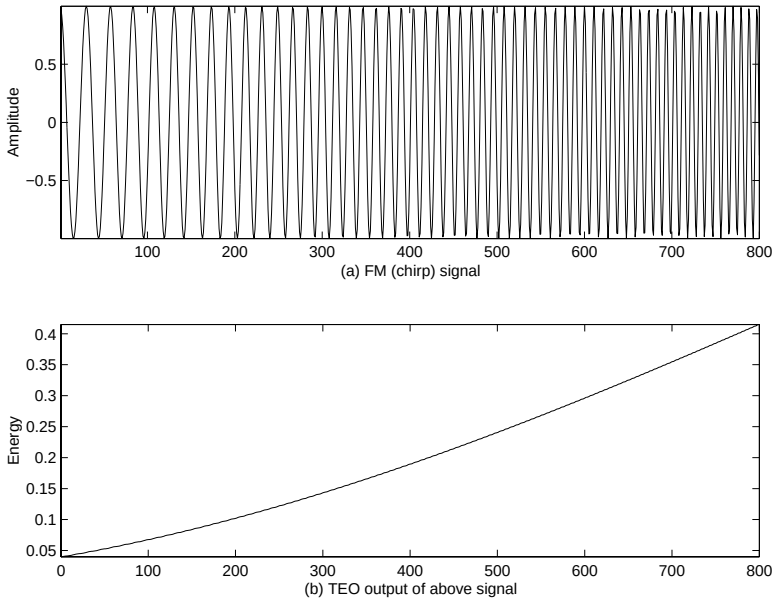

Figure 3.5: Teager Energy of a FM signal where the baseband signal is a linear function.

In figure 3.6, an AM-FM signal is shown. In this case, the operator's tracking capabilities are not apparent, although we can spot some correlation between the peaks of the Teager Energy and the zero-crossings of the AM-FM signal.

3.2.4 Extension to complex signals

The Teager Operator has been extended to cover complex signals. As the Operator is an *energy* operator, we expect the operator to always give positive values. We shall see later that this is not always the case for all signals.

In P. Maragos et. al "Image demodulation using multidimensional energy separation" [26] the definition for complex signals is

$$\Psi_C(x(t)) = ||\dot{x}(t)^2|| - Re[x^*(t)\ddot{x}(t)] \tag{3.27}$$

and in "Teager Energy and the Ambiguity Function" [22] by Hamila et. al, the operator is defined for complex signals as

$$\Psi_C(x(t)) = \dot{x}(t)^*\dot{x}(t) - \frac{1}{2}[\ddot{x}(t)x^*(t) + x(t)\ddot{x}^*(t)] \tag{3.28}$$

Although written differently, the operators above are equal because

$$||\dot{x}(t)^2|| = \dot{x}(t)^*\dot{x}(t) \tag{3.29}$$

by definition, and (omitting t for clarity)

$$\begin{aligned} Re[x^*(t)\ddot{x}(t)] &= Re[(x_r - jx_j)(\ddot{x}_r + j\ddot{x}_j)] \\ &= Re[x_r\ddot{x}_r + jx_r\ddot{x}_j - jx_j\ddot{x}_r + x_j\ddot{x}_j] \\ &= x_r\ddot{x}_r + x_j\ddot{x}_j \end{aligned} \tag{3.30}$$

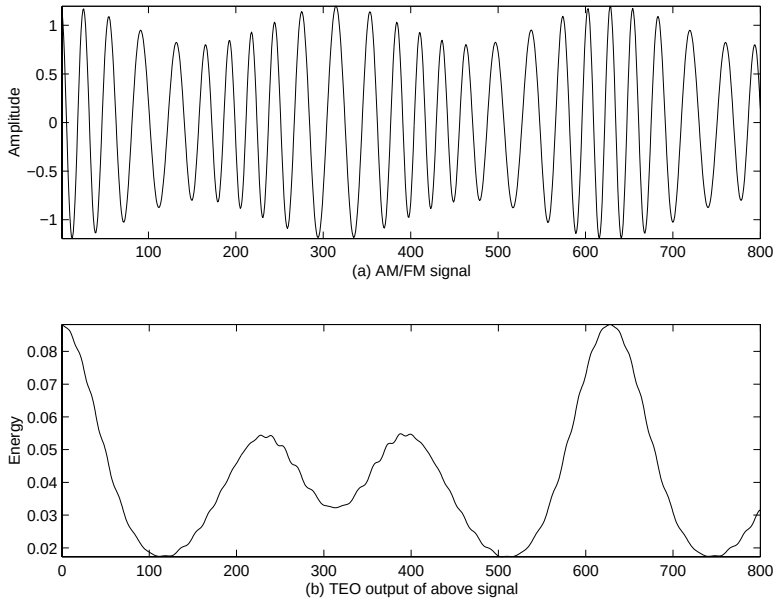

Figure 3.6: Teager Energy of an AM-FM signal.

and

$$\begin{aligned}\frac{1}{2}[\ddot{x}x^* + x\ddot{x}^*] &= \frac{1}{2}\{[\ddot{x}_r + j\ddot{x}_j][x_r - jx_j] + [x_r + jx_j][\ddot{x}_r - j\ddot{x}_j]\} \\ &= \frac{1}{2}[\ddot{x}_r x_r - j\ddot{x}_r x_j + j\ddot{x}_j x_r + \ddot{x}_j x_j + x_r\ddot{x}_r - jx_r\ddot{x}_j + jx_j\ddot{x}_r + x_j\ddot{x}_j] \\ &= x_r\ddot{x}_r + x_j\ddot{x}_j\end{aligned} \tag{3.31}$$

When $x(t)$ is a real signal, the operator reduces to the conventional definition of the Teager Operator.

We can gain more insight if we write the complex operator as the sum of the real and imaginary parts: $x(t) = x_r(t) + jx_j(t)$.

$$\Psi_C(x(t)) = \Psi_C[x_r(t) + jx_j(t)] \tag{3.32}$$

$$= \dot{x}_r(t) + \dot{x}_j^2(t) - x_r(t)\ddot{x}_r(t) - x_j(t)\ddot{x}_j(t) \tag{3.33}$$

$$= \Psi_R(x_r(t)) + \Psi_R(x_j(t)) \tag{3.34}$$

This means that the Teager Energy of a complex signal is the sum of the energy of the real and imaginary parts of the signal.

3.2.5 More generalizations of the Teager Operator

The discrete version of the Teager Operator can be generalized even more. In "A generalization to the Teager-Kaiser energy function & application to resolving two closely spaced tones" [27] by Wei Lin et.al the operator is generalized by replacing '1' with a constant M in the expression $x[n-1]x[n+1]$. The operator can then be written as

$$\Psi[x[n]] = x^2[n] - x[n-M]x[n+M] \tag{3.35}$$

where M is an arbitrary integer ≥ 1. This constant M is called the *lag parameter*. This generalization can be used to resolve two closely spaced tones and for action potential detection in neural research [17].

In "A generalization of the Teager Algorithm" by Michael Moore et. al, the Operator is generalized further, by introducing/changing the exponent of the two expressions in the discrete versions. First, a quadratic Volterra filter given as a convolution between the signal $x[n]$ and Volterra kernel $h_2[n_1, n_2]$ is constructed. We will address Volterra systems in the next chapter; the formulas below suffice for now, as we summarize the different extensions. Then the output signal is given by

$$y(n) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} h_2[k_1, k_2] \cdot x[n-k_1]x[n-k_2] \tag{3.36}$$

The above formula is the conventional Teager Energy Operator expressed as a discrete quadratic Volterra-filter, if the kernel h_2 is chosen properly. In "A new class of nonlinear filters for image enhancement" by S.K. Mitra et.al, it is shown that $y[n] \approx \mu[-x[n-1]+2x[n]-x[n+1]]$, where μ is the local mean, $\mu = (x[n-1]+x[n]+x[n+1])/3$. This means that the output from the Teager Energy Operator is approximately equal to a highpass filter weighted by the local mean. This mean value can be adjusted by changing the exponents in the original Teager Energy Operator like this:

$$y[n] = x[n]^{\frac{2}{m}} - (x[n-1]x[n+1])^{\frac{1}{m}} \tag{3.37}$$

For $m = 1$, the extension reduces to the conventional Teager Energy Operator. It should be noted that there are certain restrictions for the validity of this approximation, which we will describe in the next chapter.

By combining the two methods above, the lag and the exponent, the discrete version can finally be written as

$$y[n] = x[n]^{\frac{2}{m}} - (x[n-M]x[n+M])^{\frac{1}{m}} \tag{3.38}$$

3.3 Analytic properties

In this section, we list the most basic properties of the standard Teager Energy Operator, $\dot{x}^2 - x\ddot{x}$. These properties are regularly used throughout the thesis, and especially the result about multiplication of two functions is very useful as this result can be applied directly when we want to compute the energy of a closed-form AM signal. An extensive list of properties can be found in [28].

3.3.1 $\Psi_c(f(t), g(t))$

It is useful to define an operator that is quite similar to the Teager Energy Operator, but this new energy-like operator takes two arguments, and these arguments, $f(t)$ and $g(t)$, are combined like this:

$$\Psi_c(f, g) \triangleq \dot{f}\dot{g} - f\ddot{g} \tag{3.39}$$

This new operator has some useful properties. For instance,

$$\Psi_c(f, f) = \Psi(f) \tag{3.40}$$
$$\Psi_c(0, f) = \Psi_c(f, 0) = 0 \tag{3.41}$$
$$\Psi_c(b, f) = b\ddot{f} \tag{3.42}$$
$$\Psi_c(f, b) = 0 \tag{3.43}$$

where f is a function of t and b is a constant.

Notice that this operator is not commutative, as $\Psi_c(f, g) \neq \Psi_c(g, f)$ in general:

$$\Psi_c(f, g) = \dot{f}\dot{g} - f\ddot{g} \tag{3.44}$$

and

$$\Psi_c(g, f) = \dot{g}\dot{f} - g\ddot{f} \tag{3.45}$$

As a counter-example, we set $f(t) = at$ and $g(t) = bt^2$. Then $\Psi_c(f, g) = 2abt - 2abt$ and $\Psi_c(f, g) = 2abt$, thus the operator is not commutative.

3.3.2 $\Psi(f(t) \times g(t))$

The Teager Energy of the multiplication of two signals f and g is:

$$\Psi(g \times h) = g^2\Psi(h) + h^2\Psi(g) \tag{3.46}$$

This property is convenient to use when the Teager Energy of an AM signal is to be computed.

3.3.3 $\Psi(f(t) + g(t))$

The Teager Energy of the sum of two signals $f(t)$ and $g(t)$ is:

$$\Psi(f + g) = \Psi(f) + \Psi(g) + \underbrace{\Psi_c(f, g) + \Psi_c(g, f)}_{\text{cross terms}} \tag{3.47}$$

The cross terms are conveniently expressed using the $\Psi_c(\cdot, \cdot)$ operator, and are terms composed by both f and g.

3.3.4 $\Psi(b)$

The Teager Energy of a constant is 0. Maybe surprisingly, the Teager energy of $e^{\alpha t}$ is also 0.

3.3.5 $\Psi(b + f(t))$

Using the properties above, the energy of a constant plus a time-varying signal is

$$\Psi(b + f(t)) = \Psi(b) + \Psi(f) + \Psi_c(b, f) + \Psi_c(f, b) \tag{3.48}$$
$$= \Psi(f) + \Psi_c(b, f) + \Psi_c(f, b) \tag{3.49}$$
$$= \Psi(f) + b\ddot{f} \tag{3.50}$$

3.3.6 $\Psi(f(g(t)))$

The Teager Energy of two nested functions is given by

$$\Psi(f(g(t))) = \dot{g}^2 \left[\left(\frac{df}{dg} \right)^2 - f \frac{d^2 f}{dg^2} \right] - f \frac{df}{dg} \ddot{g} \tag{3.51}$$

This property can be used to evaluate the Teager Energy of FM signals. Combined with equation 3.46, we can evaluate the Teager Energy of AM-FM signals rather easily.

3.4 Positivity of the Energy Operator

As the Teager Energy Operator, is an *energy* operator, and energy is a positive quantity, we must investigate whether the Teager Energy Operator is zero or positive for all signals. This issue was first addressed by Petros Maragos et al in "On Amplitude and Frequency Demodulation Using Energy Operator" [9], and was later discussed more in "Conditions for Positivity of an Energy Operator" [29] by Alan C. Bovik and Petros Maragos.

3.4.1 Real-valued signals

We start by looking at the continuous version of the Teager Energy Operator:

$$\Psi(x(t)) = \dot{x}(t)^2 - x(t)\ddot{x}(t) \tag{3.52}$$

To make the right hand side less than zero, it's obvious that the only way we can achieve this is to have

$$\dot{x}(t)^2 \leq x(t)\ddot{x}(t) \tag{3.53}$$

Now, for real signals, the Teager Energy Operator is nonnegative if any of the following conditions hold

1. $x(t) = 0$
2. $\ddot{x}(t) = 0$
3. $x(t) < 0$ and $\ddot{x}(t) > 0$
4. $x(t) > 0$ and $\ddot{x}(t) < 0$

This is because in the cases above, the right hand side of equation 3.53 is negative or zero in all these cases. The left hand side is always positive, regardless of the value of $\dot{x}(t)$, as it is squared. Thus, in these cases $\Psi(x(t)) > 0$.

AM, FM, and AM-FM signals

For AM signals, we can use property 3.46 to find a condition for nonnegativity. This property implies that the Teager Energy of a product of two signals is nonnegative provided that each component has nonnegative Teager Energy. In terms of the envelope $a(t)$, this means that AM signals have nonnegative Teager Energy if $\Psi(a(t)) > 0$ for all t, as the factor $\cos(\omega_c t)$ is positive or zero for all values of t.

For the general class of AM-FM signals, we can guarantee nonnegativity provided that the parameters are bounded as below. The Teager Energy of an AM-FM signal was earlier found to be (omitting t for clarity)

$$\Psi(s_{AM-FM}) = [a\phi]^2 + \frac{1}{2}a^2\ddot{\phi}\sin(2\phi) + \cos^2(\phi)\Psi(a) \tag{3.54}$$

Clearly, the first term is always nonnegative, so a sufficient, but not necessary, condition for nonnegativity is

$$\left\{\frac{1}{2}a^2\ddot{\phi}\sin(2\phi) + \cos^2(\phi)\Psi(a)\right\}_{max} \leq [a\phi]^2_{min} \tag{3.55}$$

or expressed in terms of a and the FM baseband signal m

$$\frac{1}{2}(1+\kappa)\omega_m\dot{m}_{max} + \Psi(a)_{max} \leq (1-\kappa)^2(\omega_c - \omega_m)^2 \tag{3.56}$$

Using the above result, we can simplify the condition for FM signals, where $a(t) = 1$ for all t. We simplify as follows:

$$\omega_m\dot{m}_{max} \leq 2(\omega_c - \omega_m)^2 \tag{3.57}$$

3.4.2 Discrete version

We can get similar results for positivity for the traditional definition of the Teager Energy Operator, defined as

$$\Psi[x[n]] = x[n]^2 - x[n-1]x[n+1] \tag{3.58}$$

The discrete nature of the equations make the calculations and arguments a bit more complex, but the results are essentially the same.

3.4.3 Complex signals

We will now consider the Teager Energy Operator for complex-valued signals. This was not covered by the articles mentioned above, as the complex Teager Energy Operator had then not yet been defined.

The definition of the complex Teager Energy Operator is defined as

$$\Psi_C(x(t)) = \dot{x}(t)^*\dot{x}(t) - \frac{1}{2}[\ddot{x}(t)x^*(t) + x(t)\ddot{x}^*(t)] \tag{3.59}$$

which reduces to the traditional definition whenever $x(t)$ is real. If we write the input signal as $x(t) = x_r(t) + jx_j(t)$, the output can be written as $\Psi_R(x_r(t)) + \Psi_R(x_j(t))$. This means that the energy will be positive as long as the largest absolute term is positive. Effectively, the output for the Teager Energy Operator is positive for a broader range of signals, as one of the terms is now allowed to be negative.

To visualize the positivity constraint, figure 3.7 sketches the complex plane, with the output of $\Psi_R(x_r(t))$ on the horizontal axis and the output from $\Psi_R(x_j(t))$ on the vertical axis. Teager Energies in the shaded area are negative.

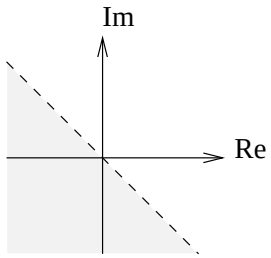

Figure 3.7: Complex plane visualizing where the Complex Teager Energy Operator is negative.

3.4.4 Signals yielding "negative Teager Energy"

Figure 3.8 shows one signal, and its Teager Energy. For some values, we get "negative" energy. This is definitely an awkward behaviour for an energy operator, and we are interested in how we can construct such signals that for some values give us this result.

We must remember that from the definition, the Teager Energy Operator tries to model the energy of the *source* of the signal, and not the signal we are actually measuring, so that is probably part of the explanation. This phenomenon is also discussed in "Teager Energy and the Ambiguity Function", by R. Hamila et. al [22].

The signal in figure 3.8 is constructed using two simple sinusoidal signals, where one has significantly higher frequency, but has less amplitude than the other signal. We can look at this as a model of two separate sources, one high and one low frequency source, where the high frequency source is placed farther away than the low frequency source. In this example, we incorrectly assume that only one source is generating the signal, so this might explain why we get negative Teager Energy

Of course, the same signal *could* be generated by one system, but this would also differ from the simple model we used in the definition of the Teager Energy Operator, as the resulting signal contains two oscillating signals.

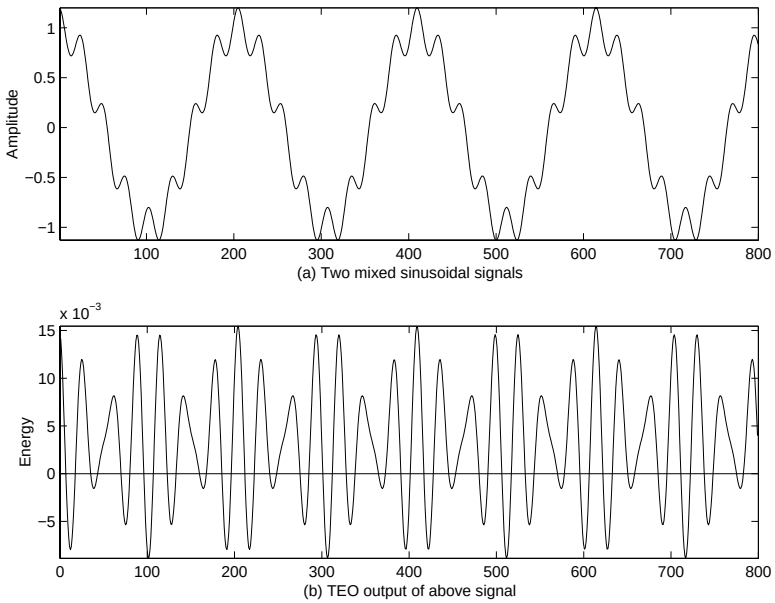

Figure 3.8: The Teager Energy Operator output of the signal is negative for some parts of the signal.

Chapter 4

Volterra systems and the Teager Energy Operator

When processing signals, one can divide the modelling and the processing into two categories: Linear models and nonlinear models. Of these two, the linear models have received most attention, as these models are easiest to understand and use. The basic definitions used in linear models are included here, but for a more extensive treatment of this, [30] should be consulted.

4.1 Operators

We can describe a general system with one input and a corresponding output as

$$y(t) = T\{x(t)\} \tag{4.1}$$

We call T an operator, transforming $x(t)$ to $y(t)$.

4.2 Linear signal models

A system is said to be linear if the following hold:

$$T\{x_1(t) + x_2(t)\} = T\{x_1(t)\} + T\{x_2(t)\} \tag{4.2}$$

and

$$T\{ax(t)\} = aT\{x(t)\}, \qquad a \text{ constant} \tag{4.3}$$

These two equations may be merged by including the scaling constant in the first equation. By an induction argument, the equations above can be extended to cover an arbitrary number of signals.

If the system does not change with time, we say that the system is time invariant, and such systems have the following property:

If

$$y(t) = T\{x(t)\} \tag{4.4}$$

and

$$y(t + \tau) = T\{x(t + \tau)\} \tag{4.5}$$

for all τ then the system is said to be time invariant.

Systems for which the above properties hold are often called linear time invariant systems (LTI systems), and these are often implemented by FIR filters.

4.3 Nonlinear signal models

Nonlinear models are systems where either the additive (4.2) or scalability (4.3) property does not hold in general. Nonlinear systems may in any case be time invariant, as we shall see below.

A simple example of a nonlinear system is

$$y(t) = \sqrt{x(t)} \tag{4.6}$$

where the output signal is simply the square root of the input. Considering that $\sqrt{a} + \sqrt{b} \neq \sqrt{a+b}$ in general for two arbitrary constants, we can immediately say that the system is nonlinear.

Nonlinear systems can be divided into subclasses, and we will look at a special class called Volterra filters, as we will show that the Teager Energy Operator can be described by a Volterra filter later.

4.3.1 Volterra filters

Vito Volterra was a mathematician born in 1860 in Italy. His interest in mathematics started early, and during his life, he published important papers on differential equations, integral equations, and functional analysis. His contributions have been used in the study of nonlinear systems, where one particular (wide) class of nonlinear operators is called Volterra systems.

An in-depth description of this class, seen from a signal processor's view, can be found in "The Volterra and Wiener Theories of Nonlinear Systems" by M. Schetzen [24].

Using Volterra system theory, we are able to construct and synthesize filter structures containing lower order systems (LTI systems, for instance), and multipliers.

Volterra operators

A first order Volterra operator is defined as the convolution

$$y(t) = \int_{-\infty}^{\infty} x(t-\tau)h_1(\tau)d\tau \tag{4.7}$$

where $h_1(t) = 0$ for $t < 0$. This constraint makes the Volterra filters causal (output values are not dependent on input values "in the future".) $h_1(t)$ is called the *first order Volterra kernel*.

We note that this is exactly the same definition as for linear filters, but we have added the causality constraint.

It can be shown that first order Volterra systems are stable in a bounded input-bounded output (BIBO) sense; if the input is bounded by some constant, then the system is bounded (and thus stable) if

$$\int_{-\infty}^{\infty} |h_1(\tau)|d\tau \leq M \tag{4.8}$$

where $M < \infty$ is a constant. This is both a necessary and sufficient condition for stable 1D systems.

We can extend the definition above in a natural way (actually, the extension is more complicated than this, but omitted here), and add a two dimensional convolution integral and get

$$y(t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(t-\tau_1)x(t-\tau_2)h(\tau_1,\tau_2)d\tau_1 d\tau_2 \tag{4.9}$$

where $h(t_1, t_2) = 0$ for $t_1 < 0$ or $t_2 < 0$. This is called a second order Volterra system, and h_2 is the *second order Volterra kernel*. The constraints put on t_1 and t_2 make the system causal.

The 2D system is stable if

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |h_2(\tau_1,\tau_2)|d\tau_1 d\tau_2 \leq M, \qquad M < \infty \tag{4.10}$$

This is a sufficient condition, but contrary to the 1D case, it is *not* a necessary condition. This is not apparent from the expression above, but examples can be constructed where the criterion above does not hold, but the stability of the filter can be shown in other ways; [24] contains an example of this.

We can extend the idea further to create higher order Volterra systems by adding more terms in a similar way to the expression above, and then get arbitrarily complex systems.

A general Volterra system of order p is the sum of p expressions H_p, where H_p is the p-dimensional convolution integral.

4.4 The Teager Energy Operator seen as a Volterra system

The connection between Volterra systems and the Teager Energy Operator was made by Michael Moore et.al in [20]. In that article, only the discrete version is discussed, and it is shown that the Teager Energy Operator can be approximated by a discrete high-pass Volterra system. In the following section, we will look at the continuous case, but we will first summarize the work by Moore and Mitra.

4.4.1 The discrete version

The discrete version of a Volterra filter can be defined by substituting the integrals of the continuous version by sums in the discrete case:

$$y[n] = \sum_{i=0}^{N-1} x[n-i]h_1[i] \tag{4.11}$$

where $h_1[i]$ is the impulse response (or Volterra kernel), and N is the length of h_1. We recognize this as the conventional convolution used to describe FIR filters. The two-dimensional discrete Volterra filter is then

$$y[n] = \sum_{j=0}^{N-1} \sum_{i=0}^{N-1} x[n-i]x[n-j]h_2[i,j] \tag{4.12}$$

where $h_2[i,j]$ is the two-dimensional Volterra kernel.

We will now analyze the Teager Energy Operator in light of the above definition. The discrete version is as usual

$$y[n] = x[n]x[n] - x[n-1]x[n+1] \tag{4.13}$$

We recognize this (almost) as a two-dimensional Volterra system with the kernel h_2 defined for instance as

$$\begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

This kernel is not symmetric, but we can make it symmetric by balancing the coefficients and get

$$\begin{bmatrix} 0 & 0 & -\frac{1}{2} \\ 0 & 1 & 0 \\ -\frac{1}{2} & 0 & 0 \end{bmatrix}$$

Notice that using the Volterra definition above, $h_2[i,j]$ is only defined for $i \geq 0$ and $j \geq 0$. This means that the output of the system is one sample delayed, because we require the Volterra system to be causal.

In [8], Mitra et.al shows that the Teager Energy Operator can be approximated by a high-pass filter weighted by a local mean filter. We will now show why this is so.

We first assume that the signal on average equals k. We then define $x_1[n] = x[n] - k$, which is a sequence with zero mean. We can now rewrite the Teager Energy Operator of the signal $x[n]$ to

$$\begin{aligned} y[n] &= [x_1[n]+k]^2 - [x_1[n-1]+k][x_1[n+1]+k] \\ &= x_1^2[n] + 2kx_1[n] + k^2 - x_1[n-1]x_1[n+1] - kx_1[n-1] - kx_1[n+1] - k^2 \\ &= \underbrace{k\{2x_1[n] - x_1[n-1] - x_1[n+1]\}}_{y_1[n]} + \{x_1^2[n] - x_1[n-1]x_1[n+1]\} \end{aligned} \tag{4.14}$$

We now assume that x_1 is a random variable uniformly distributed in the interval $[-\Delta, \Delta]$. Additionally, we define a sequence $y_2[n] = x_1[n] - x_1[n-1] - x_1[n+1]$.

