Nonlinear Modelling of Drum Sounds

Simon A. Hovell

A thesis submitted for the degree of *Doctor of Philosophy* at The University of Edinburgh

— 1994 —

Declaration of Originality

I hereby declare that this thesis and the work reported herein was composed and originated entirely by myself, in the Department of Electrical Engineering at The University of Edinburgh.

Simon A. Hovell

Abstract

The aim of this work was to design a model of a simple drum that could reproduce all the nuances found in a real drum effectively and convincingly. In the past, this approach had often failed due to an inability to regenerate the very beginning of the sound - known as the percussive attack - successfully, possibly because of nonlinear information present in this part of the sound. One tool for detecting the presence of such nonlinear information is higher order spectral analysis.

Detection of phase coupling between signals is one of the principal features of higher order signal analysis. It is shown that the presence of such phase coupling is a measure of dependence between different signal components. Furthermore, it is shown that examination of the power bispectrum can be used to detect the presence of nonlinear interactions between signals. Examination of the bispectra of a database of acoustic drum records gathered under strictly monitored conditions shows the presence of such interactions in the initial percussive attack.

In order to exploit this information, it is necessary to use a nonlinear filter structure. Two such structures are examined, the Volterra filter, and the radial basis function network. It is found that the Volterra filter is capable of accurate reproduction of the percussive attack. Both filter structures suffer from a large degree of redundancy, and two techniques for reducing the size of the filters are successfully applied. It is seen that the simple least squares noise thresholding method performs better than the established orthogonal least squares algorithm, although at the cost of significant computational overhead.

Table of Contents

List of Figures

List of Tables

List of Abbreviations and Acronyms

ADC	analogue to digital converter
AR	autoregressive
ARMA	autoregressive moving average
ERR	error reduction ratio
FFT	fast Fourier transform
FIR	finite impulse response
GWN	Gaussian white noise
LS	least square
LSNT	least squares noise thresholding
LTI	linear time invariant
MSE	mean square error
OLS	orthogonal least squares
PSD	power spectral density
RBF	radial basis function
SVD	singular value decomposition

List of Principal Symbols

*	The complex conjugation operator
$\|\cdot\|$	The modulus operator
$\|\|\cdot\|\|$	The Euclidean distance operator
$<\cdot$	The phase argument operator
$B(\omega_1, \omega_2)$	Bispectrum
$b(\omega_1, \omega_2)$	Bicoherence index
$\hat{d}(n)$	Desired filter output
ϵ	Mean square error (predictive)
$\hat{e}_{ms}$	Mean square error (regenerative)
$E[\cdot]$	The expectation operator
$F(\omega)$	The Fourier transform
j	A solution of $x = \sqrt{-1}$
$J_m(kr)$ and $Y_m(kr)$	First and second Bessel functions of order m
k	w/c
c	Drum membrane constant
H	Linear filter coefficient vector
$H_i[x(n)]$	ith order Volterra operator
H_{rbf}	Radial basis function network coefficient vector
H_{vp}	pth order Volterra filter coefficient vector
$\Phi(\omega_1, \omega_2, \ldots, \omega_n)$	Joint characteristic function
$P(\omega)$	Power density spectrum
$R(t)$	Autocorrelation function
$R(t_1, t_2)$	Triple Correlation function
X_{vp}	pth order Volterra filter input data vector

Acknowledgements

For whatsoever a man soweth, that shall he also reap.

Galatians v6:7

Tremendous thanks are due to my two supervisors, Dr. Bernie Mulgrew and Dr. Steve McLaughlin for their guidance and encouragement at all times. My time in the Anechoic Chamber was made possible by Dr Murray Campbell, and for this I am extremely grateful. In addition, I would like to give special thanks to my parents, who provided constant support.

The work contained herein was enrichened substanially through frequent enervating discussions with my colleagues in the department. In particular, Dave Cruickshank, Anne Moore, Dave Laurenson, Chris Cubiss, Iain Scott, Gary Ushaw, Rajan Bedi, David Hughes, Ed Warner, and John Thompson have all provided unstinting support.

Thanks are also due to many friends outside the department who provided much support and encouragement.

I am deeply grateful to the Science and Engineering Research Council, for the three year grant which made this work possible.

To my grandfather, Gee Kilpatrick.

with love

Chapter 1

Introduction

1.1 Historical Motivation

Since the early 1970's there has been a huge advance in the technology of the music and audio industry [3]. This has resulted in a vast improvement in the quality of the recording and reproduction of music, and a proliferation of electronic instruments with which to create such music. A significant proportion of research has been aimed at modelling traditional instruments. One of the most successful and popular outcomes of this has been the *drum machine*. While drum kits are large, unwieldy and expensive, small, cheap drum machines have proven to be a practical alternative for impecunious musicians. These drum machines, however have always had certain intrinsic faults.

Early techniques for reproducing drum sounds were based on noise shaping linear autoregressive filters. Although spectrally close to the original sound, these models did not sound true [4], and were never marketed as substitutes for full drum kits. Further research showed that drum sounds are characterised to listeners by the initial percussive attack [5, 6] - the thump heard at the commencement of the sound. Inability to model successfully this section of the drum sound has led to the dominance of sample based systems in both professional and amateur markets [7]. The disadvantage of this is that any particular sampled sound is invariant, while in practice, the sound produced by a particular drum varies from beat to beat depending on a number of factors, including the force of the blow with which the drum is struck, the dwell of the

stick upon the drum skin, and the position of the blow. A professional drummer uses all these factors and other techniques to add feeling to a performance. The reproduction of this variety of different sounds using a sample based system would require the storage of a vast number of sampled sounds. A further disadvantage of sample based systems is the discrete nature of recorded samples. Drums are frequently struck before the ending of the previous sound, the best example of this being the military drum roll. As each recorded sound must end before the following one can commence, it is impossible to reproduce this using a sample based system.

The aim of this work is to design a model of a simple drum which could reproduce all these nuances effectively and convincingly. Such a model could be used represent a range of different drums, and to represent different tunings of the same drum.

Much work has been done on the analysis and modelling of the sounds of musical instruments. The methods of modelling attempted previously can be largely split into two types; those models which attempt to reconstruct mathematically the physical properties of a given musical instrument [8–11], and those which attempt to reproduce the sound generated by the instrument using a black box technique [5,6,12]. Each approach has inherent disadvantages. Models of the physical properties are often mathematically complex, and frequently have no analytical solution. Signal regeneration models, on the other hand, although simple enough for real time simulations, are often too abstract to be able to represent with sufficient accuracy the complexities involved in actually playing an instrument.

1.2 Higher Order Spectral Analysis

Conventional statistical signal analysis concentrates on examining the mean and variance of signals. A more recently developed area of signal analysis involves the examination of higher order statistics, such as skewness and kurtosis, and their associated correlation functions and Fourier transforms. These transforms are known as higher order spectra. By examining the higher order statistics of signals, it is possible to identify both linear and nonlinear phase relationships between different frequencies within a given signal [13,14]. A variety of techniques have used the information present within higher order spectra to develop nonminimum phase models [15–18] and estimate model order [19,20].

As already noted, accurate reproduction of the initial percussive attack is essential for any realistic model of a drum. One possible reason for the lack of success that previous models have had is the presence of information in the percussive attack that cannot be regenerated using standard autoregeressive techniques. Identification and characterisation of this information would therefore be essential for any realistic model [21–24]. By analysing the higher order spectra of the signal, such information can be both detected and also presented in a comprehensible fashion [25–27].

1.3 Nonlinear Signal Modelling

An established approach to the development of a working model of a given system is through filtering [28]. A filter (see figure(1.1)) in this case refers to any system which operates on an input sequence of data samples $\{x(n)\}$ to produce an output sequence of data samples $\{d(n)\}$. In the present work, the objective is to produce, from some known input, an output that is perceptually identical in all respects to a sequence of data comprising a drum sound.

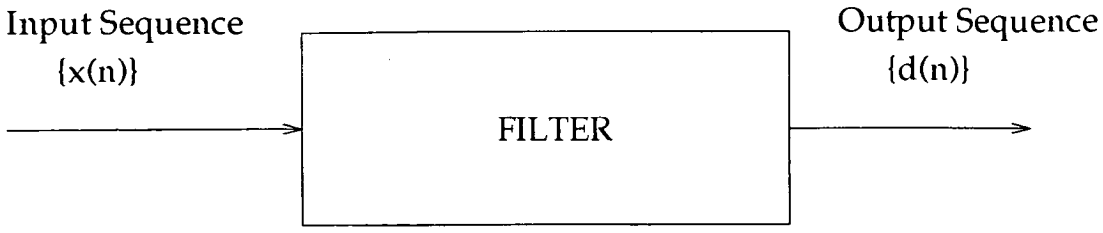


Figure 1.1: *Generic filter - block diagram*

In order to incorporate any nonlinear information found in the initial percussive attack into the model, it is necessary to use a filter structure that can exploit this information. Many nonlinear filter structures exist [29–31], but owing to their innate complexity, many of these must be trained adaptively [32–40], thus ruling out their employment in the prediction and regeneration of transient signals such as drum sounds where *a priori* training is essential.

Two nonlinear filter structures that can be configured non-adaptively are the Volterra filter, and the radial basis function network [33, 41–48]. Both of these have structures that enable implementation of the least squares algorithm, and thus optimal coefficient selection. In addition, the Volterra filter structure closely follows the representation of the signal seen using higher order statistical analysis. Any nonlinear feature discovered through such analysis is therefore likely to be reproducable using a Volterra filter.

The disadvantage of nonlinear filtering techniques - and the principle reason for their infrequent use - is their inherent computational complexity. This can be compensated for to some extent by the removal of unneccessary redundancies within the filter structure. Two techniques that perform this removal are the orthogonal least squares learning technique [49], and least squares noise thresholding. With the use of these algorithms, nonlinear filters can be reduced to a size where their use becomes a practical alternative to standard linear techniques.

1.4 Thesis Layout

As stated at the beginning of this chapter, the ultimate purpose of this work is to produce a functional model of a typical drum. Each subsequent chapter therefore describes a step taken towards the achievement of this objective.

The first of these steps is the establishment of a database of recorded sounds, sampled in an anechoic chamber, and taken from two different drums using a variety of sampling pathways. Given the transient nature of drum sounds, this large data base is necessary in order to obtain statistically significant results for any given section of the sound through the use of ensemble averaging. Chapter two details the acquisition and form of this database.

With an established database, it is possible to proceed with the second step of the work, the analysis of the drum sounds. In chapter three, bispectral analysis is used to detect the presence of phase coupling between different frequency components in the signal, and it is observed that certain bispectral patterns are indicative of nonlinear modulation between different frequency components. It is also shown that in theory, such modulation is independent of any artificial signal shaping due to the data acquisition process. By contrasting bispectra for data records gathered using acquisition paths with different phase and frequency characteristics, the transparency of quadratic phase coupling to the signal path is then confirmed in practice.

The task undertaken in Chapter 4 is to develop a model which incorporates the nonlinear information that has been shown to be present within the drum sound. To this end, two nonlinear filter structures are examined, the RBF network and the Volterra filter. Although these two filters have substantially different internal architectures, it can be shown that they share the ability to be configured using the linear least squares algorithm. Intensive testing is performed on all the data records gathered, and it is seen that the data records can be regenerated to a high degree of accuracy by using second order Volterra

filters with embedding dimensions of 12 or 16. Furthermore, it is observed that these models are unable to regenerate any points outwith the training data, and that the accuracy of the reproduced signal is maximised by using small training data lengths. It is then shown that following this highly nonlinear initial percussive attack, the sound can be modelled successfully using a simple linear AR filter with high embedding dimension.

A major drawback in the employment of the nonlinear filtering techniques described lies in the large number of coefficients employed by such models. Chapter 5 deals with the implementation of two methods for removing redundancy in the nonlinear filter structure. Using these two methods, the orthogonal least squares learning algorithm and the least squares noise thresholding coefficient reduction technique, it is shown that significant reductions in the size of the model can be achieved.

Chapter 6 concludes the thesis by presenting an overview of the work carried out in the previous chapters. Conclusions are drawn about the practical feasibility of implementing the model that has been developed, and these conclusions are used to highlight the possible areas for future work in this field.

Chapter 2

Theory and Analysis of Drum Sounds

2.1 Introduction

This chapter provides an overview of the fundamental acoustic theory which describes how sounds are generated by drums of all types. It also examines how this theory affects the ways in which different drums are used. The method used to sample the sound generated by two simple drums is examined in the light of this theory, and the resulting data records are compared with the theoretical model.

2.2 Acoustic Theory

Assuming a small initial disturbance, a vibrating circular membrane, radius a, fixed at the rim, can be represented (using a polar coordinate system with origin at the centre of the membrane, radius r, angle θ and time t) by the Helmholz equation [1],

$$\frac{\Theta \delta^2 R}{\delta r^2} + \frac{\Theta}{r}\frac{\delta R}{\delta r} + \frac{R}{r^2}\frac{\delta^2 \Theta}{\delta \Theta^2} + k^2 R\Theta = 0 \tag{2.1}$$

where $k = \omega/c$,c is a constant dictated by the surface density and tension of the membrane, and the amplitude of the displacement, $y(r, \theta, t)$ is dependent on functions of both dimensions ($\Theta(theta)$ and $R(r)$).

Assuming a standard solution of the form

$$y(r,\theta,t) = \Theta(\theta)R(r)e^{j\omega t} \tag{2.2}$$

the particular solution to the equation is

$$\Theta(\theta) = \cos(m\theta + \gamma) \tag{2.3}$$

$$R(r) = AJ_m(kr) + BY_m(kr) \tag{2.4}$$

where $J_m(kr)$ and $Y_m(kr)$ are first and second Bessel functions of order m. Physically, the result of this is standing wave vibrations at

$$f_{mn} = \frac{1}{2\pi}\frac{j_{mn}c}{a} \tag{2.5}$$

where j_{mn} are the zeros of the first Bessel Function. Fig (2.1) shows some of the resulting modes of vibration. These modes are designated by the ordered pair (m,n) where the integer m determines the number of radial nodal lines, and the integer n determines the number of azimuthal nodal circles. It should be noted that $n = 1$ is the minimum allowed value of n, and corresponds to a mode of vibration with one nodal circle occurring at the boundary of the membrane (where $r = a$). It can be seen from table (2.1) that none of the overtones are at harmonic frequencies of the fundamental (0,1) mode.

$f_{01} = 1.0f_{01}$	$f_{11} = 1.593f_{01}$	$f_{21} = 2.135f_{01}$
$f_{02} = 2.295f_{01}$	$f_{12} = 2.917f_{01}$	$f_{22} = 3.500f_{01}$
$f_{03} = 3.598f_{01}$	$f_{13} = 4.230f_{01}$	$f_{33} = 4.832f_{01}$

Table 2.1: *Relative frequencies of a circular membrane*

When considering such a circular membrane as a simple model of a drum, certain other factors have to be taken into effect. In practice, when a drum is struck, there is a damping effect due to the air [50]. As well as hastening the decay of the oscillations, this also *slows* the vibrations, and thus the overtones are actually at lower frequencies than those predicted by theory. In

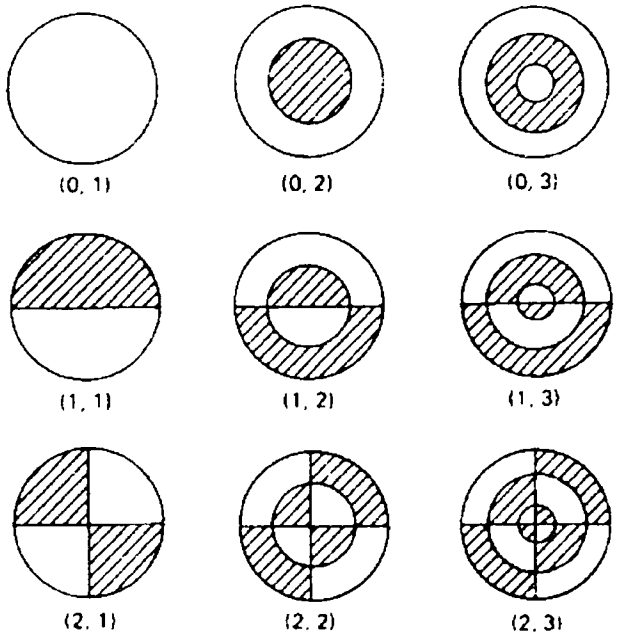


Figure 2.1: *Modes of vibration for a circular membrane [1].*

addition, the damping effect of the air is more pronounced on higher frequency oscillations and radial oscillations (i.e. those with nodal circles), leading to a preponderance of (n,1) modes. The principal contributors to the sound of a drum are therefore the (1,1), (2,1) and (3,1) modes, although many more are present in the initial percussive attack. Although this percussive attack lasts for only a few milliseconds, it is during this time that the initiating nonlinearity causing the displacement of the drum membrane occurs, and the form that this displacement takes dictates the shape of the ensuing sound. Furthermore, the transient high frequency modes provide the characteristic "thump" that is a fundamental property of all drum sounds. Succesful synthesis of this part of the drum sound is crucial to any model [5]. It is therefore necessary to consider in a little more detail the effect of striking a drum membrane.

The centre of the drum is a universal node, and hitting this would therefore excite all modes equally weakly, resulting in a short sharp pitchless thump. Hitting on the node of an (n,2) mode, however, would damp down that mode

while exciting others, leading to a pitched note. A further influence on the sound produced is the way in which the drum is hit. Striking a drum with a hard stick leads to a sharp depression in the membrane surface, and therefore a strong high frequency content. Striking with a large soft ball, on the other hand, leaves the membrane with a shallow depression, and leads to a predominance of low frequency modes, and a "booming" sound. The importance of these techniques can be demonstrated by comparing the bass drum and the orchestral timpani [50].

The bass drum has two heads of equal diameter stretched on either side of a wide shallow barrel. Striking one head results in a compression of the air inside the barrel, and thus a corresponding outwards displacement of the other membrane. By adjusting the two membranes to be at slightly different tensions from one another, the radiating frequencies are effectively heard as unpitched. The drum is struck with a soft headed stick and the striking position is about half way between the rim and the centre of the head. This contributes to the "unpitching" of the radiated frequencies, whilst also imparting most of the energy to low frequency modes, resulting in a "bass" sound. The timpani, whilst also struck by a soft headed stick, is designed to deliver a pitched note, and therefore the striking position is only about a third of the distance from the rim to the centre. This suppresses non-harmonic $(n, 2)$ modes, whilst accentuating harmonic ones.

In common with almost all drums, both the timpani and the bass drum have a closed space behind the drum membrane. This further affects the way the membrane vibrates, since pressure changes occur in this space as the volume of the entrapped gas is altered by the vibration of the membrane. It has been shown [1] that this can lead to changes both in the relative amplitude of different modes of vibration, and also in their relative frequencies. In particular, this has the effect of increasing the number of high frequency modes of vibration, and accentuating the fundamental mode. In the case of drums such as the bass

drum in which this space is enclosed by two skins, this pressure modulation is further complicated by the vibrations of the second, detuned drumhead.

2.2.1 Modelling

In order to construct a theoretical model with characteristics analogous to those of the drum, it would be necessary to make a mathematical model of each part of the drum, from the drumsticks right through to the drum barrel, and then to model the interfaces between these parts. There has been much research into this form of modelling [8–10], which, when successful, has often been able to recreate sounds more completely than other types of modelling. As can be seen from the preceeding section, a mathematical model would be extremely complicated even in the case of the simplest drum - particular difficulties being the interface between the drum membrane and drum stick, and the closed-system pressure modulation effect. Additionally, physical models suffer from the disadvantage of extreme inflexibility, any change to a system necessitating a corresponding remodelling of that part of the system.

One alternative to the physical modelling of a drum, is the construction of a "black box" model, i.e. one that reproduces the output signal without attempting to mimic the mechanics of its production. Early attempts at this, which typically involved the use of a noise shaping filter, have been criticised for their predictability, and their lack of variation with respect to amplitude and striking frequency. More importantly, despite having power spectra similar to those found in natural drum sounds, such models have never been convincing acoustically [4,8]. This implies that further information about the signal must be incorporated into any successful model.

2.3 Data Acquisition

2.3.1 Practical Considerations

While it is always important to ensure that any data collected for analysis are of high quality, where that analysis involves the testing for possible nonlinear coupling within a signal it is vital that the equipment used does not affect the data in any way. In practice, some level of filtering is inherent in the sampling process so it is necessary first to minimise this, and second, to know how it effects the data.

Most measurements are carried out in sound fields whose characteristics vary between that of the *free field* corresponding to a perfectly "dead" room or 100% open space, and that of the *diffuse field* corresponding to a perfectly reverberant room. As the qualities of a high amplitude acoustic source such as a drum can be shaped significantly by the reverberant properties of the room in which the source is sited, it was necessary to take measurements in an environment with a response as close as possible to that of a free field. In practice, this meant using an anechoic chamber. This had the added advantage of ensuring an extremely low background noise level, as the only noise source present within the chamber was the drummer himself! The sounds were recorded with a B&K 4133 Half-inch Condenser Microphone. This microphone is designed for free field measurements, and has flat frequency and linear phase responses across the whole audio spectrum (20 Hz to 40 kHz) when operating within pressure levels of 32 dB to 160 dB (fig(2.2)) [2]. At higher frequencies, the diffraction of sound waves on microphones produces an appreciable change in the resulting sound pressure acting on the microphone diaphragm. This difference, known as free field correction, depends on the orientation of the microphone with respect to the direction of propagation of the sound. For microphones intended for free field work (such as the 4133), it is possible to give the diaphragm resonance critical damping so that the normal incidence free-

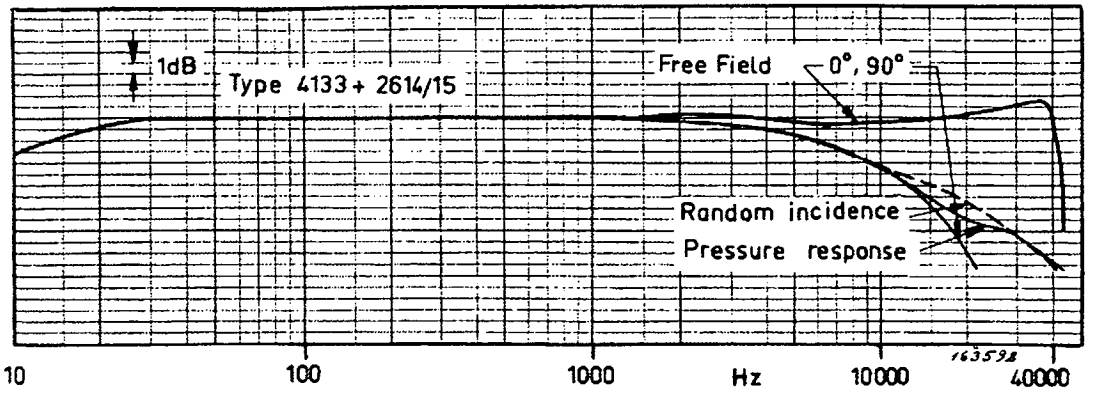


Figure 2.2: *Frequency characteristics of the 4133 Half-inch Microphone [2].*

field corrections are compensated for up to frequencies well above the point of resonance, in order to obtain the flattest possible frequency response (see fig(2.2)). The microphone was powered by a B&K 2608 measuring amplifier [51]

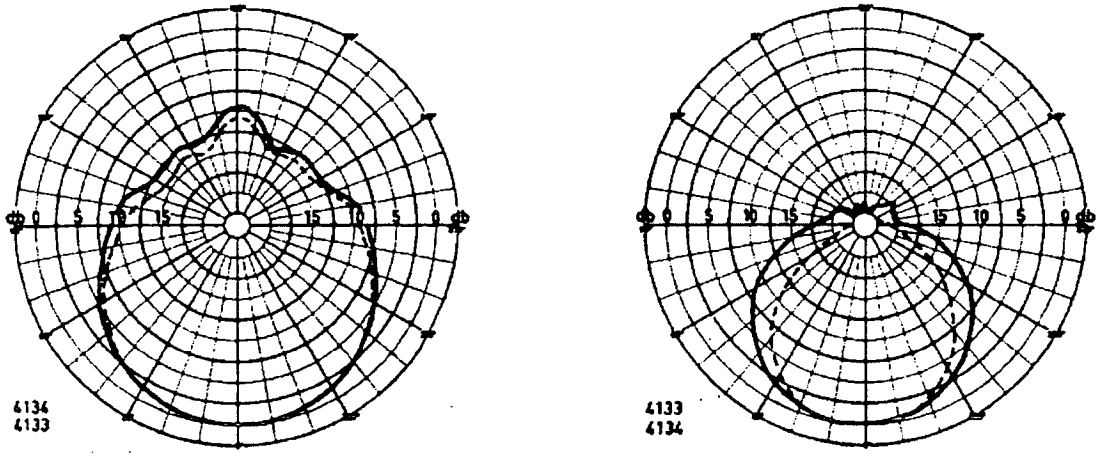


Figure 2.3: *Directional characteristics of B&K 4133 and 4134 microphones at 10 kHz and 20 kHz [2].*

in series with a B&K 2706 power amplifier [52], and the output of the amplifier was set to be precisely +/- 10v peak-to-peak. This output was then sampled with a Blue Chip ADC-42 sampling board [53] installed in a portable 486 P.C. (used to remove the need for a mains supply, and thus eliminate mains and transformer interference at the sampling point), giving 12 bit resolution. In

order to ensure that the data collected were compatible with international audio standards, a sampling rate of 44.096 kHz was desirable. This sampling rate was not, however available internally, so an external clock was built. As a control, further records were taken using the internally generated 33.33 kHz clock. The two drums to be sampled were a "Mid Tom" and "High Tom" from the "Pearl Export" series, currently one of the most popular makes of drum in Britain [7]. As this type of drum is said to have a significant frequency response up to approximately 12 kHz [3], both sampling frequencies left plenty of headroom for intial frequency transients during the percussive attack.

2.3.2 Results

Preliminary tests showed that the sound generated by the drums remained audible within the anechoic chamber for approximately three seconds. Outwith the chamber, the sound was detectable for a shorter period, due to the higher noise threshold. It was also evident that, in the hands of a professional drummer, the drum could produce a range of distinct sounds, depending on where and how hard the drum was hit. Hitting the drum hard resulted in a fast, high amplitude initial displacement to the drum membrane, and a correspondingly accentuated percussive attack. Striking it softly produced a slower, low amplitude displacement, and a less exaggerated attack. Furthermore, initial data showed that microphone position had a significant effect on the signal recorded. The position eventually chosen - to ensure both a free field response and accurate recording of the sound as heard - was approximately 1.5 metres away from the drum membrane. In order to simplify analysis of the signal, data records were limited in duration to 1.25 seconds and the drummer attempted to maintain a constant style from record to record. Data records were collected for each drum at the two different sampling rates, in conjunction with appropriate anti-aliasing filters. For the 33.3 kHz sampling rate, a 6 pole maximally flat Butterworth filter with a 12 kHz cutoff frequency was used, and for the 44.096 kHz rate, data were gathered using two different filters - a

Butterworth and a Bessel - both 6 pole, with cutoff at 16 kHz. The Bessel filter has the advantage of an approximately linear phase characteristic which was necessary to demonstrate some of the theory presented in chapter 3. Twenty data records were gathered for each case. The contents of the data base thus created can be seen in table (2.2).

Drum	Sampling rate	Filter type (cutoff)	Number of records
Low Tom	44.096 kHz	Butterworth (16kHz)	40
Low Tom	44.096 kHz	Bessel (16kHz)	20
Low Tom	33.333 kHz	Butterworth (12kHz)	20
Mid Tom	44.096 kHz	Butterworth (16kHz)	20
Mid Tom	44.096 kHz	Bessel (16kHz)	20
Mid Tom	33,333 kHz	Butterworth (12kHz)	20

Table 2.2: *Final database of recorded drum sounds.*

Figures(2.4 - 2.9) show some aspects of a typical drum sound. Each successive figure 'zooms in' on parts of the sound, in order to identify various characteristics. The whole record is displayed as captured in figure (2.4). Note the amplitude scale varies between 0 (equivalent to -10 volts) and 4095 (equivalent to +10 volts). Looking at the whole sound, one feature that is instantly visible is its rapid attenuation. A magnified plot of the beginning of the same sound, showing only the first 5000 samples (0.1 s), is shown in figure (2.5) and here some of the different constituent parts of the drum sound become visible. The figure shows ten periods of the fundamental frequency, (which is approximately 98 Hz) and upon closer inspection it can be seen that the periods increase slightly with time - evidence of a gradual lowering of the fundamental frequency. Also visible are small rapid oscillations due to the higher overtones of the signal. To examine these in more detail, figure (2.6) shows the very beginning of the sound, looking at only 1000 samples (22.68 ms). In this diagram, various features are worth noting: at the beginning of the signal, the moment of contact between the stick and the membrane is visible, and it can be seen that this is followed immediately by a large negative amplitude

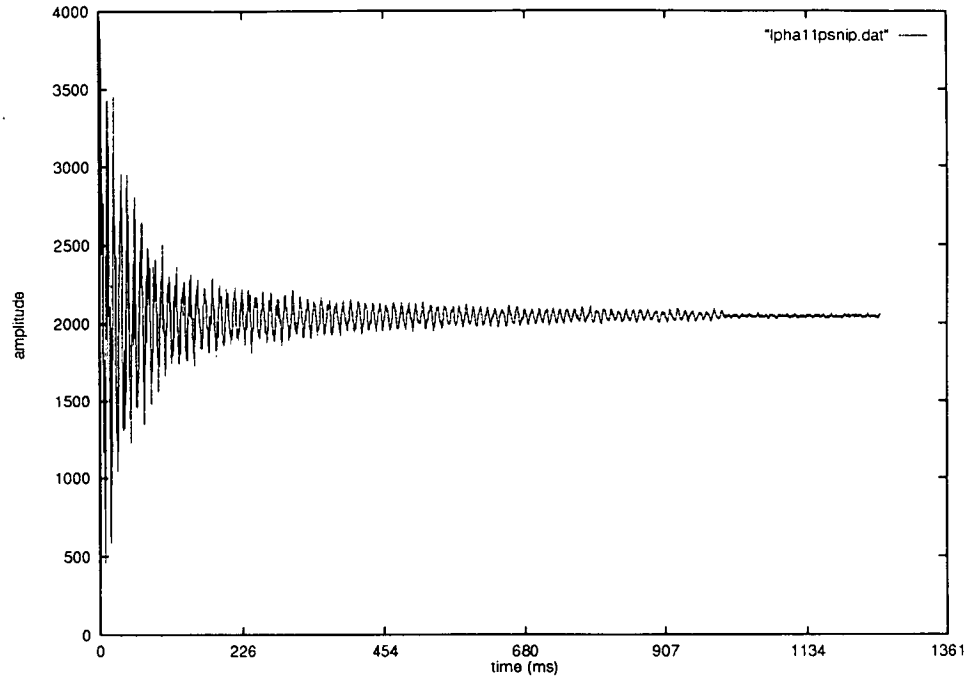


Figure 2.4: *The whole sound: 44.1 kHz record of Mid Tom (Butterworth filter).*

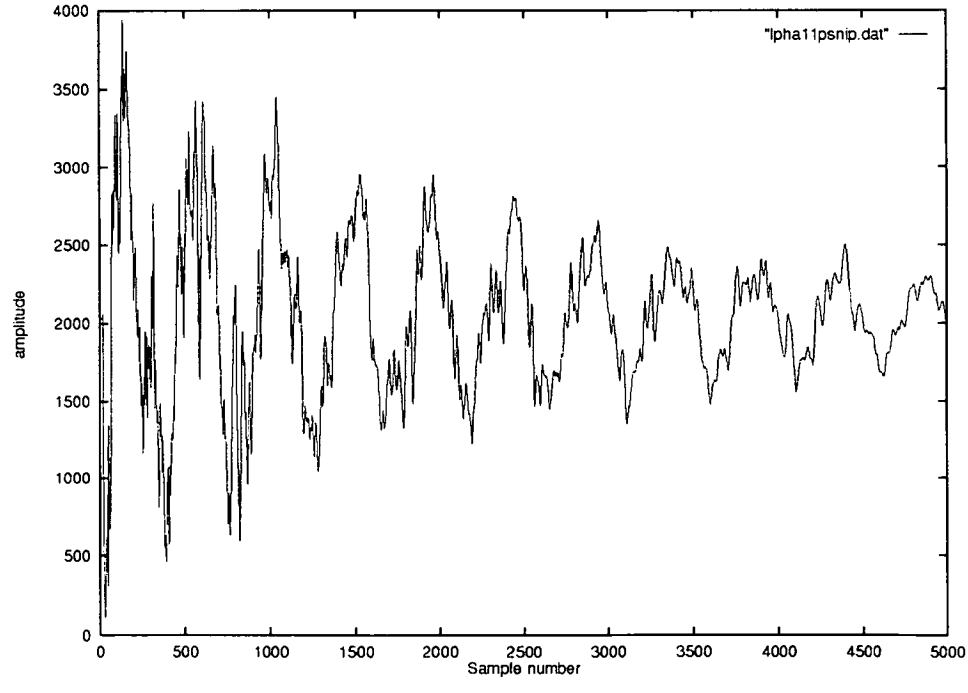


Figure 2.5: *First 5000 points: 44.1 kHz record of Mid Tom (Butterworth filter.)*

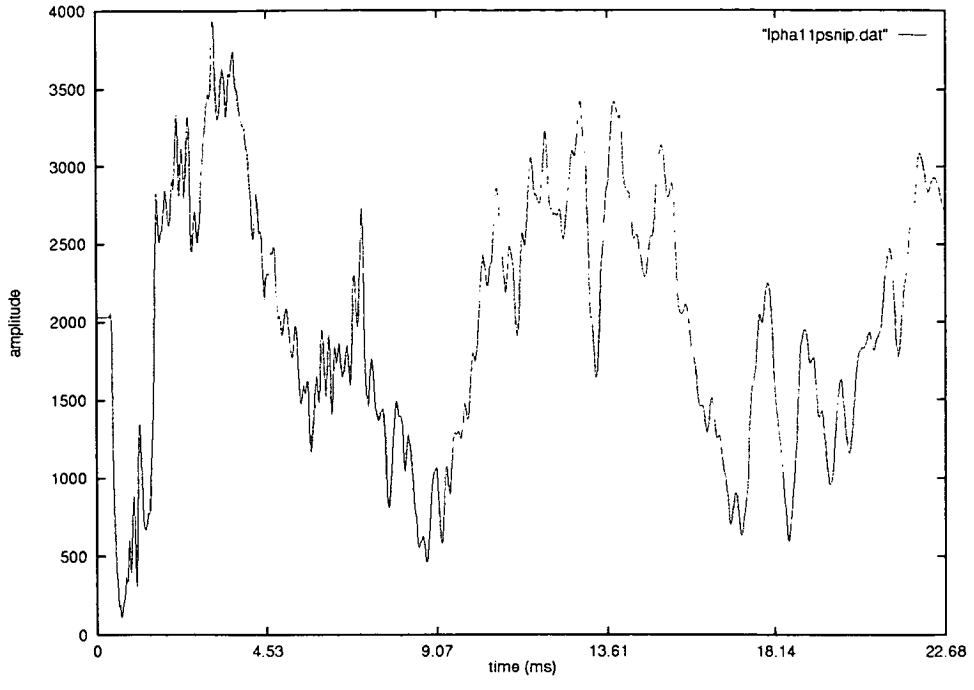


Figure 2.6: *First 1000 points (22.68ms): 44.1 kHz record of Mid Tom (Butterworth filter).*

spike. This spike has a much higher frequency than the subsequent 2.5 periods of the fundamental frequency visible on the figure, and this suggests that the spike is a result of the displacement caused by the contact of the drum stick with the membrane. Examination of all the data records gathered shows that this typically lasts for between one and two milliseconds. At the end of this time, it seems probable that the contact between stick and skin ends, and the fundamental frequency of the membrane asserts itself.