The variances of the two sequences can be found to be

$$\mathrm{Var}\{y_1\} = 2k^2\Delta^2 \tag{4.15}$$

$$\mathrm{Var}\{y_2\} = \frac{\Delta^4}{5} \tag{4.16}$$

The ratio between the two variances is then $\frac{10k^2}{\Delta^2}$. If $k >> \Delta$, then y_1 is the dominant term in equation 4.14, and we can write an approximation of $y[n]$ as

$$\begin{aligned} y[n] &= k\{2x_1[n] - x_1[n-1] - x_1[n+1]\} \\ &= k\{2(x[n]+k) - (x[n-1]+k) - (x[n+1]+k)\} \\ &= k\{2x[n] - x[n-1] - x[n+1]\} \end{aligned} \tag{4.17}$$

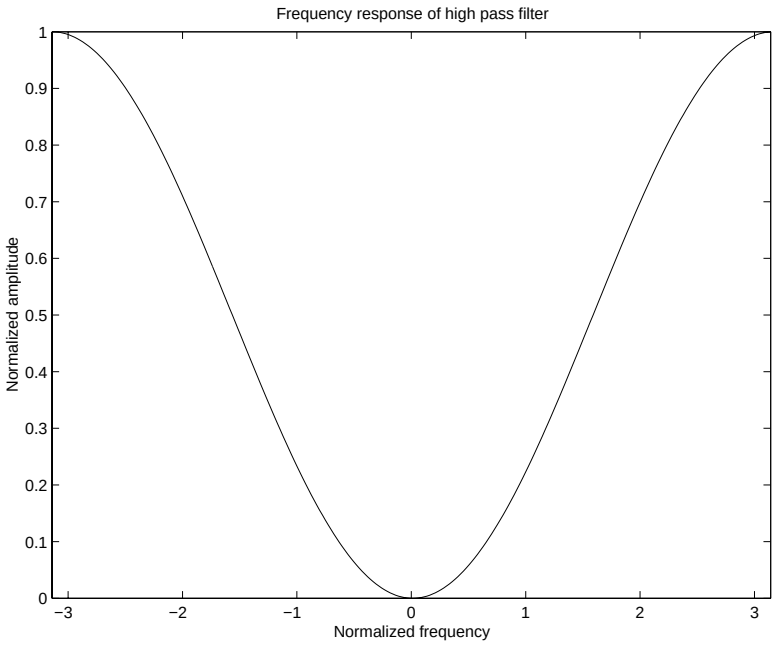

Figure 4.1: Frequency response of the FIR filter [-1 2 -1]

If we look at $y[n]$ as a sequence of length 3, then the k is still the average of the sequence. We can thus say that we only require k to be the *local* average of a possibly longer sequence, which we can express as

$$k = \frac{1}{3}(x[n-1] + x[n] + x[n+1]) \tag{4.18}$$

Thus, we can write an approximation of the Teager Energy Operator as

$$y[n] = (\text{local-mean})(2x[n] - x[n-1] - x[n+1]) \tag{4.19}$$

The result of $2x[n] - x[n-1] - x[n+1]$ can be computed as the response from a linear filter. A frequency response of such a filter is shown in figure 4.1, which shows that the filter has a high-pass frequency response.

Notice that the impulse response $\begin{bmatrix}-1 & 2 & -1\end{bmatrix}$ is equal to the negative of the approximation of a second order derivative filter (which is the convolution of $\begin{bmatrix}-1 & 1\end{bmatrix}$, the first order derivative, with itself: $\begin{bmatrix}1 & -2 & 1\end{bmatrix}$). This is not commented on in the article [20] we used as the basis for these calculations.

We can thus conclude that the Teager Energy operator can be approximated as the negative second derivate of the signal weighted by the local mean.

Regarding the validity of the approximation, it is shown that the approximation is valid if the local mean is much larger than the variance of the signal. This means that the approximation is periodically incorrect for most audio signals, which usually are zero-mean signals.

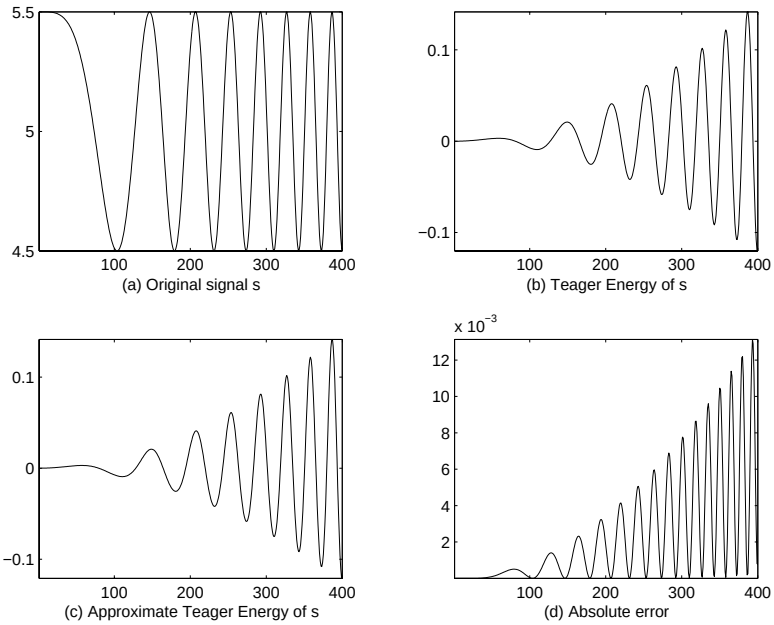

Figure 4.2: Teager Energy Operator approximation where $k >> \Delta$.

Approximation examples

Figure 4.2 shows an example of a chirp signal, the Teager Energy, its approximation, and error. In this first example the approximation is fairly good, because the chirp signal has a mean value of approximately 5 and little variance from the mean value. This means that the condition for a good approximation is met. The error is oscillating (subfigure d), and is in the order of a magnitude less than the Teager Energy.

The second example in figure 4.3 shows a signal under which the condition above is not valid. The mean value of the signal is approximately zero, thus the above k will *not* be much larger than Δ for all values in the sequence, and the condition for a good estimate is not met. The calculated error shows this clearly. Although its shape resembles the error graph in the first example, the error and the approximation are of about the same scale, which shows that the approximation in this case is bad.

4.4.2 The continuous version

In [24], a method for the synthesis of filter networks for Volterra systems is explained. We first start with what we call a basic block, shown in figure 4.4. The basic block contains three linear filters and a multiplier. The input signal is fed to the two linear filters in parallel and the results are then multiplied. The product is then fed to the last linear filter, yielding the final result. In this section, we will show how the Teager Energy Operator can be represented as the composition of multiple basic blocks.

If we can represent our desired system similar to such a structure shown in the figure, we can create a second order Volterra system that represents that the structure.

The Volterra system for a basic block can be represented as follows. We can write

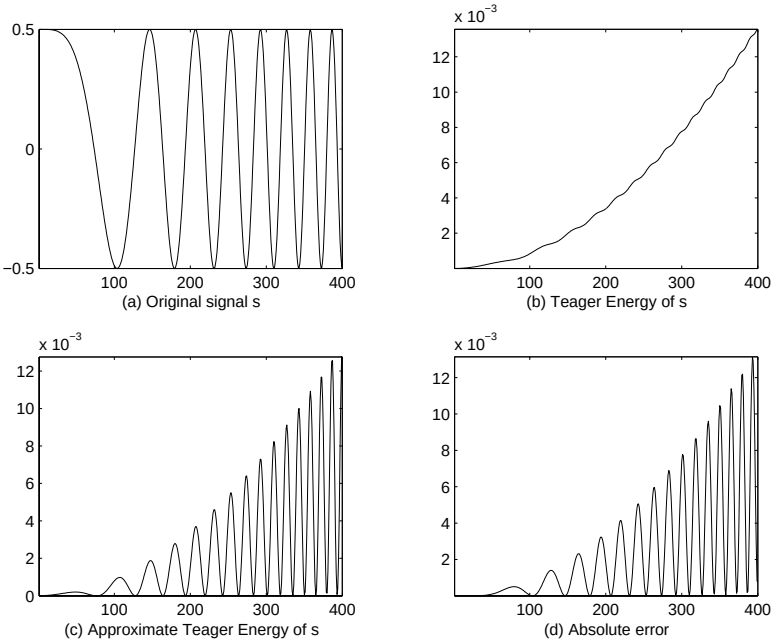

Figure 4.3: Teager Energy Operator approximation where the condition is not valid for most of the sequence.

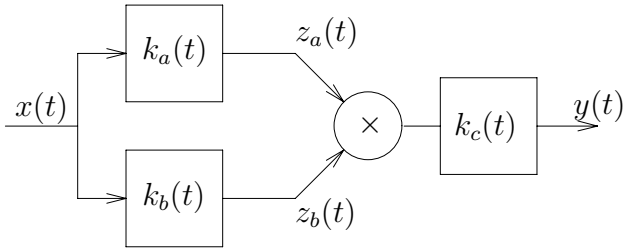

Figure 4.4: Basic block with three linear filters and a multiplier.

the result $y(t)$ as

$$\begin{aligned} y(t) &= \int_{-\infty}^{\infty} k_c(\sigma) z_a(t-\sigma) z_b(t-\sigma) d\sigma \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} k_c(\sigma) k_a(\sigma_1) k_b(\sigma_2) x(t-\sigma-\sigma_1) x(t-\sigma-\sigma_2) d\sigma_2 d\sigma_1 d\sigma \end{aligned} \tag{4.20}$$

If we set $\sigma + \sigma_1 = \tau_1$, $\sigma + \sigma_2 = \tau_2$, and then

$$k_2(\tau_1, \tau_2) = \int_{-\infty}^{\infty} k_c(\sigma) k_a(\tau_1 - \sigma) k_b(\tau_2 - \sigma) d\sigma$$

we can write the expression in a normal form. The important thing here is the kernel k_2, which we want to Laplace transform. We set $s_1 = \sigma_1 + j\omega_1$ and $s_2 = \sigma_2 + j\omega_2$, and get

$$K_2(s_1, s_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} k_2(\tau_1, \tau_2) e^{-(s_1\tau_1 + s_2\tau_2)} d\tau_1 d\tau_2 \tag{4.21}$$

If we set σ_1 and σ_2 to zero, this reduces to the Fourier transform of the kernel k_2 (if it exists).

After some restructuring of this expression, we can write K_2 as a product of three linear functions of s_1 and s_2:

$$K_2(s_1, s_2) = K_a(s_1) K_b(s_2) K_c(s_1 + s_2) \tag{4.22}$$

where

$$K_\alpha(s) = \frac{P(s)}{Q(s)}, \qquad \alpha = a, b, c \tag{4.23}$$

This is a linear system $K(s)$, written as a rational function of two polynomials P and Q, where the P represents the zeros of the system, and the Q represents the poles. In our case, we can say that our P is the product of the zeros of the three linear filters in the basic block, and the Q is composed of the poles of the three filters:

$$K_2(s_1, s_2) = \frac{P_a(s_1) P_b(s_2) P_c(s_1 + s_2)}{Q_a(s_1) Q_b(s_2) Q_c(s_1 + s_2)} \tag{4.24}$$

Systems that are similar in structure to K_2, can thus be created by employing one basic block.

In general, it can be shown that systems where the kernel $H_2(s_1, s_2)$ can be written as

$$H_2(s_1, s_2) = \frac{P(s_1, s_2)}{Q(s_1, s_2)} = \frac{\sum_{m=1}^{M} R_{a_m}(s_1) R_{b_m}(s_2) R_{c_m}(s_1 + s_2)}{F_a(s_1) F_b(s_2) F_c(s_1 + s_2)} \tag{4.25}$$

where

$$R_{\alpha_m}(s) = P_{\alpha_m}(s) \prod_{\substack{j=1 \\ j \neq m}}^{M} Q_{\alpha_j}(s)$$

$$F_\alpha(s) = \prod_{m=1}^{M} Q_{\alpha_m}(s)$$

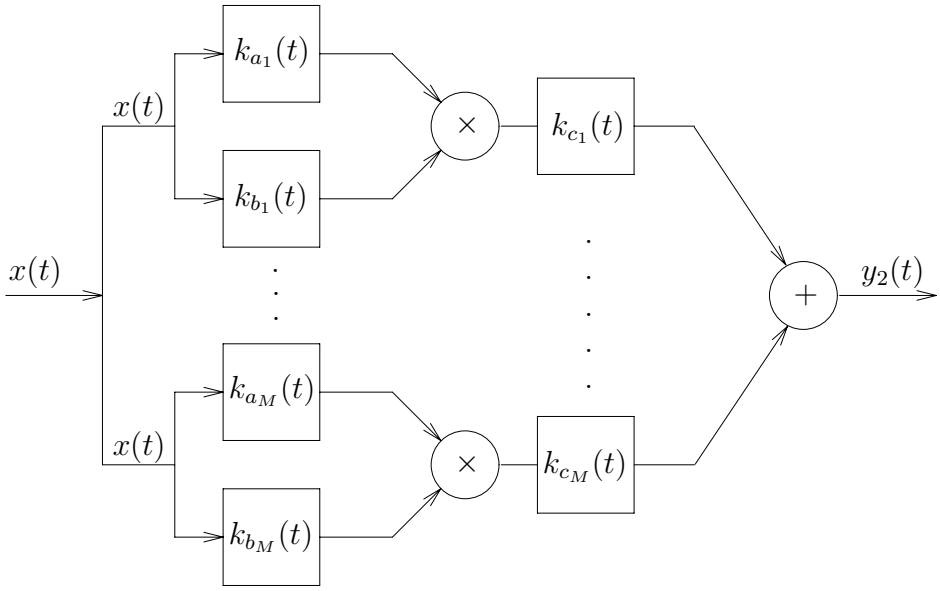

Figure 4.5: Structure of M basic blocks, each with three linear filters and a multiplier

and $\alpha = a, b$ or c, can be realized as a larger system composed by M basic blocks, linked together as shown in figure 4.5.

Figure 4.6 shows a filter network representation of the Teager Energy Operator. We see that this is similar to the structure in figure 4.5, which means that we should be able to construct a kernel $H_2(s_1, s_2)$ for this network.

We first note that the kernels k_{c_i} are removed from the filter network. In the formulas above this means that the k_{c_i} kernels equal 1. The two filters k_{a_1} and k_{b_1} are equal, which means that the poles and zeros are equal. We also note that the kernel k_{a_2} can be expressed as the convolution between two k_{a_1} filters.

The first basic block can be represented as

$$K_2(s_1, s_2) = \frac{P(s_1)P(s_2)}{Q(s_1)Q(s_2)} \tag{4.26}$$

where P and Q are the zeros and poles for k_{a_1}.

The second basic block can be represented as

$$K_2(s_1, s_2) = -\frac{P^2(s_1)}{Q^2(s_1)} \tag{4.27}$$

The final result for the complete characterization of the kernel of Teager Energy Operator should then be

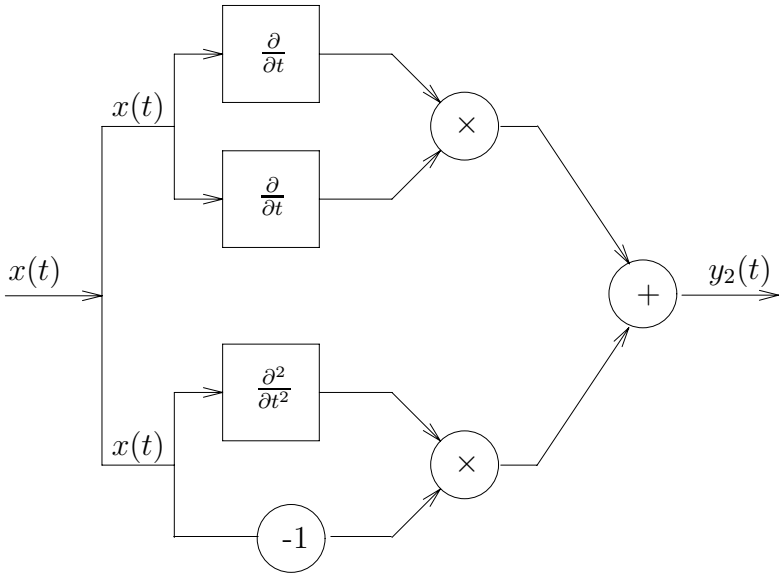

Figure 4.6: Filter network representation of the continuous Teager Energy Operator.

$$\begin{aligned} H_2(s_1, s_2) &= \frac{P(s_1)P(s_2)}{Q(s_1)Q(s_2)} - \frac{P^2(s_1)}{Q^2(s_1)} \\ &= \frac{P(s_1)P(s_2)Q^2(s_1) - P^2(s_1)Q(s_1)Q(s_2)}{Q(s_1)Q(s_2)Q^2(s_1)} \\ &= \frac{P(s_1)[P(s_2)Q(s_1) - P(s_1)Q(s_2)]}{Q(s_2)Q^2(s_1)} \end{aligned} \tag{4.28}$$

Now, the impulse response of the differentiators are neither causal nor finite. This means that the formula above *must be an approximation* if the system is a Volterra system (which must be causal). If we implement the differentiators as FIR filters, the Q's in the equation above will equal 1, and then P will be a finite product of zeros:

$$H_2(s_1, s_2) = P(s_1)P(s_2) - 1 \cdot P^2(s_1) \tag{4.29}$$

Notice the similarity to the continuous time-representation.

4.5 Summary

In this chapter we have presented two different signal models, linear and nonlinear models. As part of the theory for nonlinear models, we find the Volterra theory. This was developed by an Italian mathematician during the late 1800's. We have seen that the Teager Energy Operator may be expressed as a Volterra system, in both continuous time and discrete time. In the discrete case, the operator may be approximated as the local mean multiplied by an appropriate FIR filter. In the continuous-time analysis and synthesis we have used theory from Schetzen's work to express the Teager Energy Operator.

Chapter 5

Modulation and demodulation techniques

In this chapter we define the processes of modulation and demodulation of signals, and show examples of different types of modulations. Additionally, we will define several estimators for instantaneous frequency and amplitude, both in continuous time and discrete time, where appropriate.

It should be noted that there exists other methods to estimate the instantaneous frequency than those presented here, but we have chosen not include them here for several reasons. It is nearly impossible to create an exhaustive list of estimators, and it is beyond the scope of this thesis.

The estimators in this chapter have been selected because they have been used in the Teager Energy Operator related literature, most of them as refinements for other older techniques. Mandelstam's method is included because it shows the usage of an operator $H\{\cdot\}$ as defined in chapter 2. Shekel's method is included because of its simplicity both in derivation and implementation.

5.1 Modulation and demodulation

Modulation of a signal is the process in which a signal carrying information is transformed into another signal, typically suitable for transmission over some channel. The original signal is usually called the baseband signal, and the *modulation* process transforms the frequency range in the baseband signal to some other (higher) frequency band.

The *demodulation* process is the reverse of the modulation process; recovering the baseband signal from the modulated signal.

When modulation is used in communication systems, the receiver knows the parameters of the transmission (carrier wave frequency, for instance). This means that the demodulation process is just a matter of "tuning" the parameters (that is, the known carrier frequency) to the right values, and the result will "pop" out.

In this thesis, this is not so. Here, we aim to determine the parameters of some *unknown* signal, which we assume to be either AM, FM or both. We will see that using the Teager Energy Operator, we can get nice and simple analytic results for this. We first describe two techniques as used in communication systems.

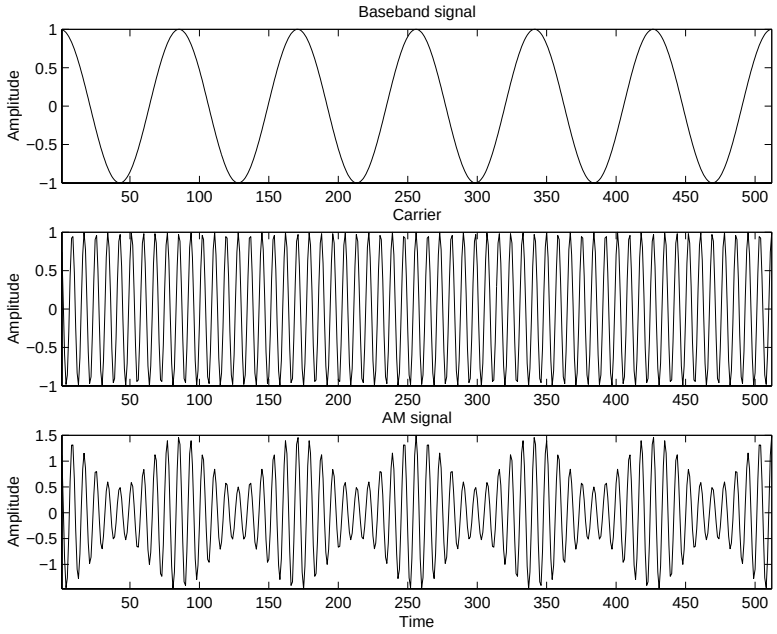

Figure 5.1: Amplitude modulation process. Baseband signal (a) is modulated with carrier (b), and we get the final modulated signal in (c).

5.2 Amplitude modulation

As we have already shown how an AM signal can be modelled in section 2.3.1, we here show an example of the modulation process. The result from such a process is shown in figure 5.1. From the figure, we can clearly see that the baseband signal is the envelope of the carrier wave; that is why we call the process amplitude modulation, as it is the amplitude of the carrier that is modified.

The demodulation process of AM signals is basically to extract the envelope of the carrier wave. We can do this by using what we call an *envelope detector* [25], which is depicted in figure 5.2. On the positive half-cycle of the input signal $s_{AM}(t)$, the diode is forward-biased, and the capacitor charges up rapidly to the input signal. When the input signal falls below its peak value, the capacitor discharges slowly through the resistor R_l until the next positive half-cycle. Then the process is repeated. The result from this, is the envelope of the carrier, with ripples. These ripples are then removed by a low-pass filter, and we have then recovered the baseband signal.

5.3 Frequency modulation

In figure 5.3, an example of the frequency modulation process is shown. Notice that in this example we can clearly see that the frequency of the signal varies. This also means that the differences between the zero-crossings are not constant, which is something that characterizes the FM signals.

The demodulation process for FM (and PM) signals is more complicated than for

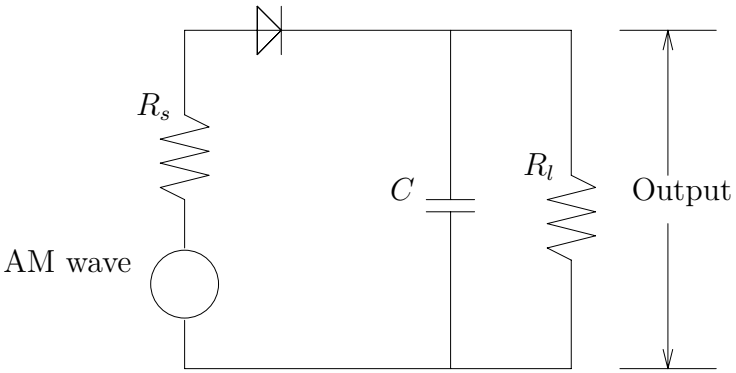

Figure 5.2: Analog envelope detector used to extract the baseband signal from AM signals [25].

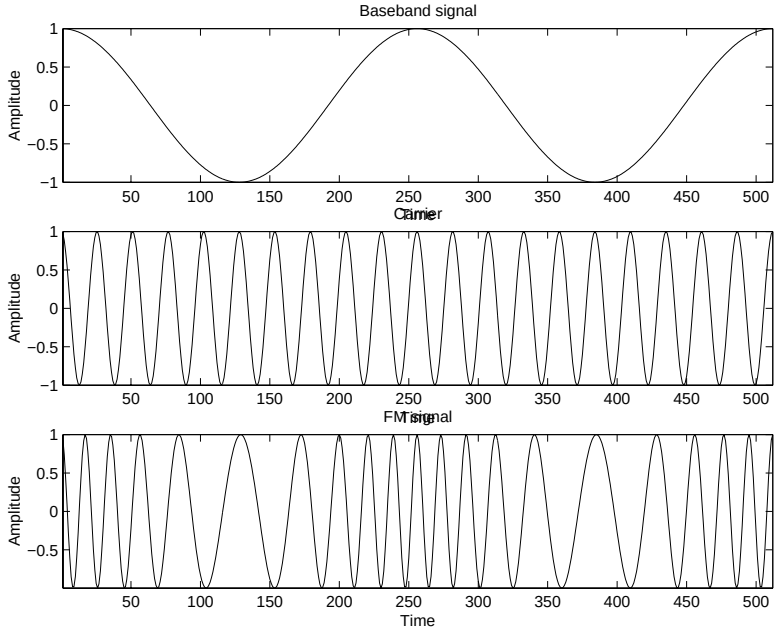

Figure 5.3: Frequency modulation process. Baseband signal (a) is modulated with carrier (b), and we get the final modulated signal in (c).

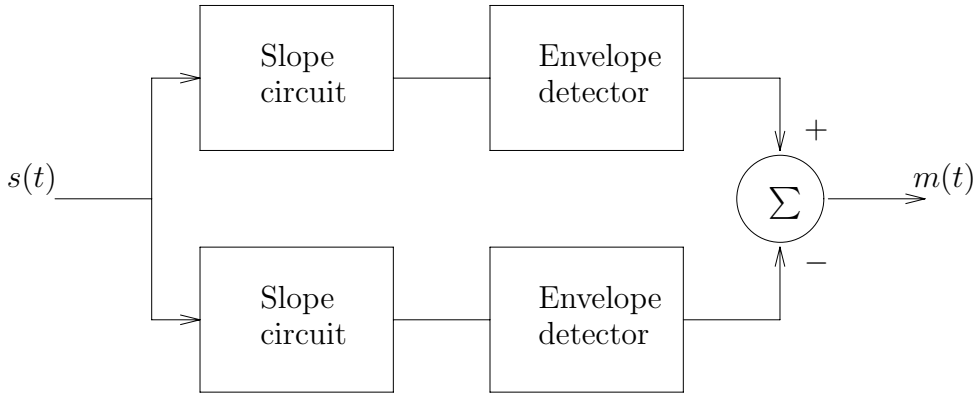

Figure 5.4: Slope circuit + envelope detector used for analog FM demodulation [25].

the AM signals. The demodulation can be done using several methods, for instance *Slope circuit + envelope detector* or *Phase-locked loop* [25]. Both these methods are more complex, and we will only summarize the first one here.