The signal immediately following this negative spike has a large high frequency content which can be seen to have attenuated significantly within one period of the fundamental frequency. These transient high frequency terms consitute the percussive attack detailed earlier in section 2.2. The attenuation of the higher frequency compónents of the sound continues for the duration of the signal, all be it in a less dramatic fashion. This can be seen in figure (2.7) which examines the same sound 5000 samples later (zooming in on the 113ms - 181 ms section from figure (2.4)). This diagram also serves to highlight the

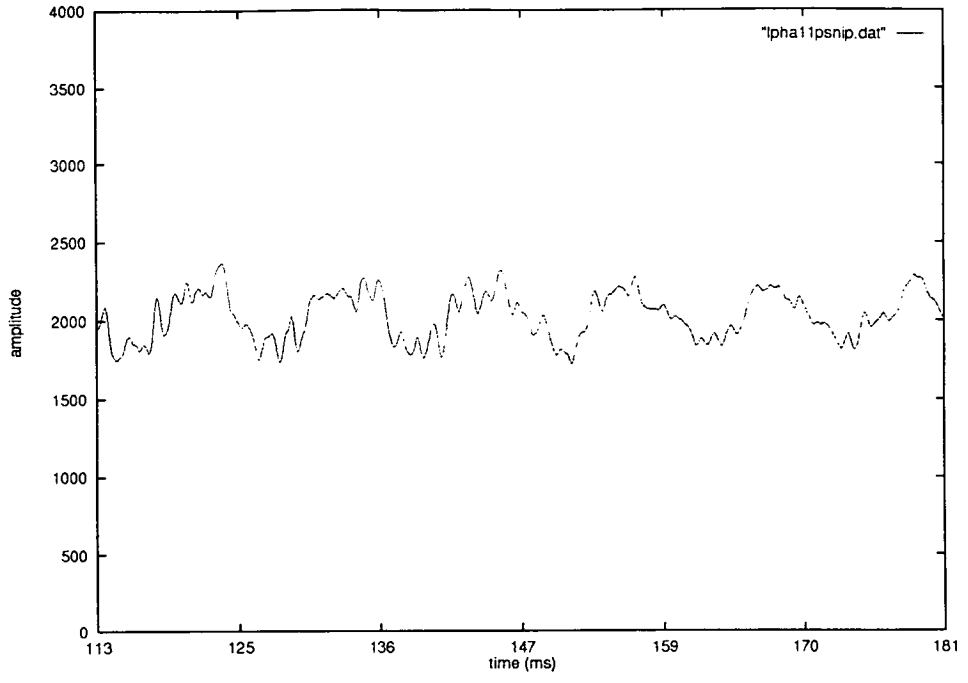


Figure 2.7: *5000 samples (113 ms) later: 44.1 kHz record of Mid Tom (Butterworth filter).*

rapidity of attenuation associated with the sound. In particular, it can be seen that the high frequency content of the signal has almost vanished.

In order to corroborate the visual evidence just discussed, it is necessary to examine the power spectrum of these two parts of the drum sound. Figure (2.8) shows a 512 point fast Fourier transform (FFT) of the initial 11 ms of the sound (as seen in figure (2.6)). This spectrum shows a large peak at the fundamental frequency, with numerous high amplitude peaks at frequencies up to 12.5 kHz. Figure (2.9) shows an equivalent 512 point FFT for the section of the sound shown in figure (2.7), 113 ms later. Although the peak at the fundamental frequency of the drum remains present, a comparison with the previous diagram verifies the reduction in amplitude of the high frequency terms in this subsequent section of the signal.

Figure (2.10) shows the results of a 16384 point FFT for the first 0.37 seconds of data, ensemble averaged over twenty different data records. The length of the FFT precludes the accurate portrayal of transient frequencies [54] - i.e.

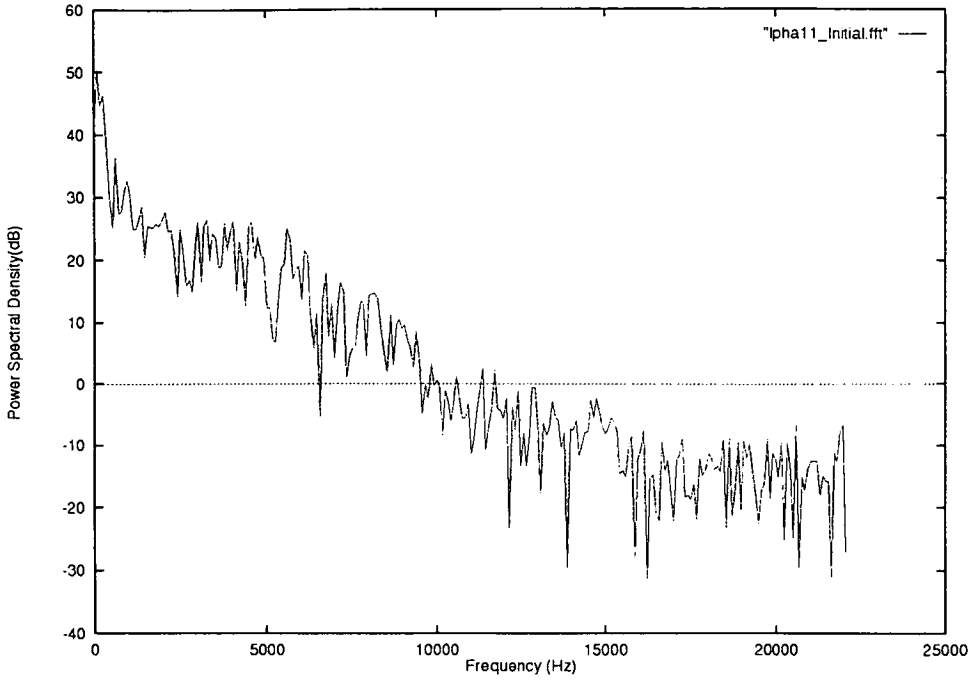


Figure 2.8: *First 512 points: Power spectrum of Mid Tom (Butterworth filter).*

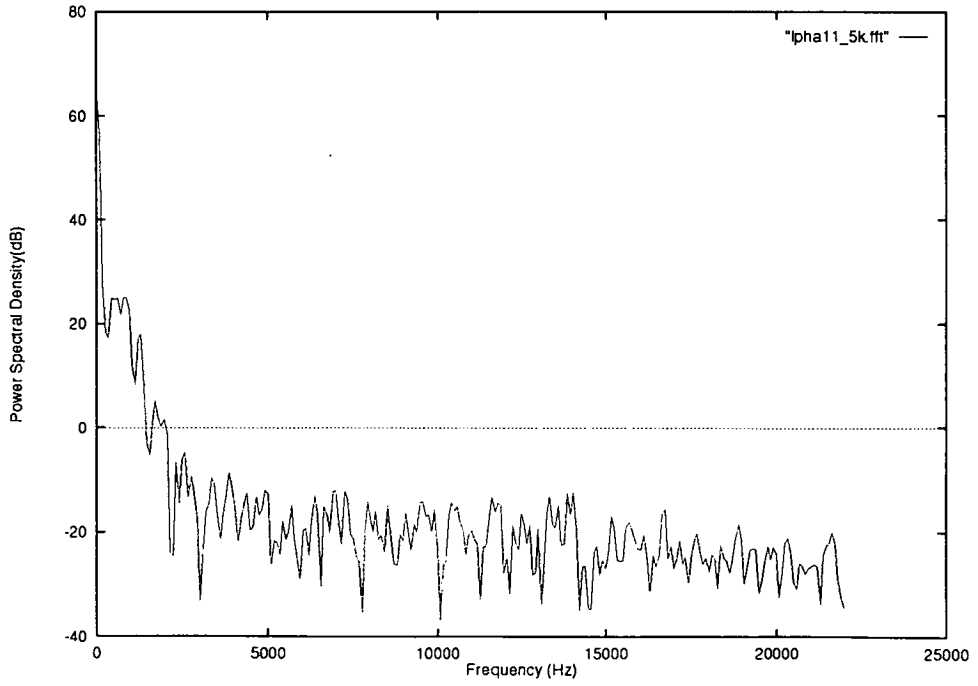


Figure 2.9: *113 ms later:512 point power spectrum of Mid Tom (Butterworth Filter).*

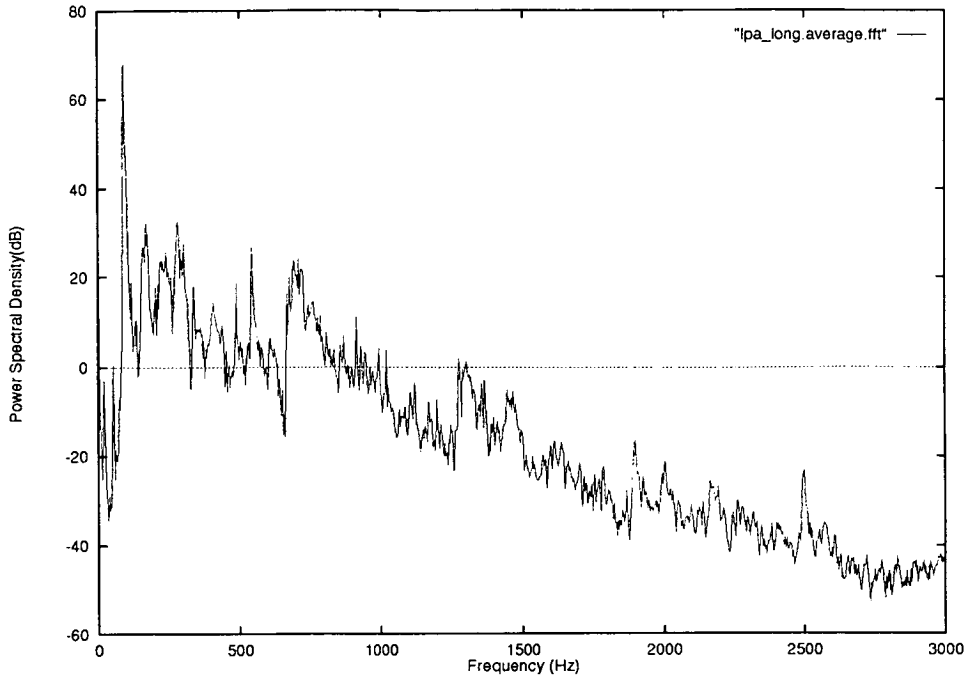


Figure 2.10: *16384 point average power spectrum of 44.1 kHz sampled Mid Tom (Butterworth Filter).*

peaks on the graph indicate both a high amplitude and a relatively long time duration at the frequency in question - but, in conjunction with the averaging, provides an exact location for the fundamental frequency (87.5 Hz +/- 0.5Hz), and can be used to discern the frequency peaks for some of the overtones as predicted by the theoretical model. Assuming that the largest frequency is a result of exitation due to the (1,1) mode of vibration, then Table (2.1) indicates that further notable frequency components should be found at f_{01} = 55 Hz, and f_{21} = 117.3 Hz. In practice, peaks can be seen at frequencies as high as 2.5 kHz, with large amplitudes up to 800 Hz. Although small peaks can be seen at 51 Hz and 113 Hz, they do not represent a large contribution to the total signal power. It is evident therefore, that the combined effect of the two detuned drumheads, in conjunction with the trapped volume of air within the drum, is enough to effect a significant change in the principal overtones of the sound, when compared to those predicted by the theory discussed earlier in this chapter.

Examination of a number of records shows that each drum sound maintains the overall characteristics discussed above, although the specifics vary from record to record. The initial 1000 points of a number of different records are shown in figures(2.11) to (2.14). Figure (2.11) shows the percussive attack

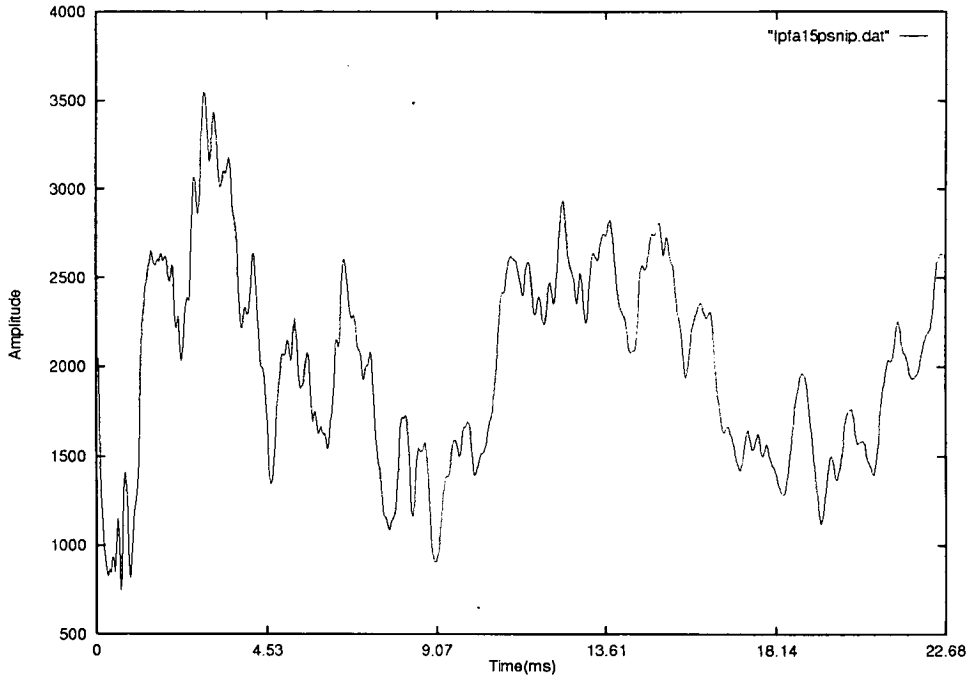


Figure 2.11: *First 1000 points (22.68 ms): 44.1 kHz record of Mid Tom (Butterworth Filter) - second example.*

of a second data record gathered using a 44.1 kHz sampling frequency and a Butterworth anti-aliasing filter. Once again, the initial negative spike indicative of contact with the drumstick can be seen, and, when comparing the signal to that shown in figure (2.6), it can be seen that the first two periods of the fundamental frequency are shaped similarly. This similarity is also present in records gathered using the Bessel anti-aliasing filter, an example of which is shown in figure (2.12). Figure (2.13) shows the initial percussive attack of a sound gathered using the lower 33.33 kHz sampling frequency. Although the lower sampling frequency leads to slightly less resolution, the underlying shape still demonstrates the visual characteristics of a drum sound, with the high frequency terms attenuating rapidly during the course of one period of the fundamental frequency.

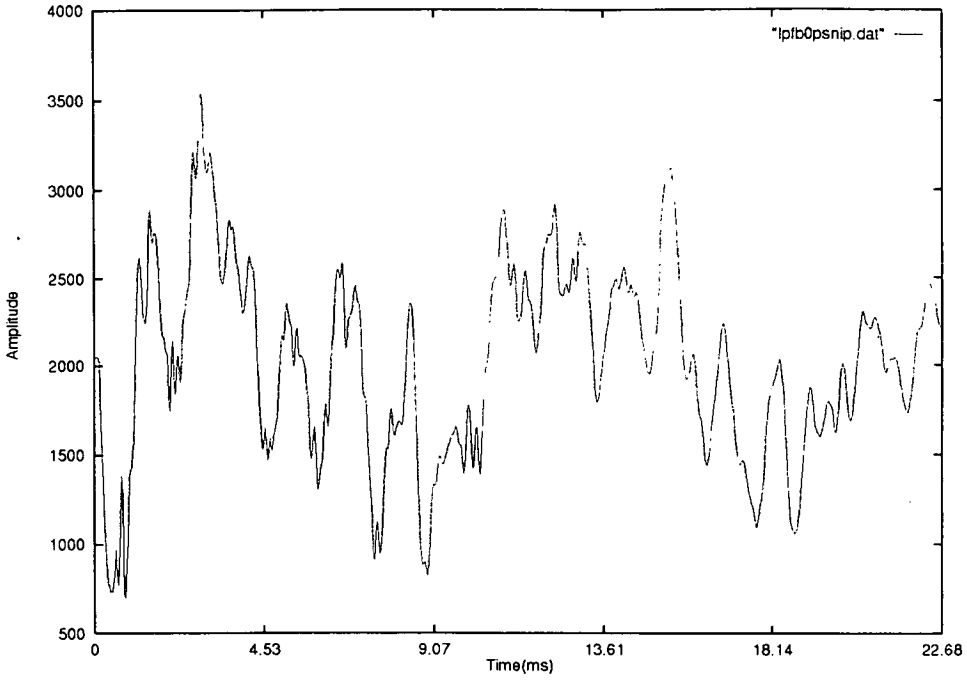


Figure 2.12: *First 1000 points (22.68 ms): 44.1 kHz record of Mid Tom (Bessel Filter).*

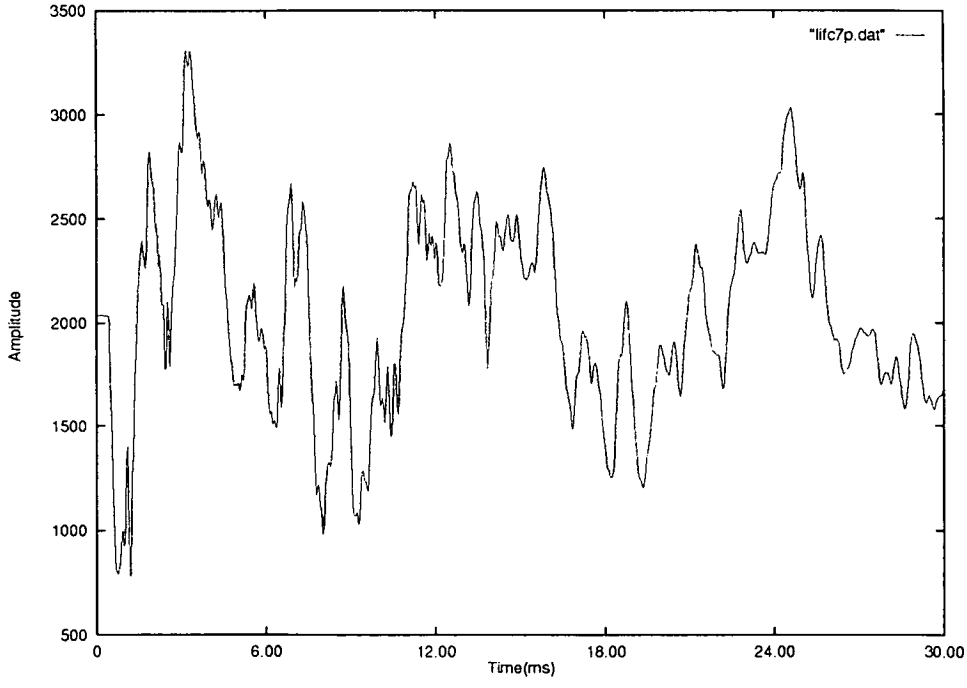


Figure 2.13: *First 1000 points (22.68 ms): 33.33 kHz record of Mid Tom (12 kHz Butterworth Filter).*

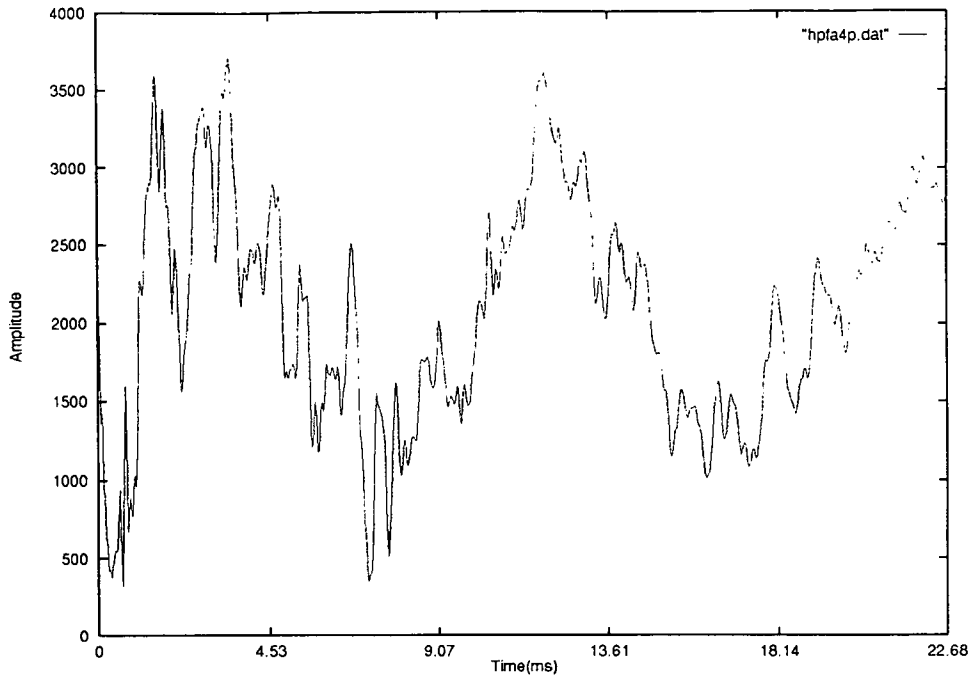


Figure 2.14: *First 1000 points (22.68 ms): 44.1 kHz record of High Tom (Butterworth Filter).*

Perhaps the most interesting case to compare with the original example is one in which the data were gathered from a different drum. Figure (2.14) shows the initial percussive attack of a record generated using the High Tom. Despite the different source, this record *still* has all the features associated with data records gathered from the Low Tom, although the fundamental frequency of the signal is at a higher frequency. From these comparisons it can be seen that all drum sounds, whilst often varying substantially in detail, contain certain features associated with their generation. In particular, the intial negative spike and the transient high frequencies present during the percussive attack are always clearly visible.

Examination of the average power spectrum for each different set of records, shown in figures (2.15) to (2.17) confirms the similarity between them. Figure (2.15) shows the power spectrum for data gathered using the Bessel filter, and figure (2.16) shows the power spectrum for data gathered at the slower 33.33 kHz sampling rate. Both these spectra show the presence of the same

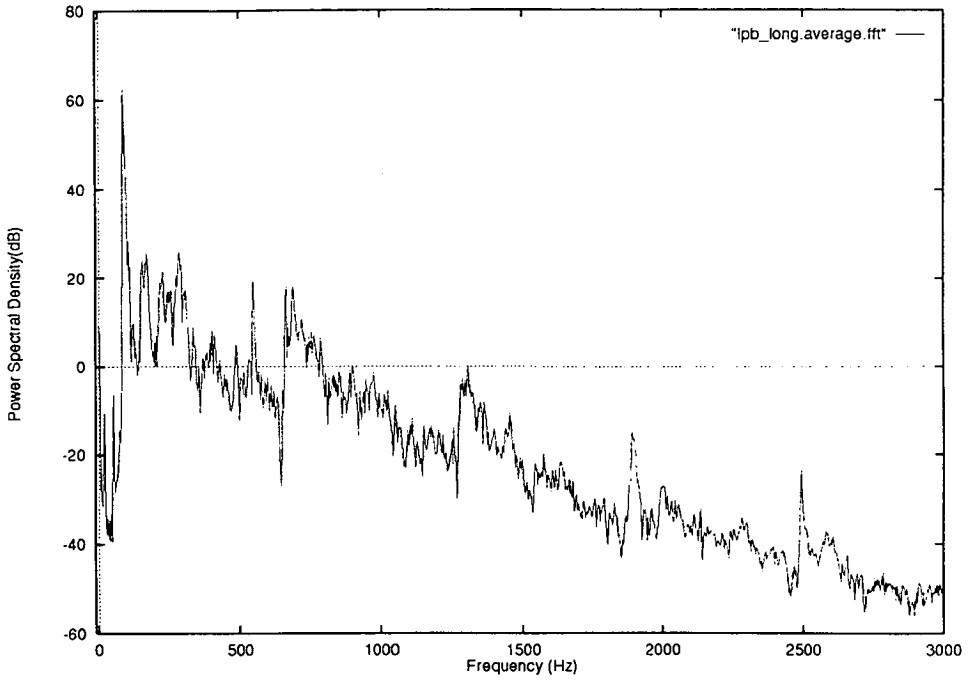


Figure 2.15: *16384 point average power spectrum of 44.1 kHz sampled Mid Tom (Bessel Filter).*

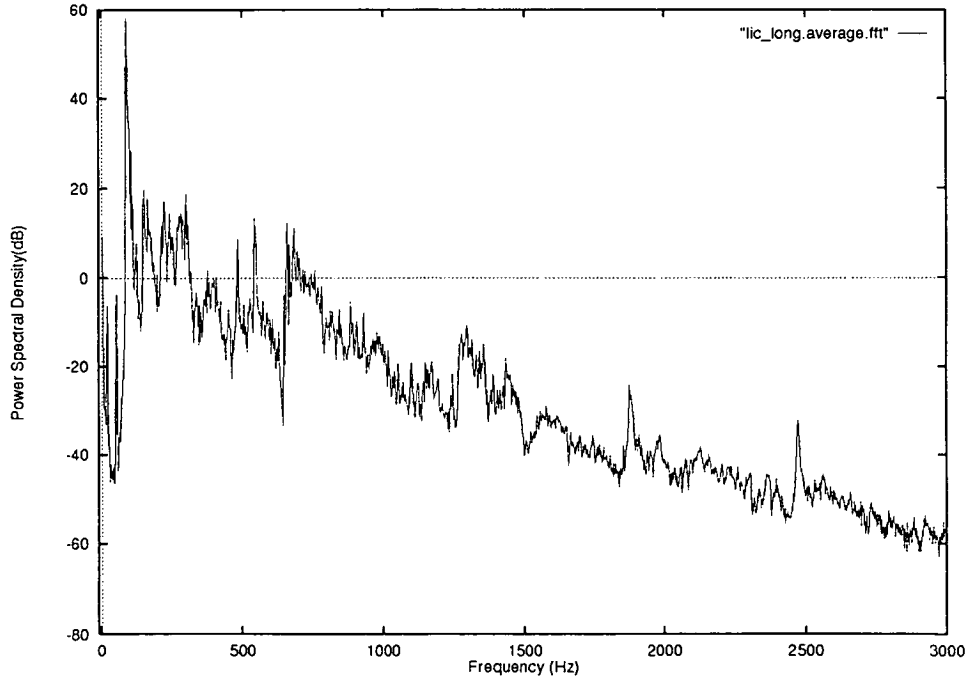


Figure 2.16: *16384 point average power spectrum of 33.33 kHz sampled Mid Tom (12 kHz Butterworth Filter).*

frequency peaks seen in fig(2.10) quite distinctly. The power spectrum for the High Tom data records, figure (2.17), also shows a similar spectral distribution, although the fundamental frequency of the sound is shifted up to 94 Hz (approximately a semitone higher), and all other frequency peaks are also shifted upwards in frequency.

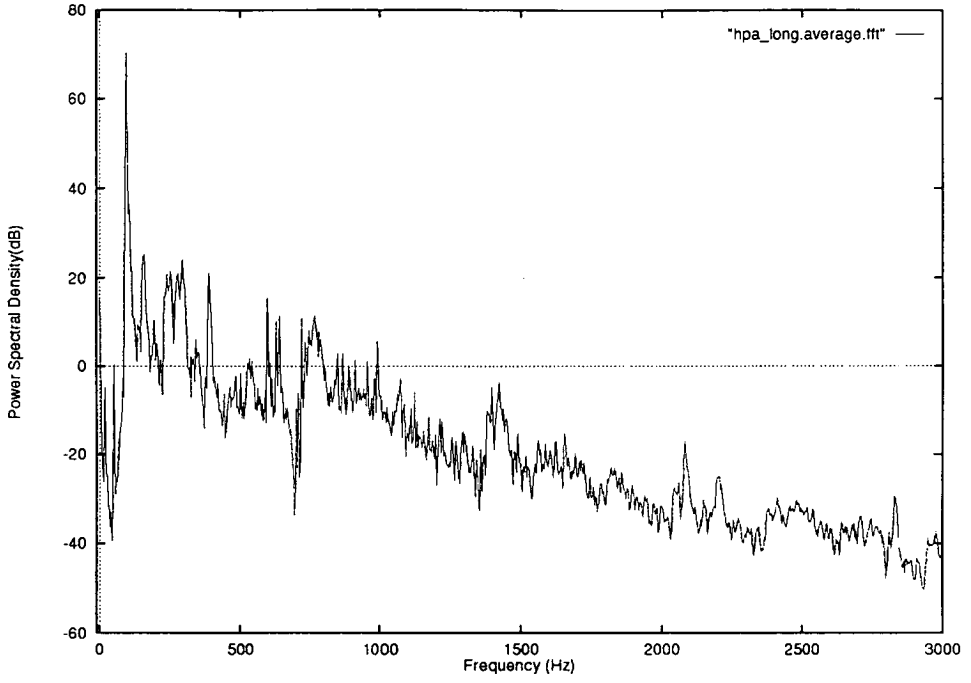


Figure 2.17: *16384 point average power spectrum of 44.1 kHz sampled High Tom (Butterworth Filter).*

2.4 Conclusion

In this chapter, the theory underlying the generation of acoustic percussion sounds has been examined. It has been shown that the apparently simple set of equations governing the generation of these sounds is in practice made mathematically intractable by the physical construction of such drums, and both the nonstationarity and nonuniformity of the initial excitation process. Following on from this, the various considerations that have to be accounted for when sampling such signals have been presented, and solutions to some

of the practical problems described. The resultant data aquisition process has been detailed, and some typical examples of recorded data have been exhibited. A simple analysis of the data records has highlighted the structure of these sounds, and shows that while the signals vary greatly from one recorded strike to the next, collectively the sounds also follow an underlying pattern, both in the frequency domain, and in the time domain. It has been observed that, as predicted in theory, the signals are highly nonstationary and extremely complicated, especially during the initial percussive attack. This percussive attack has also been shown to be of very short duration, and presents all the problems inherent in transient signals with respect to modelling and analysis.

The next chapter looks at a form of signal analysis that examines the higher order statistics of the signal. It will be shown that further information about drum sounds is revealed using this form of analysis. This information will prove to be important in the prediction and regeneration of such sounds.

Chapter 3

Higher Order Spectral Analysis

3.1 Introduction

Analysis of the power spectral density (PSD) of discrete time and stochastic processes has been a fundamental tool in digital signal processing for some time. In power spectrum estimation, the process under consideration is treated as a superposition of statistically uncorrelated harmonic components, and the distribution of power amongst these frequencies is then estimated. Phase relations between these components are suppressed, and only linear mechanisms governing the process are visible. The PSD is obtained by taking a Fourier transform of the autocorrelation function of the process, and the information it contains is essentially that which is present within the autocorrelation sequence. This autocorrelation function can be regarded as the second order term in an infinite cumulant series pertaining to the process. By examining higher order terms in this series and their Fourier transforms (higher order spectra), more information about processes can be revealed [55–58]. In particular, higher order spectra contain information about phase relations between different frequency components in a signal, and can be used to reveal the presence of nonlinearities. This chapter outlines the theory behind higher order spectral analysis, and goes on to examine in some detail the properties of the third order cumulant sequence and bispectrum. Special attention is paid to the detection of quadratic phase coupling, and the use of this information in signal analysis. It is shown that any such phase coupling remains unaffected by any

linear or nonlinear phase shifts incurred in sampling or processing of data. It is also shown that phase coupling information can be used to differentiate between linear and nonlinear signal interactions, and can be used to confirm or deny dependencies between different signals and spectral peaks. These properties are then used to analyse the acoustic data. It is shown that the initial percussive attack of the drum sound contains a significant amount of quadratic phase coupling, and that this phase coupling is linked to the generation of the high frequency terms peculiar to the percussive attack of a drum. Furthermore, it is shown that as the signal attenuates in time, these terms vanish, and the underlying process becomes linear in character.

3.2 Theory

3.2.1 Definitions

Given a set of n real random variables ($x_1, x_2, \ldots, x_n$) with joint characteristic function

$$\Phi(\omega_1, \omega_2, \ldots, \omega_n) = E\left\{e^{j(\omega_1 x_1 + \omega_2 x_2 + \cdots + \omega_n x_n)}\right\}, \tag{3.1}$$

their joint moments of order $r = k_1 + k_2 + \cdots + k_n$ are defined [59,60] as

$$m_{k_1 \cdots k_n} \equiv E\left\{x_1^{k_1} x_2^{k_2} \cdots x_n^{k_n}\right\} = (-j)^r \left.\frac{\partial^r \Phi \omega_1, \omega_2, \ldots, \omega_n}{\partial \omega_1^{k_1} \partial \omega_2^{k_2} \cdots \partial \omega_n^{k_n}}\right|_{\omega_1 = \omega_2 = \cdots = \omega_n = 0} \tag{3.2}$$

and the joint cumulants of order r of the same set of random variables are given by

$$C_{k_1 \cdots k_n} \equiv (-j)^r \left.\frac{\partial^r \ln \Phi(\omega_1, \omega_2, \ldots, \omega_n)}{\partial \omega_1^{k_1} \partial \omega_2^{k_2} \cdots \partial \omega_n^{k_n}}\right|_{\omega_1 = \omega_2 = \cdots = \omega_n = 0}. \tag{3.3}$$

The joint cumulants can be expressed in terms of the joint moments of the random variables. For a stationary process, the moment sequences of the process are related to its cumulants as follows:

Mean:

$$E\{X(k)\} = m_1 = c_1 \tag{3.4}$$

Autocorrelation (or second order cumulant) sequence:

$$E\{X(k)X(k+\tau_1)\} = m_2(\tau_1) = c_2(\tau_1) \tag{3.5}$$

Third order moment or cumulant sequence:

$$E\{X(k)X(k+\tau_1)X(k+\tau_2)\} = m_3(\tau_1,\tau_2) = c_3(\tau_1,\tau_2) \tag{3.6}$$

Fourth order moment sequence:

$$E\{X(k)X(k+\tau_1)X(k+\tau_2)X(k+\tau_3)\} = m_4(\tau_1,\tau_2,\tau_3) =$$
$$c_4(\tau_1,\tau_2,\tau_3) + c_2(\tau_1)c_2(\tau_3-\tau_2) + c_2(\tau_2)c_2(\tau_3-\tau_1) + c_2(\tau_3)c_2(\tau_2-\tau_1) \tag{3.7}$$

It is interesting to note that, assuming a zero mean, the second and third order cumulants are identical to the respective moments. In order to generate the fourth order cumulant sequence, however, both the fourth order moment and autocorrelation sequences are needed.

The N^{th} order spectrum of the process $\{X(k)\}$ is defined to be the Fourier transform of its N^{th} order cumulant sequence [59,61,62]. This is an extension of the Wiener-Khintchine theorem to higher order spectra:
Power spectrum:

$$C(\omega_1) = \sum_{\tau_1=-\infty}^{+\infty} c_2(\tau_1)e^{-j(\omega_1\tau_1)} \tag{3.8}$$

Bispectrum:

$$C(\omega_1,\omega_2) = \sum_{\tau_1=-\infty}^{+\infty}\sum_{\tau_2=-\infty}^{+\infty} c_3(\tau_1,\tau_2)e^{-j(\omega_1\tau_1+\omega_2\tau_2)} \tag{3.9}$$

Trispectrum:

$$C(\omega_1,\omega_2,\omega_3) = \sum_{\tau_1=-\infty}^{+\infty}\sum_{\tau_2=-\infty}^{+\infty}\sum_{\tau_3=-\infty}^{+\infty} c_4(\tau_1,\tau_2,\tau_3)e^{-j(\omega_1\tau_1+\omega_2\tau_2+\omega_3\tau_3)} \tag{3.10}$$

3.2.2 General Properties

The variance of a process is a measure of its spread, or divergence from the mean, and, for a zero mean signal (i.e. one in which $E\{X(k)\} = 0$) can be displayed at lag zero ($\tau_1 = 0$) of its autocorrelation function:-

$$\gamma_2^x = c_2^x(0) = E\left\{X^2(k)\right\}. \tag{3.11}$$

In a similar fashion, higher order cumulant sequences contain information about skew and kurtosis. Skew is a measure of the asymmetry of a distribution about its mean:

$$\gamma_3^x = c_3^x(0,0) = E\left\{X^3(k)\right\} \tag{3.12}$$

and kurtosis is measure of asymmetrical tightness, or the spread on each side of the mean:

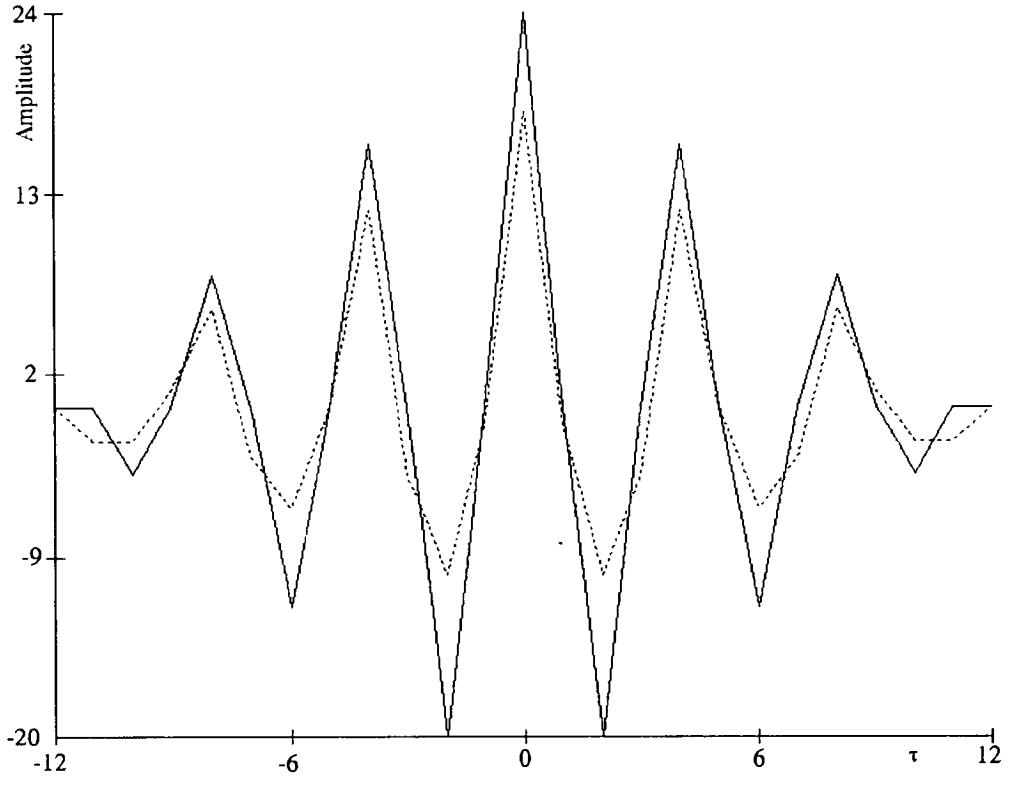
$$\gamma_4^x = c_4^x(0,0,0) = E\left\{X^4(k)\right\} - 3\left[\gamma_2^x\right]^2. \tag{3.13}$$

The relevance of these higher order terms can be clearly demonstrated by considering two simple zero mean cyclic data sequences, one skew, and one symmetric:

Skew data: 1 1 0 -2 ...
Symmetrical data: 0 2 0 -2 ...

Each cycle was repeated four times. The autocorrelation sequences of these two processes are displayed in fig(3.1). It is evident that the two processes are very similar. It is, however, easy to distinguish the two signals by examining their third order cumulant plots (fig(3.2)). It can be seen that the symmetric data sequence has a highly structured third order cumulant. This contrasts sharply with the skew sequence, which has many more peaks in a much less regular pattern.

The Nth order spectrum is defined as the Fourier transform of the cumulant sequence rather than the moment sequence for two principal reasons. Firstly, if X(k) is a stationary Guassian random process, then all its moments of order greater than 3 do not provide any further information about the process. The cumulant spectrum function demonstrates this explicitly, as higher order ($N \geq 3$) cumulants, and therefore higher order spectra, are zero for Gaussian processes. Secondly, cumulants preserve superposition, whereas moments do not. For some random process $Z(k) = X(k) + Y(k)$, if X(k) and Y(k) are



Dotted line: autocorrelation function of skew data

Continuous line: autocorrelation function of symmetrical data

Figure 3.1: *Autocorrelation functions of skew and symmetric data series.*

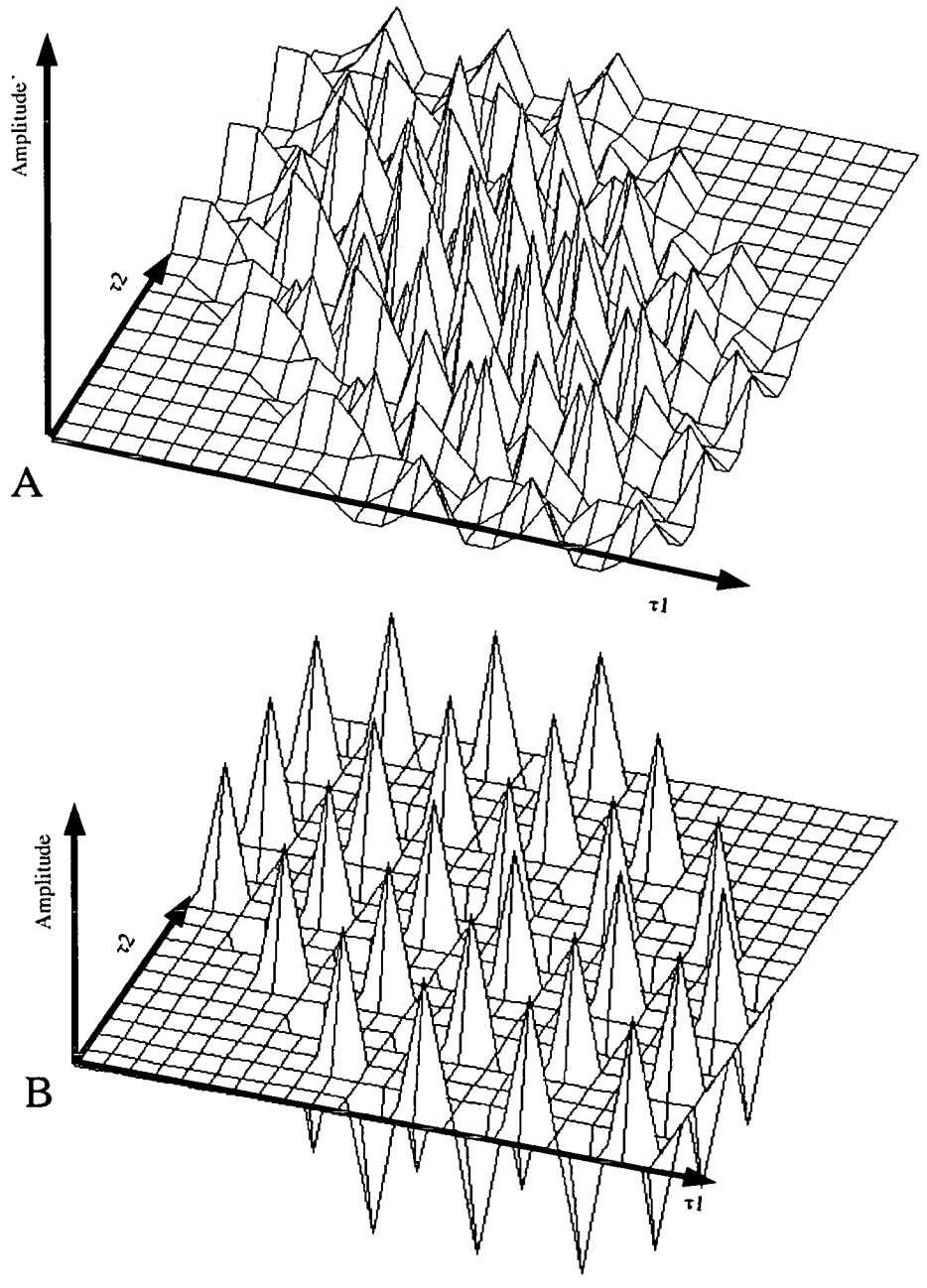


Figure 3.2: *Third order cumulants of A) skew and B) symmetric data series.*

statistically independent, then when considering their nth order cumulants,

$$c_n^z(\tau_1, \cdots, \tau_n) = c_n^x(\tau_1, \cdots, \tau_n) + c_n^y(\tau_1, \cdots, \tau_n) \tag{3.14}$$

whereas for their nth order moments,

$$m_n^z(\tau_1, \cdots, \tau_n) \neq m_n^x(\tau_1, \cdots, \tau_n) + m_n^y(\tau_1, \cdots, \tau_n) \tag{3.15}$$

In addition to these reasons, it has been pointed out [63–65] that ergodicity requirements are met more easily with cumulants.

3.2.3 Third Order Statistics

From equation(3.6), it follows [61,64,66] that the third order cumulant sequence has the following symmetry properties:

$$\begin{aligned} c_3(\tau_1, \tau_2) &= c_3(\tau_2, \tau_1) && \text{II} \\ &= c_3(-\tau_2, \tau_1 - \tau_2) && \text{III} \\ &= c_3(-\tau_1, \tau_2 - \tau_1) && \text{IV} \\ &= c_3(\tau_2 - \tau_1, -\tau_1) && \text{V} \\ &= c_3(\tau_1 - \tau_2, -\tau_1) && \text{VI} \end{aligned} \tag{3.16}$$

Using these symmetries, it is possible to reconstruct the complete third order cumulant sequence from the third cumulants in any one of the six sectors shown in fig(3.3).

From this and the definition of the bispectrum given in (3.9), certain properties of the bispectrum can be deduced:

$$\begin{aligned} B(\omega_1, \omega_2) &= |B(\omega_1, \omega_2)|\, e^{j\phi_B(\omega_1, \omega_2)} \\ &= B(\omega_1 + \pi, \omega_2 + \pi) \end{aligned} \tag{3.17}$$

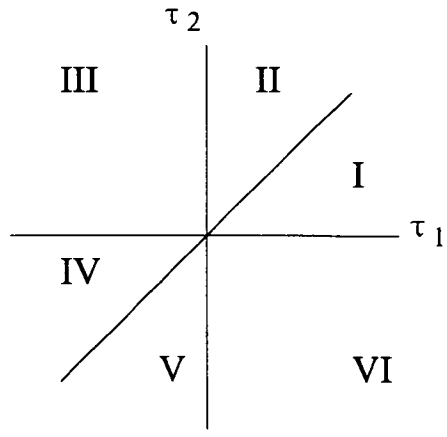


Figure 3.3: *Symmetry Regions of third order cumulants*

i.e. the bispectrum is complex with phase and magnitude, and is doubly periodic with period 2π;

$$\begin{aligned}
& B(\omega_1,\omega_2)\,(\mathbf{I}) = && B^*(-\omega_1,-\omega_2)\,(\mathbf{VII}) \\
= \; & B(\omega_2,\omega_1)\,(\mathbf{II}) = && B^*(-\omega_2,-\omega_1)\,(\mathbf{VIII}) \\
= \; & B(-\omega_2,\omega_1+\omega_2)\,(\mathbf{III}) = && B^*(\omega_2,-\omega_1-\omega_2)\,(\mathbf{IX}) \\
= \; & B(-\omega_1,\omega_1+\omega_2)\,(\mathbf{IV}) = && B^*(\omega_1,-\omega_1-\omega_2)\,(\mathbf{X}) \\
= \; & B(-\omega_1-\omega_2,\omega_1)\,(\mathbf{V}) = && B^*(\omega_1+\omega_2,-\omega_1)\,(\mathbf{XI}) \\
= \; & B(-\omega_1-\omega_2,\omega_2)\,(\mathbf{VI}) = && B^*(\omega_1+\omega_2,-\omega_2)\,(\mathbf{XII})
\end{aligned} \tag{3.18}$$

These symmetries are shown in fig(3.4). Clearly knowledge in the triangular region $\omega_2 \geq 0$, $\omega_2 \geq \omega_1$, $\omega_1 + \omega_2 \leq \pi$ is enough for a complete description of the bispectrum.

3.2.4 Theory of Quadratic Phase Coupling

As noted earlier, one of the key characteristics of higher order spectra is the preservation of phase information. The power spectrum suppresses all phase relations, and therefore cannot be used to determine the presence of nonlinear interaction between harmonically related frequency peaks. The bispectrum, however, is capable of detecting and quantifying quadratic phase coupling [60–

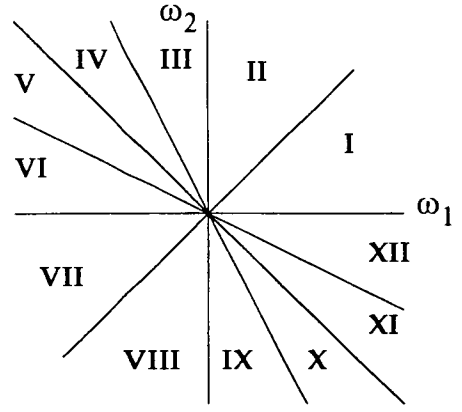


Figure 3.4: *Bispectrum Symmetry Regions.*

62,66]. Consider the process

$$X(k) = \cos(\lambda_1 k + \theta_1) + \cos(\lambda_2 k + \theta_2) + \cos(\lambda_3 k + \theta_3) \tag{3.19}$$

where

$$\lambda_3 = \lambda_2 + \lambda_1, \tag{3.20}$$

i.e. λ_1, λ_2 and λ_3 may be harmonically related. If θ_3 is independent of θ_1 and θ_2, then the three frequencies are not phase coupled, and λ_3 is an independent spectral component. If, on the other hand, $\theta_3 = \theta_2 + \theta_1$, then λ_3 is dependent on λ_1 and λ_2, and the three components are phase coupled.

A more rigorous examination of quadratic phase coupling can be undertaken by considering the Fourier-Stieltjes (Cramér) spectral representation of $x(t)$ [60,66]. To do this a brief revision of stochastic processes and probability theory is necessary.

Consider a function $\Psi(\omega)$ which represents cumulative spectral power density, i.e.

$$\begin{aligned} \Psi(-\infty) &= 0 \\ \Psi(\infty) &= p_{tot} \end{aligned}$$

where p_{tot} is the total signal power. In this case, the power density spectrum, $P(\omega)$ can be written as

$$P(\omega) = \frac{d\Psi(\omega)}{d(\omega)} \tag{3.21}$$

The autocorrelation function of $x(t)$ can then be expressed as a Lebesgue - Stieltjes integral [67–70]:

$$R(t) = E[x(s+t)x(s)^*] = \int_{-\infty}^{\infty} e^{2\pi i t\omega} d\Psi(\omega) \tag{3.22}$$

and the classical Fourier integral for spectral representation of a signal can be rewritten as the Fourier - Stieltjes integral

$$x(t) = \int_{-\infty}^{\infty} e^{2\pi j t\omega} dZ(\omega) \tag{3.23}$$

where

$$E\left[|dZ(\omega)|^2\right] = E[dZ(\omega)dZ(\omega)^*] = d\Psi(\omega) \tag{3.24}$$

this can be rewritten as

$$E\left[|dZ(\omega)|^2\right] = E[dZ(\omega+\alpha)dZ(\omega)] = d\Psi(\omega)\Big|_{\omega+(\omega+\alpha)=0} \tag{3.25}$$

Assuming $x(t)$ fulfills the Dirichlet criteria ($x(t)$ has a finite number of discontinuities, a finite number of maxima and minima and is absolutely integrable), then equation (3.23) becomes the Riemann-Stieltjes integral:

$$x(t) = \int_{-\infty}^{\infty} e^{2\pi j t\omega} Z'(\omega) d(\omega) \tag{3.26}$$

Using the fact that $F(-\omega) = F(\omega)^*$ it can be seen that equation (3.24) is a measure of PSD:

$$P(\omega) = E[F(\omega)F(\omega)^*] \tag{3.27}$$

Expanding this to the triple correlation:

$$R(t_1, t_2) = E[x(s+t_1)x(s+t_2)x(s)^*] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{2\pi j t(\omega_1+\omega_2)} d\Psi(\omega_1) d\Psi(\omega_2) \tag{3.28}$$

And similarly,

$$E[dZ(\omega+\alpha_1)dZ(\omega+\alpha_2)dZ(\omega)^*] = B(\omega_1, \omega_2)|_{(\omega+\alpha_1)+(\omega+\alpha_2)+\omega=0} \tag{3.29}$$

If we substitute $\omega_1 = \omega+\alpha_1, \omega_2 = \omega+\alpha_2$, an expression similar to equation(3.27) is derived:

$$B(\omega_1, \omega_2) = E[F(\omega_1)F(\omega_2)F(\omega_1+\omega_2)^*] \tag{3.30}$$

It is therefore apparent that the power spectrum $P(\omega)$ represents the contribution to the mean square $\overline{X}^2$ from the product of two Fourier components whose frequencies add to zero, and the bispectrum represents the contribution to the mean cube $\overline{X}^3$ from the product of three Fourier components whose resultant frequency is zero. A typical bispectrum plot can be seen in figure (3.9) of the next section, which demonstrates the application of bispectral analysis.

If $F(\omega)$ is rewritten in terms of phase and magnitude:

$$F(\omega) = |F_\omega| e^{j\theta_\omega} \tag{3.31}$$

then equation(3.30) becomes

$$B(\omega_1, \omega_2) = E\left[|F_{\omega_1}||F_{\omega_2}||F_{\omega_1+\omega_2}| e^{j(\theta_{\omega_1}+\theta_{\omega_2}-\theta_{\omega_3})}\right]. \tag{3.32}$$

Phase coupling is present if θ_{ω_3} is dependent on θ_{ω_1} and θ_{ω_2}, i.e.

$$\theta_{\omega_1} + \theta_{\omega_2} - \theta_{\omega_3} = \theta_P \tag{3.33}$$

where θ_P is constant. To verify this, it is necessary to average over a number of data records. Thus

$$\begin{aligned} \overline{B}(\omega_1, \omega_2) &= E\left[|F_{\omega_1}||F_{\omega_2}||F_{\omega_1+\omega_2}| e^{j(\theta_{\omega_1}+\theta_{\omega_2}-\theta_{\omega_3})}\right] \\ &= |F_{\omega_1}||F_{\omega_2}||F_{\omega_1+\omega_2}|\, E\left[e^{j(\theta_{\omega_1}+\theta_{\omega_2}-\theta_{\omega_3})}\right] \end{aligned} \tag{3.34}$$

assuming $X(k)$ is stationary, and that θ_i are zero mean i.i.d. between $-\pi$ and π. If equation (3.33) is true, then (3.34) simplifies to

$$\overline{B}(\omega_1, \omega_2) = |F_{\omega_1}||F_{\omega_2}||F_{\omega_1+\omega_2}| e^{j\theta_P} \tag{3.35}$$

however, in the absence of phase coupling, $E\left[e^{j(\theta_{\omega_1}+\theta_{\omega_2}-\theta_{\omega_3})}\right] = 0$ and correspondingly, there is no bispectral peak.

From this, the following property of phase coupling is revealed: consider the phase transform

$$\phi_\omega = \Phi(\theta_\omega) \tag{3.36}$$

where $\Phi()$ is some stationary non-linear function. If phase coupling is present, then equation (3.33) becomes

$$\phi_{\omega_1} + \phi_{\omega_2} - \phi_{\omega_3} = \theta_Q \tag{3.37}$$

where θ_Q is constant. Clearly, therefore the phase coupling is still present, and the average complex bispectrum at the point (ω_1, ω_2) will retain the same magnitude, but be subject to a phase shift $\theta_P - \theta_Q$.

In practice, it is reasonable to assume that the degree of phase coupling will vary. If phase coupling occurs at a frequency which is also subject to contributions from an independent component of the same signal, there would be a resulting attenuation of the magnitude of the bispectral peak:

$$\overline{B}(\omega_1, \omega_2) = \alpha_{\omega_1,\omega_2} |F_{\omega_1}||F_{\omega_2}||F_{\omega_1+\omega_2}| e^{j\theta_P} \tag{3.38}$$

where $0 \leq \alpha_{\omega_1,\omega_2} \leq 1$. In order to assess the degree of phase coupling, it is therefore desirable to normalise the bispectrum. This normalised bispectrum is known as the bicoherence index, and is defined as [22,26]:

$$b(\omega_1, \omega_2) = \frac{\left|\overline{B}(\omega_1, \omega_2)\right|}{\sqrt{\{P(\omega_1)P(\omega_2)P(\omega_1 + \omega_2)\}}}. \tag{3.39}$$

From equations (3.27), (3.31) and (3.38) it can be seen that

$$b(\omega_1, \omega_2) = \alpha_{\omega_1,\omega_2} \tag{3.40}$$

thus the bicoherence index at a point (ω_1, ω_2) represents the fraction of power at the sum frequency $(\omega_1 + \omega_2)$ owing to quadratic interactions between the two Fourier components ω_1 and ω_2. The use of the bicoherence index is demonstrated extensively in the next section, and a typical bispectral plot can be seen in figure (3.10).

In the next section, by considering some numerical examples, the properties of the bispectrum and bicoherence discussed above are demonstrated, and their application to the analysis of real signals examined.

3.2.5 Applications of Phase Coupling

At its most simple, bispectral analysis is a test of statistical dependency. For a signal $X_L(k)$ which is comprised of a linear combination of three processes:

$$\begin{aligned} x_1(k) &= A\cos(\lambda_1 k + \theta_1) \\ x_2(k) &= B\cos(\lambda_2 k + \theta_2) \\ x_3(k) &= C\cos(\lambda_3 k + \theta_3) \\ X_L(k) &= x_1(k) + x_2(k) + x_3(k) \end{aligned} \tag{3.41}$$

where $\lambda_3 = \lambda_2 + \lambda_1$, and A, B and C are constants, detection of a bispectral peak at $B(\lambda_1, \lambda_2)$ would show that the third component was dependent on the other two. Note that it does not, however, indicate whether x_1 and x_2 are independent of one another. The coupling could therefore be through some nonlinear interaction between x_1 and x_2, or could be evidence of an underlying linear system driving all three processes. This is best illustrated by considering the following numerical examples.

The following values were substituted into equation(3.41):

$$\begin{aligned} x_1(k) &= 5\cos(0.11\pi k + \theta_1) \\ x_2(k) &= 7\cos(0.34\pi k + \theta_2) \\ x_3(k) &= 2\cos(0.45\pi k + \theta_3) \end{aligned} \tag{3.42}$$

The data comprised one hundred blocks of 128 samples. The complex bispectrum and power spectrum were calculated for each block, and then averaged over the set of blocks. The average power spectrum is shown in figure (3.5). It can be seen that the frequencies appear in bin 7, bin 22, and bin 29. The first case considered was that in which no phase coupling was present. The phases of the components $x_i(k)$ were independent, and varied randomly from block to block. The resultant bispectral plots can be seen in figures(3.6 - 3.8). These and subsequent bispectra are displayed as quantised intensity plots, with ω_1 and ω_2 plotted on the x and y axes respectively, whilst the amplitude is shown by shading.

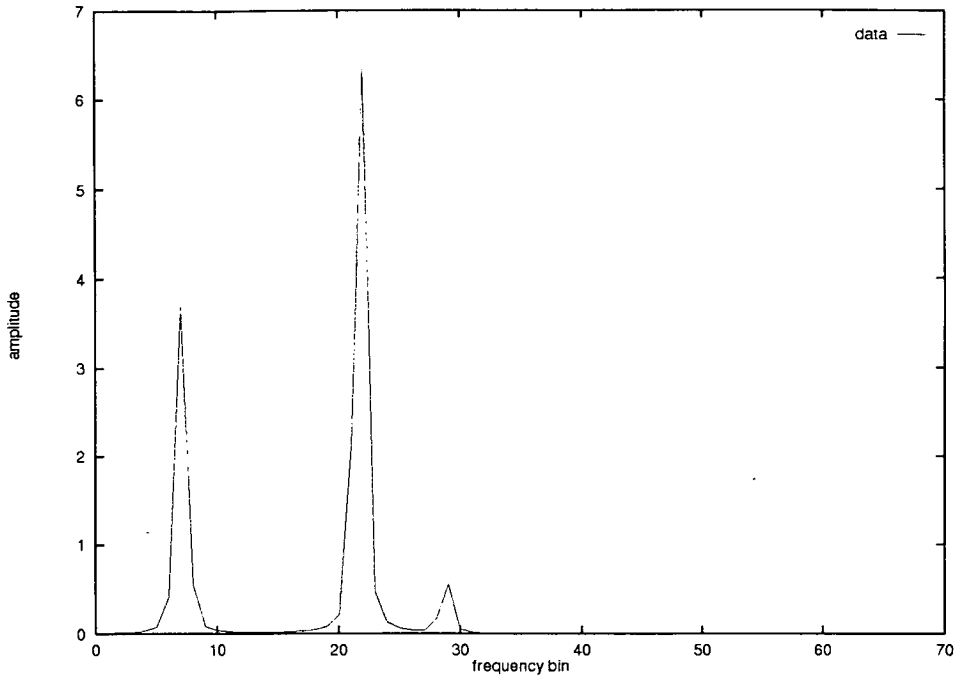


Figure 3.5: *128 point power spectrum of* $X_L(k)$.