In figure 5.4, a *slope circuit + envelope detector* is shown. The slope circuit is essentially a band pass filter, which effectively converts the FM signal into an AM-FM signal, where the AM part of the signal is the modulating signal in the original FM signal. The envelope detector is then used to extract the amplitude, and we have recovered the signal, but it is biased by the FM part. Computations show, however, that it is possible to create a mirrored system which computes a signal with a bias which when combined with the first signal creates an unbiased signal. We have thus recovered the baseband signal without any bias.

5.4 Discrete implementation details

In the sections below, explicit formulas for most methods are given, or references to articles where this is more appropriate. In the actual implementations, denominators may evaluate to zero, we may get values causing the arcus cosine and the square root function to return complex values, etc. This may be due to noise, or estimation on a signal breaking our assumption of an AM/FM signal. These values are clearly not desirable, as they are not valid frequencies. To remedy this, the implementation approximates "invalid" values by substituting them by neighbouring values, so all estimators always return real numbers.

One of our assumptions for the signal is that it should be "slowly varying", meaning that the values should not change much between each sample. It therefore makes sense to substitute invalid values using neighbouring samples. Of course, a more sophisticated algorithm may be used, such as interpolation between the neighbouring samples.

5.5 Noise

To simplify the estimates below, we assume that we have no noise in the input signal. The aim of the sections below is to show that it is possible to derive explicit exact or approximate formulas/algorithms for the amplitude and frequency of signals having AM/FM structure.

In general, noise has a severe impact on the estimates given below, mainly due to the small number of samples used to estimate each output value.

5.6 Estimation using the Teager Energy Operator

As we have now described the important modulation processes, we shall finally describe a method of how to estimate the parameters of AM and/or FM signals using the Teager Energy Operator. In chapter 6 we will investigate the performance of the operators in noise or interference from other signals.

We first assume that we have a single sinusoidal signal, $x(t) = A\cos(\omega t)$. The output from the operator is

$$\begin{aligned}\Psi(x(t)) &= \Psi(A\cos(\omega t)) \\ &= -A^2\omega^2\sin^2(\omega t) - A\omega\cos(\omega t)[-A\omega\cos(\omega t)] \\ &= A^2\omega^2\end{aligned} \tag{5.1}$$

From this, we see that given a monochromatic signal as input, we get a constant output from the operator, which is proportional to both amplitude squared and frequency squared.

Now, we can differentiate the input signal, and compute the Teager Energy for that signal:

$$\begin{aligned}\Psi(\dot{x}(t)) &= \Psi(-A\omega\sin(\omega t)) \\ &= A^2\omega^4\cos^2(\omega t) - [-A\omega\sin(\omega t)][A\omega^3\sin(\omega t)] \\ &= A^2\omega^4\end{aligned} \tag{5.2}$$

We can combine the two results above, and calculate the frequency and amplitude of $x(t)$:

$$\hat{\omega}^2(t) = \frac{\Psi(\dot{x}(t))}{\Psi(x(t))} \tag{5.3}$$

$$|\hat{A}^2(t)| = \frac{\Psi(x(t))^2}{\Psi(\dot{x}(t))} \tag{5.4}$$

These equations are exact for a sinusoidal signal $A\cos(\omega t)$, but in the following section we notice that the approximation errors for AM, FM, and AM-FM signals are small. This means that the above method is suitable for estimation of the envelope of AM signals and the instantaneous frequency of FM signals.

5.6.1 AM signals

If we describe the AM signal simplified as $a(t)\cos(\omega_c t + \theta)$, where $a(t)$ then is the envelope of the signal, we can use relation 3.46, and compute the Teager Energy:

$$\begin{aligned}\Psi(a(t)\cos(\omega_c t + \theta)) &= a^2(t)\Psi(\cos(\omega_c t + \theta)) + \Psi(a(t))\cos^2(\omega_c t + \theta) \\ &= \underbrace{a^2(t)\omega_c^2}_{D(t)} + \underbrace{\cos^2(\omega_c t + \theta)\Psi(a(t))}_{E(t)}\end{aligned} \tag{5.5}$$

Thus, the Teager Energy of an AM signal is one term varying proportionally to the envelope, plus a time-varying term, which is also a function of the Teager Energy Operator of the envelope. If the baseband signal happens to be monochromatic, then both terms reduce to periodic functions of time.

Now, the term ω_c is a constant in the expression above. If $E(t) << D(t)$, then the output from the operator is a constant multiplied by the envelope of the AM signal. We can look at $E(t)$ as an estimation error, and if this error is sufficiently small, the output from the Teager Energy Operator tracks the envelope of the AM signal nicely.

Error analysis

In the estimate above, we have to prove that the estimation error, $\cos^2(\omega_c t + \theta)\Psi(a(t))$ is relatively small compared to what we use as the approximation, $a^2(t)\omega_c^2$. This proof is rather long and non-trivial, and is omitted here. All details are covered in the article by Maragos et. al [9].

5.6.2 FM signals

When the operator is applied to an FM signal, the output from the operator can be found to be

$$\begin{aligned}\Psi(s_{FM}(t)) &= \Psi(A\cos(\phi(t))) \\ &= A^2\dot{\phi}^2(t)\sin^2(\phi(t)) - A\cos(\phi(t))[-A\cos(\phi(t))\dot{\phi}^2(t) - A\sin(\phi(t))\ddot{\phi}(t)] \\ &= A^2\phi^2(t) + A^2\cos(\phi(t))\sin(\phi(t))\ddot{\phi}(t) \\ &= \underbrace{A^2\dot{\phi}^2(t)}_{D(t)} + \underbrace{A^2\ddot{\phi}(t)\frac{\sin(2\phi(t))}{2}}_{E(t)}\end{aligned} \tag{5.6}$$

If the term $E(t)$ is small enough, $E(t) << D(t)$, we can ignore it, and we then see that the operator tracks the instantaneous frequency squared, multiplied by a constant. If we set $A = 1$, the signal is normalized, which further simplifies the expression.

Error analysis

Just as for AM signals, the proof is rather long, and is omitted here. All details are covered in the article by Maragos et. al [9].

5.6.3 AM-FM signals

When AM and FM signals are combined, the output from the Teager Energy Operator contains terms similar to the terms found in both signals, as in equation 3.26. We notice that the Teager Energy for an AM-FM signal contains the term $a^2(t)\phi^2(t)$, and if we can show that the other terms are small compared to $a^2(t)\phi^2(t)$, then we can use the Teager Energy Operator to estimate amplitude and instantaneous frequency of AM-FM signals. The details of this are covered in [9].

5.6.4 Discrete versions

Several discrete versions of the Teager Energy Operator exist [15]. There are many ways to approximate the continuous derivatives, and depending of how this is done, various discrete versions are created.

Let us first sample the signal $x(t) = A\cos(\omega t)$ as $x[n] = A\cos(\Omega n)$ where $\Omega = \omega/T$ and T is the sampling period. We then approximate $\dot{x}$ by the two-sample forward or backward differences, and the output of the continuous Teager Energy Operator can be found to be

$$\Psi[x[n]] = A^2\cos^2(\Omega n) - A^2\cos(\Omega(n-1))\cos(\Omega(n+1)) \tag{5.7}$$

which we can simplify to

$$\begin{aligned}\Psi[x[n]] &= A^2\sin^2(\Omega)/T^2 \\ &= A^2\omega^2\left(\frac{\sin(\Omega)}{\Omega}\right)^2\end{aligned} \tag{5.8}$$

So, the output of the sampled signal is the same as the output of the continuous operator, except an extra factor attenuating the result. However, we are able to compensate for this known amount. The constant Ω carries information about T, and we can assume $T = 1$ and scale the result at the end of the calculation using the relation $f = \frac{\Omega}{2\pi T} = \frac{\Omega F_s}{2\pi}$, where F_s is the sampling frequency, and f thus measured in Hz.

We now turn to the derivatives, and we look at the two-sample backward difference first. We again insert the sampled version and get

$$\begin{aligned}y[n] &= x[n] - x[n-1] \\ &= A\cos(\Omega n) - A\cos(\Omega(n-1))\end{aligned} \tag{5.9}$$

Simplifying using the trigonometric identity

$$\cos(\alpha) - \cos(\beta) = 2\sin\left(\frac{\alpha+\beta}{2}\right)\sin\left(\frac{\beta-\alpha}{2}\right) \tag{5.10}$$

we get

$$y[n] = -2A\sin(\Omega/2)\sin(\Omega n - \Omega/2) \tag{5.11}$$

Using the Teager Energy Operator on this function yields

$$\Psi(y[n]) = 4A^2\sin^2(\Omega/2)\sin^2(\Omega) \tag{5.12}$$

This expression is similar to $\Psi(x[n])$, and we can derive an expression for Ω using both of them:

$$\frac{\Psi(y[n])}{2\Psi(x[n])} = 2\sin^2(\Omega/2) = 1 - \cos(\Omega) \tag{5.13}$$

Using the formulas above, we can derive formulas for the frequency and amplitude:

$$\hat{\Omega}[n] = \arccos\left(1 - \frac{\Psi[x_n - x_{n-1}]}{2\Psi[x_n]}\right) \tag{5.14}$$

$$|\hat{a}[n]| = \sqrt{\frac{\Psi[x_n]}{1 - \left(1 - \frac{\Psi[x_n - x_{n-1}]}{2\Psi[x_n]}\right)^2}} \tag{5.15}$$

5.6.5 DESA-1a

We can do the same analysis as above for the case where $x(t)$ is an AM-FM signal. All details of this analysis are described in [15]. The main result from this is that the expressions above approximates the instantaneous frequency and the amplitude for these signals.

The particular algorithm using the expressions above is called DESA-1a, where '1' implies a two sample difference, and the 'a' means that we have used an asymmetric derivative.

5.6.6 DESA-1

If we instead use the forward difference above, and average the results, we get what we call the DESA-1:

$$y[n] = x_n - x_{n-1} \tag{5.16}$$

$$\hat{\Omega}[n] = \arccos\left(1 - \frac{\Psi[y_n] - \Psi[y_{n+1}]}{4\Psi[x_n]}\right) \tag{5.17}$$

$$|\hat{a}[n]| = \sqrt{\frac{\Psi[x_n]}{1 - \left(1 - \frac{\Psi[y_n] - \Psi[y_{n+1}]}{4\Psi[x_n]}\right)^2}} \tag{5.18}$$

As we have averaged the derivatives, the algorithm is symmetric, hence we remove the 'a' in the name (but the algorithm still uses two-sample differences).

5.6.7 DESA-2

If we instead use the three-sample derivative, $y[n] = x[n+1] - x[n-1]$, and do the same analysis as above, we get the following expressions for amplitude and frequency:

$$\hat{\Omega}[n] = \frac{1}{2}\arccos\left(1 - \frac{\Psi[x_n] - \Psi[x_{n-1}]}{2\Psi[x_n]}\right) \tag{5.19}$$

$$|a[n]| = \frac{\Psi[x_n]}{\sqrt{\Psi[x_{n+1}] - \Psi[x_{n-1}]}} \tag{5.20}$$

This algorithm is called DESA-2, where '2' implies a three-sample symmetric difference.

5.6.8 Post-processing the Teager Energy Operator

The Teager Energy of an AM-FM signal was found to

$$\Psi(s_{AM-FM}(t)) = [a(t)\phi(t)]^2 + \underbrace{\frac{1}{2}a^2(t)\ddot{\phi}(t)\sin(2\phi(t))}_{E_H} + \underbrace{\cos^2(\phi(t))\Psi(a(t))}_{E_L} \tag{5.21}$$

We notice that the term E_H, one of the error terms, contains a factor of twice the frequency of $\phi(t)$. This means that E_H is a high frequency component, and we should thus be able to remove this using a lowpass filter, making the estimator more robust in noise. This has been suggested in [31], where a binomial filter of a certain length is applied to the output from the Teager Energy Operator before the instantaneous frequency and amplitude are estimated. A binomial filter of length N is the $(N-1)th$ fold convolution of the filter $\begin{bmatrix}1 & 1\end{bmatrix}$ with itself.

5.7 Estimation using the Hilbert transform

As described in chapter 2, the Analytic Signal procedure can be used to estimate the frequency and amplitude of a signal.

We used the Hilbert transform

$$H\{x(t)\} = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{x(s)}{t-s}ds \tag{5.22}$$

previously to define a complex signal $s(t)$, whose real component was the signal $x(t)$, and the imaginary component was the Hilbert transform of $x(t)$. We can then estimate the amplitude and frequency as follows:

$$\hat{a} = \sqrt{s_r^2 + s_j^2} = |s| \tag{5.23}$$

$$\hat{\omega} = \frac{\dot{s}_j s_r - s_j \dot{s}_r}{s_j^2 + s_r^2} = Im(\dot{s}/s) \tag{5.24}$$

where all variables are implicitly dependent on t.

This method, however, requires the complete signal to estimate correctly, as the complete signal $x(t)$ is required to compute the Hilbert transform.

In the discrete case, we can approximate this method by only transforming a short period of the signal instead (for instance using a windowing function), and hope that the included signal contains enough information to make a (good) estimate of amplitude and frequency.

The Hilbert transform can be approximated in several ways. For discrete time signals, we can estimate the transform using an IIR filter such as [31, 30]

$$h[n] = \begin{cases} \frac{\pi}{2}\frac{\sin^2(\pi n/2)}{n} & n \neq 0 \\ 0 & n = 0 \end{cases} \tag{5.25}$$

This filter is neither causal nor has it a finite length. However, we can create a FIR filter approximation by truncating the filter and shift it in time, using for instance a windowing method.

The Hilbert transform can also be approximated by using a *phase splitter* [30]. First, the Fourier transform of the input sequence is calculated. Then the values of the negative frequencies are zeroed, and an inverse Fourier transform is performed. The imaginary values of the output sequence from this transform is then the Hilbert transform of the input signal.

5.8 Estimation using Mandelstam's method

Mandelstam's method was discovered in 1934, and is widely used in nonlinear oscillation theory [19]. If ω_0 is a nominal frequency of an oscillation signal $u(t)$, we can define a phase plane $(u, -\dot{u}/\omega_0)$. We then define the amplitude and phase of the signal in this plane:

$$\begin{aligned} u &= a\cos\phi \\ &= \xi\cos(\omega_0 t) - \eta\sin(\omega_0 t) \end{aligned} \tag{5.26}$$

$$\begin{aligned} \frac{-\dot{u}}{\omega_0} &= a\sin\phi \\ &= \xi\sin(\omega_0 t) + \eta\cos(\omega_0 t) \end{aligned} \tag{5.27}$$

and the conjugate signal is defined as

$$H\{u(t)\} = -\frac{\ddot{u}(t)}{\omega_0} \tag{5.28}$$

5.8.1 Discrete version

The discrete version approximates the first order derivative by a two-point forward difference, and the conjugate signal is therefore

$$H\{u[n]\} = -\frac{(x_{n+1} - x_n)}{\omega_0} \tag{5.29}$$

Having this conjugated signal, we can use the same formulas for amplitude and frequency as for the Hilbert transform.

5.9 Estimation using Shekel's method

In Shekel's method, the analyzed signal is approximated using a simple sinusoid, $a\cos(\omega t + \phi)$. Using this model and its first and second derivatives, we get three equations:

$$u(t) = a(t)\cos(\omega t + \phi) \tag{5.30}$$

$$\dot{u}(t) = -a(t)\omega\sin(\omega t + \phi) \tag{5.31}$$

$$\ddot{u}(t) = -a(t)\omega^2\cos(\omega t + \phi) \tag{5.32}$$

$$\tag{5.33}$$

Shekel's method is exact for any continuous sinusoidal signal. We first solve for $a(t)$, and we square and add the two first equations:

$$a^2 \cos^2(\omega t + \phi) - a^2\omega^2 \sin^2(\omega t + \phi) \tag{5.34}$$

This is almost equal to $a^2(t)$, except for the factor ω^2. We can remove this by multiplying this factor by $\frac{u(t)}{\ddot{u}(t)} = \frac{a(t)\cos(\omega t+\phi)}{-a(t)\omega^2\cos(\omega t+\phi)}$, and get the final expression for the amplitude:

$$a(t) = \sqrt{u^2(t) - \frac{\dot{u}^2(t)u(t)}{\ddot{u}(t)}} \tag{5.35}$$

By inspection, we see that if we divide the $\ddot{u}(t)$ by $u(t)$, we get the frequency squared, therefore:

$$\omega(t) = \sqrt{-\frac{u(t)}{\ddot{u}(t)}} \tag{5.36}$$

5.9.1 Discrete version

A direct translation of this to create a discrete version is possible, but will not be exact. We approximate the first order derivative as the backward difference, and the second order as the usual three-point estimate.

To analyze the frequency error for discrete sinusoidal signals, we let $x[n] = A\cos(\Omega n)$, where $\Omega = \omega/T$ (T being the sampling period), and then insert this into the formula for frequency above:

$$\omega[n] = \sqrt{-\frac{A\cos(\Omega n)}{A\cos(\Omega(n-1)) - 2A\cos(\Omega n) + A\cos(\Omega(n+1))}} \tag{5.37}$$

Using the trigonometric identity

$$\cos(\alpha \pm \beta) = \cos(\alpha)\cos(\beta) \mp \sin(\alpha)\sin(\beta) \tag{5.38}$$

and simplifying, we get

$$\omega[n] = \sqrt{2 - 2\cos(\Omega)} \tag{5.39}$$

which clearly is not the exact result. The expression above deviates from Ω, and in the interval $[0, \pi/4]$, the largest error is at $\Omega = \pi/4$, where it is off by approximately 2.5%.

The same analysis can be done for the amplitude, and it can be shown that the amplitude is modulated by a signal having approximately twice the frequency of the sinusoidal signal.

As mentioned, Shekel's method is exact for any continuous sinusoid, although the operator contains a division, an expression of the form $A\cos(\omega t)$ will evaluate $\hat{A}$ to exactly A for the amplitude, and to ω for $\hat{\omega}$, because of cancellations. In the discrete version, however, we may get denominators of zero, etc, and the implementation must take care of these cases. In our implementation, the first order derivative is approximated using the two-point backward difference. The second order derivative is approximated as $x[n-1] - 2x[n] + x[n+1]$, as usual.

The discrete version of Shekel's method is given as

$$\hat{\Omega}[n] = \sqrt{-\frac{x_n}{x_{n-1} - 2x_n + x_{n+1}}} \tag{5.40}$$

$$|\hat{a}[n]| = \sqrt{x_n^2 + \frac{[x_n - x_{n-1}]^2 x_n}{x_{n-1} - 2x_n + x_{n+1}}} \tag{5.41}$$

This method is the simplest one to implement, but it has its limitations. For instance, if x_n is small or zero, then the frequency estimate will deviate severely. Simulations have confirmed this. A good solution to this is to use a median filter to post-process the estimates. This way, spikes are removed from the estimate, and simulations have proven this method to be efficient.

5.10 Estimation using the modified Covariance method

The paper "Instantaneous Frequency Estimation Using Linear Prediction with Comparisons to the DESAs" by L. B. Fertig ad J. H. McClellan [32] introduces linear predictors for instantaneous frequency estimation in connection with the Teager Operator. This application for linear predictors was already known in 1975 [33], but a more complicated signal environment was used there, which then required a different and computationally expensive approach.

The paper assumes that the real-valued discrete signal can be modelled as

$$x[n] = A\rho^n \cos(\Omega n + \phi) \tag{5.42}$$

The estimator is then calculated by finding the roots of the predictor filter, which is a quadratic polynomial, based on the input data samples.

The two coefficients of the predictor filter associated with the covariance method are given by

$$\begin{bmatrix} c_{1,1} & c_{1,2} \\ c_{2,1} & c_{2,2} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = - \begin{bmatrix} c_{1,0} \\ c_{2,0} \end{bmatrix} \tag{5.43}$$

The coefficients $c_{j,k}$ are estimated as

$$c_{j,k} = \frac{1}{2(N-2)} \left(\sum_{n=2}^{N-1} x_{n-j} x_{n-k} + \sum_{n=0}^{N-3} x_{n+j} x_{n+k} \right) \tag{5.44}$$

where N is the number of samples used in the estimate. The instantaneous frequency of the signal can be found from the roots of the quadratic predictor polynomial $1 + a_1 z^{-1} + a_2 z^{-2}$, and can then be estimated as

$$\hat{\Omega}_{Cv}[n] = \arccos\left(\frac{-a_1}{\sqrt{2a_2}} \right) \tag{5.45}$$

As N is the length of data samples used to estimate the frequency, we should expect that larger N produces better estimates. This method only estimates the instantaneous frequency, so to estimate the amplitude as well, some other method must be used, for instance one of the DESAs, as we shall see below.

Explicit versions have been calculated for $N = 4$ and $N = 5$. The four-point version can be found to be [32]

$$\hat{\Omega}[n] = \arccos\left(\frac{x_{n-2}x_{n-1} + 2x_{n-1}x_n + x_n x_{n+1}}{4(x_{n-1}^2 + x_n^2)}\right) \tag{5.46}$$

whereas the five-point version can be found to be

$$\hat{\Omega}[n] = \arccos\left(\frac{x_{n-2}x_{n-1} + 2x_{n-1}x_n + 2x_n x_{n+1} + x_{n+1}x_{n+2}}{2(x_{n-1}^2 + x_n^2 + x_{n+1}^2)}\right) \tag{5.47}$$

5.10.1 AM/FM signals

Both the modified covariance method and Prony's method are designed using a sinusoidal signal model as described above. However, simulations in [32] show that both methods handle AM-FM signals very well. We will do a similar test in chapter 9.

5.11 Estimation using Prony's method

Instantaneous frequency can also be estimated using Prony's method [34, 32]. For the four-point estimator, the coefficients for the predictor polynomial are found from

$$\begin{bmatrix} x_{n-1} & x_{n-2} \\ x_n & x_{n-1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = - \begin{bmatrix} x_n \\ x_{n+1} \end{bmatrix} \tag{5.48}$$

To find a_1 and a_2, we invert the matrix on the left hand side of the equation, and write this as

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = -\frac{1}{x_{n-1}^2 - x_n x_{n-2}} \begin{bmatrix} x_{n-1} & -x_{n-2} \\ -x_n & x_{n-1} \end{bmatrix} \begin{bmatrix} x_n \\ x_{n+1} \end{bmatrix} \tag{5.49}$$

Expanding the right hand side gives us

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \frac{1}{x_{n-1}^2 - x_n x_{n-2}} \begin{bmatrix} -x_{n-1}x_n & x_{n-2}x_{n+1} \\ x_n^2 & -x_{n-1}x_{n+1} \end{bmatrix} \tag{5.50}$$

We then define a new operator $\Upsilon[x_n] \equiv x_{n-1}x_n - x_{n-2}x_{n+1}$, and simplify the above equation to get

$$a_1 = \frac{-\Upsilon[x_n]}{\Psi[x_n]\Psi[x_{n-1}]} \tag{5.51}$$

$$a_2 = \frac{1}{\Psi[x_n]\Psi[x_{n-1}]} \tag{5.52}$$

To find the roots of the predictor polynomial, we insert a_1 and a_2 into $r = \frac{-a_1}{2\sqrt{a_2}}$:

$$\begin{aligned} r &= \frac{-\frac{-\Upsilon[x_n]}{\Psi[x_n]\Psi[x_{n-1}]}}{2\sqrt{\frac{1}{\Psi[x_n]\Psi[x_{n-1}]}}} \\ &= \frac{\Upsilon[x_n]}{2\sqrt{\Psi[x_n]\Psi[x_{n-1}]}} \end{aligned} \tag{5.53}$$

If we assume that the Teager Energy Operator does not vary too fast, we can approximate $\sqrt{\Psi[x_n]\Psi[x_{n-1}]}$, which is the "geometric mean", by the arithmetic mean $\frac{\Psi[x_n]+\Psi[x_{n-1}]}{2}$. The estimated frequency can then finally be written as

$$\hat{\Omega}_{Prony}[n] = \arccos\left(\frac{\Upsilon[x_n]}{\Psi[x_{n-1}] + \Psi[x_n]}\right) \tag{5.54}$$

This estimator uses a window of four samples to estimate the frequency.

The five-point version of Prony's method is hard to find directly, but we can approximate it by combining shifted versions of the four-point version:

$$\hat{\Omega}[n] = \arccos\left(\frac{\Upsilon[x_n] + \Upsilon[x_{n+1}]}{4\Psi[x_n]}\right) \tag{5.55}$$

Just as for the covariance method, only instantaneous frequency is estimated.

5.12 Estimating the amplitude using Prony-4/5 or LP-4/5

In the above sections, only the instantaneous frequency could be estimated using Prony's method or the modified covariance method. The Teager Energy Operator gives us an expression for the estimated amplitude as well, and we can use one of these expressions to estimate the amplitude when we already have estimated the frequency using some other method.

The basic expression for the energy of a sinusoidal signal $x[n]$ is as usual

$$\Psi[x[n]] = \Psi[A\cos(\Omega n)] = A^2\sin^2(\Omega) \tag{5.56}$$

which we can turn around, and express the amplitude as

$$|A| = \sqrt{\frac{\Psi(x[n])}{\sin^2(\Omega)}} \tag{5.57}$$

Assuming that we already have an estimate of the frequency, we can use that estimate, compute the Teager Energy of the signal, and then estimate the amplitude of the signal. If the estimate of the frequency using the Prony (or Modified covariance method) is good, we should expect that we should get a good estimate of the amplitude as well [32].