Figure (3.6) shows the power bispectrum for the case where the signal components are not phase coupled (i.e. θ_3 is independent of θ_1 and θ_2). As predicted in theory, no peak is present in the power bispectrum, which is thus empty. Likewise, the bicoherence index (shown in figure (3.7)) is empty, as the proportion of signal power due to phase coupled components is zero for all frequencies. Together, these two plots demonstrate conclusively that no phase coupling is present. As a result, the phase bispectrum should contain no information. The phase bispectrum is shown in figure (3.8) with the frequencies ω_1 and ω_2 plotted on the x and y axes, with the biphase angle shown in radians as a quantised intensity plot. It is interesting to note the presence of a definite pattern in the biphase plot, with lines at the three interacting frequencies, as the appearance of these seems to be contrary to the theory derived from equation(3.34). Referring to this equation, the bispectral phase is given by

$$B_\theta = E[\theta_1 + \theta_2 + \theta_3] \tag{3.43}$$

and this should be zero for all points where the three phases are zero or random

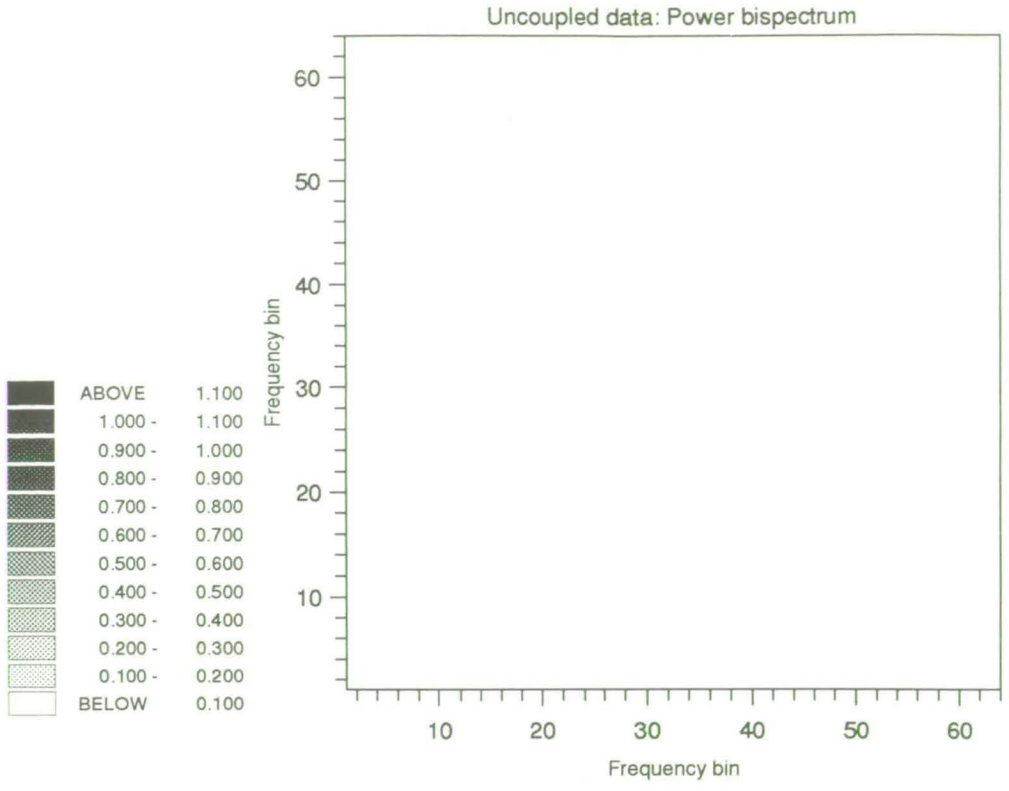


Figure 3.6: *128 point power bispectrum: uncoupled case.*

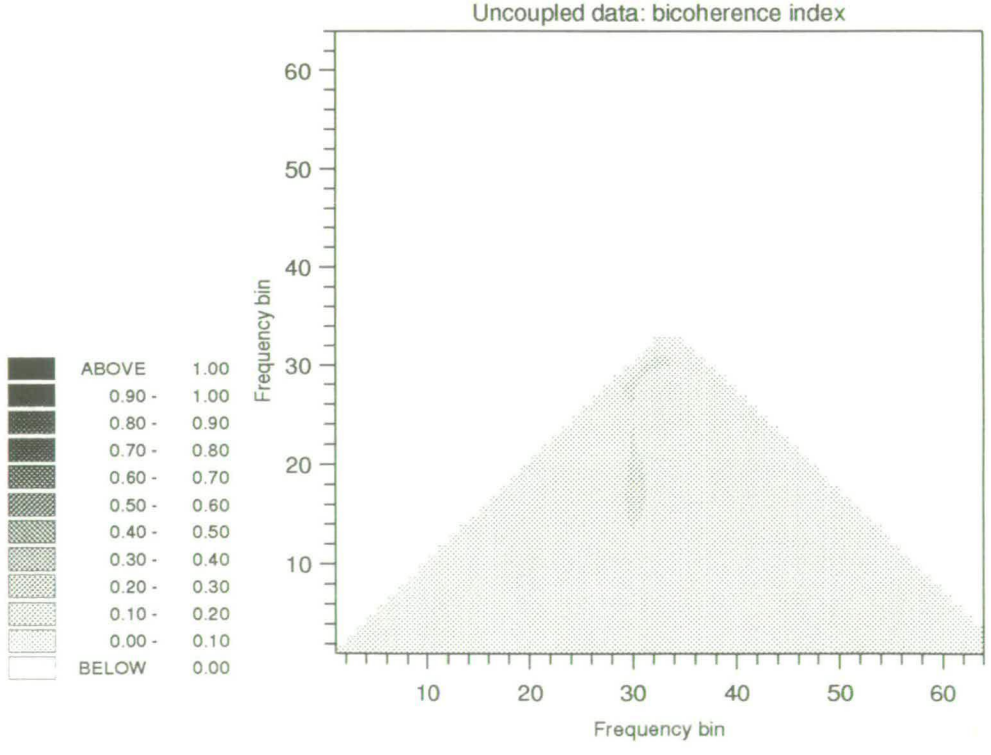


Figure 3.7: *128 point bicoherence index: uncoupled case.*

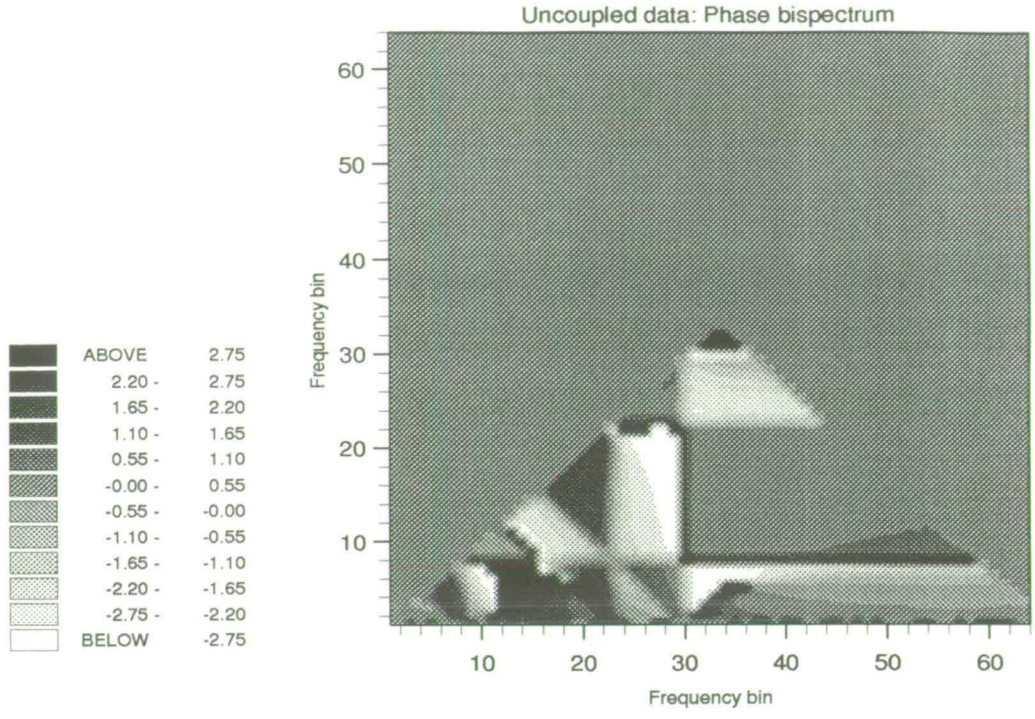


Figure 3.8: *128 point phase bispectrum: uncoupled case.*

with zero mean. However, the practical aspects of the experiment must be taken into account - the very small values of the complex bispectrum at points other than bispectral peaks remove any accuracy of the resulting phase calculation at these points. The phase bispectrum is therefore meaningless at points with no bispectral magnitude. As the plots discussed above are dealing with the uncoupled case, there are no bispectral peaks, and therefore the biphase portrait contains no information, despite appearances to the contrary.

The second case considered was that in which $x_1(k)$ and $x_2(k)$ were independent random variables, but $x_3(k)$ was a result of phase coupling between $x_2(k)$ and $x_1(k)$. The phases of $x_1(k)$ and $x_2(k)$ therefore varied randomly from block to block, but $\theta_1 + \theta_2 - \theta_3$ remained constant (at a value of 0.55π). This is shown in figures(3.9 - 3.11). Figure (3.9) shows the power bispectrum for this case, and a large peak can be seen at (22,7). This indicates the presence of phase coupling between the signal components at bins 7, 22, and 7+22=29. It is therefore evident that part of the signal component in bin 29 is dependent on

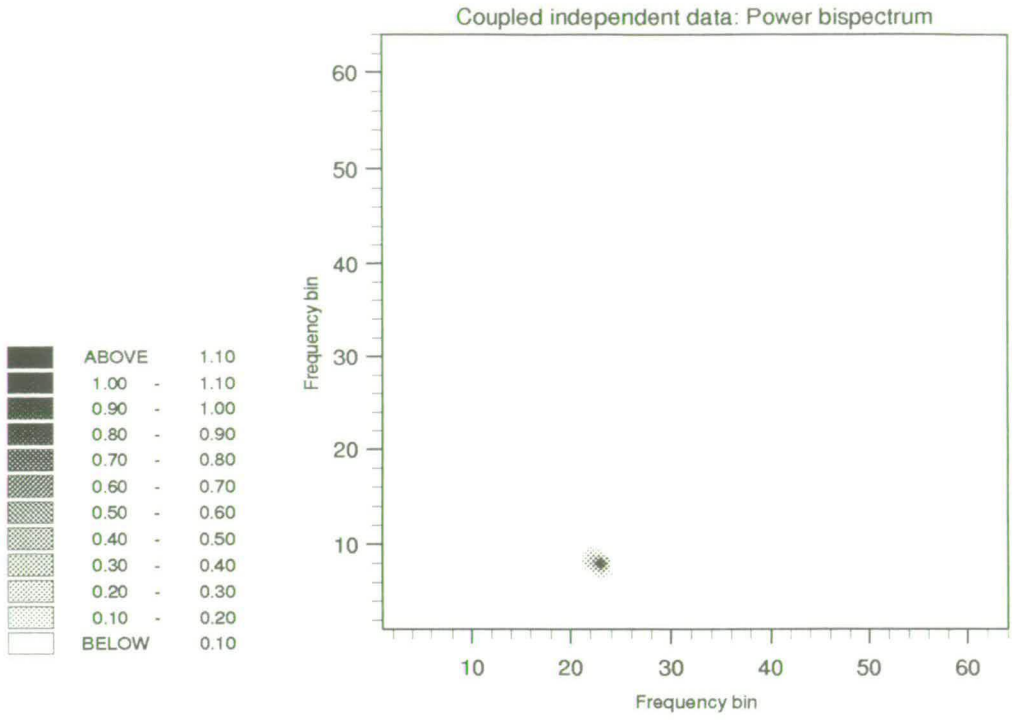


Figure 3.9: *128 point power bispectrum: coupled independent case.*

those components in bins 7 and 22. In order to find out what proportion this is, the bicoherence index of the signal, shown in figure (3.10), can be examined. As with the biphase plot, the bicoherence index is only meaningful at points where the signal has a bispectral component. In this case, the only such point found in the power bispectrum is at point is at (22,7). Examination of the bicoherence index here shows a value of unity, indicating that all the signal power present in bin 29 is dependent on the signal components in bins 7 and 22. The biphase of the signal is shown in figure (3.11). As stressed before, the only point of interest is at the location of the bispectral peak, at (22,7), where the biphase can be seen to be in the band 1.65 – 2.20. As mentioned earlier, $\theta_1 + \theta_2 - \theta_3$ was kept at a constant value of 0.55π (1.728), so it can be seen that the biphase plot explicitly shows this constant phase difference.

In the third case (figures (3.12 - 3.14)), both $x_2(k)$ and $k_3(k)$ were dependent on $x_1(k)$, with $\theta_2 = \theta_1 + 0.5\pi$ and $\theta_3 = \theta_1 + 0.15\pi$. This models the situation where all three frequencies are generated by the same underlying

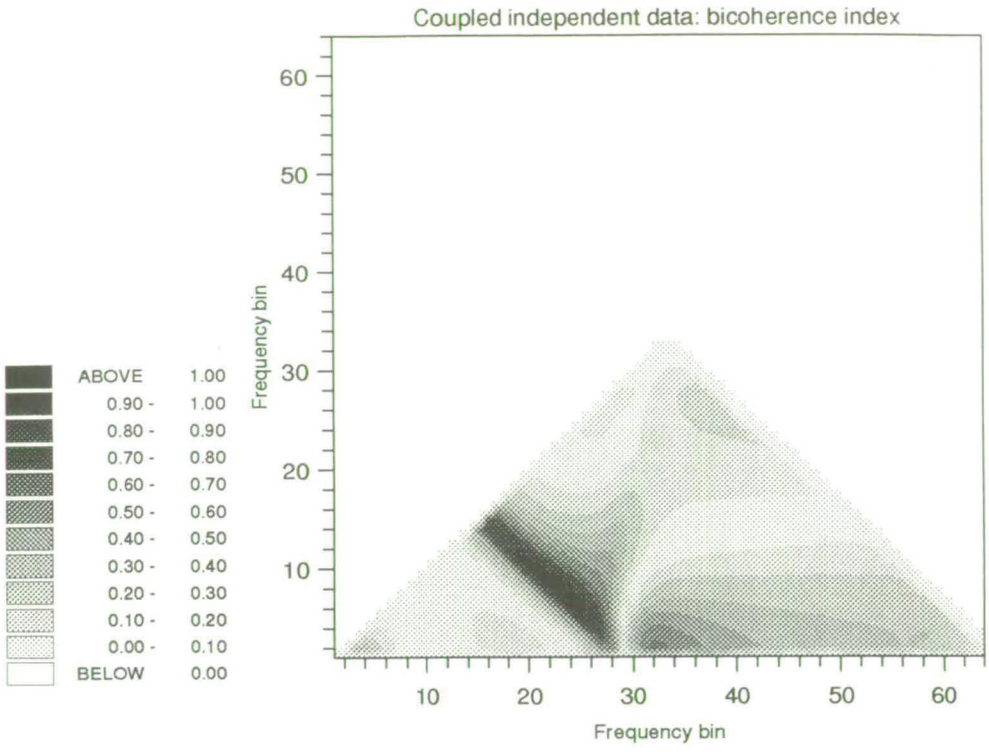


Figure 3.10: *128 point bicoherence index: coupled independent case.*

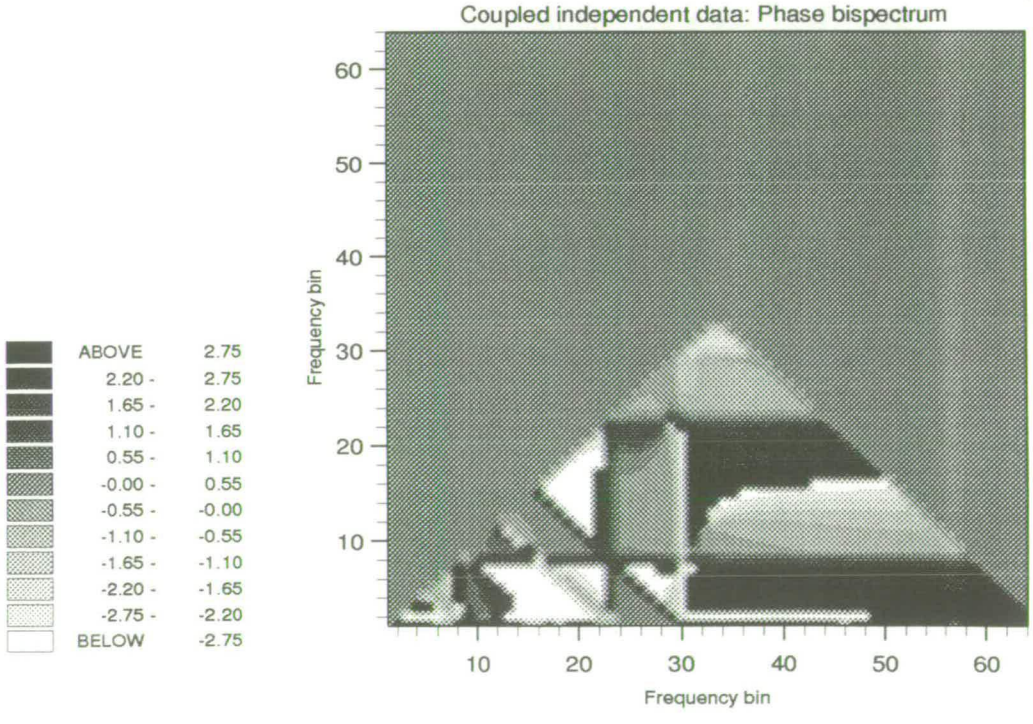


Figure 3.11: *128 point phase bispectrum: coupled independent case.*

system. Once again, the power bispectrum, which is shown in figure (3.12),

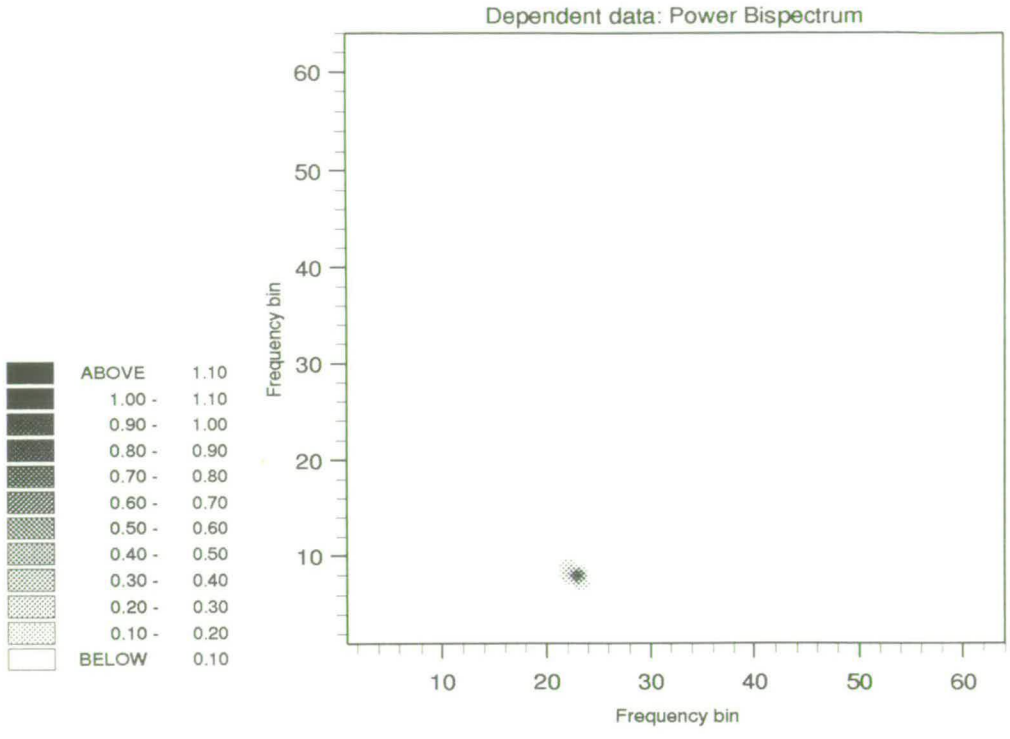


Figure 3.12: *128 point power bispectrum: coupled dependent case.*

has a peak at (22,7). The bicoherence index for the signal, can be seen in figure (3.13). As with the bicoherence index for the independent data in the previous example, there is a unity value at (22,7), indicating that all the power at the signal component in bin 29 is due to phase coupling between bins 7 and 22. The only difference that can be seen between these two cases (in both of which phase coupling is present), is in the phase bispectra (figures (3.11) and (3.14)) of the two processes at the point (22,7), as a result of the $0.65\pi - 0.55\pi$ biphase difference between the two sets of data. These three examples are graphical evidence of the information present in bispectral plots. In addition, the last two examples demonstrate explicitly one important fact that is not always appreciated: although detection of phase coupling in the bispectrum is indicative of dependence between spectral components of a signal, it does not in itself offer any further information about the underlying process or processes generating the signal in question.

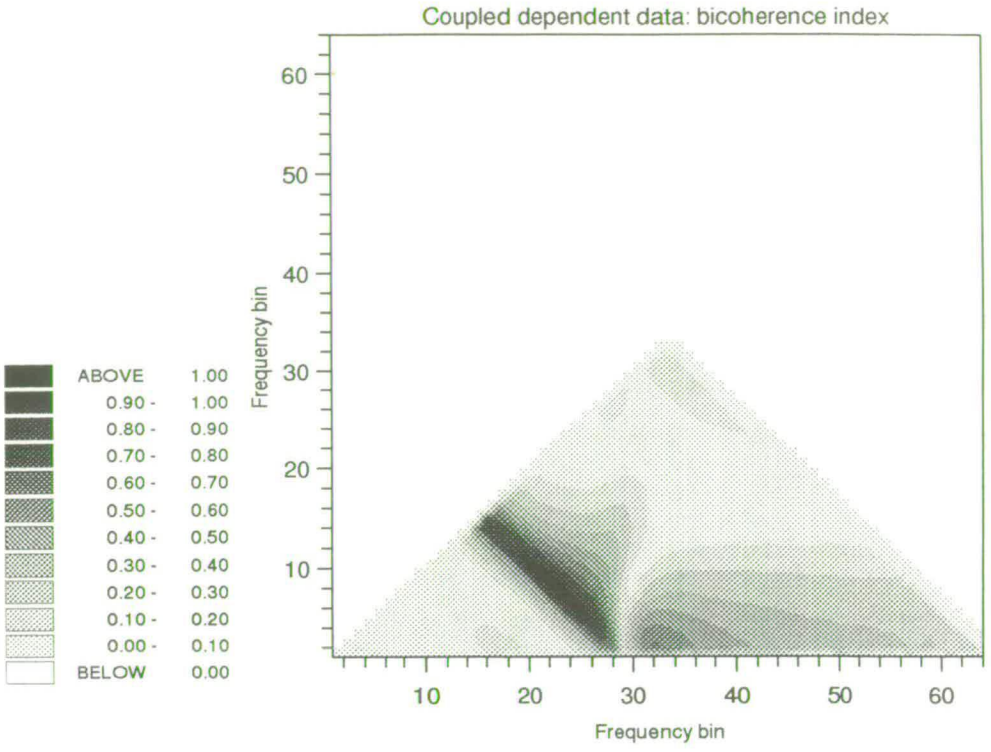


Figure 3.13: *128 point bicoherence index: coupled dependent case.*

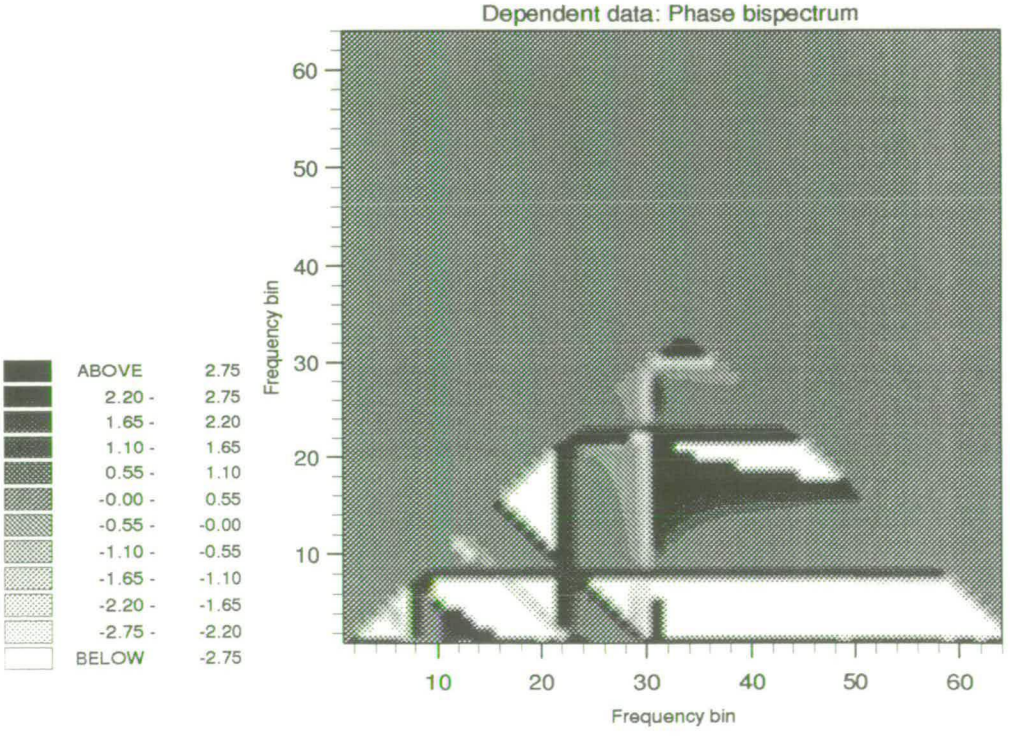


Figure 3.14: *128 point phase bispectrum: coupled dependent case.*

In order to determine whether the signal is being generated by one or more processes, and whether these processes are combining in a linear or nonlinear fashion, it is necessary to look at the *pattern* that the signal displays in the bispectrum. A typical example of this can be demonstrated using a second process X_Q comprising the two random variables $x_1(k)$ and $x_2(k)$ with a cross modulation term:

$$\begin{aligned} X_Q(k) &= x_1(k) + x_2(k) + x_1(k)x_2(k) \\ &= \cos(\lambda_1 k + \theta_1) + \cos(\lambda_2 k + \theta_2) + \cos(\lambda_1 k + \theta_1)\cos(\lambda_2 k + \theta_2) \end{aligned} \tag{3.44}$$

The cross modulation term can be expressed as

$$0.5\left[\cos(\lambda_{3a} k + \theta_{3a}) + \cos(\lambda_{3b} k + \theta_{3b})\right] \tag{3.45}$$

where

$$\begin{aligned} \lambda_{3a} &= \lambda_1 + \lambda_2 \\ \lambda_{3b} &= \lambda_1 - \lambda_2 \\ \theta_{3a} &= \theta_1 + \theta_2 \\ \theta_{3b} &= \theta_1 - \theta_2 \end{aligned} \tag{3.46}$$

The signal bispectrum would thus have peaks at (λ_1, λ_2) and $(\lambda_1 - \lambda_2, \lambda_2)$. Furthermore, the biphase angles at these points are identically zero:

$$\begin{aligned} &< B(\lambda_1, \lambda_2) = \theta_1 + \theta_2 - \theta_{3a} = 0 \\ &< B(\lambda_1 - \lambda_2, \lambda_2) = \theta_{3b} + \theta_2 - \theta_1 = 0 \end{aligned} \tag{3.47}$$

This is shown in figures(3.15 - 3.17). The power bispectrum can be seen in figure (3.15). As anticipated, bispectral peaks are present at the two coupled frequencies; (22,7), corresponding to (λ_1, λ_2), and (15,7), corresponding to $(\lambda_1 - \lambda_2, \lambda_2)$. The bicoherence index for $X_Q(k)$ is shown in figure (3.16), and examination of the bicoherence index at the two peaks found in the power bispectrum show both have values of unity, indicating total phase coupling at those frequencies. The biphase plot for $X_Q(k)$ is then shown in figure (3.17), and also corroborrates the result anticipated in theory, with the biphase at both (22,7) and (15,7) being 0. To generate a similar signal from a linear combination

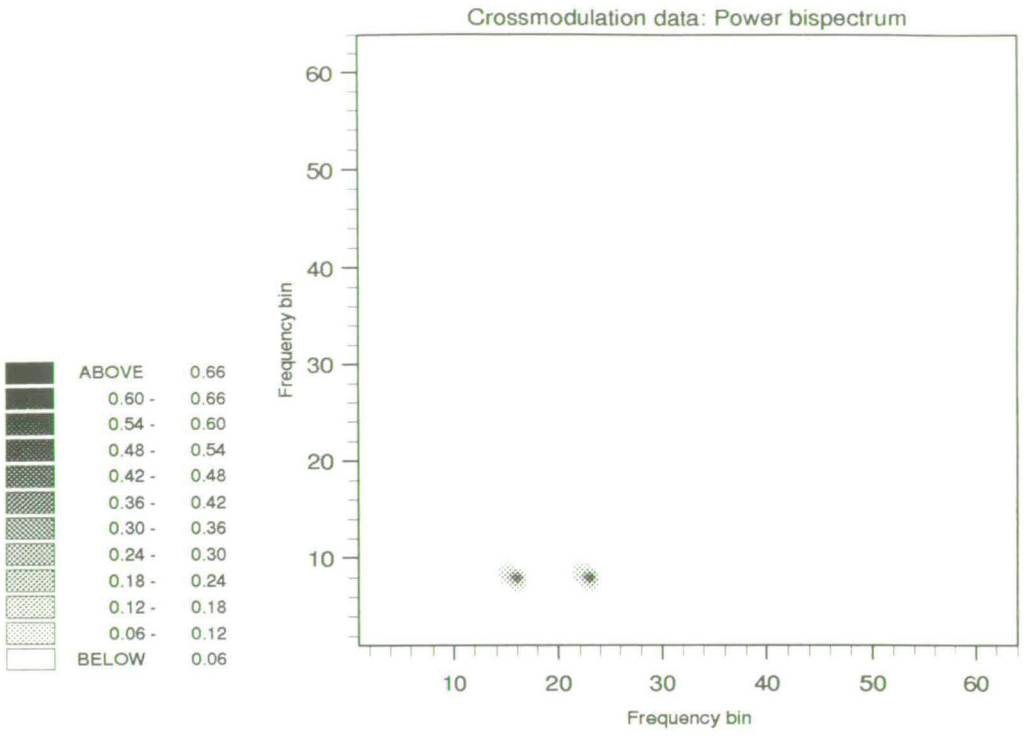


Figure 3.15: *128 point power bispectrum: quadratic cross modulation example*

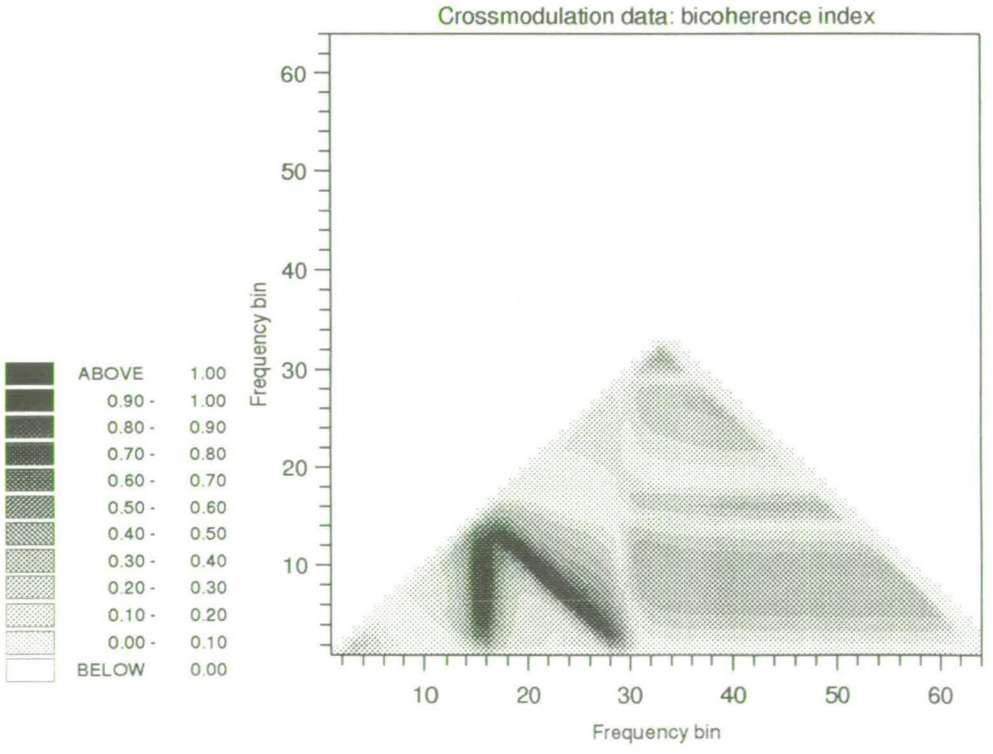


Figure 3.16: *128 point bicoherence index: quadratic cross modulation example*

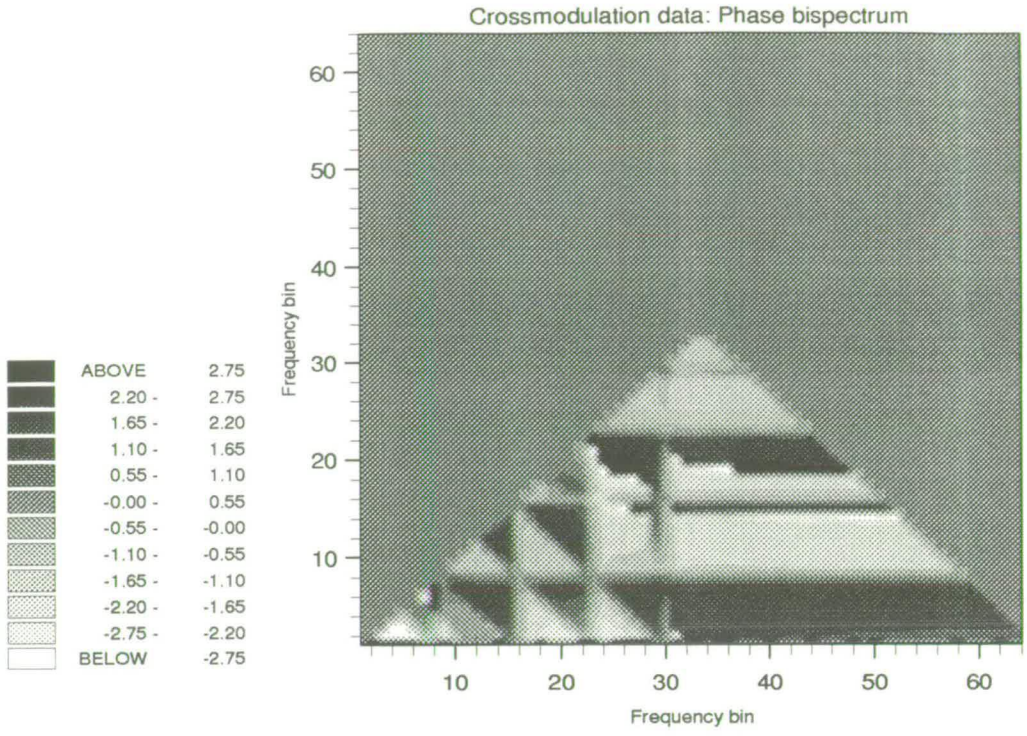


Figure 3.17: *128 point phase bispectrum: quadratic cross modulation example*

of random variables is impossible, as the assumption of linearity infers that all bispectral frequency components are dependent on λ_2.

Having examined both the theory of higher order spectral analysis, and its application to some typical signals, it is now possible to make a meaningful and detailed analysis of the bispectra of some real drum data.

3.3 Bispectral Analysis of Drum Sounds

Data records were available for two drums, a "Pearl Export" Mid Tom, and a "Pearl Export" High Tom. Both drums were sampled at two rates, 44.096 kHz and 33.33 kHz, and the input signal was filtered using six-pole Butterworth anti-aliasing filters. A further set of data records were taken for the Mid Tom, using a six-pole Bessel filter in order to preserve the signal phase information. Further details of the records and the acquisition process can be found in chapter 2.

Twenty records were taken in each case. Calculation of the bispectrum was complicated in this case by the nonstationarity of the drum sound. This made it impossible to calculate a meaningful bispectrum estimate across an entire data record. Instead, the the mean of each record was removed, and they were split into length 512 segments, across each of which stationarity was assumed. A 512 point FFT was then calculated for each segment using a Hamming window. From these, single segment PSD and bispectrum estimates were calculated using the formulae derived in equations (3.21) to (3.30):

$$P(\omega) = E[F(\omega)F(\omega)^*] \tag{3.48}$$

$$B(\omega_1, \omega_2) = E[F(\omega_1)F(\omega_2)F(\omega_1 + \omega_2)^*] \tag{3.49}$$

Final estimates were then calculated by ensemble averaging across the twenty data records for each segment. It must be stressed that whilst the PSD as shown in equation (3.48) is a real valued function, the bispectrum estimate shown in equation (3.49) is complex, and as a consequence bispectral ensemble averaging is also complex. Figures (3.18) to (3.23) show the resultant bispectral plots for the data records gathered.

The first two bispectra shown, figures (3.18) and (3.19) are both for the Mid Tom. These two plots have been printed in colour for the purpose of clarification, as they demonstrate in practice the theory set out in equations (3.36) and (3.37). Figure (3.18) shows the average power bispectrum for the data records gathered using a Butterworth filter, whilst figure (3.19) shows the average power bispectrum for those records gathered using a Bessel filter.

Bispectral amplitude on these and subsequent bispectral power graphs is plotted using a log scale, in order to maximise the information revealed by the diagrams. The fact that human auditory perception functions on an approximately logarithmic scale is a further reason for the use of a log scale. It must be noted however, that as the power bispectrum is the Fourier transform of the triple correlation of a signal, rather than the autocorrelation, the resulting amplitude cannot be considered to be a decibel scale.

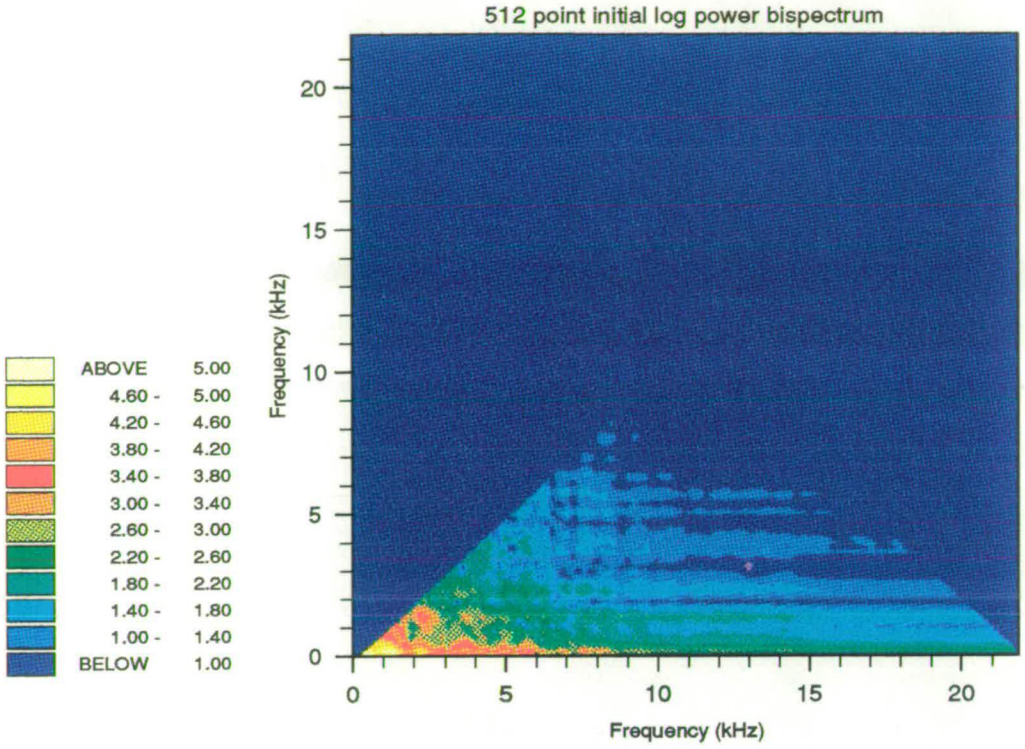


Figure 3.18: *512 point initial log power bispectrum: 44.1kHz Mid Tom (Butterworth).*

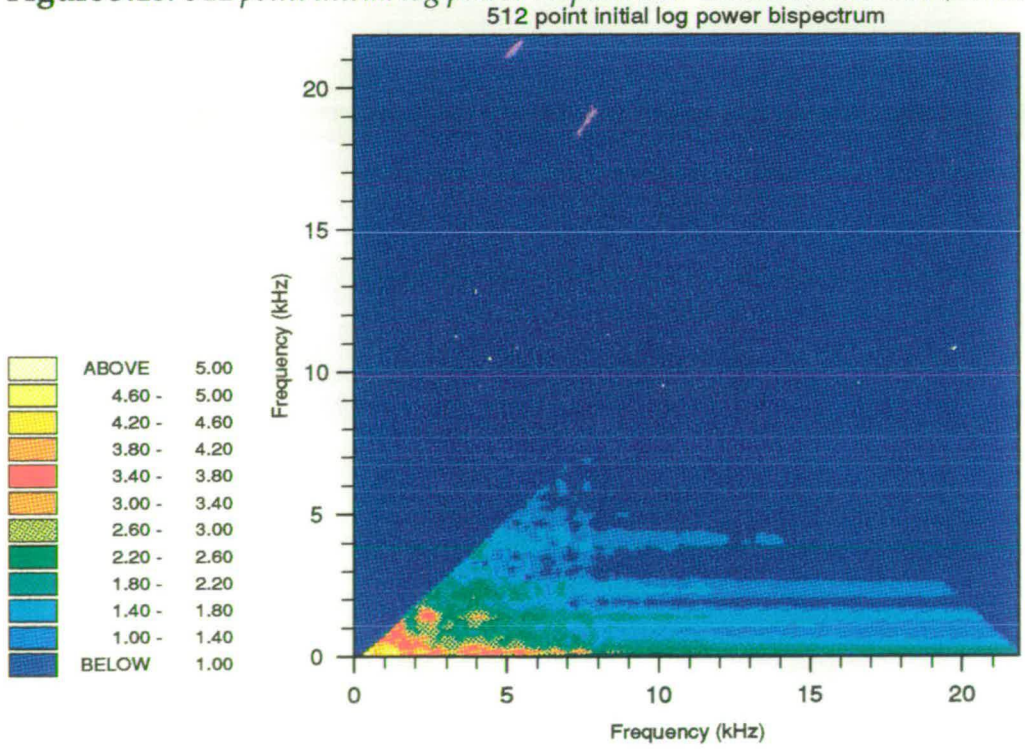


Figure 3.19: *512 point initial log power bispectrum: 44.1kHz Mid Tom (Bessel).*

Examination of these two figures reveals not only that there is a definite pattern present in both signal bispectra, but that this pattern is independent of the anti-aliasing filter phase response. Following from the simple example shown in figures (3.9) and (3.12), this is therefore a practical demonstration of the robustness of the power bispectrum to phase transforms. The biphase shift, seen in the earlier examples in figures (3.11) and (3.14), is also present in the actual drum biphase portraits. Figures (3.20) and (3.21), also plotted in colour, show these biphase plots for the Mid Tom data records gathered using the Butterworth and Bessel filters respectively.

The biphase shift between these two diagrams is most evident when considering the long horizontal bands (from 10 – 20 kHz on the x-axis and 0 – 5 kHz on the y-axis) in both diagrams. Those in figure (3.21) are shifted upwards slightly with respect to the equivalent bands in figure (3.20), an indication of a largely uniform biphase shift in that area. It is important to realise that the biphase offset is not neccessarily uniform across the entire bispectrum, as the value of that offset depends on the difference in phase response between the two filters. This itself is not uniform across the bandwidth of the signal.

Figure (3.22) shows the power bispectrum for the data records gathered from the High Tom, and figure (3.23) shows the power bispectrum for those gathered using the 33.33 kHz sampling rate. It can be seen that all these bispectra have the same overall shape. In addition, a similar pattern to that seen in the cross modulation example shown in figure (3.15) is evident along the fundamental frequency of the drum (at (1.12,0.098), (2.20,0.098) on the colour power bispectra), indicative of interaction between two independent signal components at the frequencies 98 Hz and 220 Hz in the Mid Tom (also seen at 110 and 234 Hz in the High Tom).

Figure (3.24) shows the phase bispectrum for the High Tom, and figure (3.25) shows the phase bispectrum for the Low Tom sampled at 33.333 kHz. For both these and the previous phase bispectra, the two biphase values at the peaks detailed above are in the same intensity band as one another, which acts as

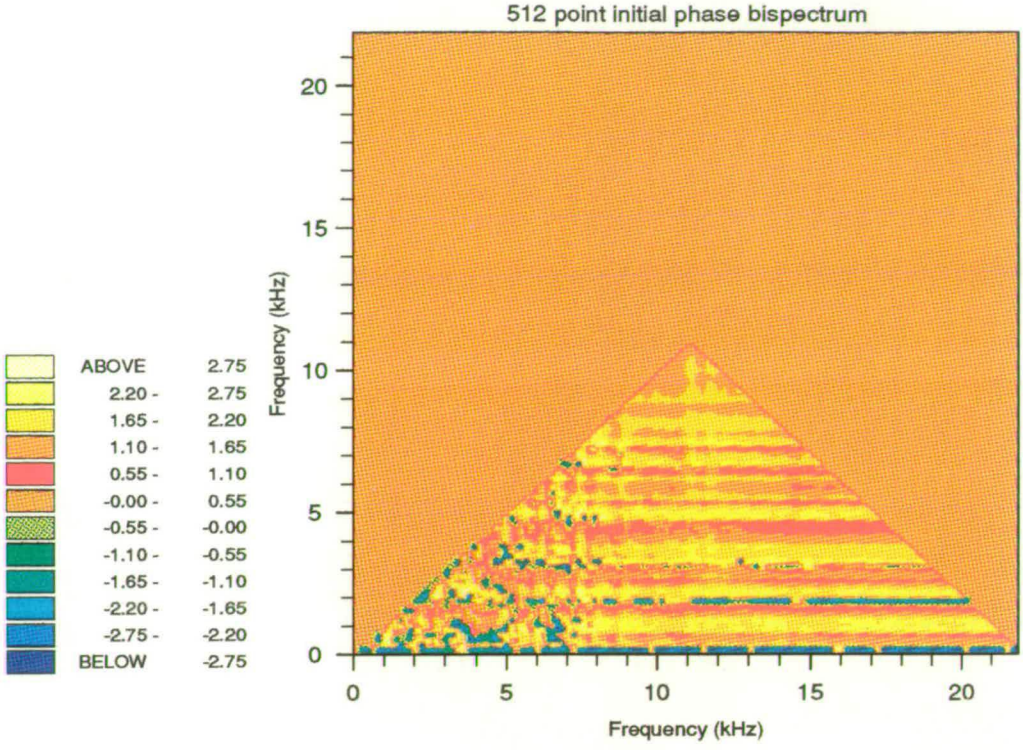


Figure 3.20: *Initial phase bispectrum: 44.1kHz Mid Tom (Butterworth).*

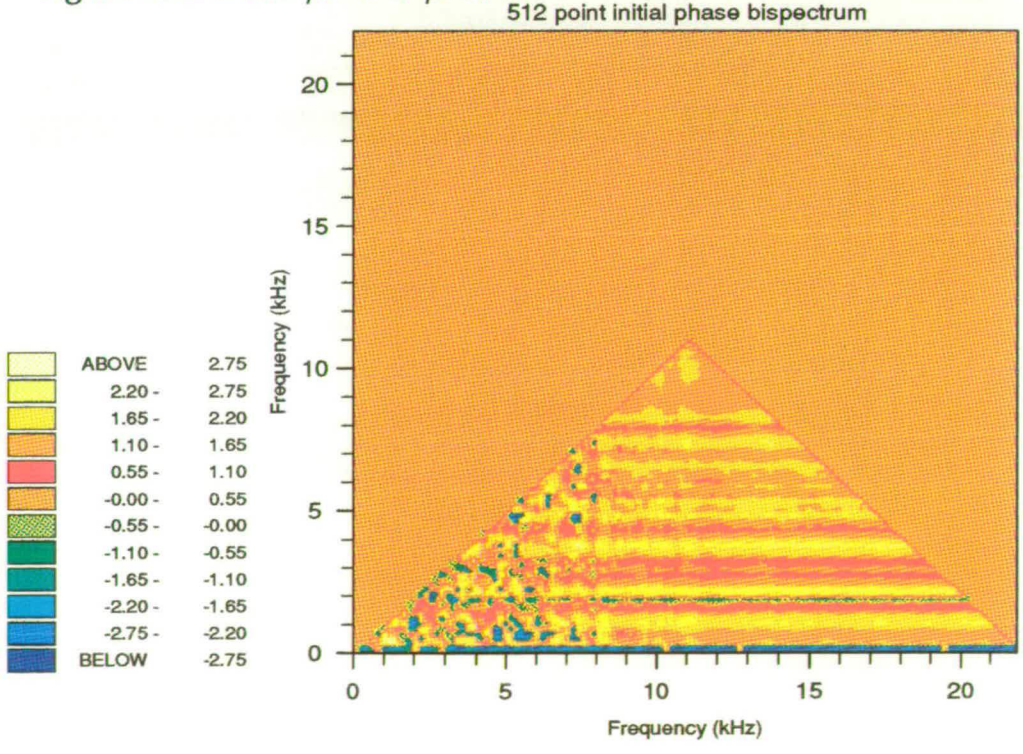


Figure 3.21: *Initial phase bispectrum: 44.1kHz Mid Tom (Bessel).*

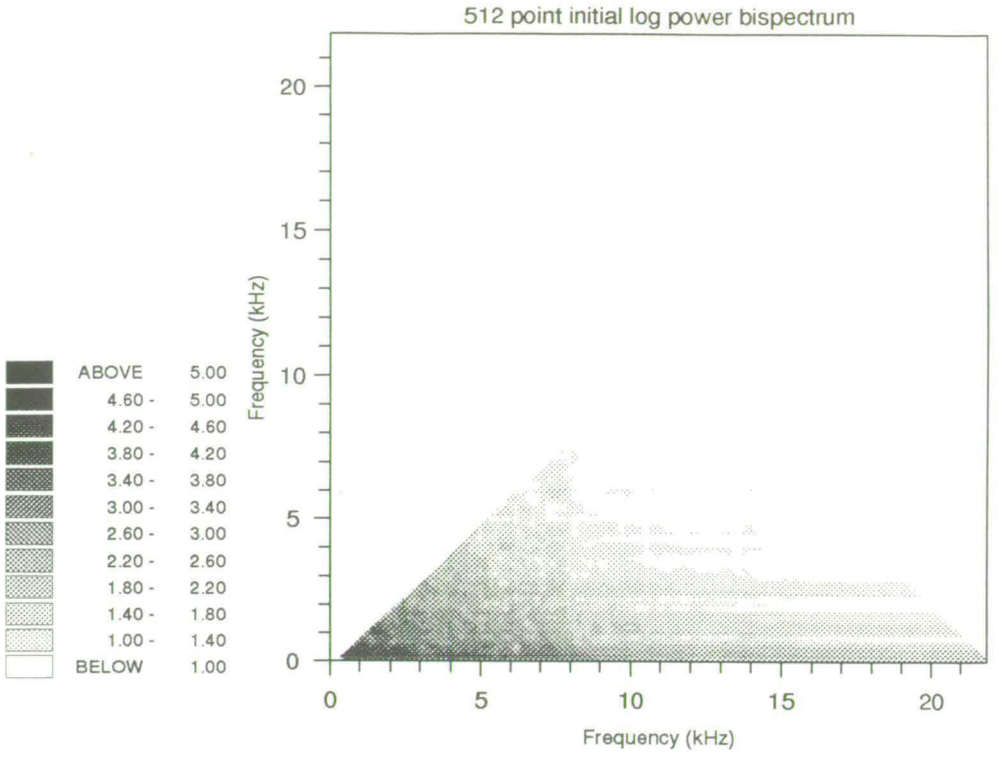


Figure 3.22: *Initial log power bispectrum: 44.1kHz High Tom (Butterworth).*

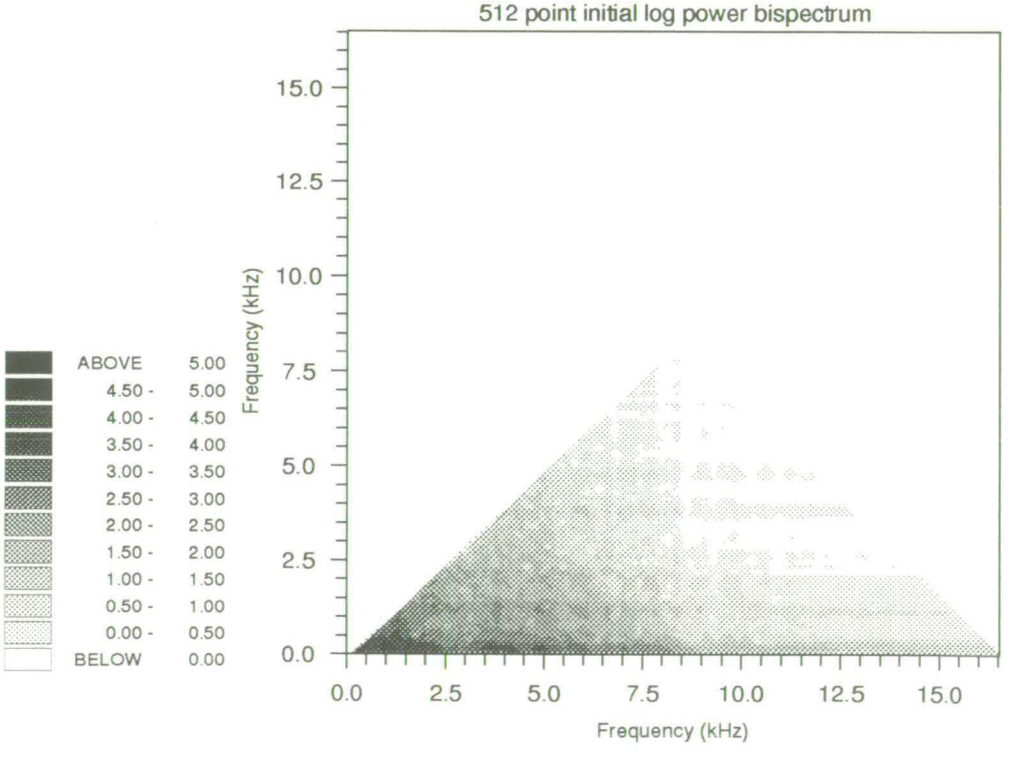


Figure 3.23: *Initial log power bispectrum: 33.33kHz Mid Tom (Butterworth).*

a further indication of an underlying nonlinear interaction between the two signal components.

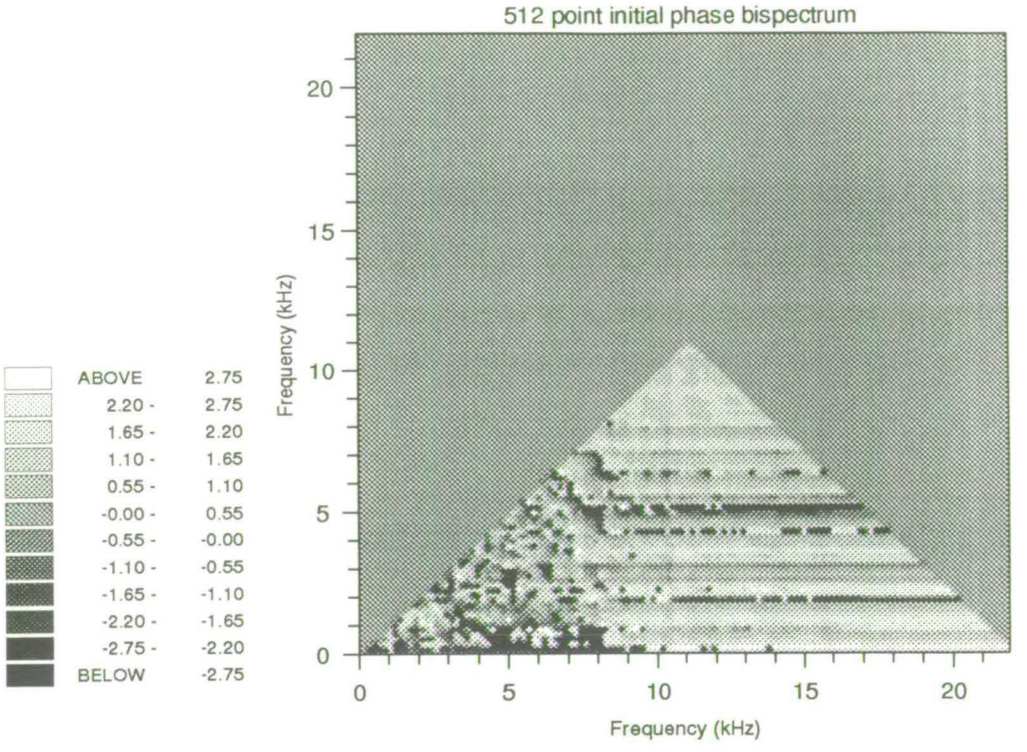


Figure 3.24: *Initial phase bispectrum: 44.1kHz High Tom (Butterworth).*

As shown in the previous section, it is necessary to look at the power bispectrum in conjunction with the bicoherence index in order to gain complete understanding about the bispectrum of each set of records. Figures (3.26) - (3.29) show the bicoherence indices for the four different data sets. Examination of these indices at both cross modulation peaks spotted in the power bispectra reveals values close to unity in all the graphs. Hence, the signal power at the sum frequencies corresponding to those peaks (112 Hz and 124 Hz for the Low and High Toms respectively) is almost entirely due to quadratic phase coupling.

Further examination of the bicoherence indices reveals more information of interest. The overall shape is the same for each of the data records. With the exception of the quadratic modulation just noted, there is very little phase coupling at lower frequencies, indicating that the proportion of signal power