5.13 Spline ESA

In "An improved energy demodulation algorithm using splines", by Dimitriadis and Maragos [35], the signal is first smoothened using splines, and then the instantaneous energy is estimated using the *continuous* version of the Teager Energy Operator.

Splines are built from so-called B-splines, where B means *basic* or *basis*. A spline is a segmented, continuous function where each segment is built from a polynomial. The smoothness of the spline depends on the degree of the polynomials. Signal processing using splines is covered in [36], and we give an outline of the algorithm below.

Given data points $x[n]$, we can create a continuous function using the synthesis function

$$x_v(t) = \sum_{n=-\infty}^{\infty} c[n]\beta_v(t-n) \tag{5.58}$$

The sequence $c[n]$ are the spline coefficients and $\beta_v(t)$ is the v*th* order B-spline. We can calculate this as the v*th*-fold convolution of the zeroth-order B-spline. The zeroth-order B-spline $\beta_0(t)$ is defined as

$$\beta_0(t) = \begin{cases} 1 & \text{if } |t| < \frac{1}{2} \\ \frac{1}{2} & \text{if } |t| = \frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \tag{5.59}$$

The original sequence can be recreated by a convolution. Using the discrete B-spline $b_v[n] = \beta_v(n)$ (the B-spline sampled at n), we can express $x_v[n]$ as $(c * b_v)(n)$. We can transform this to the Z-domain and get $C(z) = \frac{X(z)}{B(z)}$. In this way, we can calculate the coefficients. An efficient algorithm to calculate these exists and is given in [36, 35]. In [36], an explicit formula to calculate the v*th*-fold convolution of β_0 is also given, which is needed for an efficient and flexible implementation.

The above equation reconstructs the signal exactly, such that $x_v(n) = x[n]$. As shown in [35], the robustness of the energy measure is *not* improved if the signal is reconstructed *exactly* (simulations show that the Prony ESA performs better). This means that if a noisy signal is analysed, the continuous version will also contain the noise, and the energy estimate will be affected.

A solution to this is to use smooth splines where we do not require the reconstruction to be exact. An error criterion is presented in the article, which modifies the system to be

$$C(z) = \frac{X(z)}{B_v(z) + \lambda(-z + 2 - z^{-1})^{\frac{v+1}{2}}} \tag{5.60}$$

where λ is a parameter to adjust the smoothness of the function. Notice that if λ is zero, the expression reduces to the exact interpolation.

5.13.1 Calculating the energy

Using the continuous expansion of the sequence $x[n]$, we can calculate its energy using the continuous Teager Energy Operator. The operator involves the signal and its first and second order derivative. Additionally, we need to compute the energy of the first order derivative, thus we also need the third order derivative of $x(t)$ to estimate frequency and amplitude.

Using one of the rules for B-spline manipulation found in [36],

$$\frac{\partial \beta_v(t)}{\partial t} = \beta_{v-1}(t - \frac{1}{2}) + \beta_{v-1}(t + \frac{1}{2}) \tag{5.61}$$

we can easily find the derivatives of the signal. These derivatives can then be used in the continuous ESA, and the expressions for amplitude and frequency can be sampled to yield the final estimate. This whole process is what Dimitriadis and Maragos refers to as the *Spline ESA* method.

One potential problem with this method, is that it actually requires a "complete" signal to compute the coefficients, just as the Hilbert transform method needs the complete sequence to calculate the Hilbert transform. This is because in the recursive algorithm described in [36, 35], boundary conditions are calculated at both the start and end of the coefficient sequence. This suggests that the algorithm may be better suited for non-realtime systems, or systems where the processing is frame-based (for instance speech processing).

5.14 Summary

In this chapter, we have looked at the problem of modulation and demodulation of signals. We have also looked at examples of AM and FM signal modulation. Furthermore, we have summarized several methods to demodulate signals and estimate their amplitude and instantaneous frequency.

Chapter 6

Noise and interference

As we have written earlier, noise sensitivity is the main problem for the Teager Energy Operator. The reason for this is that discrete differentiators are very sensitive to noise. To deal with the problem of noise, we seek to use statistics to analyse and remedy this situation.

Several articles have addressed the problem; the first article by Kaiser [3] briefly mentioned results for the discrete operator in zero-mean white Gaussian noise. Later, Alan C. Bovik et. al published "AM-FM Energy Detection and Separation in Noise Using Multiband Energy Operators" [16], where the statistics of $\Psi(\cdot)$ were developed. The results obtained below were later used in an application, described in "Teager Energy Based Feature Parameter for Speech Recognition in Car Noise" by Firas Jabloun et al [37].

The most recent achievement in making the estimation process more robust in noise, has been by using the Spline ESA method we have described, where one to some extent can adjust the process parameters to suppress noise.

6.1 The Teager Energy Operator in noise

Assume that we have the signal

$$x(t) = s(t) + w(t) \tag{6.1}$$

where $s(t)$ is the noise-free signal component, and $w(t)$ is zero-mean white Gaussian noise with variance σ^2.

We use the usual definition

$$\Psi(x) = \dot{x}^2 - x\ddot{x} \tag{6.2}$$

and would like to compute the expected value given by

$$E\{x\} = \int_{-\infty}^{\infty} x(\alpha) f_x(\alpha) d\alpha \tag{6.3}$$

We insert $x(t)$ into the expectation operator $E\{\cdot\}$, and get [37]

$$\begin{aligned} E\{\Psi(x)\} &= E\{\dot{x}^2 - x\ddot{x}\} \\ &= E\{\dot{x}^2\} - E\{x\ddot{x}\} \\ &= E\{[\dot{s} + \dot{w}]^2\} - E\{[s + n][\ddot{s} + \ddot{w}]\} \\ &= E\{\dot{s}^2 + 2\dot{s}\dot{w} + \dot{w}^2\} - E\{\dot{s}\ddot{s} + \dot{w}\ddot{s} + \dot{s}\ddot{w} + \dot{w}\ddot{w}\} \end{aligned} \tag{6.4}$$

As n is independent of s we can remove all cross terms in the equation above. This means that we can write the expected value as

$$E\{\Psi(x)\} = E\{\Psi(s)\} + E\{\Psi(w)\} \tag{6.5}$$

For the discrete Teager Energy Operator the expected value can be found to be [3]

$$E\{\Psi[x]\} = E\{\Psi[s]\} + \sigma^2 \tag{6.6}$$

This means that the Teager Energy estimate is biased by the variance of the noise, and this skews the estimates of both amplitude and frequency in presence of noise.

6.2 Suppressing noise

The paper "AM-FM Energy Detection and Separation in Noise Using Multiband Energy Operators" [16] has the most extensive treatment of *multiband demodulation*, where noisy signals are considered, and a multiband strategy for the analysis is presented to suppress the noise.

The article only covers signals with *one* information signal, and is mostly concerned with the filter selection to get the best estimate of the information signal. Although its extensive treatment of its topic, it does not exactly cover the *multiple* signal detection/tracking problem, but the other articles that address this are based on this one.

6.2.1 The filter bank

Figure 6.1 shows a schematic representation of the analysis process (and n would be 1 in the case we are searching for only one signal). The input signal s is fed into a bank of N bandpass filters. Then an energy measure is calculated for each band, and on the basis of a selector function Ψ^*, one or more bands are selected as carrying an information signal. The instantaneous frequencies and amplitudes are then estimated.

6.2.2 Selector function

The energy measure function is usually the Teager Energy Operator, but could in principle be any kind of measure. As we are looking for AM/FM signals, we know that the Teager Energy Operator is a suitable operator. The selector function depends on the design goal, and several good functions are available. For instance, one could choose

$$\Psi^*(t) = \max_{1 \le i \le N} \frac{\Psi(f_i)}{|G_i(\omega_i)|^2} \tag{6.7}$$

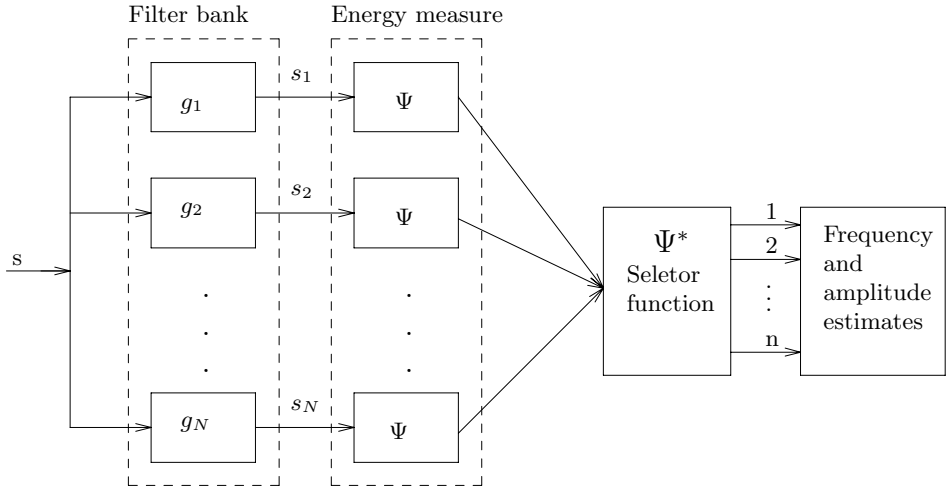

Figure 6.1: Schematic overview of the multiband analysis process.

as this function uses the maximum value of the frequency-amplitude product squared to select a band. Another option could be to only consider amplitude, and thus normalize by frequency

$$\Psi^*(t) = \max_{1 \leq i \leq N} \frac{\Psi(f_i)}{(\omega_i |G_i(\omega_i)|)^2} \tag{6.8}$$

As the Teager Energy Operator tracks the amplitude and frequency product squared, the function should thus be able to approximately track the amplitude.

To capture the signal with the highest instantaneous frequency, the selector function could, for instance, be a version of the ESA, for instance DESA-1, and then select the band containing the highest frequency.

Using one of the selector functions above, it can be shown that this function will select the correct filter. The main point here is that the output from Teager Energy Operator for a given filter will be essentially zero if the signal is not in the passband of the filter.

6.2.3 The Gabor filter

We use a bandpass filter to extract each of the various frequency bands containing each signal. We can construct the bandpass filter using a Gabor filter, which for our purpose has some nice properties.

A Gabor filter is described in time and frequency as [7]

$$h(t) = \exp(-a^2 t^2) \cos(\omega_c t) \tag{6.9}$$

$$H(\omega) = \frac{\sqrt{\pi}}{2a} \left(\exp[-\tfrac{(\omega-\omega_x)^2}{4a^2}] + \exp[-\tfrac{(\omega+\omega_c)^2}{4a^2}] \right) \tag{6.10}$$

where ω_c is the center frequency of the band and a is the width of the filter in Hertz.

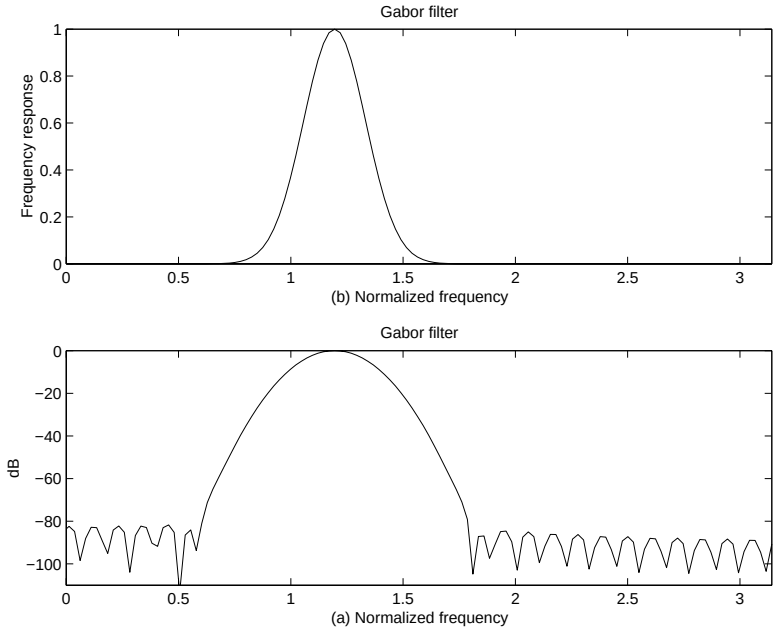

Figure 6.2: Gabor bandpass filter example.

We can create a discrete version of the filter by sampling the continuous version. We let F_s be the sampling frequency, and $T = 1/F_s$. Then $h[n]$ is

$$h[n] = \exp(-a^2(nT)^2)\cos(\Omega_c nT) \tag{6.11}$$

where Ω_c is the digital frequency. The length of the filter, N, must be chosen such that the factor $\exp(-(aT)^2n^2)$ is "small enough". If we choose N to be dependent on a and T such that $N = 2.5/(aT)$, as in [7], then we get that $\exp(-(2.5^2)) \approx 0.002$.

Figure 6.2 shows an example of a Gabor filter. In the frequency domain, the filter is shaped as a Gaussian bell curve, as depicted in subfigure (a). The filter is shown using a logarithmic scale on the y axis in subfigure (b).

6.2.4 Placement and bandwidth

In [16], it is stressed that the location and bandwidth of the filters should be logarithmically spaced and have accordingly different bandwidths in the frequency domain. This is to ensure a uniform (worst-case) predicted performance across the filters.

Given the highest frequency in the range one want to analyze, Ω_c, the center frequencies for the $N-1$ subband filters are given by

$$\omega_m = 3 \times 2^{-(m+1)}\Omega_c, \text{ for } m = 1, \cdots, N-1 \tag{6.12}$$

The baseband filter is placed at $\omega_N = 0$. Notice that the indexing of the filters are from higher to lower center frequencies.

For the Gabor filters, the bandwidth can be chosen such that two neighbouring filters intersect at half-peak. To achieve this, the subband filter bandwidth should be

chosen such that

$$\alpha_m = \frac{\omega_m}{2\sqrt{\ln 2}}, \text{ for } m = 1, \cdots, N-1 \tag{6.13}$$

If the baseband filter also should intersect at half-peak, we must choose its bandwidth to be $\omega_N = 2\omega_{N-1}$.

As our final aim is to track several signal components, wide band filters may capture several components, and we may then get interference and poor estimates as a result. The solution to this problem is to use a large filter bank with uniformly spaced filters with equal bandwidths. So, whether we should use a logarithmically spaced or uniformly spaced filterbank depends on the application.

6.2.5 Example – a simple chirp in noise

Figure 6.3 and 6.4 shows the analysis of a chirp sampled at 16000Hz sweeping the frequency range from 500 to 3500Hz. The signal-to-noise ratio in this example is 15dB.

In the first figure, 6.3, subfigure (a) shows the actual signal as a function of time. A conventional spectrogram of the signal is shown i subfigure (b). The filter bank used in the multiband analysis is shown in subfigure (c), and the selector function output band is shown in subfigure (d).

In the second figure, 6.4, the estimated values for the instantaneous frequency are shown. The first subfigure (a) shows the raw estimated frequency as calculated by our DESA-1 implementation. As shown in the figure, the frequency estimate is very poor, but a linear trend can be spotted. For all practical purposes, this estimate is useless. The subfigure to the right (b) is a filtered version of the first estimate, where a median filter of length 7 is used to smoothen the original estimate. We see that this significantly improves the estimate, although there are still large deviances in the lower frequencies of the signal.

Subfigure (c) shows the estimate when proper bandpass filters have been employed to suppress the noise using the multiband strategy. The bandpass filter used at each time instant is shown in figure 6.3 (d). In this estimate, we see that the deviances are much smaller, so this suggests that the multiband method performs well. Subfigure (d) shows a filtered version of (c), where again a median filter of length 7 is used. We see that the estimate is improved further by the smoothening filter.

The example above suggests that filtering improves the estimate significantly. Figure 6.5 shows the relationship between the RMS error and the signal-to-noise ratio for the chirp signal. This example, however, does not say anything about the multiband strategy in general, but as it sweeps most of the relevant frequency spectrum, it is a good indication of how other signals will perform.

The four graphs show the RMS error for the four corresponding estimates in figure 6.4. As expected, the raw estimate has a high RMS error for a small SNR. We also notice that both (a) and (b) are better than the multiband versions for very high SNR. We have experienced small deviances in the transitions between the different filters (when the selector function decides to switch to another band), and this contributes to error in (c) and (d), which makes (a) and (b) better in these cases.

We also notice that (b) is better than (a) in all cases for the chirp signal. This suggests that it is a good choice to add a short median filter to post-process the estimate.

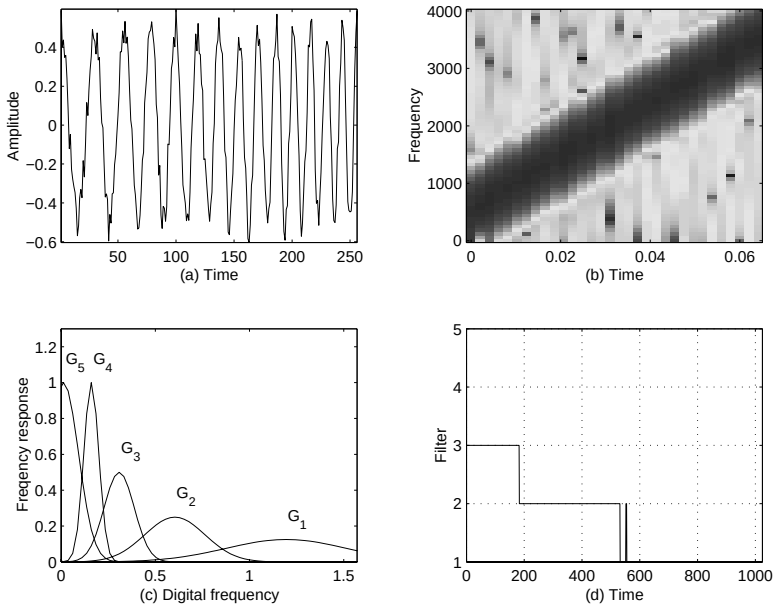

Figure 6.3: Multiband analysis applied to chirp signal. (a) Time representation of chirp in SNR = 15dB. (b) Spectrogram of chirp. (c) Filter bank of bandpass filters. (d) Filter selection as done by selector function.

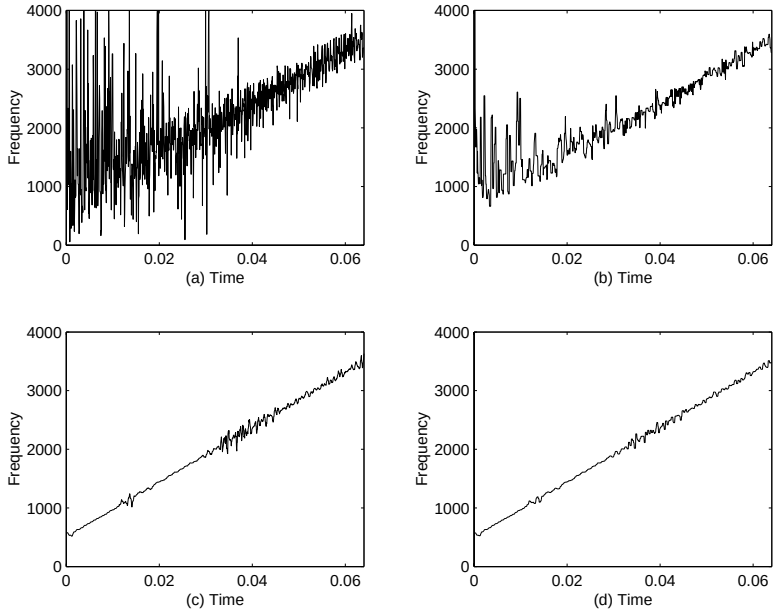

Figure 6.4: Estimated instantaneous frequency. (a) Raw estimate using DESA-1. (b) Median filtered version of (a). (c) Result from multiband analysis using DESA-1. (d) Median filtered version of (c).

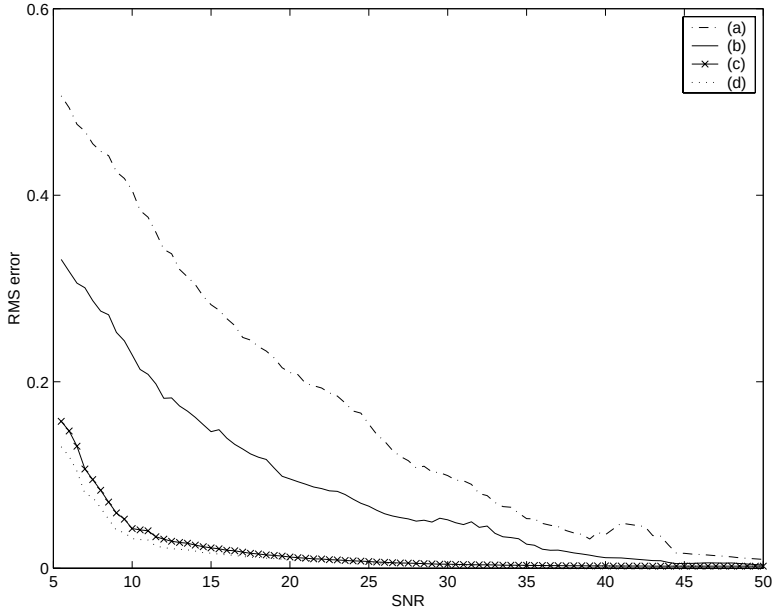

Figure 6.5: RMS error as a function of SNR for the estimates in figure 6.4.

The median filter adds latency and complexity to the overall system, and does affect time resolution, but an RMS error some 30 percent lower might be worth it.

The overall best estimation of the four is done using the multiband strategy and then post-processing the frequency estimate using a median filter. However, the difference between the multiband demodulation with and without filtering is minimal, so the post-filtering is not the main factor in the overall improvement.

To conclude, the multiband solution should be used in environments where noise is expected, and here this method provides an RMS error 60 percent lower than the raw estimate. If the SNR is known to be high, then a direct estimate of the frequency without any filtering will perform best.

6.3 On the effect of bandpass filtering

If several information signals are located within a narrow frequency band in the frequency domain, the energy estimate of the signal may not be what we expect it to be, as the Teager Energy of the sum of to signal is *not* equal to the sum of the Teager Energies of the two individual signals. The solution to this, is to use bandpass filters to remove one of the components and calculate the energy of the remaining signal.

Using the bandpass filtering technique above, this implies that the bandwidth of our filters must be small enough, so we do not get cross-signal interference. In this section, we explore the effect of this cross-signal interference, and see whether we can find some rule for choosing the bandwidth of our filters.

In [16], the constant-Q property is used to select the filter bandwidths, but other papers seem to use experimental results to choose the bandwidths of the filters. This

section therefore aims to give an idea of how much the estimate is affected by the filter bandwidth.

We will use synthetic signals, created by adding two simple AM signals (and no noise) in this experiment. Other signals (FM or AM-FM) should also have been investigated, but it is plausible that they will yield approximately the same results, as they are both bandlimited and we know that the behaviour of the Teager Energy Operator on these signals is approximately the same as for the case of AM signals.

We first look at the Teager Energy Operator for a sum of two signals:

$$\Psi(f+g) = \Psi(f) + \Psi(g) + \underbrace{\Psi_c(f,g) + \Psi_c(g,f)}_{\text{cross terms}} \tag{6.14}$$

which is the sum of the Teager Energy of f and g and several cross-terms expressed using the $\Psi_c(\cdot,\cdot)$ operator. We now define our two information signals, and use the above equation to calculate the energy of the signals s_1 and s_2 defined as

$$s_i(t) = a_i(t)\cos(\omega_{c_i} t) \tag{6.15}$$

where $i = 1, 2$, $a_i(t)$ is the envelope function and ω_{c_i} is the carrier frequency. The baseband signals in our test are simple sinusoidal signals, such that $a_i(t) = 1+\kappa_i \cos(\omega_i t)$, where κ_i is the modulation factor and ω_i is the frequency of the baseband.

Then the energies of these signals are

$$\begin{aligned}\Psi(s_i(t)) &= a_i^2(t)\Psi(\cos(\omega_{c_i}t)) + \cos(\omega_{c_i}t)^2\Psi(a_i(t)) \\ &= a_i^2(t)\omega {c_i}^2 + \Psi(a_i(t))\cos(\omega_{c_i}t) \\ &= (\kappa^2 + \kappa\cos(\omega_i t))\cos(\omega_{c_i}t)^2\omega_i^2 + (1 + 2\kappa\cos(\omega_i t) + \kappa^2\cos(\omega_i t)^2)\omega_{c_i}^2\end{aligned} \tag{6.16}$$

From the above expression, we see that we have one constant term, $\omega_{c_i}^2$. The other terms are oscillations with frequencies ω_{c_i} or ω_i.

The energy of the sum $s_1 + s_2$ is a complicated expression including $\Psi(s_1(t)) + \Psi(s_2(t))$ as above, and the cross terms between s_1 and s_2, which we seek to eliminate by filtering the signal first.

6.3.1 The filter

The bandpass filters, $H(\omega)$, are different versions of the Gabor filter described above. The Gabor filter is dependent on two parameters; center frequency and bandwidth. We'll see how these parameters affect the energy estimate in our examples.

6.3.2 Some clarifying examples

In these examples, we choose the sampling rate to be 2000Hz, and choose two AM signals, s_1 and s_2, with carrier frequencies $f_{c_1} = 150Hz$ and $f_{c_2} = 450Hz$. The bandwidths of s_1 and s_2 are chosen to be 20 and 40 Hz, respectively.

Example showing cross terms of Teager Energy Operator

In figure 6.6, the two AM signals are shown first in subfigures (a) and (c). Subfigures (b) and (d) show the Teager energies of these signals, and (e) shows the sum of the two signals, with the computed Teager Energy of it in figure (f).