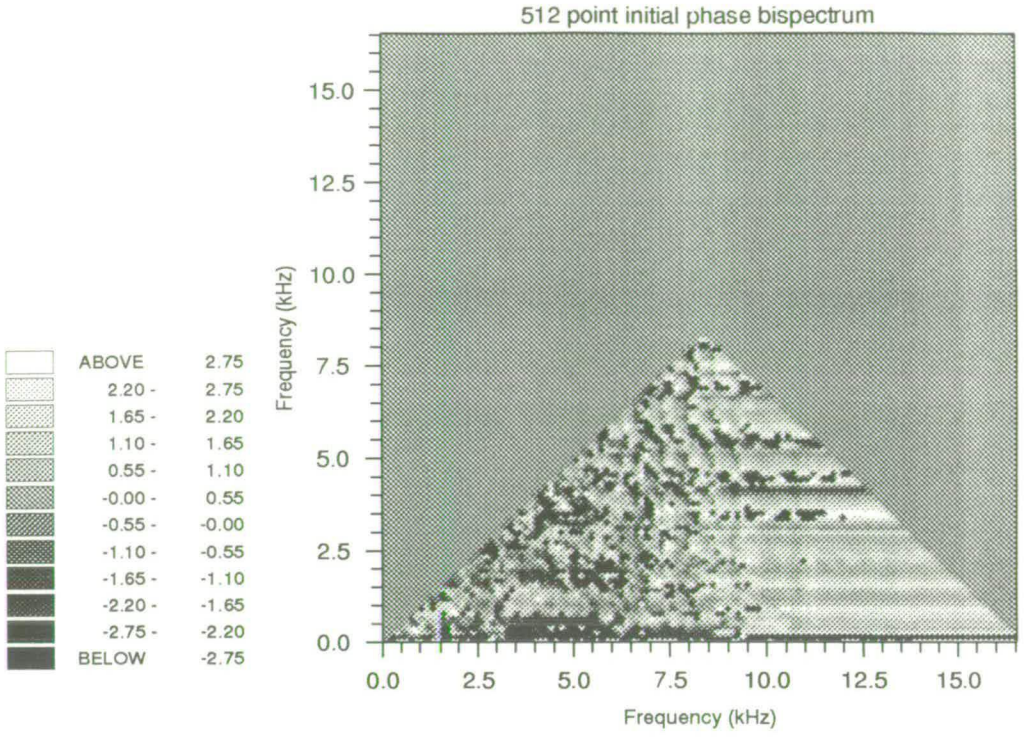


Figure 3.25: *Initial phase bispectrum: 33.3kHz Mid Tom (Butterworth).*

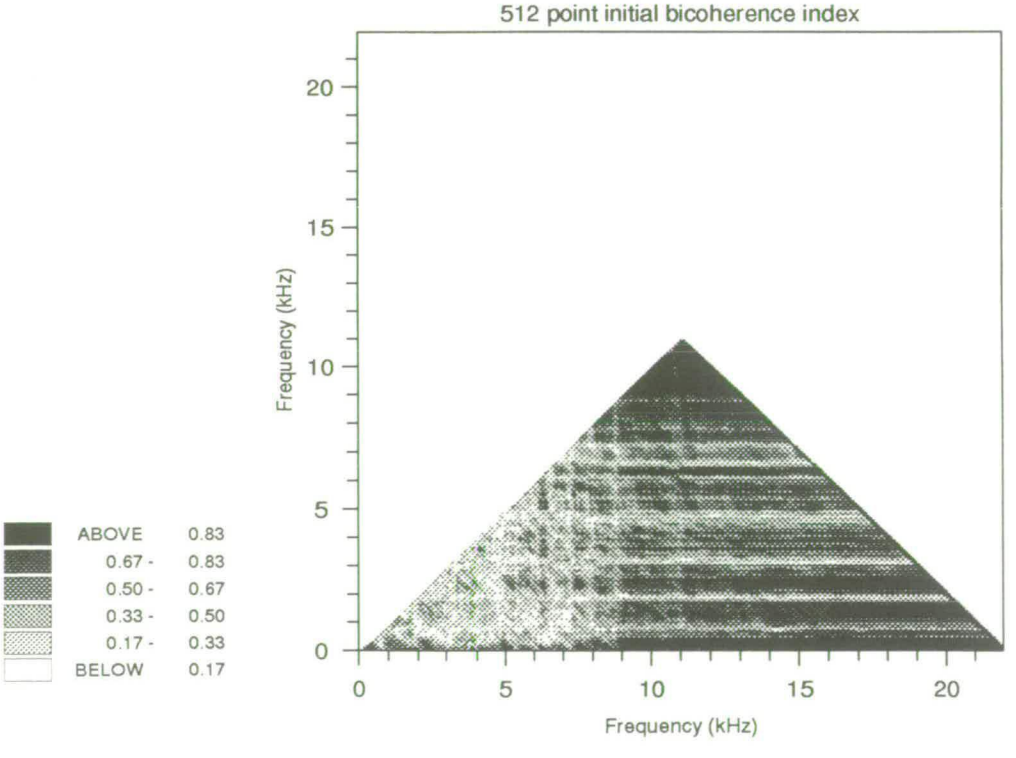


Figure 3.26: *Initial bicoherence index: 44.1kHz Mid Tom (Butterworth).*

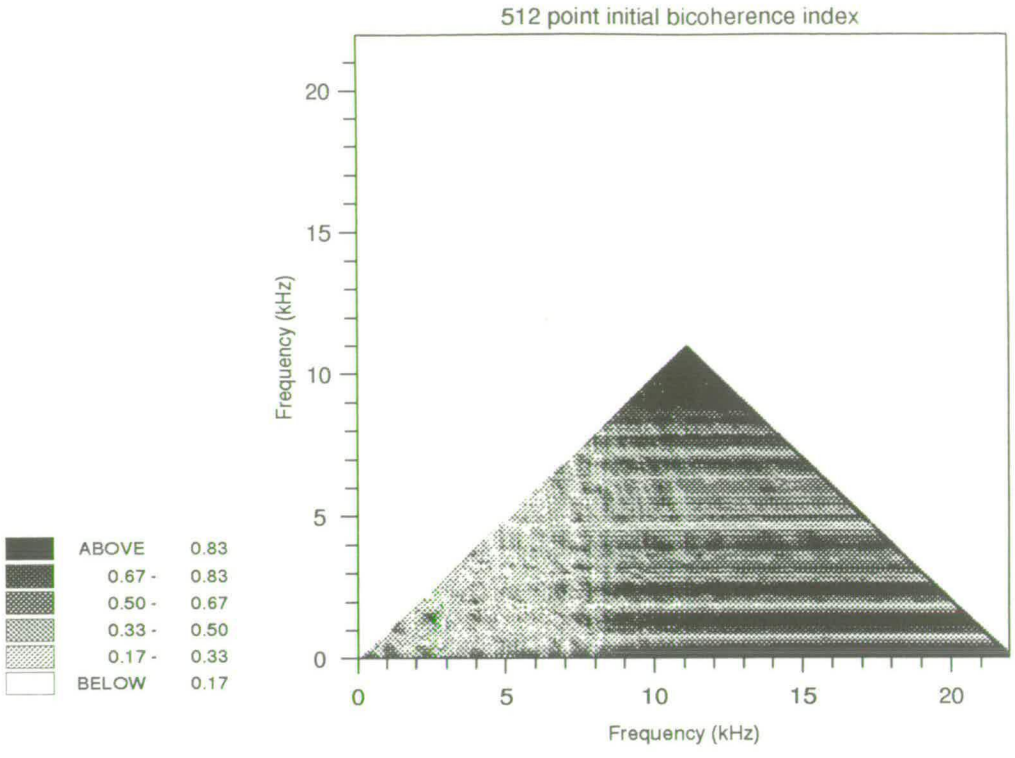


Figure 3.27: *Initial bicoherence index: 44.1kHz Mid Tom (Bessel).*

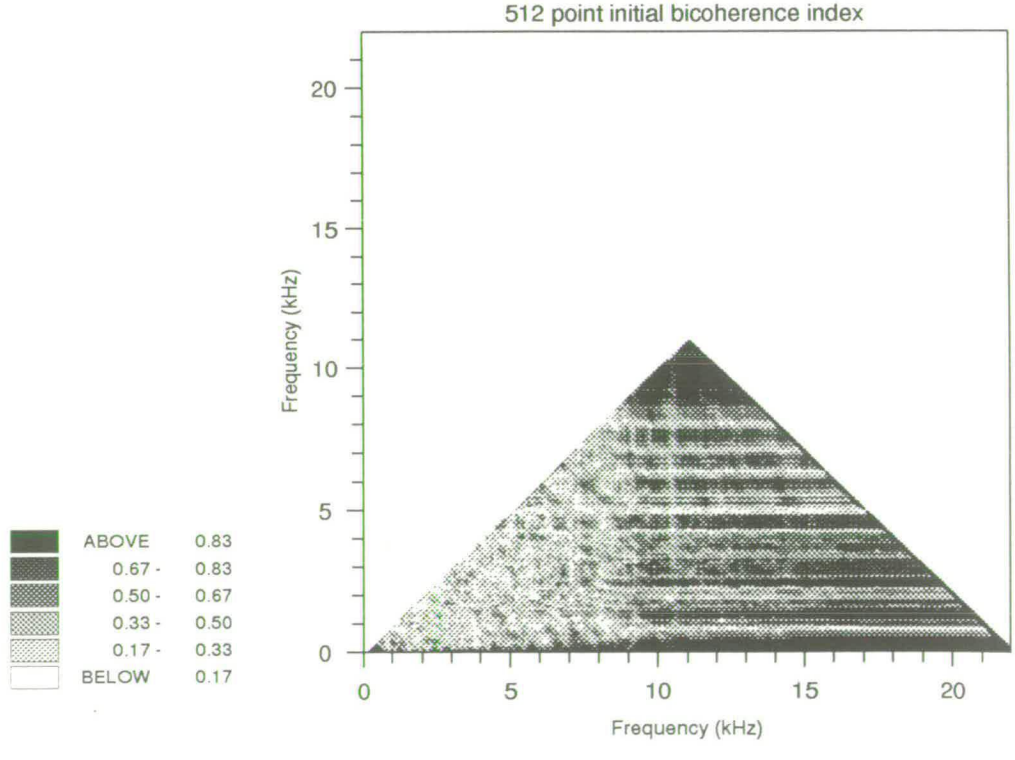


Figure 3.28: *Initial bicoherence index: 44.1kHz High Tom (Butterworth).*

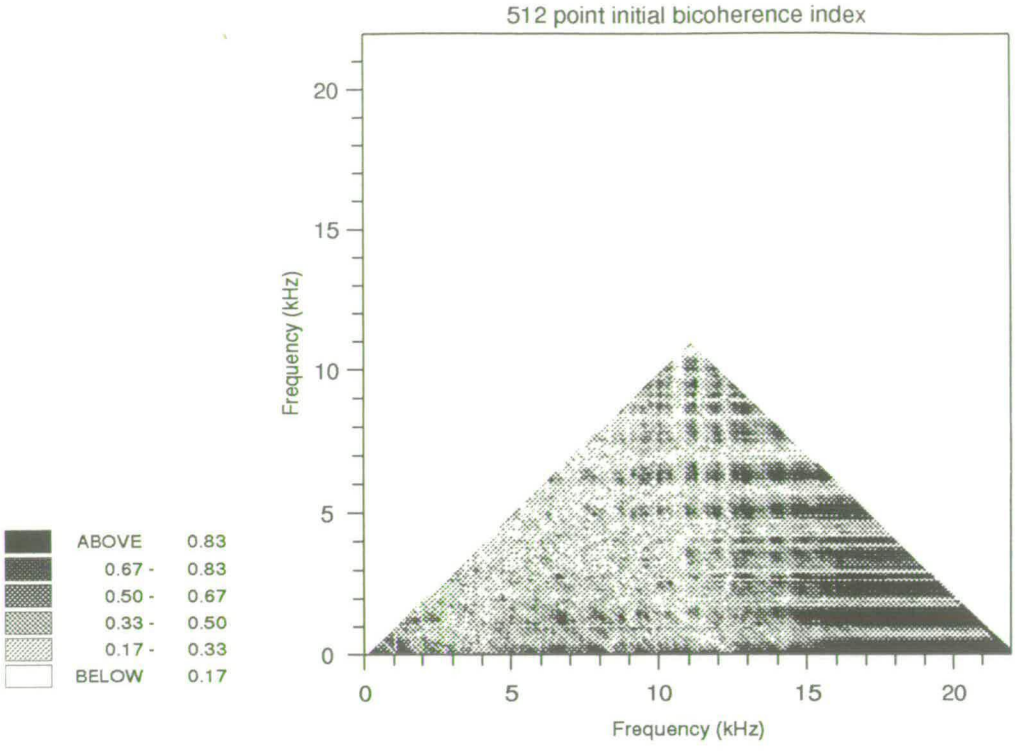


Figure 3.29: *Initial bicoherence index: 33.3kHz Mid Tom (Butterworth).*

present at lower frequencies due to phase coupling is negligible. At higher frequencies (values above approximately 10 kHz on the x-axis), it can be seen that the bicoherence index for each data record is dominated by high values, indicating that these components of the sound are entirely dependent on lower frequencies.

The initial percussive attack of a drum sound therefore comprises a number of independent low frequency oscillations which interact in a nonlinear fashion to produce high frequency harmonics. This behaviour is a significant departure from the Helmholz equations traditionally used to model drum vibrations (as discussed in chapter 2) which assumed a single underlying model generating all the signal harmonics.

Figures (3.30) and (3.31) show the bispectra and bicoherence index for the Mid Tom (sampled at 44.1 kHz, using a Butterworth filter) 10 ms later (i,e, after 20 ms of the sound have passed). The reduction in the amplitude of the power bispectrum evident in figure (3.30) can be anticipated from the signal

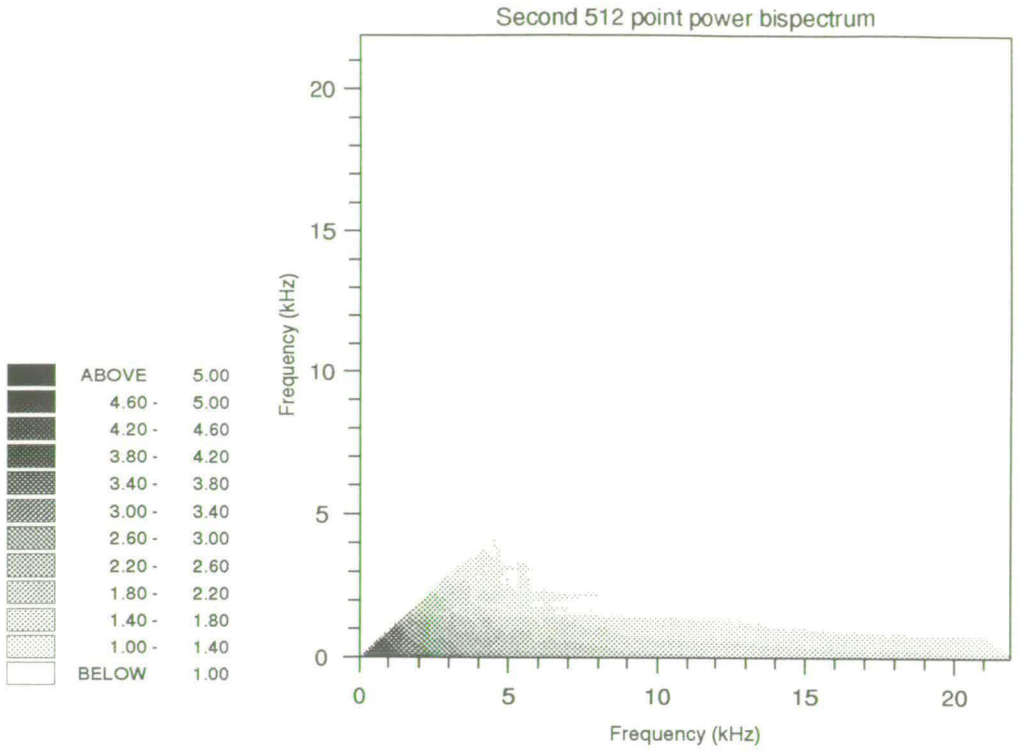


Figure 3.30: *512 point log power bispectrum after 20 ms*

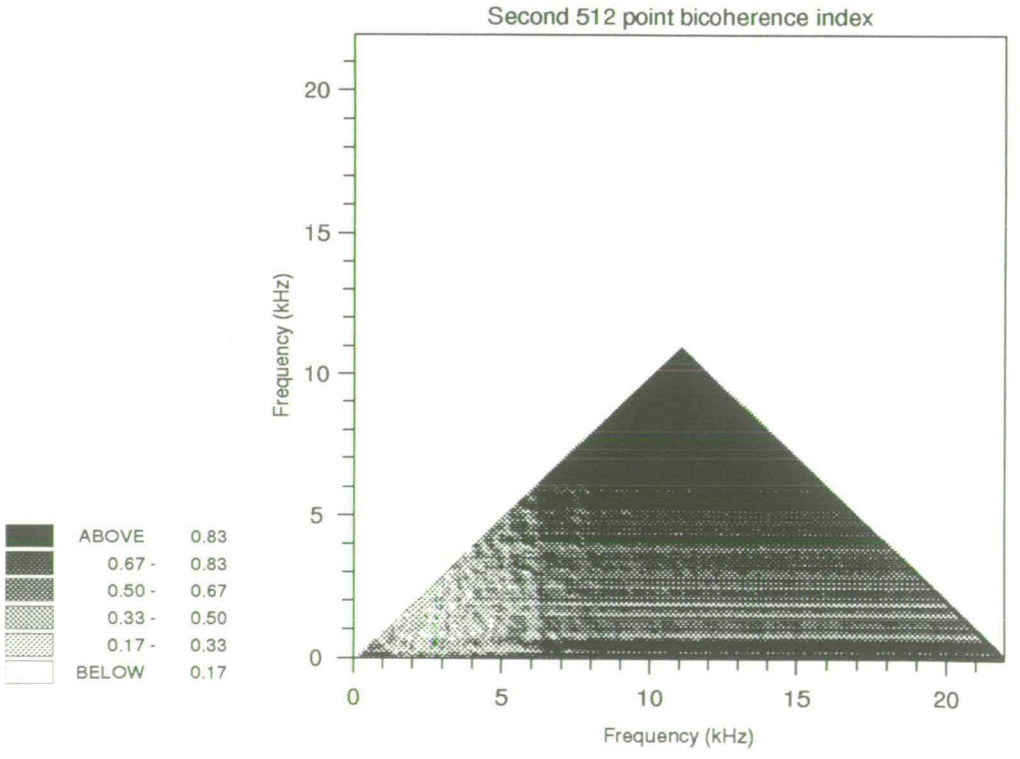


Figure 3.31: *Bicoherence index after 20 ms*

attenuation (especially at high frequencies) seen in the signal power spectrum. Having taken this into account, it can also be seen that the drum power bispectrum has become much smoother. Figure (3.31) shows that bicoherence has also changed noticably, with much more of the signal becoming inter-dependent, and having a correspondingly high bicoherence index. This trend continues in figures (3.32 and (3.32), which show the power bispectrum and bicoherence for the Mid Tom after 300 ms have passed. By this time, the power bispectrum has become a smooth slope, and the bicoherence index is almost uniformly unity. As discussed in the previous section, this is evidence of strong inter-frequency dependence between all components of the sound, and thus a single underlying system generating the entire signal. It is worth noting that the attenuation in signal power necessitates the different scaling of figure (3.32).

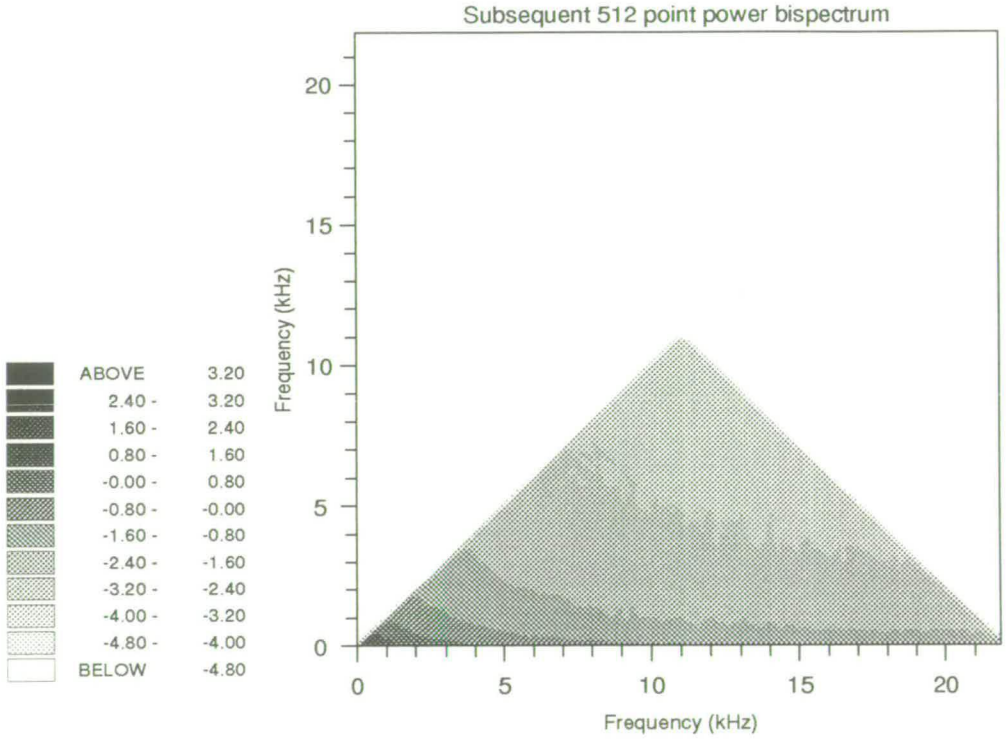


Figure 3.32: *512 point log power bispectrum after 300 ms.*

Having analysed the signal at different points in time, and discussed the reasons for the changes between these points, it is now possible to collate all

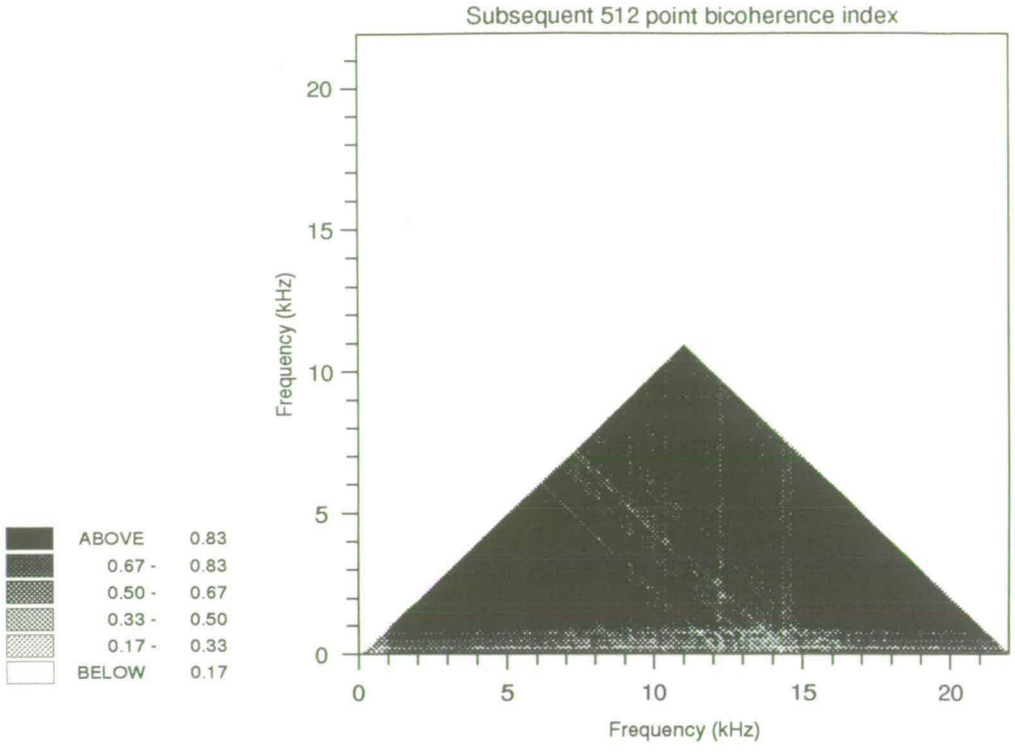


Figure 3.33: *Bicoherence index: 300 ms.*

the results together into a coherent picture describing a new model of the signal produced by the drum, and the underlying processes driving it.

The initial contact between the drumhead and the drumstick appears to last for approximately 1 ms. The excitation caused by this is highly nonlinear, and results in a number of independent, low frequency oscillations in the drum membranes. These interact in a nonlinear fashion, modulating one another to produce a wide range of high frequency harmonics. This forms the percussive attack. The damping effects of friction and air cause the signal to attenuate. In particular, high frequencies and thus the interactive nonlinearities are affected, resulting in the linearising of the signal, and thus in the membrane oscillation tending towards the Helmholz model described in chapter 2. In this model, the vibrations of all different modes are inter-dependent, resulting in a unity bicoherence index across the spectrum.

3.4 Conclusion

In this chapter the theory behind higher order spectral analysis has been examined, and the properties of the third order cumulant sequence and bispectrum studied. In particular, special attention has been paid to the phenomenon of quadratic phase coupling, and its relevance to the detection of nonlinearities and signal dependencies. It has been shown that any such phase coupling remains unaffected by any linear or nonlinear phase shifts incurred in sampling or processing of data. It has also been shown that phase coupling information can be used to differentiate between linear and nonlinear signal interactions, and to spot inter-signal dependencies.

These properties were then used to analyse the acoustic data sampled from two different drums, using various different sampling paths to ensure signal integrity. It has been shown that the initial percussive attack of the drum sound contains a significant amount of quadratic phase coupling, and that this phase coupling is linked to the generation of the higher frequencies during the initial percussive attack, indicating that the presence of high frequency vibration on the drum membrane is through nonlinear harmonic generation. Furthermore, it has been shown that the drum sound collapses to a linear model based on the Helmholz equations with the progression of time.

Chapter 4

Nonlinear Signal Modelling

4.1 Introduction

Linear filtering methods have had an enormous impact on the development of techniques involved in signal detection, estimation, and processing. The optimum solution to a modelling problem is often a nonlinear function of the observed data. Nevertheless, the simplicity - in terms of design and implementation - of a linear model frequently outweighs the inaccuracies caused by ignoring the nonlinearities. With the ever increasing computing power available to modern engineers, the complexity of the nonlinear modelling problem is no longer such a drawback. For those cases where linear solutions are inadequate, a nonlinear solution is now a practical possibility [35–39,41,42,44,71,72].

It has been shown that there is a significant degree of nonlinear phase coupling in acoustic percussion sounds, and that reproduction of this phase coupling is desirable in order to create a model that is convincing to the ear. This chapter commences with a brief overview of least squares methods, a mathematical tool well established in linear filter theory, and then goes on to look at two nonlinear structures, radial basis function networks and the Volterra series, both of which can be implemented as filters. The advantage that both of these structures possess is the ability to be trained using the least squares algorithm. It is shown that these nonlinear filters can, in a manner analogous to their linear counterparts, be turned round and used as models with which to generate a desired signal [43]. These techniques are applied to

the acoustic data, and it is shown that a second order Volterra filter is the more successful at regenerating the initial percussive attack of the drum sound. The drawbacks inherent in both nonlinear structures are discussed, and, in the light of these results, some new difficulties in succesfully synthesising drum sounds are observed.

4.2 Least Squares Filter Optimisation

Suppose $x(n)$ is a sample sequence from a stationary process from which it is desired to predict some desired response $d(n)$ (typically the desired response is the next sample in the sequence, i.e. $d(n) = x(n+1)$). A linear filter with embedding dimension p (one employing the current sample, $x(n)$ and the previous $p-1$ samples) used to predict $d(n)$ then has the form

$$\hat{d}(n) = \sum_{k=0}^{p-1} h_k x(n-k) \tag{4.1}$$

where $\hat{d}$ denotes an estimate, and $\{h_k\}$ is the filter coefficient associated with $x(n-k)$. This can be expressed more compactly in vector form:

$$\hat{d}(n) = H^T X(n) \tag{4.2}$$

where $H^T = [h_0, h_1, \cdots, h_{p-1}]$ and $X(n)^T = [x(n), x(n-1), \cdots, x(n-(p-1))]$. The error between the desired signal, $d(n)$, and the filter output, $\hat{d}(n)$ can be expressed as

$$\epsilon(n) = d(n) - \hat{d}(n) \tag{4.3}$$

In order to find the optimal linear filter for a given system it is necessary to minimise the power of this error signal [73].

$$\varepsilon = E\left[\epsilon^2(n)\right] \tag{4.4}$$

This mean-square value is a real and positive scalar quantity, representing the average power of the error signal $\epsilon(n)$. Substituting equation(4.3) into (4.4),

this becomes

$$\varepsilon = E\left[d^2(n)\right] - 2E\left[d(n)\hat{d}(n)\right] + E\left[\hat{d}^2(n)\right] \tag{4.5}$$

and a further substitution of equation (4.2) into (4.5) leads to

$$\varepsilon = E\left[d^2(n)\right] - 2HE\left[X(n)^T d(n)\right] + HE\left[X(n)X(n)^T\right] H^T \tag{4.6}$$

This can be rewritten as

$$\varepsilon = C_d - 2HC + HR_xH^T \tag{4.7}$$

Where C_d is equal to the mean-square value of $d(n)$, C is the cross correlation vector whose elements consist of the correlation between the desired response $d(n)$ and the input vector X, and R_x is the autocorrelation matrix $E[XX^T]$ comprising the mean square value of the individual filter inputs.

The mean squared error ε obtains its minimum value when its derivatives with respect to the filter weights are simultaneously zero. Differentiation of the expression for mean squared error displayed in equation(4.7) gives

$$\frac{\partial \varepsilon}{\partial H} = -2C + 2HR_x \tag{4.8}$$

Setting this result to zero, the optimum value of H is obtained:-

$$H_oR_x - C = 0 \tag{4.9}$$

rearranging to solve for H_o, this equation becomes the well known Wiener-Hopf equation:

$$H_o = R_x^{-1}C \tag{4.10}$$

This equation gives a method (known as least squares) of calculating the optimal coefficients for a desired filter response directly from the signal data. It has one drawback, namely the necessity of calculating the inverse autocorrelation matrix R_x^{-1}. This is typically achieved using singular value decomposition [73–75].

The next two sections examine two nonlinear functions, the Volterra series and radial basis function networks, and show how each, when used as a predictive filter, can be optimised using the least squares method.

4.3 The Volterra Series

4.3.1 Definitions

Equation (4.1) presented the structure of a non-recursive linear time invariant (LTI) filter. It is interesting to note that this LTI system has *memory*. The contribution to the present value of the output by the input k samples in the past is determined by h_k. If the system has no memory, then $h_k = 0$ if $k \neq 0$ and the output of the system is simply

$$\hat{d}(n) = h_0 x(n) \tag{4.11}$$

If we consider a similar no-memory series in which higher order moments of the input affect the output, then the system becomes a Taylor series.

$$y(t) = \sum_{p=0}^{\infty} h_p x^p(n) \tag{4.12}$$

An extension of this, to nonlinear systems with memory, is the Volterra series:

$$\begin{aligned} y(n) = \; & h_0 \\ & + \sum_{\tau_1^1=0}^{\infty} h_1(\tau_1^1)\, x(n-\tau_1^1) \\ & + \sum_{\tau_1^2=0}^{\infty} \sum_{\tau_2^2=0}^{\infty} h_2(\tau_1^2, \tau_2^2)\, x(n-\tau_1^2)\, x(n-\tau_2^2) \\ & + \sum_{\tau_1^3=0}^{\infty} \sum_{\tau_2^3=0}^{\infty} \sum_{\tau_3^3=0}^{\infty} h_3(\tau_1^3, \tau_2^3, \tau_3^3)\, x(n-\tau_1^3)\, x(n-\tau_2^3)\, x(n-\tau_3^3) \\ & + \ldots \end{aligned} \tag{4.13}$$

In a Volterra series such as the one shown above, the functions $h_i(\tau_1, \ldots, \tau_i)$ are called the Volterra kernels of the system. In the same way that a LTI system

is completely characterised by its unit impulse response, so a nonlinear system represented by a Volterra series is completely characterised by its Volterra kernels. With an argument similar to that for linear systems [76], it can be shown that the nonlinear system is causal if and only if $h_n(\tau_1, \ldots, \tau_n) = 0$ for any $\tau_j < 0$ where $j = 1, \ldots, n$.

A Volterra filter of order p (where p is a positive integer) is a nonlinear system with an input-output relationship that can be described by a finite Volterra expansion of the type shown in equation (4.3.1). In such a filter, $x(n)$ and $y(n)$ would denote the filter input and output respectively, and $H_i(\tau_1^i, \ldots, \tau_i^i)$ would represent the i- dimensional filter weight sequence associated with the ith Volterra kernel.

The constant h_0 corresponds to $k = 0$ and is often omitted, or set to zero [77]. When this is the case, it can be seen that by setting $p = 1$, the expression obtained is a standard convolution, characterising a linear filter.

4.3.2 Properties of the Volterra series

In analysing Volterra systems, it is frequently necessary to perform operations that involve reordering the τ's within Volterra kernels. Analysis would therefore be simplified if only symmetric kernels need to be considered, as the specific order of τ's then becomes unimportant. Considering the second order kernel, for example

$$h_2(\tau_1, \tau_2) = h_2(\tau_2, \tau_1) \tag{4.14}$$

Fortunately, it has been shown [78] that any asymmetric kernel can be symmetrised. There is therefore no loss of generality in considering only symmetric kernels.

Another way of considering the Volterra series is :

$$y(n) = h_0 + H_1[x(n)] + H_2[x(n)] + \cdots + H_i[x(n)] + \cdots \tag{4.15}$$

where

$$H_i[x(n)] = \sum_{\tau_1=0}^{\infty} \cdots \sum_{\tau_n=0}^{\infty} h_n(\tau_1, \ldots, \tau_n) x(n-\tau_1) \cdots x(n-\tau_i) \tag{4.16}$$

In this representation, the symbol H_i is called an ith order Volterra operator. A system represented by Volterra operators up to order p is therefore known as a pth order Volterra series.

The Volterra series is a power series with memory. This can be shown by increasing the input by a gain factor c. Using the Volterra series representation in equation (4.15), we can calculate the new output:

$$\begin{aligned} y(n) &= h_0 + \sum_{n=1}^{\infty} H_n[cx(n)] \\ &= h_0 + \sum_{n=1}^{\infty} c^n H_n[x(n)] \end{aligned} \tag{4.17}$$

which is a power series in the amplitude factor c.

As a consequence of its power series character, there are some inevitable limitations to the use of Volterra series in filtering problems. In particular, the Volterra series suffers from the same convergence problems traditionally associated with the Taylor series [76], which is to be expected, as the Taylor series is a subset of the Volterra series (see section 4.1). A second problem associated with Volterra filters is the measurement of the Volterra kernels of a given system [79]. In order to characterise a given Volterra system, it is necessary first to separate the contribution of each operator from the total system response. Characterisation of linear systems is found from the impulse response. The system impulse response for the Volterra series can be found by substituting $x(n) = Au_0(t)$ in (4.15). The resultant nonlinear system impulse response is

$$y(n) = h_0 + Ah_1(n) + A^2h_2(n,n) + A^3h_3(n,n,n) + \cdots \tag{4.18}$$

The impulse response of each operator is determined by $h_i(\tau_1, \ldots, \tau_i)$ only for $\tau_1 = \tau_2 = \cdots = \tau_i$, and therefore does not completely specify the associated kernel. To specify a kth order kernel, the system response to a string of k impulses with varying delays between them must be measured. This involves a total of 2^p separate measurements [76,78]. As a result, measurement of the Volterra kernels of a system becomes increasingly complicated for higher order systems, and is only possible where the number of kernels is finite. Fortunately it is possible to rewrite the Volterra series in a form to which the least squares algorithm discussed earlier can be applied. This is demonstrated in the next section.

4.3.3 Calculation of Volterra Filter Coefficients Using the Least Squares Algorithm

In order to use the least squares algorithm with a nonlinear series such as the Volterra series, it is necessary to express this series as a vector inner product, in a form similar to that shown in equation (4.2). For a pth order filter, this can be displayed as:

$$\hat{d}(n) = H_{vp} X_{vp}(n)^T \tag{4.19}$$

where H_{vp} is a vector comprising all the coefficients of the Volterra filter from the p Volterra kernels, and X_{vp} is a vector containing the associated combinations of the input terms. This technique is best illustrated with an example. Consider a 2^{nd} order Volterra filter used to predict the next sample $x(n+1)$ of a drum data record. Such a filter might have an input observation vector of length 3:

$$X(n)^T = [x(n), x(n-1), x(n-2)] \tag{4.20}$$

Being second order, the filter would have two operators; a linear (first order) operator

$$H_1^T = [h_1(1), h_1(2), h_1(3)] \tag{4.21}$$

and a quadratic (second order) operator:

$$H_2 = \begin{bmatrix} h_2(11) & h_2(21) & h_2(31) \\ h_2(12) & h_2(22) & h_2(32) \\ h_2(13) & h_2(23) & h_2(33) \end{bmatrix} \tag{4.22}$$

A block diagram of this can be seen in figure (4.1).

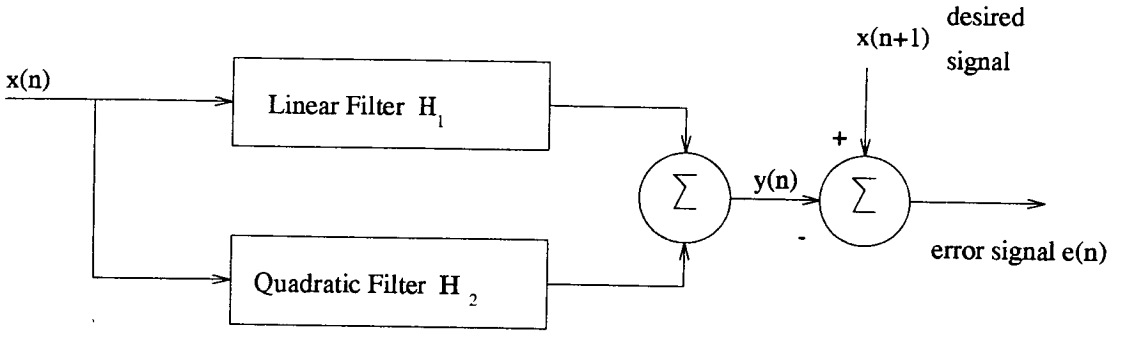


Figure 4.1: *Second order Volterra filter - block diagram*

From equations (4.15) and (4.16) it can be seen that the corresponding combinations of terms from $X(n)$ for H_1 would be

$$X_1(n)^T = [x(n), x(n-1), x(n-2)] \tag{4.23}$$

and for H_2 would be

$$X_2 = \begin{bmatrix} x(n)x(n) & x(n-1)x(n) & x(n-2)x(n) \\ x(n)x(n-1) & x(n-1)x(n-1) & x(n-2)x(n-1) \\ x(n)x(n-2) & x(n-1)x(n-2) & x(n-2)x(n-2) \end{bmatrix}. \tag{4.24}$$

The second order Volterra kernel H_2 contains redundant terms as described in equation (4.14). They are $h_2(12)$ (equivalent to $h_2(21)$), $h_2(13)$ (equivalent to $h_2(31)$) and $h_2(23)$ (equivalent to $h_2(32)$). The corresponding terms in X_2 are thus also unneccessary.

Upon combining H_1 and the non redundant terms from H_2 as described in equation (4.19),the resulting vector inner product becomes

$$\hat{d}(n) = H_{v2} X_{v2}(n)^T \tag{4.25}$$

where

$$H_{v2} = [h_1(1), h_2(11), h_1(2), h_2(21), h_2(22), h_1(3), h_2(31), h_2(32), h_2(33)] \quad (4.26)$$

and

$$\begin{aligned} X_{v2} = [&x(n), x(n)x(n), x(n-1), x(n-1)x(n), x(n-1)x(n-1), \\ &x(n-2), x(n-2)x(n), x(n-2)x(n-1), x(n-2)x(n-2)] \end{aligned} \quad (4.27)$$

As H_{v2} and X_{v2} are both vectors, the least squares algorithm detailed in section (4.2 can be applied to equation (4.25) as it was to equation (4.2) in order to find the optimal Volterra filter coefficients H_{v2o}. The resultant filter is displayed in figure (4.2).

It is worth noting at this point that the total number of coefficients required for a pth order Volterra filter increases dramatically as the value of p grows - the kth kernel within such a filter is a symmetric n^k tensor for an observation vector of length n, and as such contains $\frac{n(n+1)(n+2)\cdots(n+k-1)}{k!}$ non-redundant terms. The total number of coefficients required for the series is therefore

$$N_v p = \sum_{k=1}^{p} \left[\frac{\prod_{i=0}^{k-1}(n+i)}{k!} \right] \quad (4.28)$$

The next section looks at another nonlinear function, the radial basis function network. It examines how that too can be configured as a filter, and subsequently trained using a similar least squares strategy to that for the Volterra series.

4.4 Radial Basis Function Networks

4.4.1 Definitions

A radial basis function (RBF) network (figure (4.3)) can be regarded as a two layer neural net which is constrained to be linear in all parameters by confining all nonlinearities to a hidden layer. This hidden layer performs a fixed nonlinear

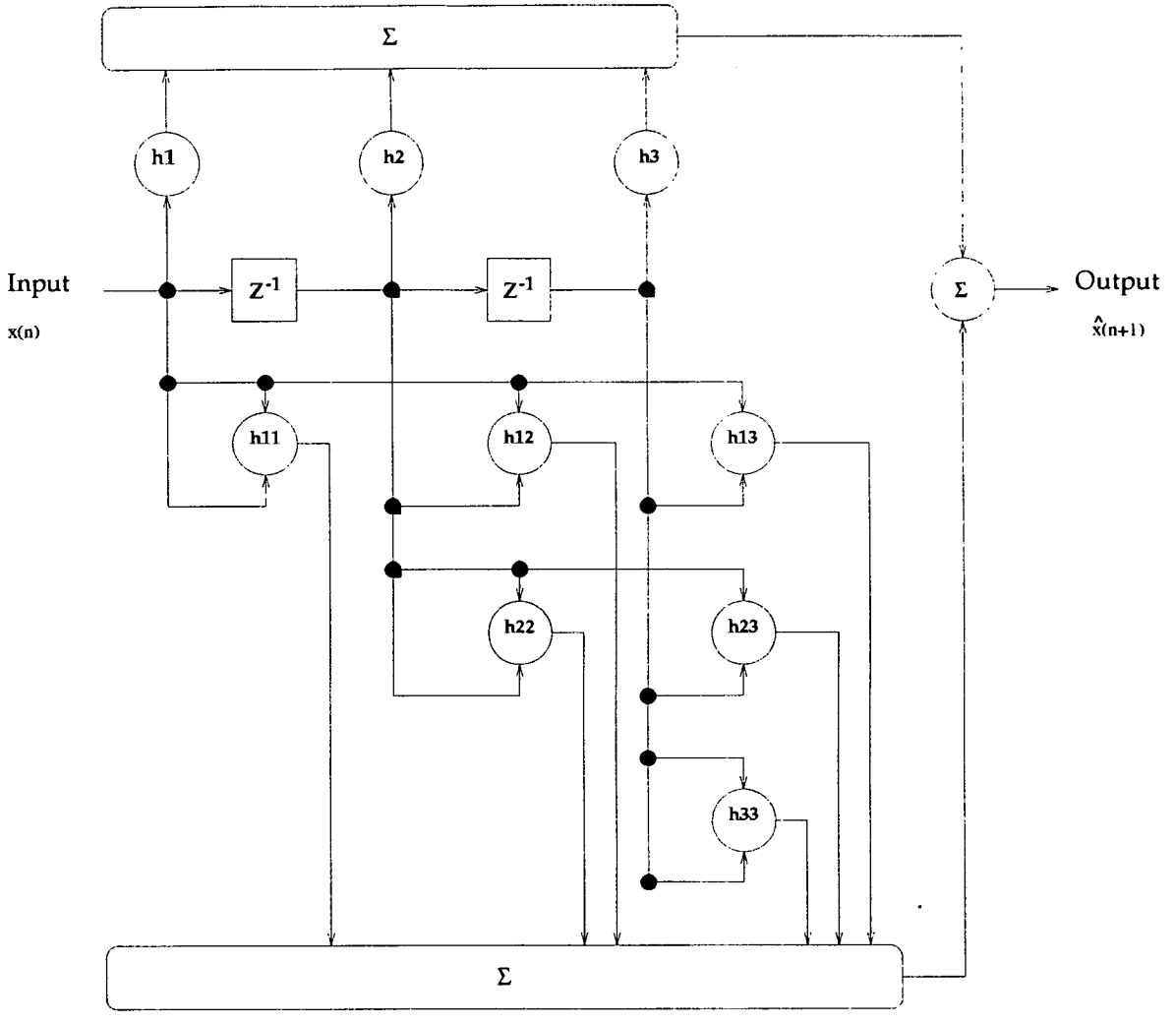


Figure 4.2: *Second order Volterra filter with redundancy removed*

transformation with no adjustable parameters, and maps the input space onto a new space. The output layer then implements a linear combiner on this new space. The weights of this combiner are the adjustable parameters of the system, and these can be chosen using a least squares technique.

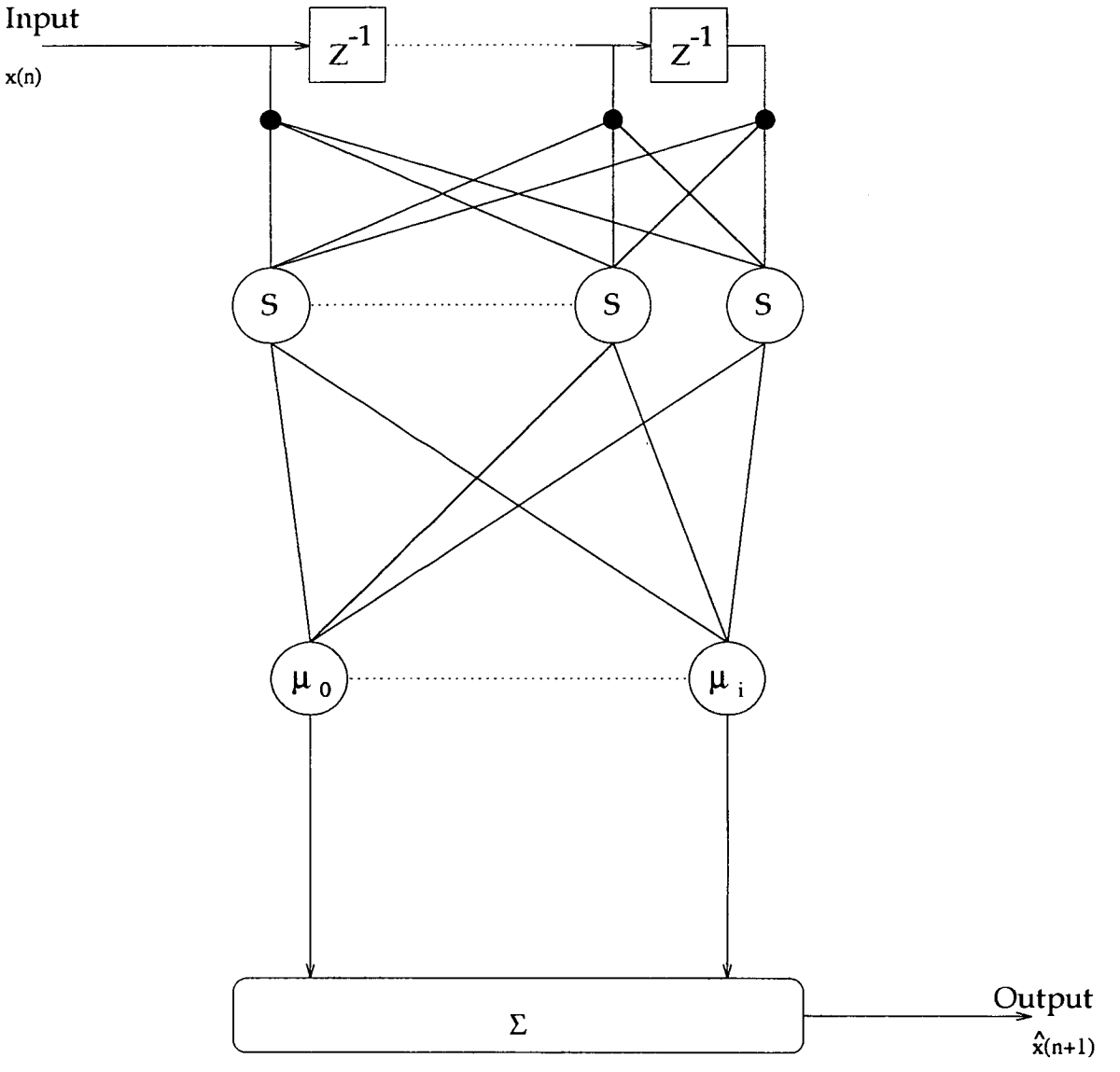


Figure 4.3: *Schematic of a radial basis function network*

An RBF network with N centres can be defined as follows [46]:

$$\hat{x}(n+1) = h_0 + \sum_{i=0}^{N-1} h_i s(||X_n - C_i||) \tag{4.29}$$

where X_n is the nth input vector used to predict the next sample x_{n+1}, C_i is the ith "centre" vector, $||.||$ represents the Euclidean norm, and $s(.)$ is an appropriately chosen nonlinear function. Although the length of the centre vectors is dependent upon the embedding dimension of the filter, it can be seen that the number of centres is not. Choice of an appropriate number of centres is therefore a parameter that must be considered when employing RBF networks.

As the values of the centres and the nonlinearity $s(.)$ are fixed, the only trainable parameter is the weighting h_i associated with each nonlinear mapping. From equation (4.29), the output from each node can be seen to be

$$l_k(n) = s\left(||X_n - C_k||\right) \tag{4.30}$$

It can be seen that the relevance of the output from a particular node is determined by the distance between X_n and that centre. $\hat{d}(n)$ therefore depends largely on those nodes with centres which are close to the input, i.e. the basis functions in the hidden layer only produce a significant response when the input falls within a small localised region of the input space. It is consequently important that an appropriate set of centres is chosen in order to cover the input signal space completely, and in the past, much consideration has gone into different methods of selection. Standard iterative techniques such as the popular κ-means clustering algorithm [80,81] are unfortunately inapplicable, as the severe nonstationarity of the data set rules out the availability of enough material with which to train any such algorithm. Most commonly used non-iterative methods are based around choosing some arbitrary points from the input signal space as centres [46,47,49]. Due to the random nature of this technique, it is frequently necessary to choose a large number of centres in order to ensure optimum performance.

Another apparently important consideration is in the choice of the nonlinearity $s(.)$. In practice, however, it has been shown that this is not crucial to the performance of the RBF network [45,49]. Typical functions used in the past

include the the thin plate spline [48]

$$s(x) = x^2 \log(x) \tag{4.31}$$

and the Gaussian function [45]

$$s(x) = \exp(-x^2/\beta^2) \tag{4.32}$$

where β is a real constant. In equation (4.31), $s(\infty) \rightarrow \infty$, and in (4.32) $s(\infty) \rightarrow 0$. Although these two nonlinearities have quite different properties, both the resulting RBF networks have good approximation capabilities [49].

4.4.2 Calculation of RBF Network Coefficients using the Least Squares Algorithm

As with the Volterra series, it is necessary to express the RBF network as a vector inner product. In this case, this is far more intuitively obvious, as the RBF network is a linear sum of all the nodes. The node outputs $l_i(n)$ as expressed in equation (4.30) can be written in vector form, and equation(4.29) rewritten as a vector inner product:

$$\hat{d}(n) = H_{rbf} L(n)^T \tag{4.33}$$

where $H_{rbf} = [h_1, \cdots, h_N]^T$, and $L(n) = [l_1(n), \cdots, l_N(n)]$.

Having thus linearised the filter equation, the least squares technique can then be applied to calculate the optimal RBF coefficients, H_{orbf}.

4.5 Testing

Volterra filters and RBF networks were designed as specified in the previous section. To test their performance relative both to one another and also to a simple linear (i.e. first order Volterra) filter and more importantly, to verify the functionality of each filter, it was necessary to run them with a selection of test signals. Given the nature of the signal that these filters were ultimately intended to model, various criteria were important. First, the filters had to be able to perform well with only a small training set. Second, the filters should require only a small embedding dimension (i.e. observation vector) from which to predict the signal. Overall, the most important factor was the ability of the nonlinear predictor to perform *regeneratively*. When used predictively, the original signal is input to the filter, which calculates an estimation of the next sample value. Used regeneratively, this estimate of the next sample value is fed back to the filter input at each step, and is then used as the next input sample. The filter thus regenerates an estimation of the original signal, and acts as a model of that signal. The difference between these two filter structures is shown in figure (4.4).

Figures (4.5) to (4.9) show Volterra filters and RBF networks run on a variety of different test signals. All the filters performed well when used for next step prediction. The differences between the filters began to show, however, when they were used to resynthesise the test signals regeneratively. The first signal used was a straightforward sinewave. It is known ([82,83]) that this can be modelled successfully using a linear filter with embedding dimension two. Results (shown in figures (4.5) and (4.6)) were measured for two different signals - a high frequency sinewave, $y(x) = \sin(\frac{3x}{5\pi}\pi)$ and a low frequency wave, $y(x) = \sin(\frac{x}{15\pi}\pi)$. Note that in each case, irrational multiples of π were used to ensure that no value of $y(x)$ was ever repeated. Volterra filters of orders up to five successfully modelled the high frequency signal, but diverged rapidly on the low frequency signal. The RBF network performed poorly in both cases,

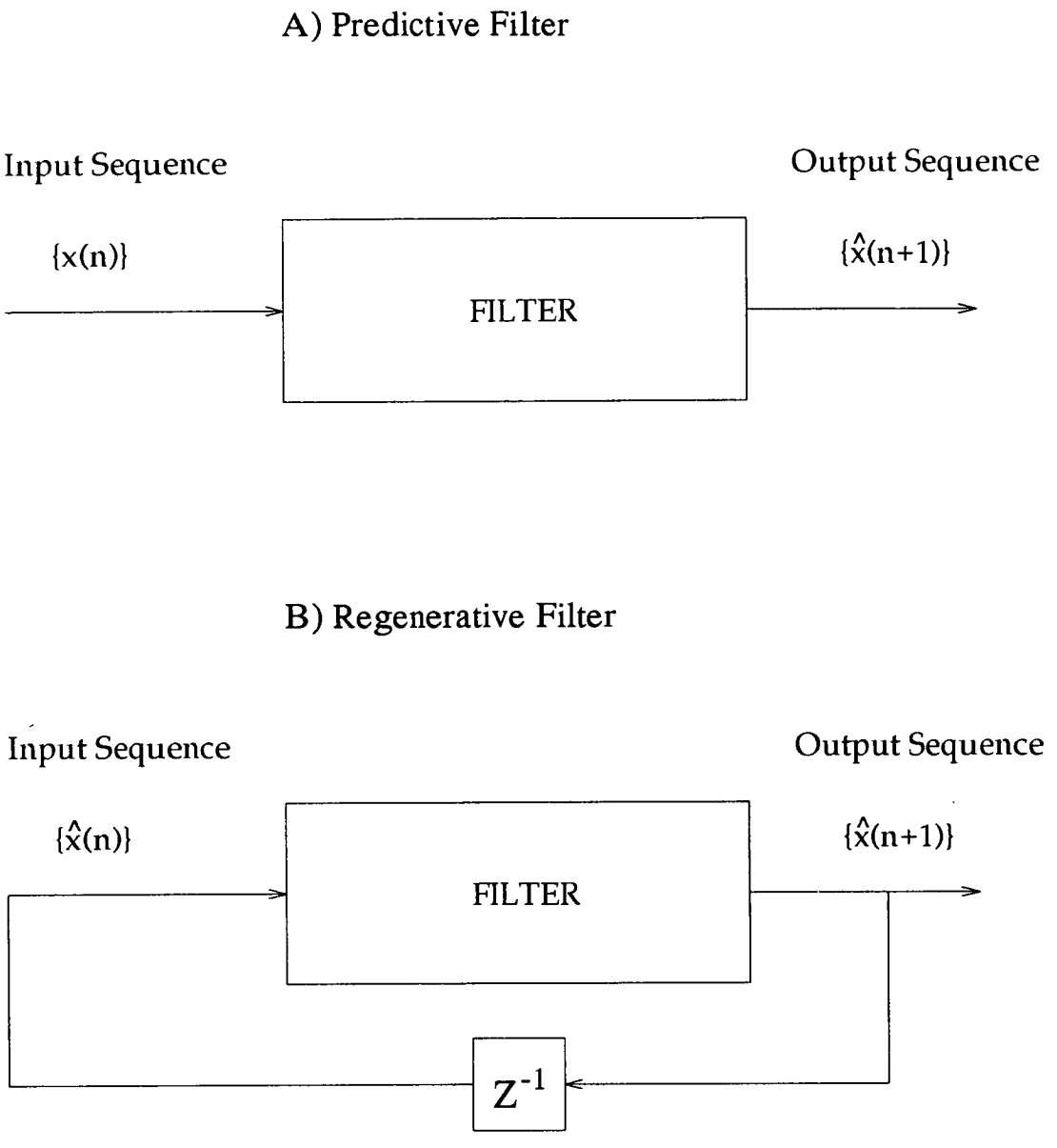


Figure 4.4: *Block diagrams of a) predictive and b) regenerative filters*

also finding the low frequency signal harder to track. It should be borne in mind at this point that the RBF network suffers from the necessity of choosing its centres on a random basis:- in order to ensure a reasonable spread of centres across the signal vector space, it is necessary to choose a large number of centres. The resultant surplus of non-critical centers serves little purpose except to add noise to the network output. Using coefficient reduction techniques such as orthogonal least squares learning or noise threshold weight reduction to pare down the number of centers can markedly improve the performance of the RBF network dramatically. Further discussion of these can be found in chapter 5.

The second test signal was a squared sine wave: $y(x) = 0.75\left(\sin(\frac{3x}{32\pi}\pi)\right)^2$. Using standard trigonometric identities, this can be expressed as: $y(x) = 0.375\left[\cos(0) - \cos(\frac{6x}{32\pi}\pi)\right]$. Consequently this test measures the ability of the filters to track a non zero mean signal. Figure (4.7) shows some of the results of this test using an embedding dimension of 3.

Again, the RBF network was unable to regenerate the signal. The linear filter shown in the diagram also failed to regenerate the signal, but by increasing the embedding dimension to 7 a stable linear model could be generated. The most successful model shown is that of a second order Volterra filter, as could be expected given prior knowledge of the filter's design. This would indicate that a linear filter is capable of reproducing non-linear signals, but requires a larger embedding dimension then an equivalent higher order filter. This contention is borne out by the results obtained from the final signal, a quadratic crossmodulation of the type examined in chapter 3, which has been shown to appear during the percussive attack of drum sounds. Figure (4.8) shows the results obtained for this test. The signal used was $y(x) = \sin(\frac{x}{27\pi}\pi)\sin(\frac{x}{11\pi}\pi + \frac{\pi}{2})$ In this case, the results shown are for a linear filter with embedding dimension 14, an RBF network with embedding dimension 14, and a second order Volterra filter with embedding dimension 8. Once again, the RBF network failed to regenerate the signal, whilst the two other techniques were successful. For this more complex test, it is worth examining the error signal $e(x) = y(x) - \hat{d}(x)$ for

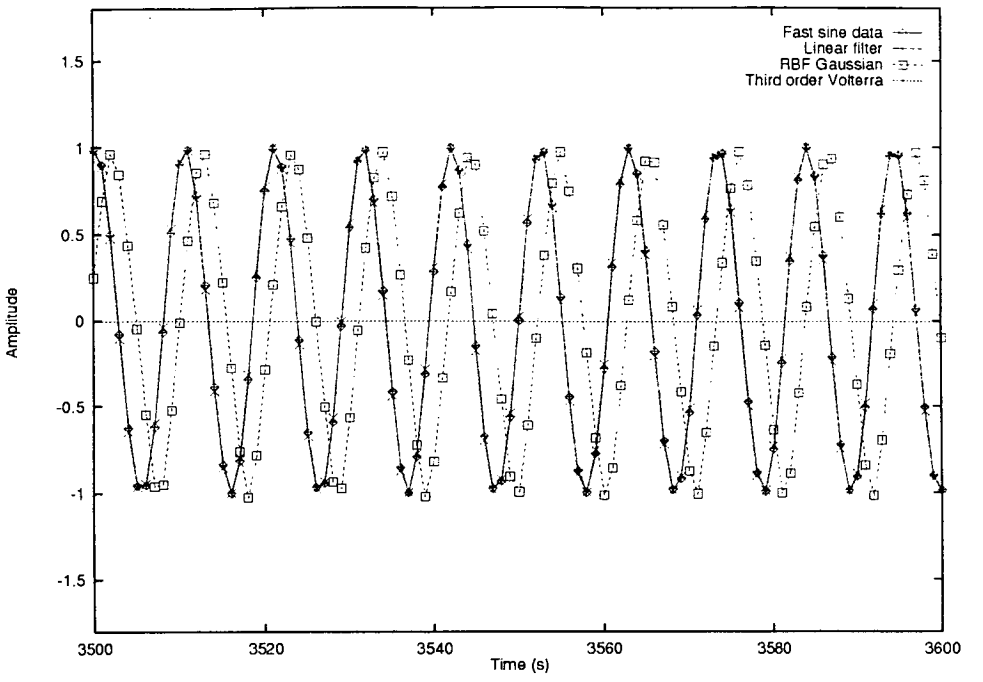


Figure 4.5: *Performance test: regeneration of fast sine wave. Each filter was trained on 3500 data points. All filters used an embedding dimension of 2. A third order Volterra filter is shown here, although higher orders modelled the signal successfully. The RBF network had 4 centres, and used a Gaussian nonlinearity with* $\beta = [||X_n - C_i||]_{max}$

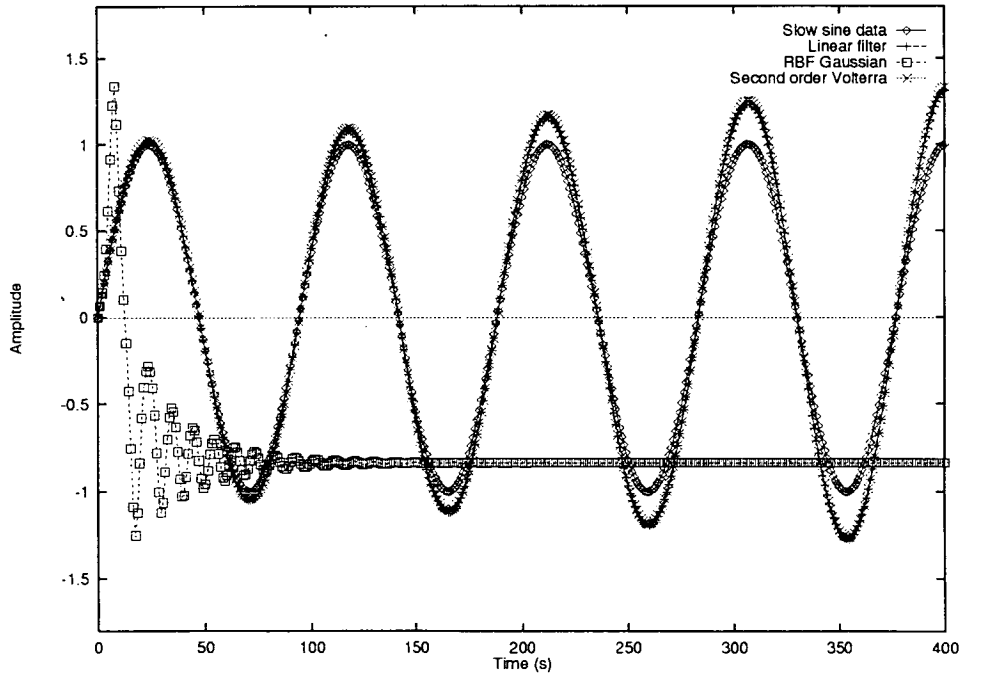


Figure 4.6: *Performance test: regeneration of slow sine wave. Each filter was trained on 3500 data points. All filters used an embedding dimension of 2. The Volterra filter shown here is second order. Higher order Volterra filters were unable to regenerate the signal with any degree of success.*

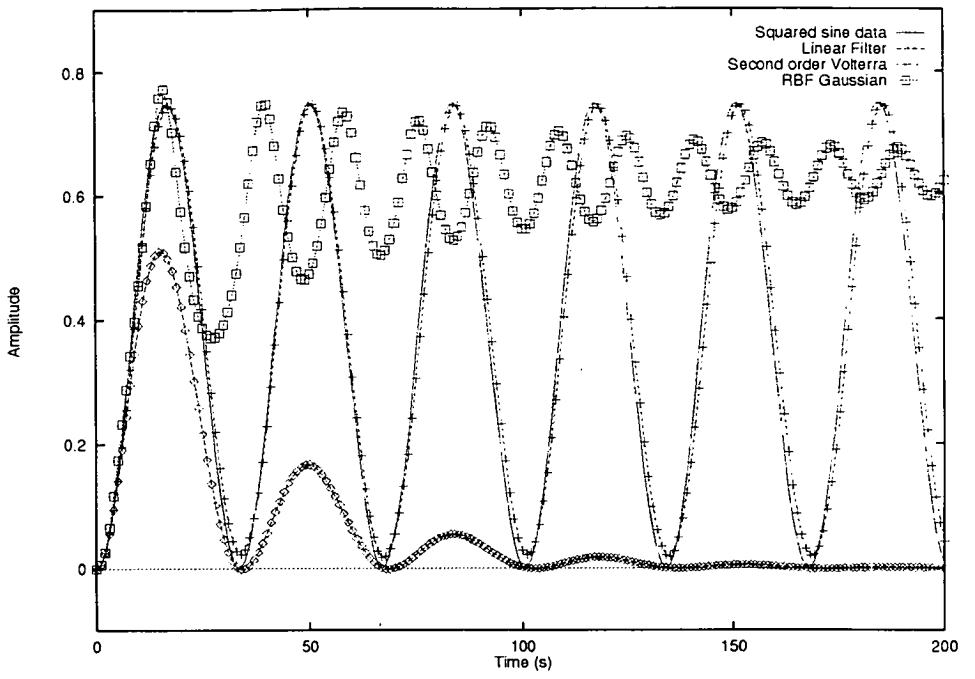


Figure 4.7: *Performance test: regeneration of squared sine wave. Each filter was trained on 3500 data points using an embedding dimension of 3.*

the linear filter and the Volterra filter. This is shown in figure (4.9). It can be seen that the error signals for the linear and second order filters are comparable for the first 300 - 350 points (a more detailed analysis shows that the quadratic error is in fact slightly smaller than the linear error). The quadratic nature of the second order filter's error, however, becomes dominant after 350 points, and the filter rapidly becomes unstable. The linear filter's error also increases, but in a linear fashion, and the linear filter models the signal reasonably accurately for a further 2000 points.

From these tests it can be concluded that the different filter types have both advantages and disadavantages. For the purposes of transient signal modelling, the RBF network suffers from a lack of coherent centre selection techniques. Nonlinear Volterra filters can present a system model with a small embedding dimension, but when used regeneratively, the generation of higher order moments from the input signals can lead to rapid instability.

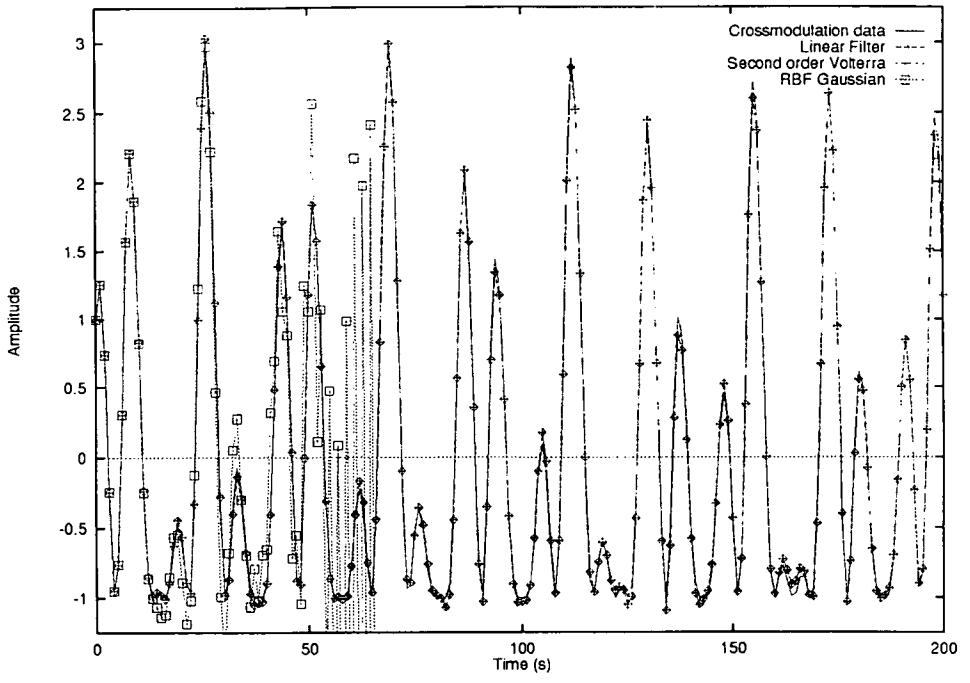


Figure 4.8: *Performance test: regeneration of crossmodulation data. Each filter was trained on 3500 data points. The linear filter and RBF network used an embedding dimension of 14, whilst the second order Volterra filter used an embedding dimension of 8.*

Non-recursive linear filters do not suffer from this drawback, but often need long embedding vectors for more complicated signals. This may prove to be impractical when dealing with transient or nonstationary signals.