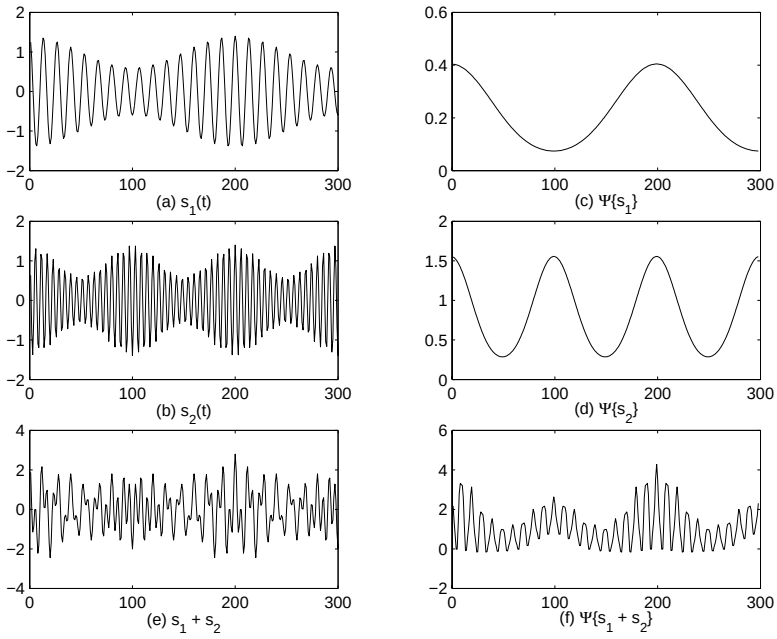

Figure 6.6: Two separate AM signals and their sum with the respective Teager Energies.

As expected, the Teager Energy of the sum is not the sum of the two Teager Energies in figure (c) and (d). Some features of the two signals appear in figure (f); the energy has components of the carrier frequency of s_2, and the periodicities of both the baseband signals can be seen.

Example showing effects of bandpass filtering

Figure 6.7 shows the estimate done by employing a suitable bandpass filter of 200 Hz (this bandwidth is chosen somewhat arbitrarily, and is not optimal, but it suffices in this example). In this example, we seek to remove signal s_1, and the bandwidth of the bandpass filter is chosen properly, such that it actually removes most of the other signal.

Subfigure (a) shows the filtered signal, (b) shows the estimated energy of s_2. The differences between the filtered signal and s_2 are shown in (c). The large deviances on both ends are transients due to the filtering process. In this case, the difference is rather small (order of a magnitude less), but we can also see the interference from signal s_1.

Subfigure (d) shows the estimated Teager Energy of the filtered version of the input signal $s_1 + s_2$, and (e) and (f) contains the frequency plots of $s_1 + s_2$ and $H\{s_1 + s_2\}$, with the filter H overlaid.

6.3.3 Filter bandwidth

In this section we investigate the energy estimation error when we vary the bandwidth of the filter. In this example, we use the same two signals as in the first examples. We use the RMSE as the error measure for the difference between the two estimates.

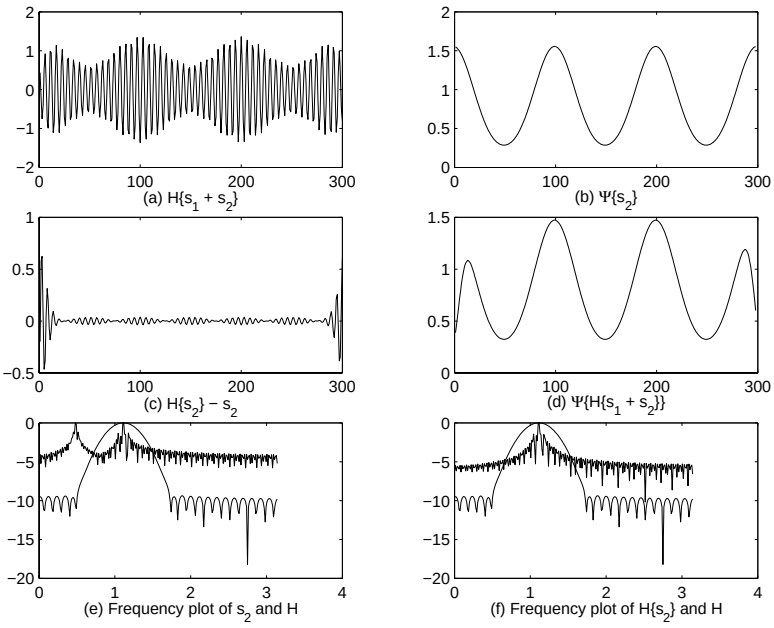

Figure 6.7: Estimation of energy using a suitable bandpass filter to remove one of the signals.

Example showing effect of too wide bandpass filter

Figure 6.8 shows the same information as figure 6.7, but here the filter bandwidth is much higher; 1000 Hz. In subfigure (d) we can clearly see that the estimated Teager Energy deviates much from the expected Energy signal, mainly due to a high frequency component, which is due to the interference from signal s_1.

Figure 6.9 shows a plot of filter bandwidth versus calculated RMS error. The solid line describes the error of the estimate, and as expected, the curve has a minimum. Large values below the minimum are most likely because the filter bandwidth is too low to capture the AM signal completely, or at least it is distorted severely by the low bandwidth.

The larger values above the minimum are due to the interference with the other AM signal, as it is no longer actually removed by the filter, because the bandwidth is too large.

Example of post-filtering of the Teager Energy output

Experiments have shown that the energy sometimes contains high-frequency components, and these are effectively removed by a lowpass filter [31]. We show an example of this, where a too wide bandpass filter has been chosen.

In figure 6.10, subfigure (a) shows the filtered version of the composite signal. Again, the filter bandwidth is 1000Hz, so some of the signal s_1 is included in the estimate of $\Psi(s_2)$. Subfigure (b) shows the computed Teager Energy and the actual Teager Energy from figure 6.6 (d). The difference between the filtered and the original signal is shown

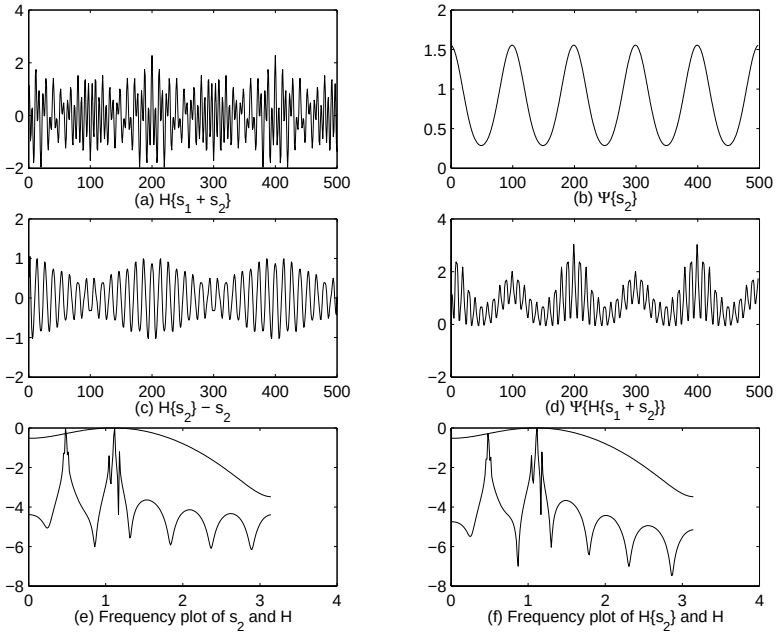

Figure 6.8: Energy estimate where the filter bandwidth is too large.

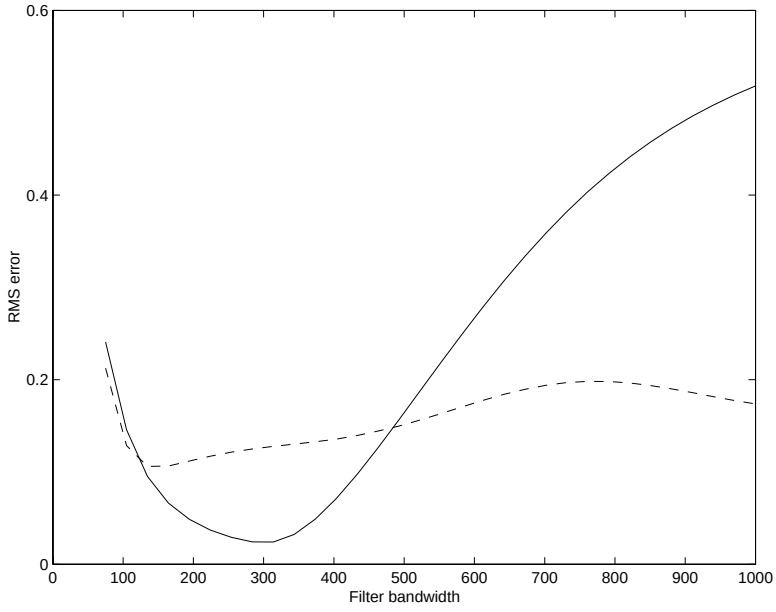

Figure 6.9: Estimated RMS error for different filter bandwidths.

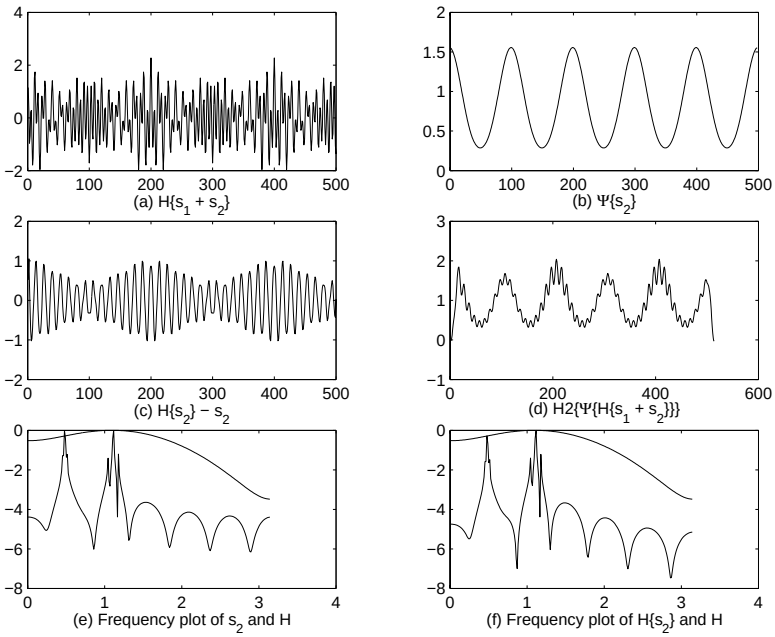

Figure 6.10: Energy estimate where the energy is post-filtered using a lowpass filter.

in subfigure (c). In (d), a lowpass filtered version of the Teager Energy is shown, and in subfigure (e), the frequency plot of the signals are shown, with the filter response overlaid. Subfigure (f) shows a frequency plot of (a) with the filter overlaid.

The dashed line in figure 6.9 is the computed RMS values of the filtered estimate. We see that the estimate is improved by the lowpass filter in the case that the bandwidth is larger than the optimum filter bandwidth in this case.

6.3.4 Filter bandwidth versus frequency difference

In this example, we use the same signal parameters for s_1 as in the first example, but we will vary the bandwidth of the filter and the carrier frequency of the second signal, s_2. We will then compute the Teager Energy of the filtered version and compare this to the expected estimate.

Figure 6.11 shows a surface where the value is the calculated RMS-value for a given filter bandwidth and the distance between the two AM carrier frequencies. In this figure, two areas contain relatively large error values:

- At low bandwidth *and* small carrier distance. In this area, the filter is probably unable to actually remove the other signal, as they are too close. The filter bandwidth does not actually matter much here; if it is too large, both signals are kept, if it is too small, none of the signals are kept.

- At high bandwidth *and* small carrier distance. Due to the large filter bandwidth, the filter does not remove the unwanted signal because they are too close, thus the cross terms are included in the energy estimate.

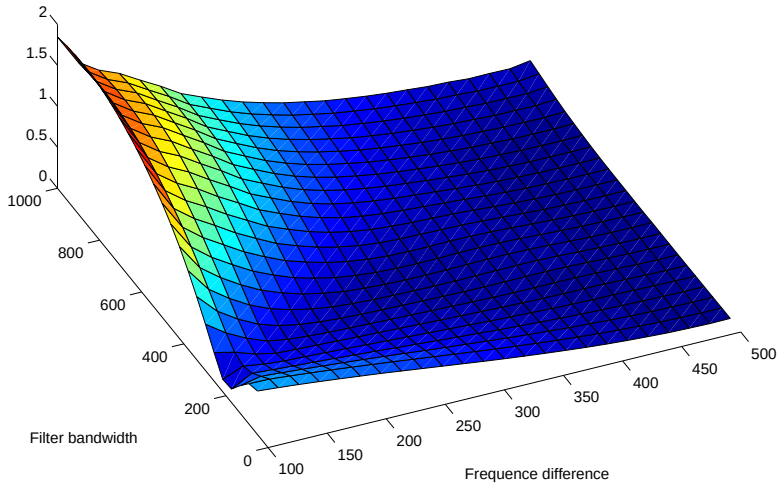

Figure 6.11: Filter bandwidth versus frequency difference.

We also notice that for large bandwidths *and* large carrier differences, the error is small. This is intuitive, since the carrier frequencies are so far from each other, the large bandwidth does not capture the unwanted signals, and the cross terms will not be included in the energy estimate.

6.4 Summary

In this chapter we have looked at the process of demodulation of speech signals, which as of today is the most common application of the Teager Energy Operator. During this chapter, we have

- Presented some statistics with regard to the Teager Energy Operator.
- Looked at a multiband method for the analysis of noisy signals.
- Looked at the problem of interference of other signals during energy estimation due to the cross terms in the Teager Operator for the sum of two (or more) signals.

Chapter 7

Discrete differentiators and their performance in noise

The major disadvantage of the Teager Energy Operator is its sensitivity to noise. Several measures have been taken to refine the operator to make it more robust, for instance [35]. The sensitivity is intuitively mainly due to the differentiators working over a very short time window.

In this chapter, we look at the design of differentiators, and their performance in noise. Ultimately, the knowledge from this chapter might be used to create new Teager Energy Operator approximations that are more robust in presence of noise.

7.1 First order linear differentiators

Let $x(t)$ be our signal and let its Fourier transform be $X(j\omega)$. By using integration by parts it can be shown that the derivative of $X(j\omega)$ is proportional to ω:

$$\dot{X}(j\omega) = j\omega X(j\omega) \tag{7.1}$$

From this, we see that the amplitude of the Fourier transform of $\dot{x}(t)$, is varying linearly as a function of ω. The factor j does not affect the amplitude of the frequency response.

We now want to create a FIR filter implementation of the differentiator operator. The ideal impulse response of such a (bandlimited) system can be found to be

$$h[n] = \frac{\cos(\pi(n - M/2))}{n - M/2} - \frac{\sin(\pi(n - M/2))}{\pi(n - M/2)^2}, \qquad \text{for } -\infty < n < \infty \tag{7.2}$$

and a truncated version is shown in figure 7.1.

There are several ways to approximate this filter and we will look into how this can be done in the following sections. But before we do that, we have to define what we mean by a "good" approximation, and thus define what makes one filter better than another.

There are two standard methods to measure the filter approximation; minimax bound and least-squared error. We can formalize these, and say that the minimax error

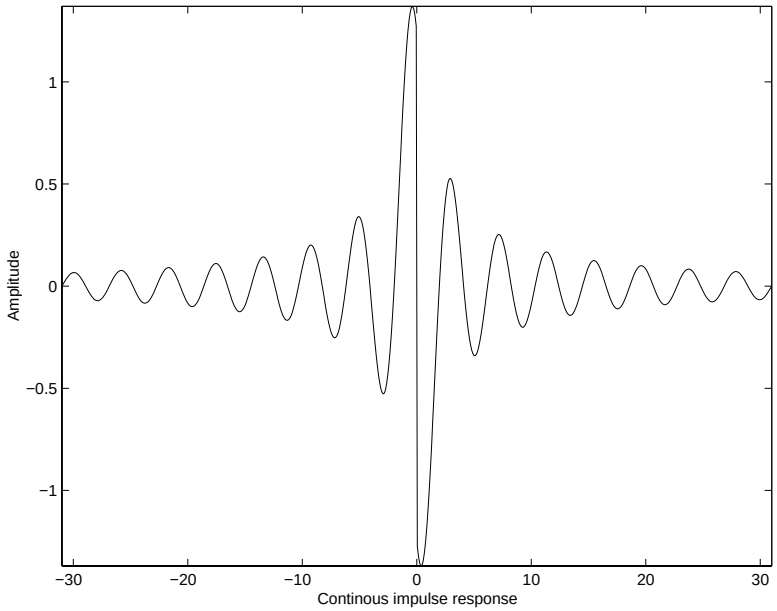

Figure 7.1: Truncated impulse response for differentiator (approximation of first order derivative).

bound for a filter is the largest absolute deviation from the ideal (wanted) frequency response:

$$\epsilon_{minimax} \triangleq max(|H(j\omega) - H_{ideal}(j\omega)|), \qquad \pi < \omega < \pi \tag{7.3}$$

The least-squared error can be written as

$$\epsilon_{ls}^2 \triangleq \frac{1}{2\pi} \int_{-\pi}^{\pi} |H(j\omega) - H_{ideal}(j\omega)|^2 d\omega \tag{7.4}$$

The filter design problem thus becomes an optimization problem; the best filter is the filter where the coefficients are chosen in such a way that the error is minimized. Notice that when these constraints are given, we do not need to know the ideal impulse response, as suggested above, as we are just trying to find coefficients that best approximates the frequency response.

The problem with the both methods above, is that there is no way to specify error bounds for specific frequency bands in the filter. To achieve this, it is possible to extend the definitions, for instance by introducing weights for the different bands.

7.1.1 Generalized linear phase filters

In the following sections and examples, we will use the term *generalized linear phase filters*. This term is defined in [30], and is used to describe a special class of filters. A system is said to be a *generalized linear phase system* if the Fourier transform $H(j\omega)$ can be written as

$$H(j\omega) = A(j\omega)e^{-j\alpha\omega+\beta} \tag{7.5}$$

where α and β are constants, and $A(j\omega)$ is a real function of ω.

Causal systems with generalized linear phase are divided into four classes, and we call these type I, II, III and IV. These types are characterized by the whether the filter lengths are odd or even, and whether the impulse responses are symmetric or antisymmetric.

Symmetric impulse responses are responses $h[n]$ such that

$$h[n] = \begin{cases} h[M-n] & \text{for } 0 \leq n \leq M \\ 0 & \text{otherwise} \end{cases} \tag{7.6}$$

whereas antisymmetric impulse responses are responses such that $h[n]$ is

$$h[n] = \begin{cases} -h[M-n] & \text{for } 0 \leq n \leq M \\ 0 & \text{otherwise} \end{cases} \tag{7.7}$$

Type I

A Type I filter has a symmetric impulse response, and M is even. This means that the filter delay $M/2$ is an integer.

Type II

A Type II filter has a symmetric impulse response, and M is odd, which results in a delay which is not an integer. These systems always have a zero at $z = -1$.

Type III

A type III filter has an antisymmetric impulse response, and M is even. These systems always have a zero at $z = -1$.

Type IV

A type IV filter has an antisymmetric impulse response, and M is odd. From figure 7.1, we see that the impulse response is antisymmetric. As we know that the frequency response of a differentiator is proportional to ω, a good approximation should *not* contain a zero a $z = -1$, if we want the differentiator to give good estimates in the interval $\omega \in [0, \pi]$. This suggests that the best first order differentiators are type IV filters.

7.1.2 The windowing method

The impulse response for a differentiator is infinite, and we somehow have to truncate it to make it a causal system with finite impulse response. The truncation can be done by multiplying the impulse response by a symmetric window of length $M + 1$. We can then easily show that $h[n] = -h[M - n]$. This means that the system is either a type III or type IV generalized linear phase filter.

The truncation causes ripples in the frequency discontinuities (Gibb's phenomenon), and the windowing method is used to spread this truncation error over time. To do this, we multiply the truncated impulse response by a symmetric sequence, which we call a window. It can be shown that this window is the same as a cyclic convolution in

the frequency domain. This means that the resulting frequency response is smeared by the frequency response of the window, thus smoothing discontinuities in the frequency response for the truncated impulse response.

Several well known windows exist; Rectangular, Hamming, Hanning, Triangle, Bartlett, Kaiser. All of these have their pros and cons, and we will see examples of the rectangular one and the Kaiser window (which reduces to the rectangular for a certain parameter setting). We will use the Kaiser window below.

The Kaiser window was originally created to design simple filters with one discontinuity in frequency, such as low-pass filters. There exists simple formulas, which can be used to predict filter length and the filter parameter β, based on the simple transition band and error constraints.

In filter design, the result is always a trade-off between two or more conflicting requirements; usually the result is a trade-off between main lobe width and side lobe height. The parameter β is used to adjust this trade-off; this is what makes the Kaiser window useful as a window in the windowing method.

As an example, we choose $\beta = 2.4$ and $M = 10$, and calculate the impulse response. The value of β is selected "arbitrarily" for this example; too low values cause severe ripples in the response, and too high values make the responses deviate more and more from the ideal response, although the responses are smooth, as they should be. The value of β is therefore a compromise. The frequency response is shown in figure 7.2. As M is an even number, $H(z)$ will have zeroes both at $z = +1$, that is for $\omega = 0$, and $z = -1$, or $\omega = \pi$. The first zero is just as it should be, as the frequency response should be zero for $\omega = 0$. The second, however, is far off the ideal response, as shown in the figure.

If we choose M to be an odd number (we then get a type IV filter), we can avoid the zero at $z = -1$. An example where $\beta = 2.4$ and $M = 7$ is shown in figure 7.3. The price we have to pay for this, is that we get a sample delay of $M/2$, which is not an integer. If the filter is part of a filter network, this might not be a problem, as we might compensate for the half delay in other linear phase filters. The value of β does not cause the response to deviate as much as in the first example, and it is primarily the filter length that determines the response for the high frequency components.

Choosing M and β

In the example above, both M and β were chosen "arbitrarily", and we would like to see if we can choose the best values for both of these.

Figure 7.4 shows two plots of M versus the β parameter for a type III filter. The upper graph is the measured error (deviation) from the ideal response when the mean-squared error criterion was used. In lower graph, the minimax error bound was used. The figures suggests that error is highly dependent on the error criterion:

- If the mean-squared error criterion is used, the error is mostly dependent on the value of M. The error varies little for different values of β, but a small value of β is best.

- If the minimax error criterion is used, the error is mostly dependent on the choice of β, as the error varies very little with the choice of M.

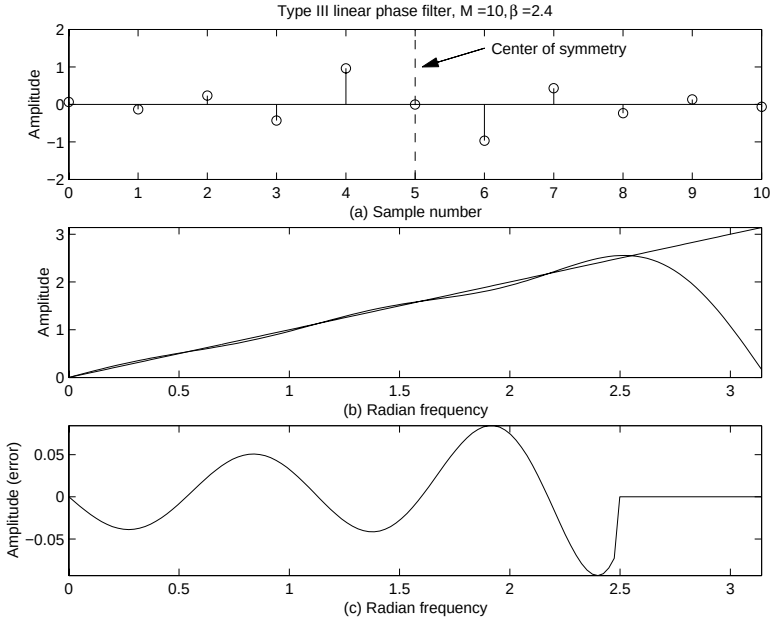

Figure 7.2: (a) Impulse response for differentiator where $\beta = 2.4$ and $M = 10$. (b) Frequency response. The dotted line shows the ideal response. (c) Approximation error (80% of response; last 20% deviates too much.)

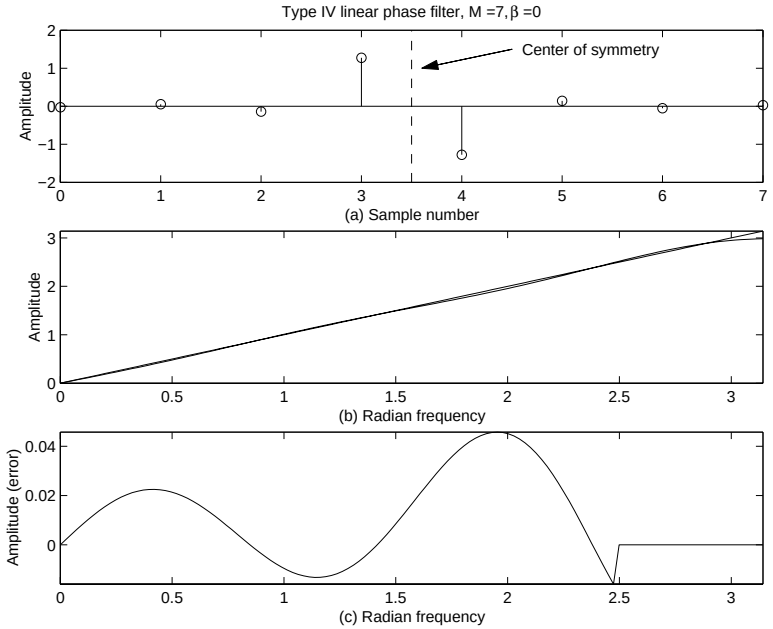

Figure 7.3: (a) Impulse response for differentiator where $\beta = 2.4$ and $M = 7$. (b) Frequency response. The dotted line shows the ideal response. (c) Approximation error (80% of response; last 20% deviates too much.)

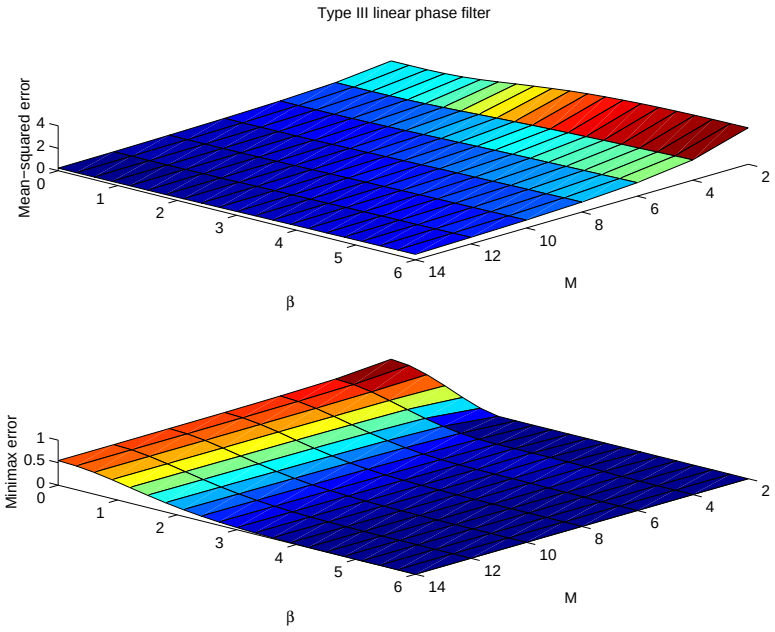

Figure 7.4: M versus β for type III filters.

Figure 7.5 shows the same graphs for type IV filters. This figure suggests that tuning both parameters will minimize the error:

- If the mean-squared error criterion is used, choosing a small β will get a smaller error for the same M.
- If the minimax error criterion is used, the error is reduced if both M and β is sufficiently large.

7.1.3 Least-squares error approximation

Another approach we can take to design the differentiator filter, is to start with the frequency response constraint; the amplitude should be linearly dependent on ω. To find the impulse response we can solve the problem by using Matlab's function `firls`, which searches for the impulse response using least-squares error minimization described above.

The `firls` has a special mode for differentiators, which assigns weights to each frequency band, in such a way that it is the inverse of the frequency squared, and proportional to an optional weight array. The result of this is that the filter has much better fit at low frequency than at high frequency.

As an example, figure 7.6 shows a differentiator filter created using the `firls` command, where the order of the filter is 10. Just as expected, the filter has a zero at $\omega = \pi$, as this filter is a generalized phase filter (and thus of type III).

Figure 7.7 shows another differentiator where the order is 11. This filter does not have the zero at $\omega = \pi$, as expected (because this is a type IV filter).

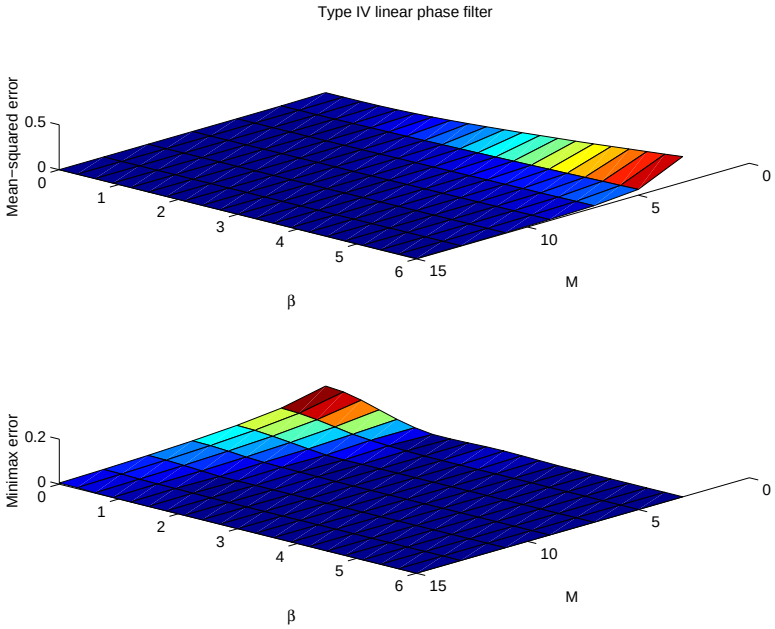

Figure 7.5: M versus β for type IV filters.

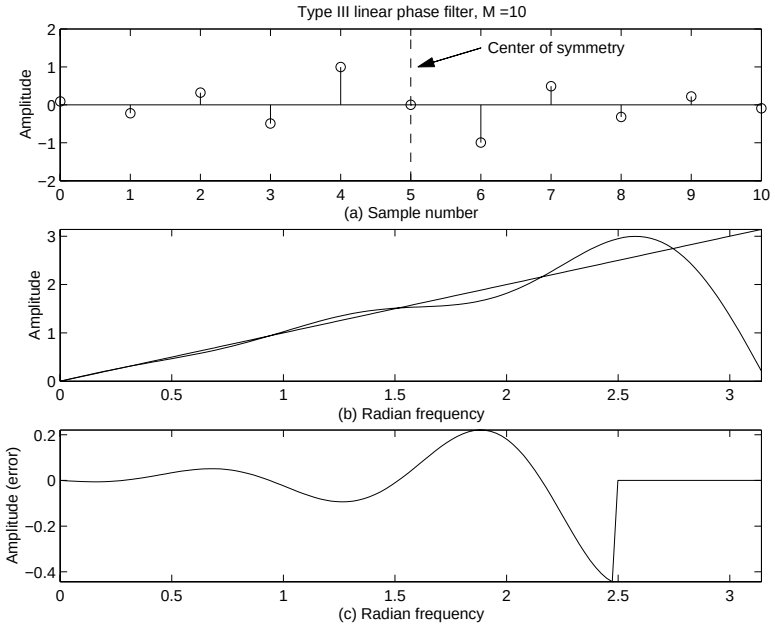

Figure 7.6: 10. order FIR filter designed using the `firls` method in Matlab: `firls(10, [0, 1], [0, 1], 'differentiator')`.

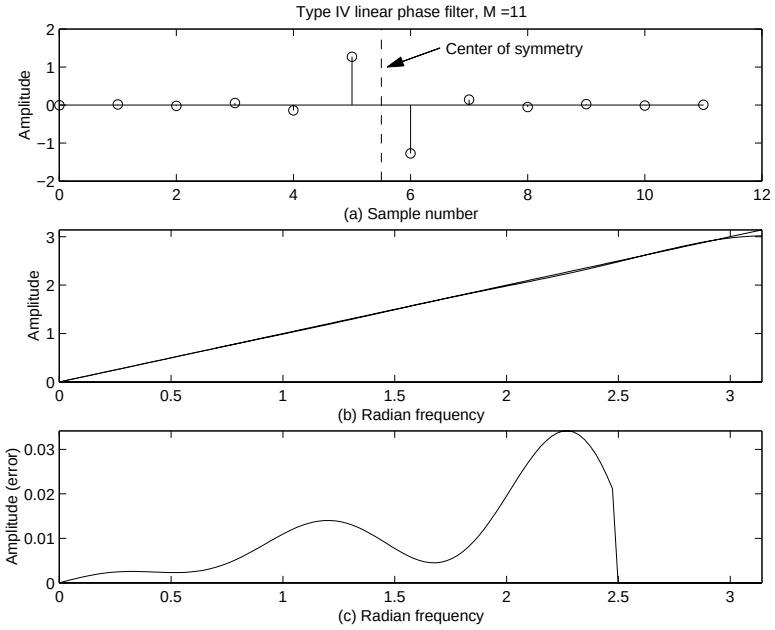

Figure 7.7: 11. order FIR filter designed using the `firls` method in Matlab: `firls(11, [0, 1], [0, 1], 'differentiator')`.

7.1.4 Second order linear differentiators

We now turn to second order differentiators, and see how to construct FIR filters that will yield the desired frequency response.

The frequency response for the second order derivative can of course be found by differentiation of $\dot{X}(j\omega)$. We can also think of the second order derivative as two cascaded differentiators, where the input $x(t)$ is filtered, yielding $\dot{x}(t)$, and then that result is filtered again, yielding $\ddot{x}(t)$. So, we can combine two LTI systems, and the resulting system is the convolution of the two impulse responses. In the frequency domain, the convolution becomes a multiplication, which means that the Fourier transform of $\ddot{X}(j\omega)$ is $\dot{X}(j\omega)^2 = \omega^2 X(j\omega)$

We can use similar techniques as described above to create second order differentiators, or we can just convolve two first order differentiators. Notice that second order differentiator will be of odd length. Using a first order differentiator of type III or type IV, will result in a second order differentiator of type I, as it will always be symmetric and of odd length (M is even).

7.1.5 Example and test of noise sensitivity

This section includes one example, and a test of the differentiators' noise sensitivity.

The signal used in the plot below is a simple sinusoidal signal. The sample rate was set to $F_s = 4000$ Hz, and the frequency of the signal was set to 250 Hz. Different signal frequencies have been tested, yielding approximately the same results.

Figure 7.8 shows the signal with additive Gaussian noise, and its estimated first

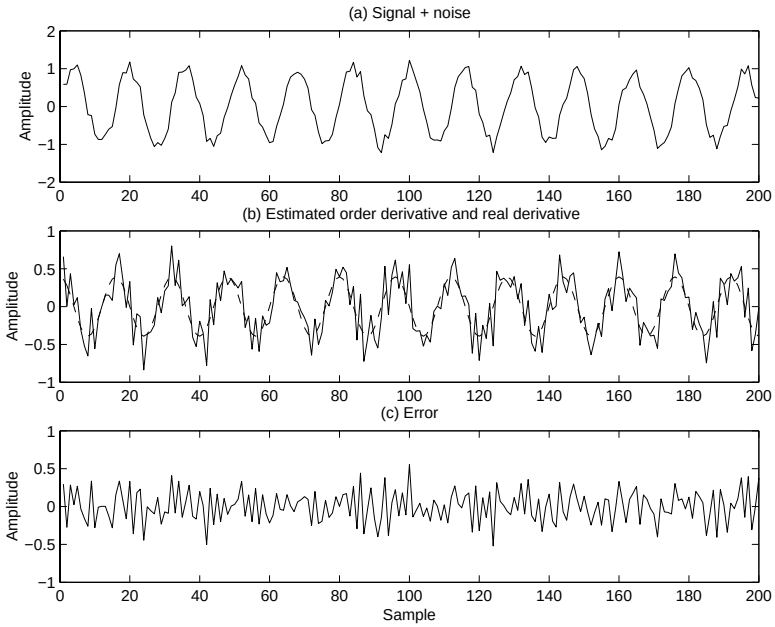

Figure 7.8: Sinusoidal signal and noise (SNR = 15dB), with its first order derivative and the estimation error.

order derivative (using a filter of length 2). The signal-to-noise ratio was set to 15 dB. Subfigure (a) contains the signal plus noise, subfigure (b) shows the estimated derivative and the analytic derivative (dashed lines), and subfigure (c) shows the difference between the analytic derivative and the estimated derivative. In this case, the estimate and the error are in the same range, and we thus conclude that this differentiator is not very robust in presence of noise.

Figure 7.9, subfigure (a) shows a plot of the RMS error between the estimated signal and the estimated derivative as a function of filter length and SNR. The results from this plot are discouraging, as it clearly shows that the error does *not* depend on the filter length. In practice, this means that none of the longer filters perform any better *in noise* than the two-point filter that we initially wanted to improve.

7.1.6 An alternative FIR implementation

Linear FIR differentiators do not necessarily have to be based directly on differentiation of the Fourier transform, as above. An alternative implementation is the filter [38, 39]

$$h[n] = \frac{6}{(2N+1)(2N+2)}\left(1 - \frac{n}{N}\right), \qquad n = 0, \cdots, 2N \tag{7.8}$$

This filter is of length $2N + 1$, and has an impulse and frequency response similar to the filter and response depicted in figure 7.10. The filter is of odd length, and is antisymmetric, and it is thus a generalized phase filter of type III. Notice that the amplitude response has a lowpass characteristic, and the response is thus not in general proportional to the frequency. However, the filter response *is* proportional to the

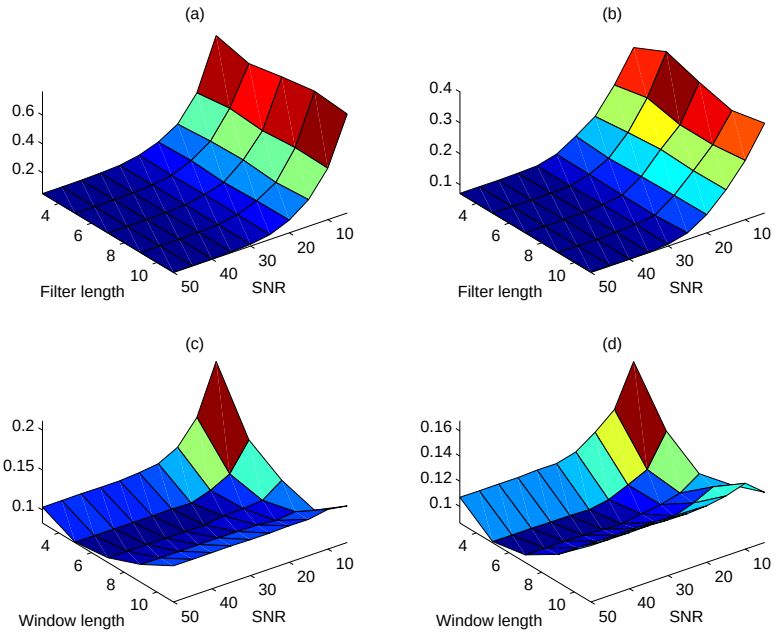

Figure 7.9: RMS errors as functions of filter/window length and SNR.

frequency for very small frequency values. This suggests that the differentiator is only suited for signals which are heavily oversampled, or else the scaling of the estimated signal will deviate significantly from the real derivative.

Figure 7.11 shows an example, where a simple sinusoidal signal of 70Hz is sampled at 4000Hz. White Gaussian noise has been added, so the SNR is in this case 15dB. The filter in figure 7.10 was used to estimate the derivative. Compared to the other FIR differentiators, it is evident that this differentiator is very robust in the presence of noise.

7.1.7 Linear FIR differentiators – summary and conclusion

In this section, we have looked at the filter class of linear first and second order differentiators. Examples have shown the different types of filters, and some error estimates have been done. The section ended by two examples, where it is shown that the linear differentiators based on the frequency response will not help us to suppress the noise and give us better estimates. Longer filters should therefore only be used if we want a better frequency response, when we already have a high SNR.

7.2 Linear regression differentiators

The problem of differentiators can be solved by using other methods than linear filters. In "Nonlinear methods in differentiators" by Sari Siren and Pauli Kuosmanen [38] some of the work done on discrete differentiators is summarized. The method of linear regression differentiators is explained, and we will state the results here.

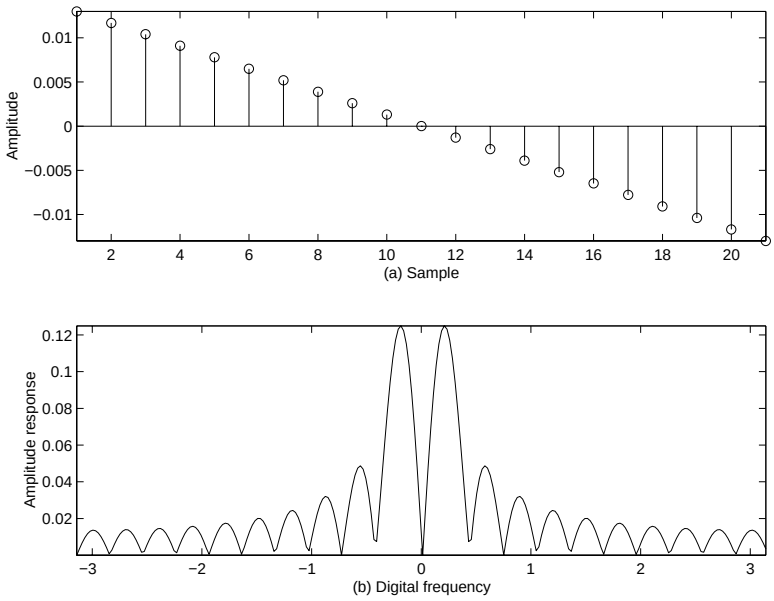

Figure 7.10: (a) Impulse response of filter in section 7.1.6. (b) Frequency response of the filter in (a).

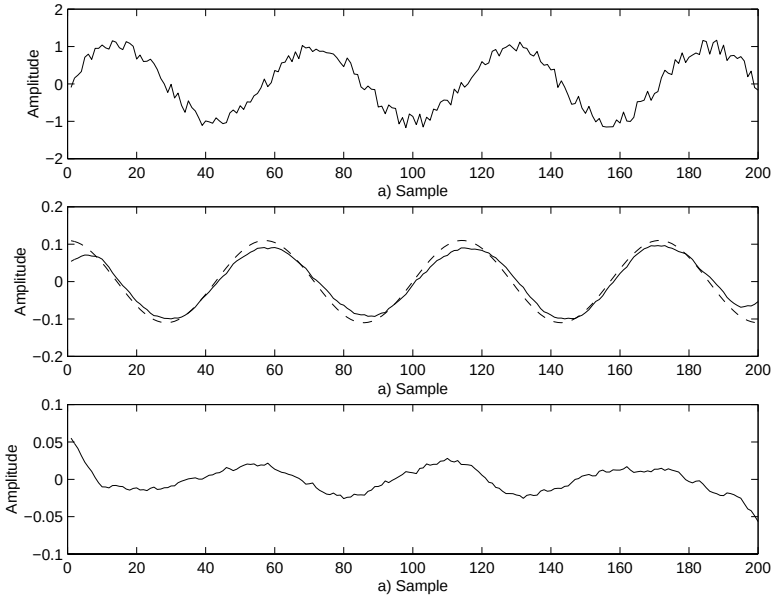

Figure 7.11: (a) Sinusoidal signal with noise. SNR = 15dB. (b) Solid line is estimated derivative, dashed line is real derivative. (c) Absolute error between real and estimated derivative.

If we assume a sliding window of N samples, we can try to approximate the signal in this window by a straight line. We use linear regression to solve this, and use the least squares method to find the slope of the line. If we assume that the window is chosen such that the estimated line is centered in this window, then we can use the estimate of the line's slope as the estimated derivative in that point.

If the line is given as $\hat{y} = ax + b$, then

$$a = \frac{E\{XY\} - E\{X\}E\{Y\}}{E\{X^2\} - E\{X\}^2} = \frac{\mathrm{cov}(X,Y)}{\mathrm{var}(X)} \tag{7.9}$$

$$b = E\{Y\} - aE\{X\} \tag{7.10}$$

where X and Y are the two random variables, and Y is dependant on X. Because we have a limited number of samples, we can estimate the expected value of X and Y as the mean of x and y in the window. $E\{XY\}$ can also be estimated as the mean of the product of the two sequences: $E\{XY\} \approx \frac{1}{N}\sum_{i=1}^{N} x[i]y[i]$, where the indices are relative to the window, and not the whole sequence.

This estimator handles uniform noise nicely, but it is still sensitive to peaks in the noise, spikes. A possible solution to this is to somehow do a nonlinear filtering of the data before the linear regression is done.

When using the linear regression model, the window length must be adjusted to fit the sampling frequency and desired frequency range of analysis. The Nyquist theorem states that we need at least two data points per period of an oscillating signal to be able to reconstruct it properly. This implies that the regression window should never be more than roughly half of the number of samples used to represent one period in the desired frequency range, that is, the highest frequency we want to track. If more samples are used, there is a greater risk that the data points in the window no longer resembles a straight line (it might be the upper or lower half of an oscillation, for instance, and the estimated slope will deviate severely, as the best fit to such curve will most likely be almost horizontal).

7.3 Nonlinear differentiators

The nonlinear filtering or selection process can be done at least either as some pre-filtering process before the slope estimation is done, or the slope estimator itself may reject samples that it deem to be outliers (which are then rejected from the estimate calculation). The difference between these strategies is that in the first case the estimate is based on the same number of data points as the window length, and in the second case, the number of data points considered usable may vary.

A good candidate for a pre-filter is the median filter. In [38], a median filter of length three was found to perform well. Median filters of such a short length also have little performance penalty, as it is easy to implement as custom three-point version if necessary.

The nonlinear pre-filter process also helps the linear differentiators described above. In figure 7.9 (b), the same test as in subfigure (a) is done, but the signal is pre-filtered using a three point median filter first. As expected, the error does not vary with the filter length, but the error has dropped considerably to roughly 60 percent of the error without the pre-filtering applied.

Subfigure (c) of figure 7.9 shows the error estimate when linear regression is used. In subfigure (d), the values of y has been pre-filtered using a three-point median filter. We notice that this estimation procedure outperforms the other methods: comparing (c) to (a), the RMS error in (c) has dropped to about a fourth (worst case) of (a). The results in (d) are even better, although the differences between (c) and (d) are not as apparent as the difference between the linear filters and the linear regression method.

A new nonlinear estimator is presented in [38], and here the slope estimator itself accepts/rejects samples based on the residuals, $\hat{e}^2[i] = (y[i] - \hat{y}[i])^2$, of the estimated line and the actual values. This estimator is based on the WMMR (Weighted Majority with Minimum Range) filter described in [40]. Based on the ideas from these articles, we have tested a similar, although simpler, estimator, described below.

First, an initial line is estimated using the standard linear regression explained above. Then the residuals are calculated, and $m < N$ new points are selected and a final estimate is done using the linear regression. The m points are selected such that the points having the largest residuals are removed.

The value of m is important here, and depends on the noise one wants to remove. If single peaks in the input are to be removed, then m can be chosen to be $N - 1$, as this methods removes one sample from the final calculations. If there are no peaks, for instance if the signal *does not* contain any noise in the current regression window, then one good sample is thrown away. However, as we assume that the window is short and the samples in general should lie on a straight line, this should not distort the results very much.

Figure 7.12 shows the error estimate as a function of regression window length and SNR when the algorithm above is implemented, and one sample is discarded per regression window. First of all, we notice that this algorithm outperforms the linear filters as expected. We also note that its performance is not much better than the conventional regression filter (the overall mean value of the RMS is about the same for both filters). The difference between the two regression filters is the sensitivity to the regression window length. The nonlinear regression filter is more tolerant, as it throws away samples that it deems to be noise. The filter used in figure 7.12 has its lowest RMS errors when the window length is five, whereas the new filters has acceptable RMS error values when the window length is between five and nine.

7.4 Nonlinear differentiators – summary and conclusion

In this section we have seen that differentiators can be implemented successfully without using linear filters. We have implemented an estimator based on ideas from [40, 38], and our examples have shown that this estimator outperforms the linear filters in noise.

7.5 Summary and discussion

In this chapter, we have looked at discrete differentiation in noise. Several methods have been considered, including linear differentiators implemented as FIR filters, linear regression filters, and a combination of nonlinear filters and linear regression.

We have found that differentiators implemented as FIR filters approximating the

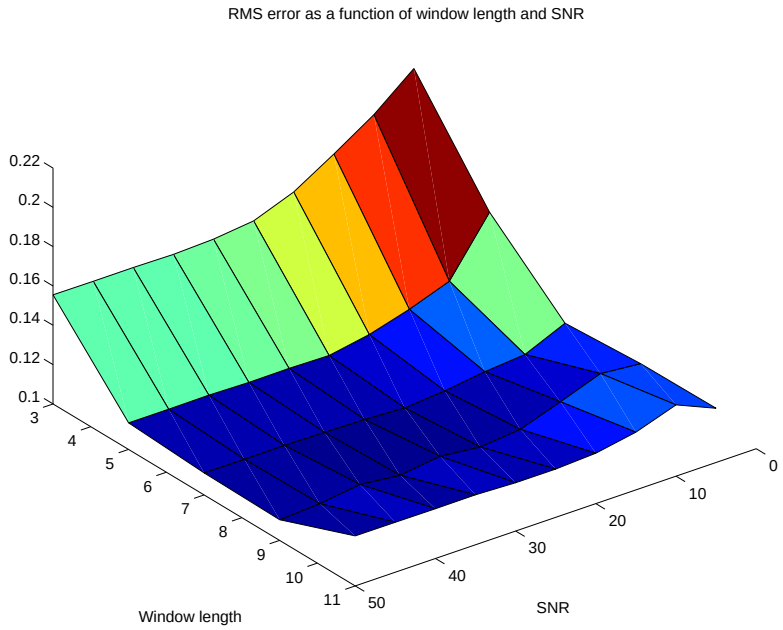

Figure 7.12: A new linear regression method, where one sample is rejected per slope estimate.

frequency response are not more robust in noise. Other FIR differentiators exists, but these are only correct for small frequencies compared with the sampling rate.

Differentiators using linear regression have been implemented and tested, and we have found that these are more robust in presence of noise.

A continuous expansion of the discrete signal using (smoothened) B-splines has not been considered here, as it is used by one of the estimators already.

Chapter 8

Speech signal applications

In this chapter, we show one application of the Teager Energy Operator, namely tracking several information signals in speech signals. Various methods have been proposed for this, and this chapter seeks to implement the multiband approach described in "Speech Formant Frequency and Bandwidth Tracking Using Multiband Energy Demodulation". We seek to implement some of the simulations in that paper, albeit with different test signals.

8.1 Speech models

The vowels in speech are composed by what we call *formants*. A formant is, simply put, a relatively narrow band of frequencies carrying information in a speech signal. The number of significant formants varies, but most, if not all, vowel-like sounds can be classified and discriminated by the two lowest formant frequencies. Figure 8.1 shows a map which can be used to discriminate vowel-sounds based on only the two lowest formant frequencies.

It has been proposed in [15, 7, 9] that each formant can be modelled as a damped AM-FM signal

$$s_f(t) = Ar^t \cos(t) \cos(\omega_c t + \beta \sin(\omega_f t)) \tag{8.1}$$

where A is the amplitude, r is a damping factor, β is the frequency deviation, ω_c is the carrier frequency, and ω_f is the baseband frequency.

A vowel can thus be modelled as a sum of the above damped AM-FM signals:

$$v(t) = \sum_{i=1}^{N} A_i r_i^t \cos(t) \cos(\omega_{c_i} t + \beta_i \sin(\omega_{f_i} t)) \tag{8.2}$$

8.2 Tracking formants in speech

Tracking the formants described above in actual speech signal is no simple task. Doing this automatically poses several questions:

- Is the analysed signal actually a vowel?
- How many formants are there, if any?

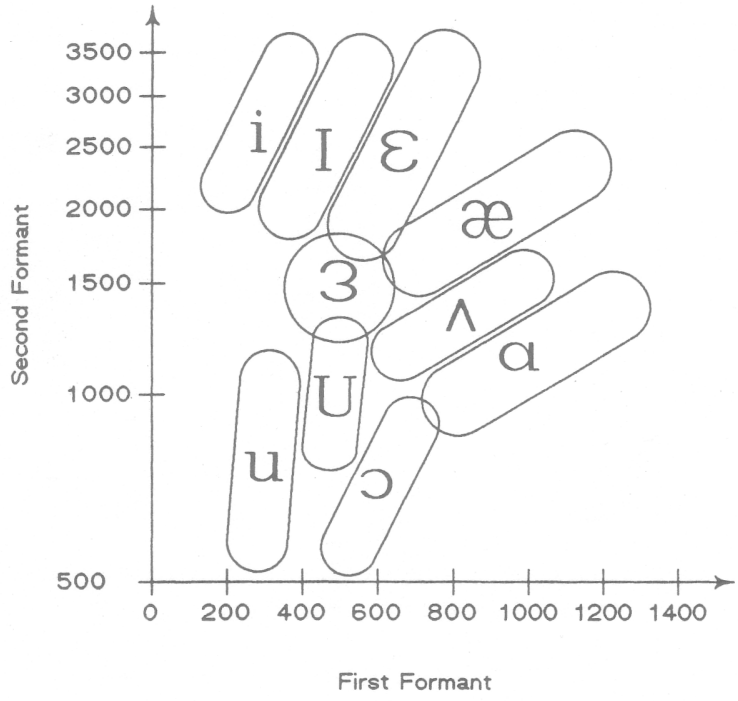

Figure 8.1: Vowel segmentation map using the two lowest formant frequencies [41].

- How can the center frequencies and bandwidths of the different formants be estimated?

The first question can probably be answered by trying to find the formants in the signal; we can try to estimate the number of formants by trying to find them. There are several methods to do this. The standard method is LPC, where the roots of the estimator polynomial are used as the formant center frequencies. Another approach is taken in [18], where a formant model similar to the above is formulated in a statistical language and an extended Kalman filter is then used to estimate the center frequencies.

A third approach is taken in [7, 15, 16]. The main idea is to find the center frequency of each formant by bandpass filtering the signal to suppress "uninteresting" parts of the signal.

The other formants (or other components or noise) in the signal will usually interfere significantly with the formant of interest with regard to energy computation. This can clearly be seen from property 3.47 of the Teager Energy Operator, as cross terms between the functions are introduced. It is therefore vital to remove everything but the "real" energy term for each formant.

8.3 Tracking formants

Formant tracking using the Teager Energy Operator has been treated in a number of papers, for instance [16, 21, 42]. We will in this chapter implement a multiband method for formant tracking.

In the multiband approach the problem is to find the right filters and center frequencies of the formants. Several strategies for filter center frequency can be chosen for the bandpass filter procedure. The simplest one is to manually set the center frequency and bandwidth, but this clearly is not feasible. An automatic system can do at least one of the following:

- An a priori estimate of the center frequency is given, and an iterative algorithm is used to make the initial estimate converge to the "true" center frequency [13].
- A bank of bandpass filters is created, where the filters are relatively closely spaced. The energy of each band is then computed, and compared to the other estimates. A selection function as described in section 6.2.2 is then used, based on a specific criterion (maximum energy, for instance), and a subset of the filters is selected. The center frequencies of the filters in the subset are then used as the center frequencies of the formants. This procedure may also be iterative, as the estimated frequencies may be used as new estimates for the filter center frequencies in the next iteration.

8.3.1 Selecting more than one band

When multiple signals are to be detected, the selector functions described in section 6.2.2 must be adjusted, as they only pick one band. A specious choice would therefore be to just choose the n highest frequency bands (the bands containing most energy at a given instant), as in [21]. This method works as long as two signals are separated by at least one other band, and as long as there is not too much leakage to the other neighbouring bands. The article [21] mentions that the the uniformly spaced Gabor filters performs best, and that might be explained by the leakage introduced in the other method where wider, logarithmically-spaced Gabor filters are used.

A different approach that would remedy these problems would be to do peak-picking of the energy estimates across all bands, and choose the n highest peaks as bands carrying the signals.

8.3.2 Another approach to the selection problem

In "Speech Formant Frequency and Bandwidth Tracking Using Multiband Energy Demodulation" [42], another approach to formant tracking is taken. In this paper, the tracking procedure is split into the following steps

- Run the multiband analysis on the input signal.
- Estimate the instantaneous frequencies and amplitudes.
- Split the estimated values into frames of a given length, and estimate mean frequencies and bandwidths.
- Estimate "raw" formant tracks for all frames.
- Post process the "raw" formant tracks and create the final formant tracks and bandwidth estimates.

The first step in this procedure is just as described above, and uses a filterbank similar to the one depicted in figure 6.1. Instead of using the selector function, the frequencies and amplitudes are estimated for all bands. The estimation of frequency and amplitude is done using one of the ESAs. DESA-1 is used in the examples later.

We then calculate the mean frequencies and bandwidths. This is done on a frame-by-frame basis. Splitting the signal into frames will most likely give us more robust estimates, at the cost of better time resolution. A frame-by-frame approach will also give us a bandwidth estimate, which we otherwise can not get an instantaneous measure of, at least not if the estimated bandwidth is defined as below.

We divide the signal into frames of length T, and estimate the frequency and bandwidth of each frame. It makes sense to define the short time frequency and bandwidth as the mean frequency during a frame and the variance of the frequency estimate, respectively [42]. More formally,

$$F_1(n) = \frac{1}{T}\int_{t_0}^{t_0+T} f_i(t)dt \tag{8.3}$$

$$[BW_1]^2(n) = \frac{1}{T}\int_{t_0}^{t_0+T} [f_i(t) - F_1(n)]^2 dt \tag{8.4}$$

where n is the frame number, T is the length of one frame, t_0 is the start time of frame n. The formulas above are for the continuous case, but these can be translated more or less directly to the discrete case in the actual implementation.

According to [42], we can also use the first and second moments of f_i to estimate the frequencies and bandwidths,

$$F_2(n) = \frac{\int_{t_0}^{t_0+T} f_i(t)a^2(t)dt}{\int_{t_0}^{t_0+T} a^2(t)dt} \tag{8.5}$$

$$[BW_2]^2(n) = \frac{\int_{t_0}^{t_0+T} (a(t)/2\pi)^2 + (f_i(t) - F_2(n))^2 a^2(t)dt}{\int_{t_0}^{t_0+T} a^2(t)dt} \tag{8.6}$$

These two methods of frequency and bandwidth estimation have different properties, which are exemplified in [42] (which signal they will "lock" on to under certain conditions, for instance).

When the frequencies and bandwidths are estimated, we can visualize these by using what we call a *pyknogram*. A pyknogram is a time-frequency plot of the signal, and can be compared to the conventional spectrogram. A spectrogram is composted by a number of short-time Fourier transforms, and will usually be a compromise between time and frequency resolution (although methods exist that try to preserve both [43, 44]). In addition, a spectrogram shows the "amount" (amplitude) of the frequencies at given points in time. A pyknogram differs from this, in that it does not show any amplitude of the frequencies. Instead, it shows the different "interesting" frequency bands, AM-FM signals, as dense areas in the plot.

The frequency and bandwidth estimates are used to select the formant candidate frequencies. A formant candidate is a frequency estimate which is deemed to be included for further processing in the formant tracking routine. It is likely that there is a formant

at this frequency in the analysed signal, however, being included here does not mean that it will be included in the final estimate created by post-processing all candidates.

If we take one frame and plot filter band (the frequency, f) versus estimated frequency ($F_m(n)$, where m is either 1 or 2 to choose mean frequency estimation method), we can pick the high density areas rather easily; wherever the estimated frequency is equal to the center frequency of a band ($F_m(n) = f$) and $dF_m(f)/df < 1$, we choose the estimated frequency as a formant candidate. This is depicted in figure 8.5. The points marked by a star are formant candidates for the given frame. Additionally, strong formants may shadow weaker neighbouring formants. If this is the case, then the frequency estimate has a local maximum nearby the stronger formant, and we can find these by including the local maxima of $F_m(f) - f$ to the set of formant candidates if $F_m(f) < f$ and the difference between them is below a given threshold (to avoid including all local maxima).

The formant candidate picking can also be done by using the pyknogram. An automatic picking routine can be implemented by estimating $dF_m(f)/df$, and finding the minima of this function. The paper states that as the pyknogram is used, it has a finite resolution, which means that there is a limit on the precision one can archive in the formant estimation.

When the formant candidates are selected, the last step is to post process the estimate to remove spurious candidates and connect the different estimates and estimate the final bandwidths of the formant tracks. This is done using linear prediction. This last step has not been implemented in the experimental results presented below.

8.4 Experimental results

Several signals have been analysed using the method described above. All signals we have tested were real speech signals, and we will present two of them here; one simple signal consisting of a single vowel, and a more complex one consisting of a few words.

In general, the signals have been recorded under good conditions, so the original signals are assumed to have a good signal to noise ratio. The test signals have been re-sampled to 16000Hz, and this is because the Teager Energy Operator is best suited for signals with frequency less than one fourth of the sampling frequency, that is 4000Hz in our case. The two lowest formant frequencies are both well below this limit, so we should have no problem discriminating the different vowel sounds.

The filter bank used was composed by Gabor filters uniformly spaced from 200 to 4000 Hz, 50 Hz apart. In total, the bank consisted of approximately 80 Gabor filters. The bandwidths of all filters were also chosen to be uniform. This method of formant tracking is found to be very sensitive to the filter bandwidth, and the filters must be rather wide [13].

Figure 8.2 shows the pyknogram of the vowel sound 'i', as spoken by a 25 years old male speaker. Notice the three distinct bands of high density in the plot. By inspection, we find that these bands are at approximately 250, 2250 and 3500 Hz. The two lowest formants are therefore at 250 and 2250 Hz. By using the segmentation map in figure 8.1, the map says that the formant should be an 'i', which is exactly what we expected.

Figure 8.3 shows a plot of the estimated formant candidates. Notice the three lines found as formant candidates at 250, 2250 and 3500 Hz. There are also some dots

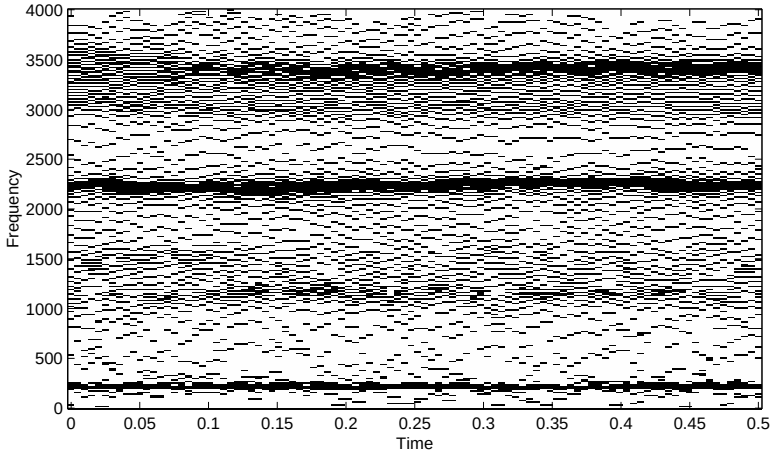

Figure 8.2: Pyknogram of the vowel sound 'i'.

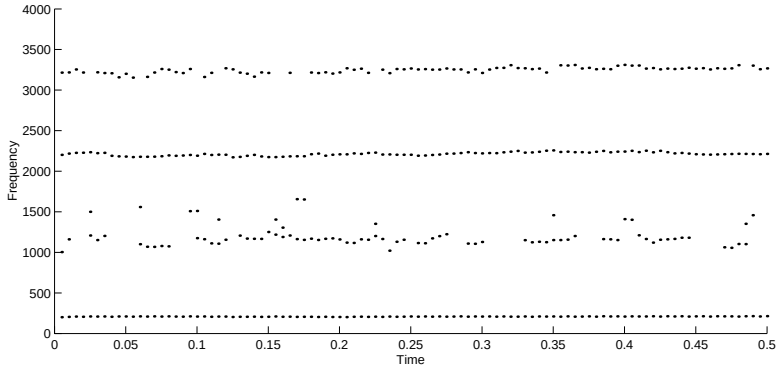

Figure 8.3: The formant candidates for the vowel sound 'i'.

between 1000 and 1500 Hz, but these do not make up a real formant. A higher density of dots can also be seen in the pyknogram, but this area is not as dense as the areas for three other formants. The reason for the inclusion of these candidates is that they are all boundary cases for the candidate picker (found by inspection of the candidate formant picking routine). These could either be removed by a more robust candidate picker, by adding more constraints to the algorithm, or by the post-processor that connects the formant candidates and estimates bandwidths.

Figure 8.4 shows both the pyknogram and two spectrograms of a longer speech signal. The actual speech is 'Ja, rev på to-tjue' (a sample from a joke by the two Norwegian comedians Lystad and Mjøen). The signal contains six vowels, including two instances of an 'e', and they are all marked above the pyknogram.

One frame as analysed by the formant picking routine is shown in figure 8.5. Subfigure (a) shows the mean estimated frequency across one frame of data for all bandpass filters (placed along the x axis). The dashed line is drawn through all center frequencies. Subfigure (b) is a zoom-in of subfigure (a), centered at the dashed vertical

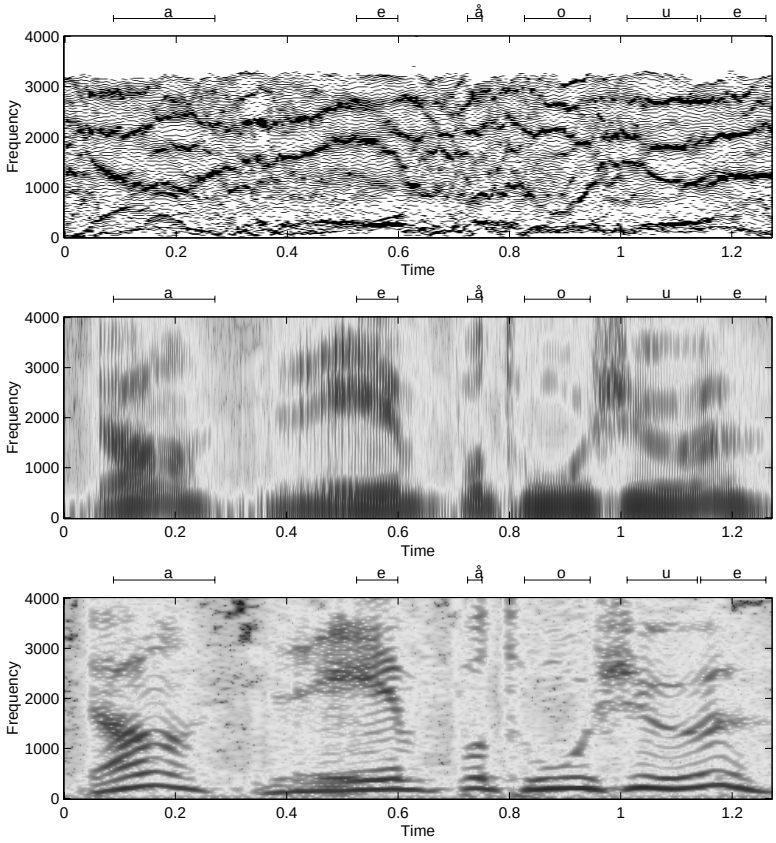

Figure 8.4: Pyknogram and wide/narrow band spectrograms showing the densities for 'Ja, rev på to tjue'.

line. In this figure we clearly see that the estimated frequency curve intersects the center frequency curve. As the derivative of the estimates frequency curve at the intersection is less than one, the algorithm decides that a formant candidate is found. Subfigure (c) shows the estimated bandwidths, the frequency variances, for each bandpass filter.

The results from the formant picking routine for the longer speech signal are shown in figure 8.6. In this figure, we can see clear tracks of the formants, but clearly spurious estimates are also included. Again, these could be removed by a more robust candidate picker, or by the post-processing routine.

8.5 Summary

In this chapter we have looked at the process of demodulation of speech signals, which as of today is the most common application of the Teager Energy Operator. During this chapter, we have

- Introduced the problem of formant tracking, and seen that the Teager Energy Operator can be used successfully for this purpose.

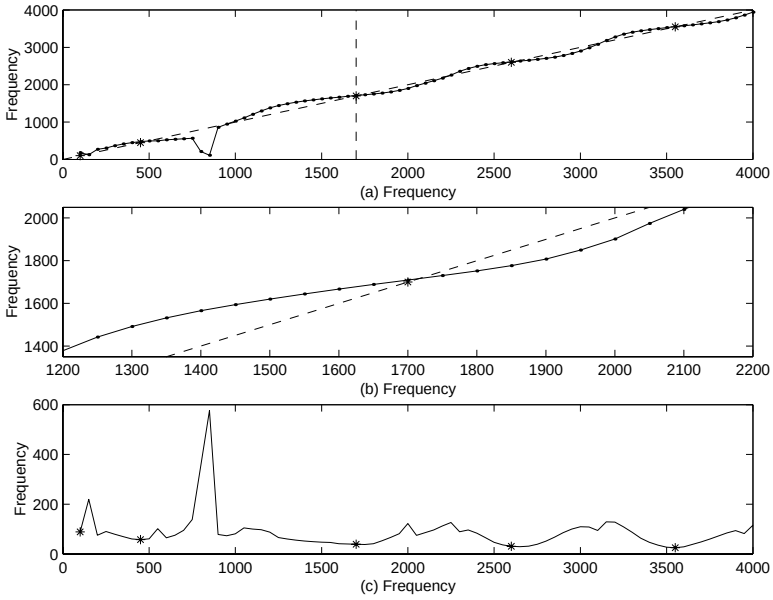

Figure 8.5: Selection of formant candidates. (a) Estimated frequency in frame. (b) Zoom-in centered at vertical dashed line of subfigure (a). (c) Estimated bandwidths.

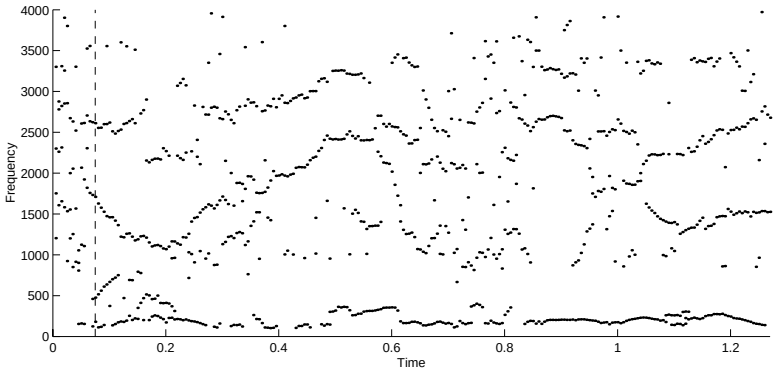

Figure 8.6: Formant estimate for the longer speech signal. The vertical dashed line shows the analysed frame in figure 8.5.

- Looked at a multiband method for the analysis of signals composed by more than one component.
- Explained and implemented a framework for formant tracking as described in "Speech Formant Frequency and Bandwidth Tracking Using Multiband Energy Demodulation" [42].

Chapter 9

A performance comparison

In this chapter, we will do a performance comparison of the instantaneous frequency estimators that we defined in chapter 5.

9.1 Estimators

The following estimators will be tested:

- DESA-1a
- DESA-1
- DESA-2
- Modified covariance method, 4 and 5 point versions
- Prony's method, 4 and 5 point versions
- Spline ESA
- Shekel's method
- The Hilbert transform method

Mandelstam's method will not be tested, as our preliminary tests showed that it did not reach up to the expected performance level of the other estimators.

9.2 Test signals

During the research of the Teager Energy Operator, a certain standard set of test signals has been created [15]. This means that for each claimed improvement, a special range of test signals has been used to show empirical evidence that it is better or more robust than its preceding algorithms.

The standard signals that have been used are:

$$s_{i,j}[n] = \left[1 + \kappa \cos\left(\frac{\pi}{100}n\right)\right] \times \cos\left[\frac{\pi}{5}n + 20\lambda \sin\left(\frac{\pi}{100}n\right)\right] \tag{9.1}$$

with $(\kappa, \lambda) \in \{(0.05i, 0.05j) : i, j = 1, \ldots, 10\}$, $N = 401$, and $n = 0, \ldots, N-1$.

These signal are AM-FM signals, where the i variable generates ten different AM variants, and j generates ten FM variants, totalling 100 AM-FM signals when combined.

9.3 Estimation error measures

Traditionally, the estimation error has been measured in two ways. First, we have the Frequency Mean Absolute Error, which is measured in percent. For the signals $s_{i,j}[n]$ above, this error can be formulated as follows:

$$E_{mean}[i,j] = \frac{\frac{1}{N}\sum_{l=0}^{N-1} |\hat{f}[l] - f[l]|}{\frac{1}{N}\sum_{l=0}^{N-1} f[l]} \tag{9.2}$$

where $f[l]$ is the exact frequency and $\hat{f}[l]$ is the estimated frequency for signal $s_{i,j}[n]$. The factors $\frac{1}{N}$ are included to show that we actually compute the mean value of the absolute error. The final estimation error is the average value of E_{mean} multiplied by 100 to get a percentage.

The other error measure is the RMS error, also in percent, which we calculate as

$$E_{rms}[i,j] = \frac{\text{RMS}\{\hat{f} - f\}}{\text{RMS}\{f\}} \tag{9.3}$$

for each signal $s_{i,j}[n]$, where $f[l]$ and $\hat{f}[l]$ are as described above. The final estimation error is the average value of E_{rms} multiplied by 100 to get a percentage.

9.4 Noise

In practical applications, noise will always corrupt the signal. It is therefore of great interest to see how well the different estimators perform in noisy environments. To find out this, we will therefore test the estimators for the different signals in different levels of noise. In our tests, we used a white Gaussian noise model, with zero mean and variance adjusted to get specific signal-to-noise ratios.

In the tests below, the SNR varied between 10 dB and 70 dB.

9.5 Simulation results

We here present the simulation results using the noise model and the test signal set described above. All plots use the same scaling for easy comparison, as the results are spread across several figures. This is done to avoid too many graphs in one figure, and it makes comparison between estimators of equal length easier. A consequence of using the same scale for all plots is that for some estimators, the estimate is clipped for low values of the SNR. However, this means that the error is quite large, and the estimator should thus not be used in environments with that SNR.

In the following error plots, only the RMS error is shown. Each figure is divided into two subplots, where the lower one is a zoom-in of the upper plot. This is done for better visualization of the error for high SNR values.

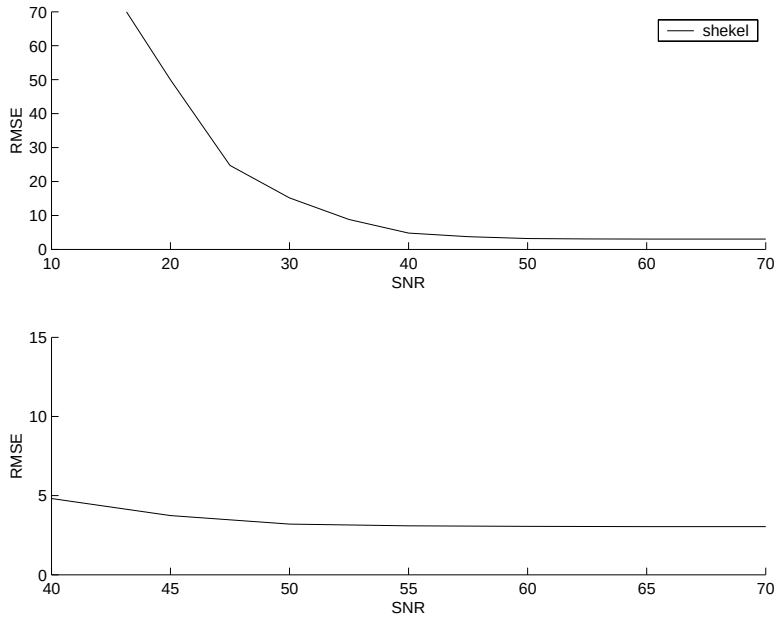

Figure 9.1: Mean RMS values as a function of SNR for Shekel's method.

9.5.1 Estimators of length 3

Only one estimator uses three samples to estimate frequency, and that is Shekel's method. However, it is possible to construct a Spline ESA of order $v = 3$ and a Hilbert transform method of the same length, but we have not done that here. We will use longer versions of these below.

In figure 9.1, the RMS estimation error as a function of SNR. Here, we see that there is a lower bound for the estimation error, which is about 3% for our signals. This is reasonable, as our discrete implementation of Shekel's method is never exact, according to section 5.9.1. From the figure, we see that Shekel's method is only suitable in high SNR environments.

9.5.2 Estimators of length 4

In figure 9.2, our 4-point estimators are shown. In this plot, we see that both the Modified Covariance and Prony method have better performance than DESA-1A in low SNR. For high SNR, the Prony method performs best, but for SNR values larger than 55dB, the differences are marginal.

9.5.3 Estimators of length 5 and above

The estimators of length five are shown in figure 9.3. In this plot, we see that the differences are marginal in high SNR, but empirical evidence shows that the Prony estimator of length five is the best in high SNR environments [32]. Notice that its performance degrades rapidly, and Prony's method is outperformed by all others in presence of noise (SNR $<$ 65-70dB). Modified covariance is the overall best estimator,

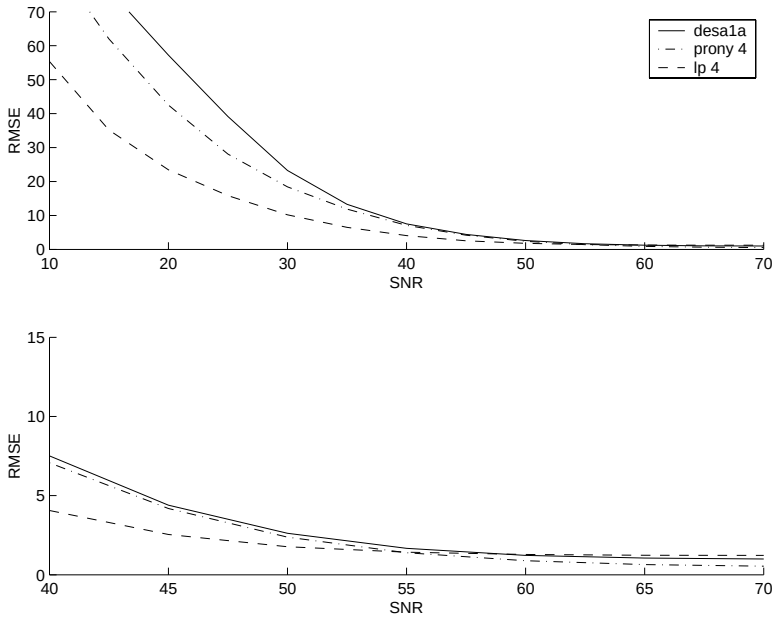

Figure 9.2: Mean RMS values as a function of SNR for the estimators of length four.

while DESA-1 and DESA-2 perform approximately equally well. To conclude, none of the estimators are particularly good in presence of noise, and should not be used in SNR less than 40dB, although they perform better than their four-point counterparts.

Different lengths of the Hilbert transform method

The Hilbert Transform method may use different lengths for the local Hilbert transform. In figure 9.4, different versions of this estimator are shown, with window length varying from 5 to 140 samples. The performance of the Hilbert Transform in this test is poor, but consistent with regard to noise. These results do not correspond to tests done in [31], which shows that the performance of the Hilbert transform method is approximately equal to the DESAs. The reason for this difference might be differing FIR filters for the Hilbert transform, and the effect of significant transients on the filter edges in our implementation.

Different parameters in the Spline ESA

The most recently developed estimator, the Spline ESA, is showed in figure 9.5. This estimator is dependent on two parameters; spline order and smoothness. We have only tested splines of order $v = 5$ here, as it can be shown [35] that splines of order $v = 3$ always have a larger error. In the figure below, we have tested different values of the smoothness factor. In this plot, we see that the exact interpolation method introduces large errors compared to the smoothened splines, and that the performance of the smoothened splines is acceptable in SNR values as low as at least 30dB.

Because the Spline ESA is dependent on two parameters, it is difficult to say

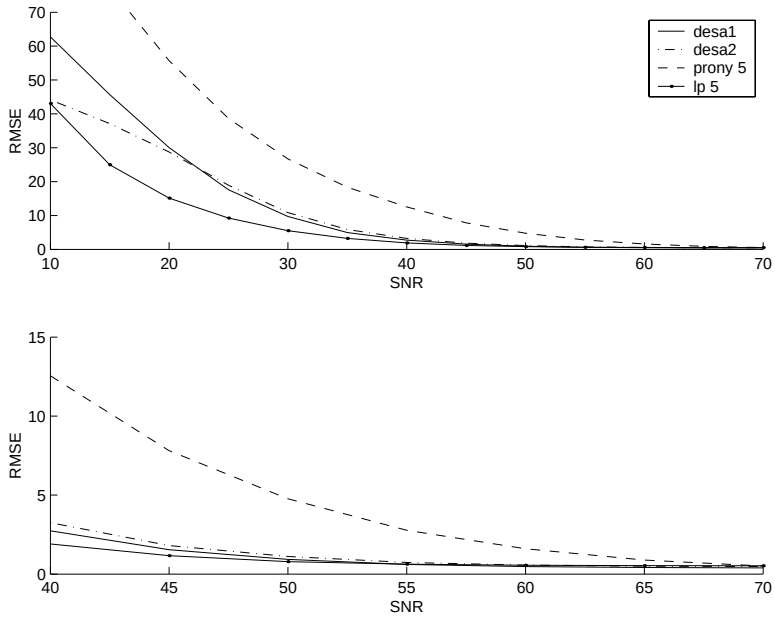

Figure 9.3: Mean RMS values as a function of SNR for five-point estimators.

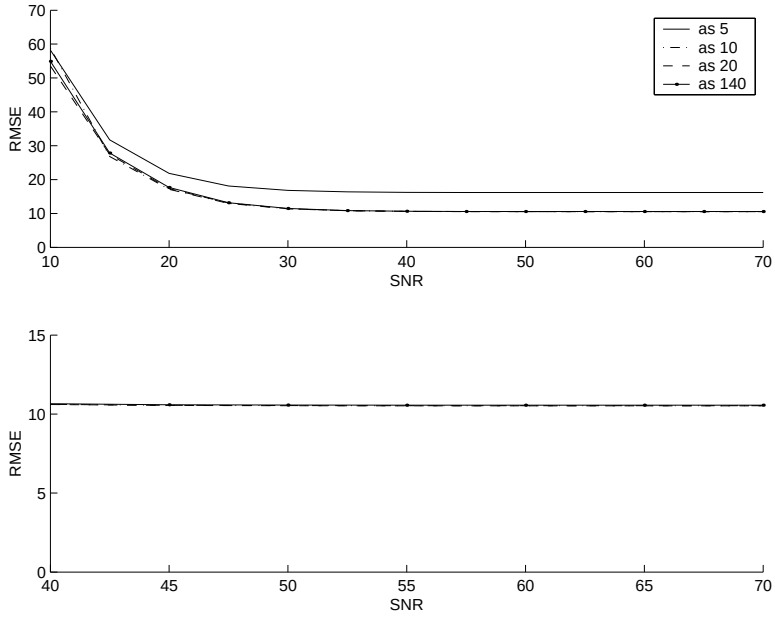

Figure 9.4: Mean RMS values as a function of SNR for the Hilbert Transform method.

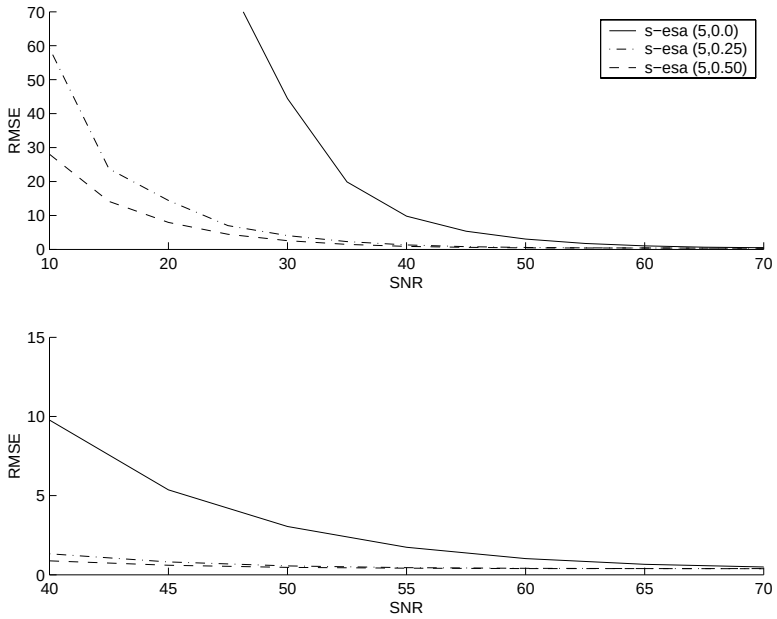

Figure 9.5: Mean RMS values as a function of SNR for the Spline ESA method with different values of λ.

something about the performance in general, other than it performs well on our data set with the parameters given below.

In the Spline-ESA test, the value of λ was constant for each test, and not a function of SNR. If the SNR is approximately known, then the value of λ may be adjusted. It was found in [35] that the error was lower than for the DESAs if λ was varied proportionally to the inverse of the SNR. This is intuitive, since for low SNR, the signal should be smoothened heavily to suppress noise, and for higher SNR, λ should be close to zero, so the continuous expansion of the original signal could be used for the energy estimate.

In general, both the order and smoothness factor are dependent on the noise and signal type, which may differ from application to application.

9.5.4 Effect of multiband filtering

One would expect that utilization of the multiband filtering strategy would improve the estimates further. In figure 9.6 the signal has been filtered using a logarithmically spaced bank of Gabor filters, $G(\omega)$. The filter bank were created according to section 6.2.1, and the filter band selector function was

$$\Psi^*(t) = \max_{1 \leq i \leq N} \frac{\Psi(f_i)}{|G_i(\omega_i)|^2} \tag{9.4}$$

which is the maximum Teager Energy normalized by the filter gain. The scale of the figure is completely wrong, and no differences between the estimators are shown, but is kept for comparison with the other tests. The figure clearly shows that the multiband demodulation outperforms all other estimators, especially for low SNR. Even for a SNR

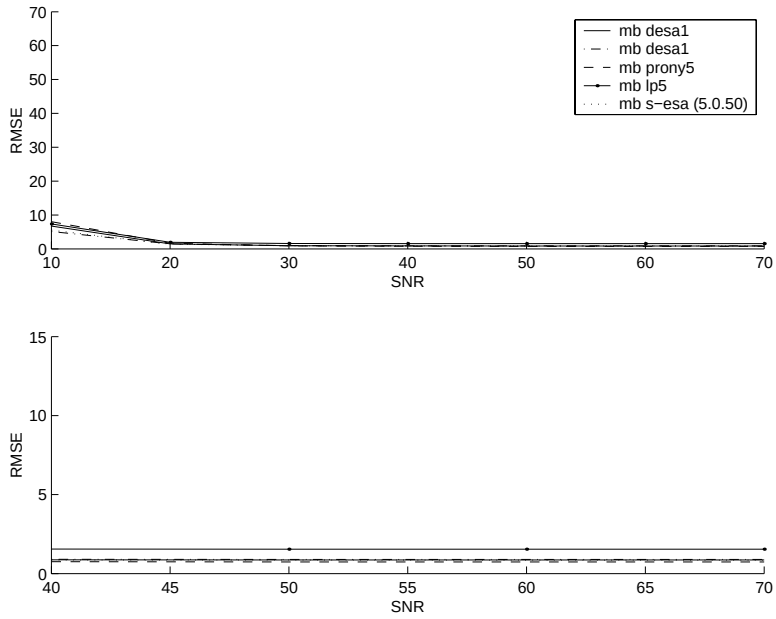

Figure 9.6: Mean RMS values as a function of SNR for five-point estimators using the multiband approach.

of only 10dB, the mean RMS error is as low as 7%. We also notice that there are only marginal differences between the estimators we used in this experiment. Surprisingly, the DESA-1 is best of them all at SNR = 10 dB. Prony's method has the largest RMS error for low SNR, and this is not unexpected since pervious tests have also shown this.

9.6 Post-processing of estimated values

For the estimation methods not directly designed to be (more) robust in presence of noise, we can use filters to smoothen the estimated value, as in [31]. Experimental results suggests an improvement of 30-50% on the estimated values.

9.7 Computational complexity

The table below shows the computational complexity of the different operators when frequency is estimated (the estimation of amplitude is *not* included). When consecutive values of the estimators are to be calculated, it is not always necessary to recalculate all sub-expressions. For instance, the value of $x^2[n+1]$ might be reused as $x^2[n]$ in the next estimate, if the formula contains both terms. Therefore, only "necessary" calculations are counted and given in the table below.

In our Matlab implementation, all necessary terms are re-calculated for each sample. This is done because all operators are very easy to express using matrix operations in Matlab. We have thus not focused on speed, but ease of implementation.

Estimator	Add/sub	Mul	Div	Arccos	Sqrt
Shekel's	3	1	1	0	1
DESA-1a	4	5	1	1	0
DESA-1	5	5	1	1	0
Prony 4	3	4	1	1	0
MCov 4	3	3	1	1	0
DESA-2	4	5	1	1	0
MCov 5	5	3	1	1	0
Prony 5	3	5	1	1	0
Hilbert transform	$N+4$	$N+4$	1	0	0
Spline ESA	$4(2v+1)+8$	$8v+7$	1	0	1

Table 9.1: Complexity of the different estimators

In systems doing real-time analysis, it might be feasible to implement the operators using only the necessary calculations given below, especially if processing power is limited, and store temporary values in a ring buffer to allow reuse of the previously calculated values.

9.7.1 Complexity of the Hilbert transform method

The analytic signal requires the Hilbert transform to be computed, and this is approximated by a FIR filter of length N in our implementation. This FIR filter may be pre-computed, and is thus not included in the calculations. The convolution results in N multiplications and $N-1$ additions per output sample. Additionally, 5 additions/subtractions and 4 multiplications are required in the calculation of the instantaneous frequency.

9.7.2 Complexity of the Spline ESA

For a given order v and smoothness factor λ, the filter used to calculate the spline coefficients can be pre-calculated, as it is independent of the input sequence. We therefore only need to find the complexity of the convolutions and the continuous time ESA. The convolution is between the signal and the B-spline of order v, which when implemented results in a filter of length $2v+1$. The complexity of this process is therefore $2v+1$ multiplications and $2v$ additions per output sample. We have to compute this for the signal and its first three derivatives. The computations of these three derivatives require one, two, and three subtractions per sample, and zero, one, and two multiplications, respectively. The continuous ESA for instantaneous frequency requires one square root operation and one division, and two estimates of the Teager Energy. The continuous Teager Energy Operator requires two multiplications and one subtraction.

9.8 Summary and discussion

In this chapter we have implemented and tested all estimators described in chapter 5. All estimators were tested in white Gaussian noise, with SNR in the range 10 to 70dB. The Spline ESA was found to have best performance alone. When the estimators were combined with multiband demodulation, the performance of the tested estimators increased, and the differenced between the estimators were marginal for all SNRs.

We have not tested pre- and post-filtering of the estimates, as this method will help any of the above estimators. The multiband demodulation was tested because the method was specifically designed to give better estimates in presence of noise. A performance increase of 30-50% by using additional filtering has been found empirically in the literature.

Chapter 10

Conclusion

In this thesis, we have studied nonlinear methods for demodulation of AM-FM signal. Numerous papers have been read during the work with this thesis, primarily focusing on the Teager Energy Operator, and its usefulness to demodulation.

In addition to a brief history of the Teager Energy Operator, we began the thesis by some general ideas and constraints regarding amplitude, phase and frequency, and then defined the Teager Energy Operator. All known one-dimensional extensions to date have been included, and some of the more useful and often-used properties were included.

As we have seen, the Teager Energy Operator has been identified to be a discrete 2D Volterra system. In this thesis, we have extended this, and have included the Laplace transform of the continuous Teager Operator, using Volterra synthesis theory. Restricting the Volterra system to require causality, we have seen that the continuous system must be an approximation.

In chapter 5, we briefly mentioned conventional analog methods for demodulation, and then looked at how the Teager Energy Operator could be utilized to estimate the instantaneous frequency and amplitude of an unknown AM-FM signal. We also introduced several other similar demodulation algorithms.

The Teager Energy Operator's sensitivity to noise was part of the motivation and problem definition for the thesis, and we have investigated the current known strategies for this. The model used to establish the Teager Energy Operator assumes that the signal is a single sinusoidal signal. When the Teager Energy Operator is employed on signals where this assumption is not true, for instance when two signals are present instead of one, cross-terms between the two signals appear in the calculated Teager Energy. We have shown examples of this, and the effect of bandpass filtering to remove the cross terms.

Demodulation of AM-FM signals using a bank of bandpass filters have also been treated. Assuming an AM-FM signal, the input signal is fed to a bank of bandpass filters, and a single bandpass filter at each instant of time is selected based on a given selector function. From this, the instantaneous frequency and amplitude are estimated.

Another measure to make the Teager Energy Operator more robust to noise, is to create a continuous expansion of the discrete signal using B-splines, and then use the continuous Teager Energy Operator to estimate frequency and amplitude.

In chapter 8 we looked at one application of the Teager Energy Operator: Speech

analysis. We first introduced our model for vowel sounds, formants, and the problem of estimating these. During the work with this thesis, a system for estimation of formant candidates has been implemented, following known algorithms found in the literature. We showed two examples of the algorithm's output.

Finally, a performance comparison was done between the different estimators. A well-known set of test signals was created, and Gaussian noise was added to get different signal-to-noise ratios. The experimental results showed that the conventional short-time estimators performed well in high SNR, but degraded rapidly in noise. The Spline ESA, using smooth splines, was the most successful short-time estimator, providing usable estimates in SNR as low as 30 dB. The test also showed strong empirical evidence that the multiband strategy works well, and that it can handle signals in signal-to-noise ratios as low as 20 dB.

10.1 Concluding remarks

A variety of applications have been developed using the Teager Energy Operator, most of them concerning speech analysis, but it is remarkable and fascinating that such applications has sprung out of such a simple definition in the recent years.

The work with this thesis has been interesting, although it has taken some time to finish it. A considerable amount of time has been spent in the library, searching for and reading about the Teager Energy Operator in the literature. The process of writing the thesis has been interesting and learning, as well as challenging.

Bibliography

[1] Herbert M. Teager and Shushan M. Teager, *A Phenomenological Model for Vowel Production in the Vocal Tract*, ch. 3, pp. 73–109. San Diego, CA: College-Hill Press, 1983.

[2] H. M. Teager and S. M. Teager, *Evidence for Nonlinear Sound Production Mechanisms in the Vocal Tract*, vol. 55 of *D*, pp. 241–261. France: Kluwer Acad. Publ., 1990.

[3] James F. Kaiser, "On a Simple Algorithm to Calculate the 'energy' of a Signal," *IEEE Proc. ICASSP-90*, 1990.

[4] "Private communications," 2002.

[5] "James Kaiser Oral History Transcript." `http://services3.ieee.org/organizations/history_center/sloan/ASSR_Oral_%Histories/jkaiser_transcript.html`.

[6] J. F. Kaiser, "On Teager's Energy Algorithm and its generalization to continuous signals," September 1990.

[7] Petros Maragos, Thomas F. Quatieri, and James F. Kaiser, "Speech Nonlinearities, Modulations, and Energy Operators," in *Proc. IEEE ICASSP-91*, vol. 1, (Toronto, Canada), pp. 421–424, May 1991.

[8] S. K. Mitra, H. Li, I.S. Lin, and T. H. Yu, "A New Class of Nonlinear Filters for Image Enhancement," in *Proc. IEEE ICASSP-91*, (Toronto, Canada), pp. 2525–2528, May 1991.

[9] Petros Maragos, James F. Kaiser, and Thomas F. Quatieri, "On Amplitude and Frequency Demodulation Using Energy Operators," *IEEE Transactions on Signal Processing*, vol. 41, pp. 1532–1550, April 1993.

[10] Alan C. Bovik, Joseph P. Havlicek, and Mita D. Desai, "Theorems for Discrete Filtered Modulated Signals," in *Proc. IEEE ICASSP-93*, vol. 3, pp. 153–156, April 1993.

[11] R. B. Dunn, T. F. Quatieri, and J. F. Kaiser, "Detection of transient signals using the Energy Operator," in *Proc. IEEE ICASSP-93*, vol. 3, pp. 145–148, April 1993.

[12] J. T. Foote, D. J. Mashao, and H. F. Silverman, "Stop Classification using DESA-1 High Resolution Formant Tracking," in *Proc. IEEE ICASSP-93*, vol. 2, pp. 720–723, April 1993.

[13] Helen M. Hanson, Petros Maragos, and Alexandros Potamianos, "Finding Speech Formants and Modulations via Energy Separation: With Applications to a Vocoder," in *Proc. IEEE ICASSP-93*, vol. 2, pp. 716–719, April 1993.

[14] G. S. Ying, C. D. Mitchell, and L. H. Jamieson, "Endpoint Detection of Isolated Utterances Based on a Modified Teager Energy Measurement," *Proceedings of the 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. II, pp. 732–735, April 1993.

[15] Petros Maragos, James F. Kaiser, and Thomas F. Quatieri, "Energy Separation in Signal Modulations with Application to Speech Analysis," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3024–3051, October 1993.

[16] Alan C. Bovik, Petros Maragos, and Thomas F. Quatieri, "AM-FM Energy Detection and Separation in Noise Using Multiband Energy Operators," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3245–3265, December 1993.

[17] Choi-JH and Kim-T, "Neural action potential detector using multi-resolution TEO," *Electronics-Letters*, vol. 38, pp. 541–543, June 2002.

[18] Shan Lu and Peter C. Doerschuk, "Nonlinear Modeling and Processing of Speech Based on Sums of AM-FM Formant Models," *IEEE Transactions on Signal Processing*, vol. 44, pp. 773–782, April 1996.

[19] D. Vakman, "On the Analytic Signal, the Teager-Kaiser Energy Algorithm, and other Methods for Defining Amplitude and Frequency," *IEEE Transactions on Signal Processing*, vol. 44, pp. 791–797, April 1996.

[20] Michael Moore, Sanjit Mitra, and Reinhard Bernstein, "A Generalization of the Teager Algorithm," in *Proc. 1997 IEEE Workshop on Nonlinear Signal Porcessing*, (Ann Arbor, Michigan), September 1997.

[21] Ridha Hamila, Markku Renfors, Moncef Gabbouj, and Jaakko Astola, "Time-Frequency Signal Analysis Using Teager Energy," in *Proc. Fourth International Conference on Electronics, Circuits and Systems*, (Cairo, Egypt), pp. 911–914, December 1997.

[22] R. Hamila, J. Astola, F. Alaya Cheikh, M. Gabbouj, and M. Renfors, "Teager Energy and the Ambiguity Function," *IEEE Transactions on Signal Processing*, vol. 47, pp. 260–262, January 1999.

[23] D. Gabor, "Theory of Communication," *J. Inst. Elec. Eng*, vol. 93, pp. 429–457, 1946.

[24] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*. New York: Wiley & Sons, 1980.

[25] Simon Haykin, *Communication Systems*. John Wiley & Sons, Inc, 1994.

[26] P. Margos and A.C. Bovik, "Image Demodulation Using Multidimensional Energy Separation," *Journal of the Optical Society of America*, vol. 12, pp. 1867–1876, September 1995.

[27] Wei Lin, Chris Hamilton, and Prabhakar Chitrapu, "A Generalization to the Teager-Kaiser Energy Function & Application to Resolving Two Closely-Spaced Tones," in *Proc. IEEE ICASSP-95*, (Detroit, Michigan), pp. 1637–1640, May 1995.

[28] James F. Kaiser, "Some Useful Properties of Teager's Energy Operators," *Proc. IEEE ICASSP-93*, vol. 3, pp. 149–152, 1993.

[29] Alan C. Bovik and Petros Maragos, "Conditions for Positivity of an Energy Operator," *IEEE Transactions on Signal Processing*, vol. 42, pp. 469–471, February 1994.

[30] Alan V. Oppenheim and Ronald W. Schafer, *Discrete-time signal processing*. Prentice Hall, 1989.

[31] A. Potamianos and P. Maragos, "A comparison of the energy operator and the Hilbert transform approach to signal and speech demodulation," *Signal Processing*, vol. 37, pp. 95–120, 1994.

[32] L. B. Feertig and J. H. McClellan, "Instantaneous Frequency Estimation Using Linear Prediction with Comparisions to DESAs," *IEEE Signal Processing Letters*, vol. 3, pp. 54–56, February 1996.

[33] L. J. Griffiths, "Rapid measurement of digital instantaneous frequency," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 207–222, Apr 1975.

[34] M. H. Hayes, *Statistical Digital Signal Processing and Modeling*. New York: John Wiley & Sons, Inc, 1996.

[35] Dimitriadis D and Maragos P, "An improved energy demodulation algorithm using splines," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, vol. 6, (IEEE, Piscataway, NJ, USA), pp. 3481–4, 2001.

[36] M. Unser, "Splines: A Perfect Fit for Signal and Image Processing," *IEEE Signal Process. Magaz.*, vol. 3, pp. 22–38, Nov 1999.

[37] Firas Jabloun, A. Enis Cetin, and Engin Erzin, "Teager Energy Based Feature Parameters for Speech Recognition in Car Noise," *IEEE Signal Processing Letters*, vol. 6, pp. 259–261, 10 1999.

[38] Sari Siren and Pauli Kuosmanen, "Nonlinear methods in differentiators," *Proc. of the IEEE/EURASIP Workshop on Nonlinear Signal and Image Processing*, September 1997.

[39] O. Vainio, M. Renfors, and T. Saramäki, "Recursive Implementation of FIR Differentiators with Optimum Noise Attenuation," *Proc. of the Joint Conference - 1996: IEEE Instrumentation and Measurement Technology Conference & IMEKO Technical Committee 7*, vol. 1, pp. 344–349, June 1996.

[40] H. Longbotham and D. Eberly, "The WMMR Filters: A Class of Robust Edge Enhancers," *IEEE Transactions on Signal Processing*, vol. 41, pp. 1680–1685, April 1993.

[41] C. Schmandt, *Voice Communication with Computers*. 115 Fifth Avenue, New York, NY 10003: Van Nostrand Reinhold, 1994.

[42] Alexandros Potamianos and Petros Maragos, "Speech Formant Frequency and Bandwidth Tracking Using Multiband Energy Demodulation," in *Proc. IEEE ICASSP-95*, (Detroit, Michigan), pp. 784–787, May 1995.

[43] Patrik J. Loughlin, James Pitton, and Blake Hannaford, "Approximating Time-Frequency Density Functions via Optimal Combinations of Spectrograms," *IEEE Signal Processing Letters*, vol. 1, pp. 199–202, 12 1994.

[44] Shiufun Cheung and Jae S. Lim, "Combined Multiresolution (Wide-band/Narrow-band) Spectrogram," *IEEE Transactions on Signal Processing*, vol. 40, pp. 975–977, 4 1992.