4.6 Predictive Filtering

The first decision to be made when considering the modelling and estimation of the sampled data was to choose an appropriately sized embedding vector. There has been a considerable body of research in this field with regard to linear filter structures [73, 82, 83], but very little which is applicable to other filter structures. The test results presented in the previous section indicated that an embedding vector of length 8 might be an appropriate starting point for

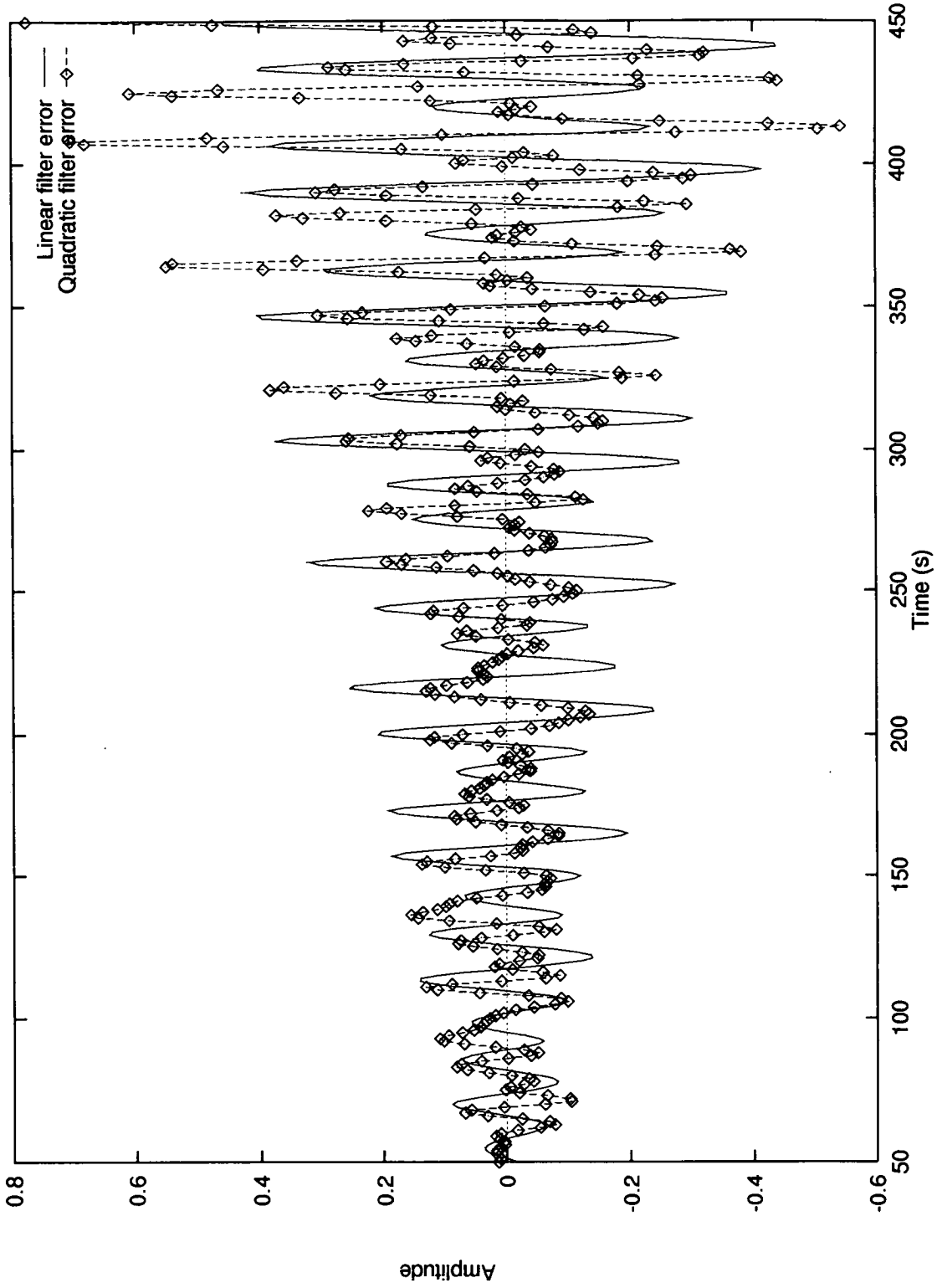


Figure 4.9: *Performance test: crossmodulation data regeneration error curves*

the sampled acoustic data, as one of its characteristic features has been shown to be crossmodulation. Figure (4.10) shows the resultant output of linear and second order Volterra filters used as next step predictors. Each graph shows two plots, one for a filter trained on one thousand points, and the other for one trained on two hundred points. In these and subsequent plots, the mean square error (MSE) over n samples between the predicted output and the actual output is defined to be:

$$\bar{e}_{ms} = 10 \log_{10} \frac{E\left[e(n)^2\right]}{E\left[d(n)^2\right]} \tag{4.34}$$

where

$$E\left[d(n)^2\right] = \frac{1}{n} \sum_{i=1}^{n} d(n)^2 \tag{4.35}$$

and

$$E\left[e(n)^2\right] = \frac{1}{n} \sum_{i=1}^{n} e(n)^2 \tag{4.36}$$

A quick glance at these two graphs is enough to show that the linear filter easily outperforms the second order filter. MSE values for these and subsequent graphs of the filters operating predictively can be found in section A.2of appendix A.

In chapter 2, it was observed that the initial dip in the signal amplitude was probably indicative of the contact between the drum membrane and drum stick. By training filters only on that part of the signal in which the drum membrane is oscillating freely, improved results can be obtained. These can be seen in figure (4.11). Despite this improvement, the second order Volterra filter is still visibly worse at predicting the signal. In order to verify these results, 1800 similar tests were carried out, for all the data records collected. As well as varying the training length, the effect of changing the embedding dimension was also examined. The full results of all these trials can be found in section A.3 of appendix A. Various conclusions can be drawn from these tests. As observed above, the linear (first order) Volterra filters consistently outperformed second order filters. Closer examination of the results also shows

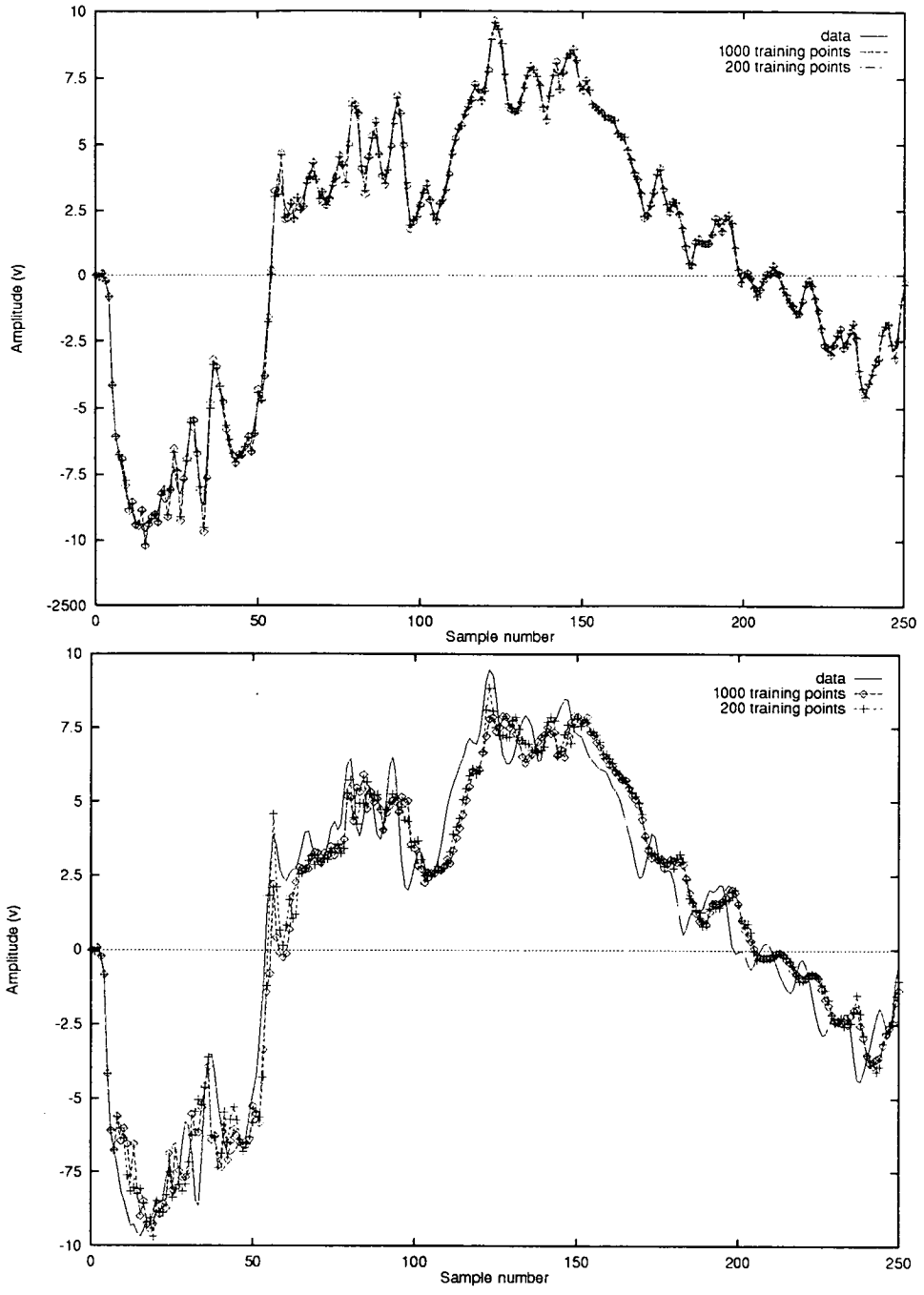


Figure 4.10: *Comparison of predicted signal against original signal with embedding dimension 8. The top graph shows first order Volterra filters with training lengths 1000 and 200. The lower graphs shows second order Volterra filters with similar training lengths.*

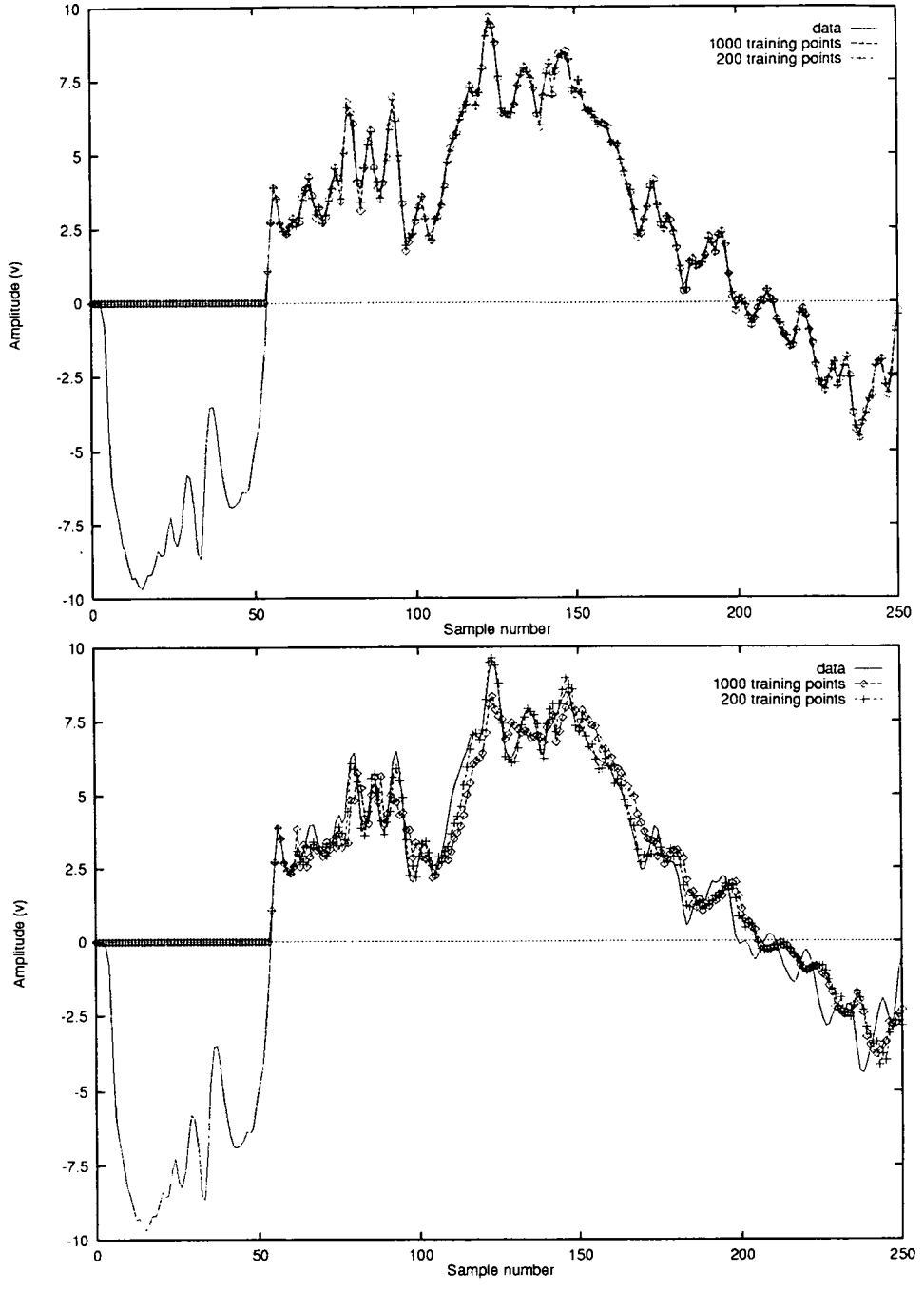


Figure 4.11: *Performance of filters trained on data without initial excitation period. The top graph shows first order Volterra filters, and the lower graph second order filters. Again, training lengths of 1000 and 200 points have been used.*

that for a given linear filter, the MSE after 100 samples is very close to the MSE after 2000 samples, regardless of training data length. It can also be seen that varying the filter embedding dimension had little effect in the linear case. Given the nonstationarity of the data, it can be concluded that the linear filters predictive capability relies on the close correlation between neighbouring samples. Conversely, the performance of the second order filter depends to a large extent on the embedding dimension and the training data length, with MSE rising rapidly when the filter attempted to predict points beyond those that it was trained on. Training the second order filter on fewer data results in a more accurate prediction of that data.

As indicated earlier, tests using RBF networks were also carried out on the data. Figures (4.12) and (4.13) show plots for RBF networks comparable to those shown previously for the Volterra filters. As with the Volterra filters, the RBF networks used an embedding dimension of 8 with training lengths of 1000 and 200 points. The two graphs in each figure show results for RBF's with eight centres and sixteen centres.

It can be seen that the RBF networks also performed poorly in comparison with linear filters. Once again, to verify these results, 1080 further trials were performed on all the data records, varying training data length and embedding dimension, and these results can also be found in appendix A. From these, it can be seen that, like the second order Volterra filter, the RBF network is sensitive to both training length and embedding dimension, although its performance is consistently worse than the second order Volterra filter.

4.7 Regenerative Filtering

As has been stressed previously, the Volterra filters and RBF networks were tested with the aim of using these filters regeneratively (as described at the beginning of section 4.5). The results obtained in the previous section indicated

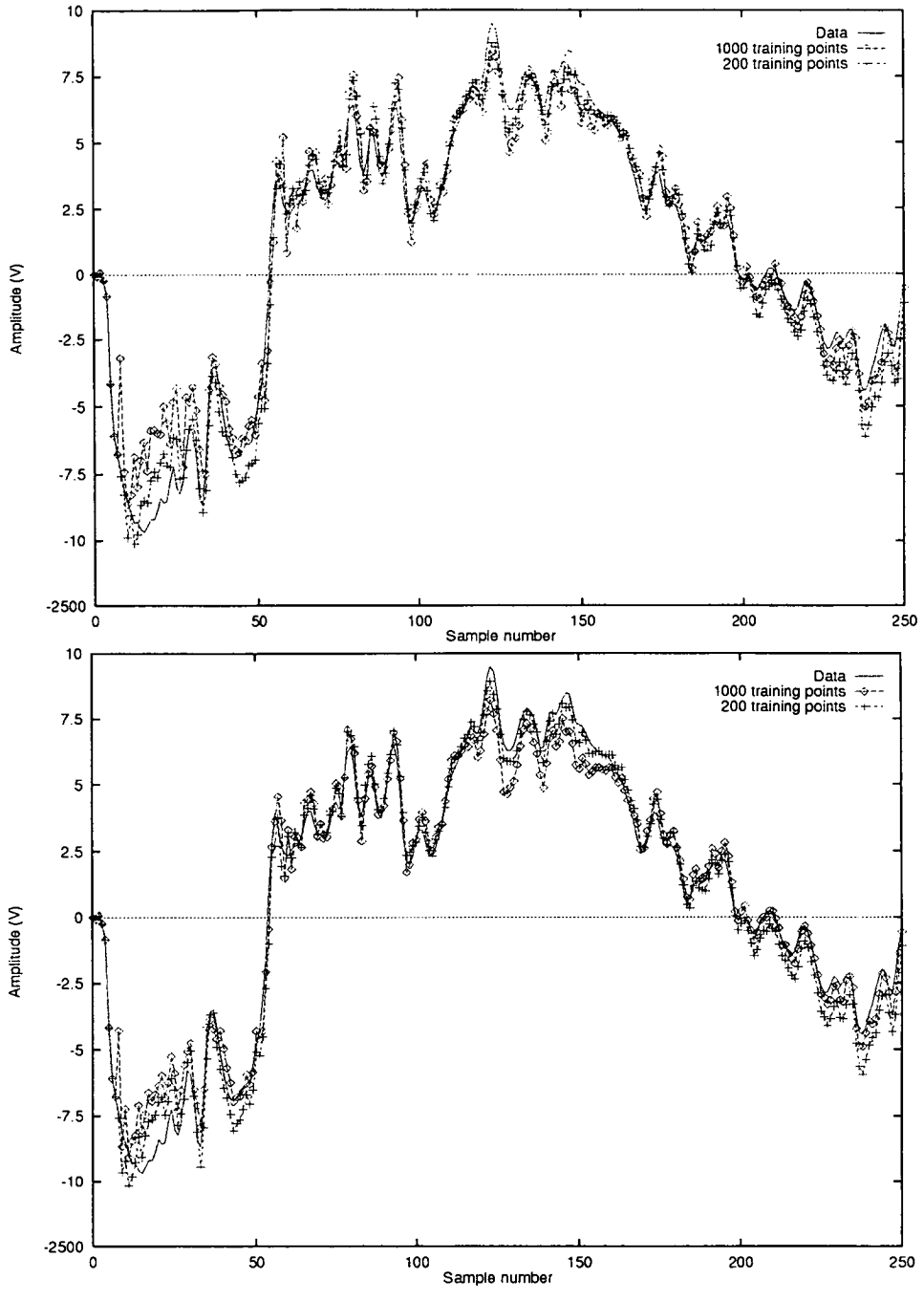


Figure 4.12: *Comparison of predicted signal against original signal with embedding dimension 8. The top graph shows 8 centre RBF networks using a Gaussian nonlinearity with training lengths 1000 and 200. The lower graphs shows similar 16 centre RBF networks.*

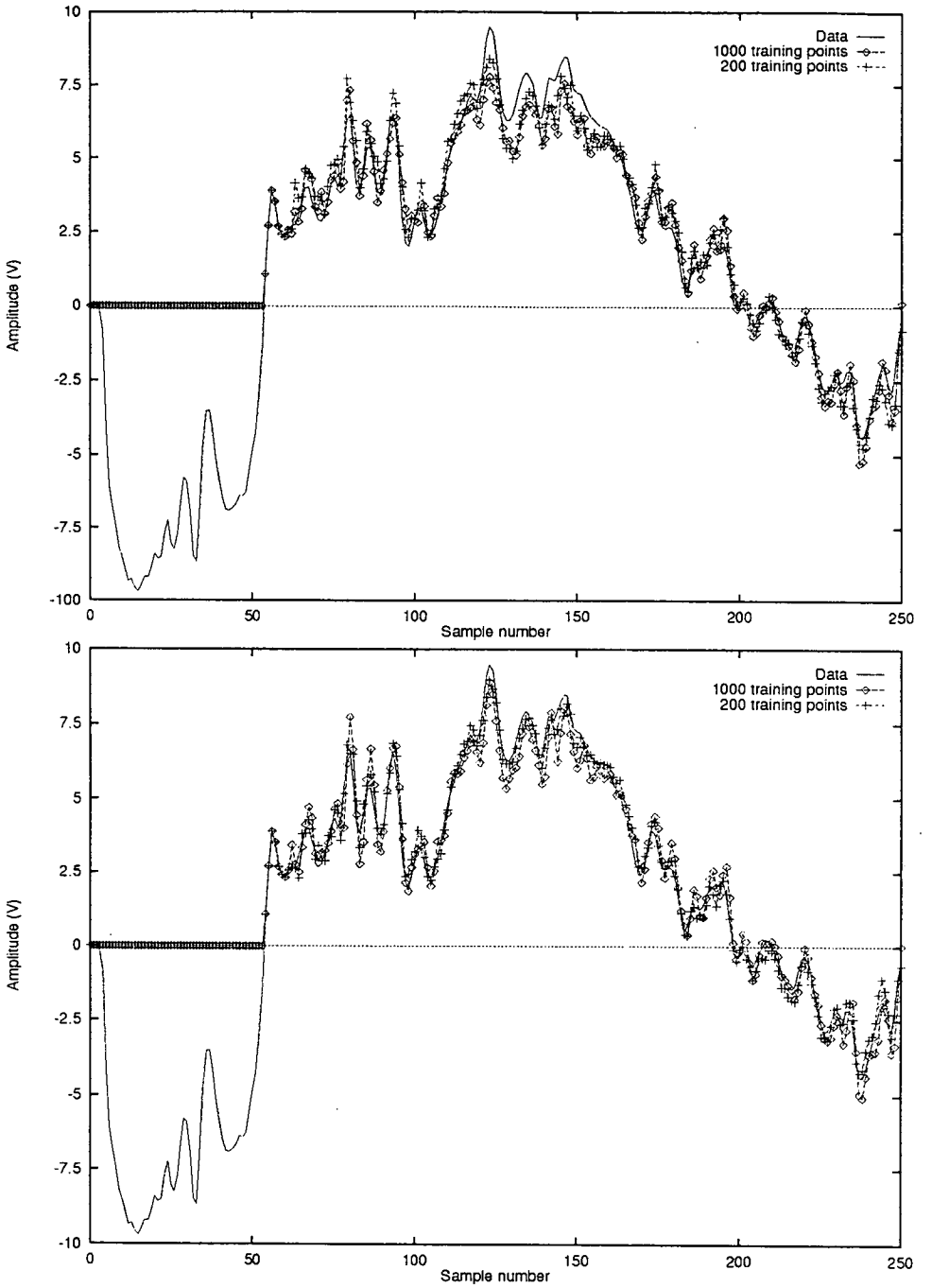


Figure 4.13: *Performance of RBF networks trained on data without initial excitation period. The top graph shows 8 centre networks, and the lower graph 16 centre networks. Again, training lengths of 1000 and 200 points have been used.*

that the nonlinear filters performed best when using small training data lengths. Using this information as a starting point, first and second order regenerative Volterra filters were trained and run on all the data records using the same three different embedding vectors as before, and trying different data training lengths, of 30, 50 and 70 samples. The results of these 1800 tests are shown in full in appendix A. Some typical results for Volterra series with embedding dimensions 8 and 16 are shown in figures (4.14) and (4.15).

In both graphs, the second order filter is much more succesful than the linear filter at regenerating the data. It can also be seen that, as predicted in section 4.5, the second order filter diverges rapidly following the onset of instability. It was hypothesised in section 4.6 that the success of the first order filter was largely due to the close correlation between adjacent samples. This is clearly not the case when using second order filters, as second order filters are unable to regenerate data beyond that upon which they are trained. Increasing the training length degrades the performance of the filter, since the assumption of stationarity becomes inaccurate.

Figure (4.16) shows results for a Volterra filter with embedding dimension 16 trained upon the same signal, this time with initial excitation data. It can be seen that the filter is unable to regenerate the signal beyond the initial excitation period. Increasing or decreasing the training length resulted in performance degradation, as explained above. In addition, the initial excitation period, as noted earlier, is completely different from the rest of the percussive attack as the drumstick is still in contact with the membrane. Taking these factors into consideration leads to the conclusion that it is neccessary to skip the initial excitation when assessing filter performance in order to give a more accurate indication of the performance of the same filter structures when trained on subsequent sections of the percussive attack.

From section A.3 in appendix A, it can be seen that the best results were obtained using embedding dimensions of 12 or 16 with training lengths of 30 and 50 respectively. Since Volterra filters of these dimensions contain 91

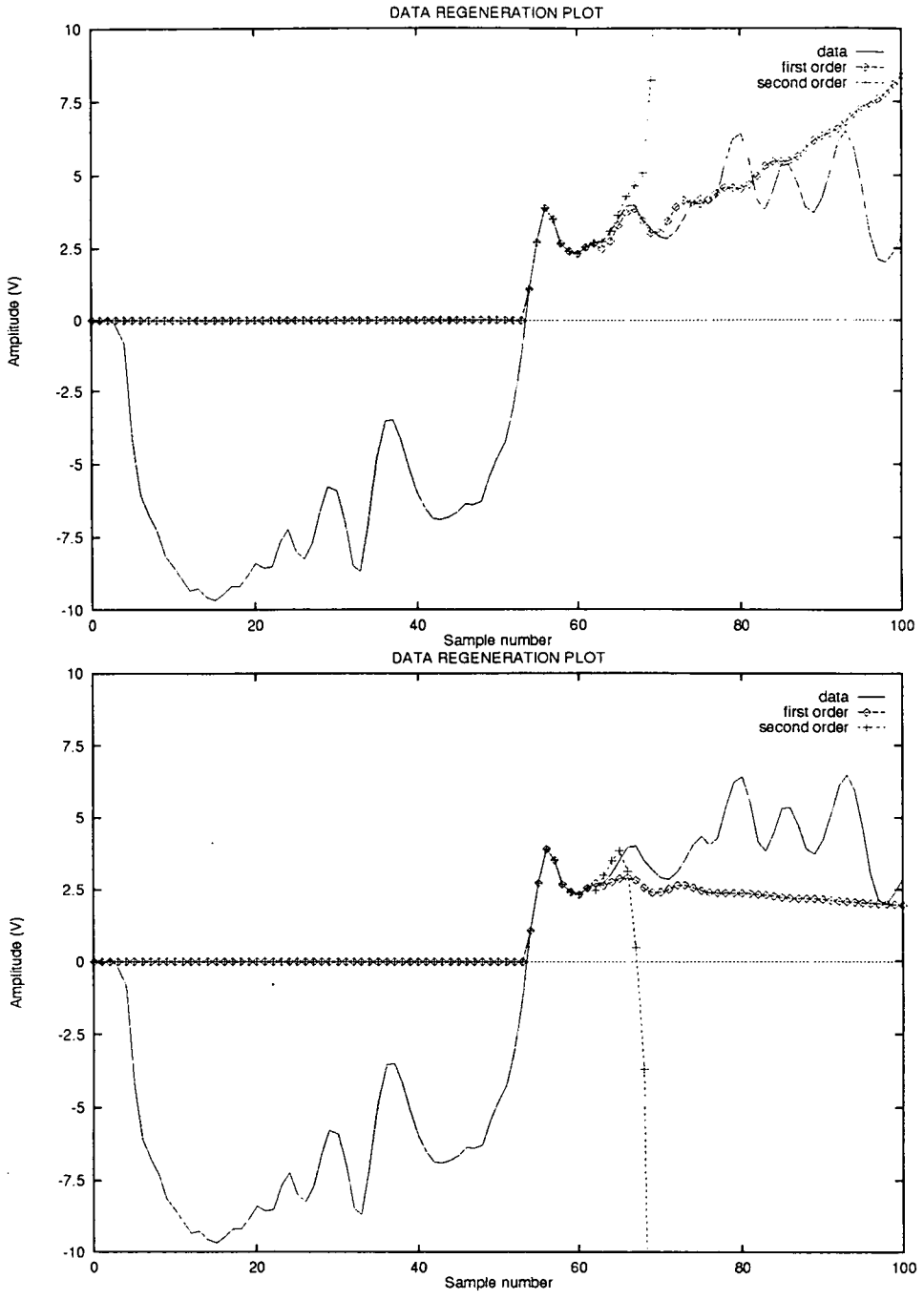


Figure 4.14: *Performance of Volterra filters with embedding dimension 8 trained on data without initial excitation period. The top graph shows results for filters with training length 30, and the bottom one shows results for filters trained on 50 samples.*

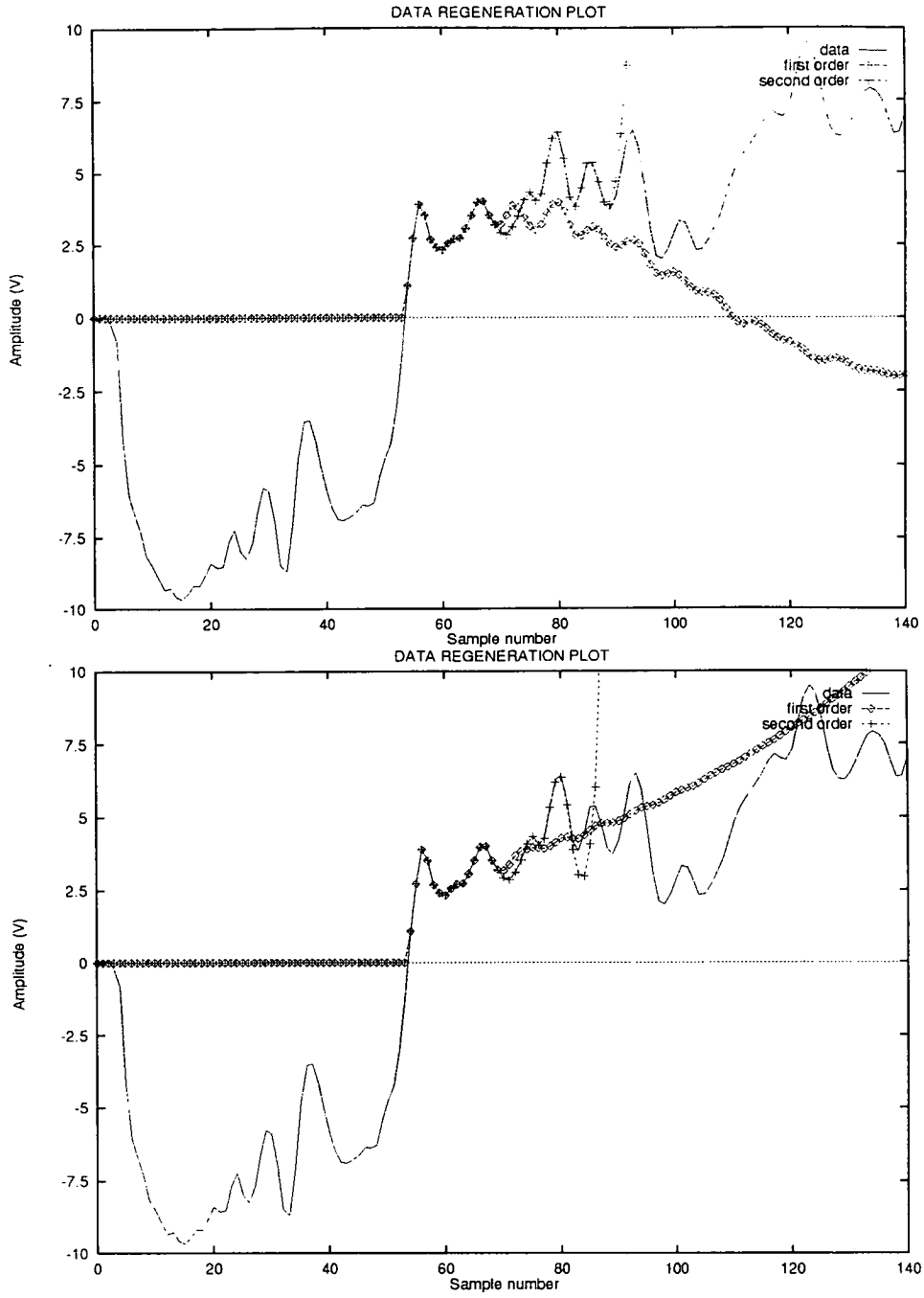


Figure 4.15: *Performance of Volterra filters with embedding dimension 16 trained on data without initial excitation period. The top graph shows results for figures with training length 50, and the bottom one shows results for filters trained on 70 samples.*

and 153 coefficients respectively (see equation (4.28)), it can be concluded that, although the model is able successfully to regenerate the acoustic signal, it does so very inefficiently, requiring approximately 3 coefficients for every sample regenerated. The very nature of the Volterra series makes it likely that there is significant redundancy present in the model. In the next chapter, methods of removing these redundancies and hence improving the efficiency of the model are examined.

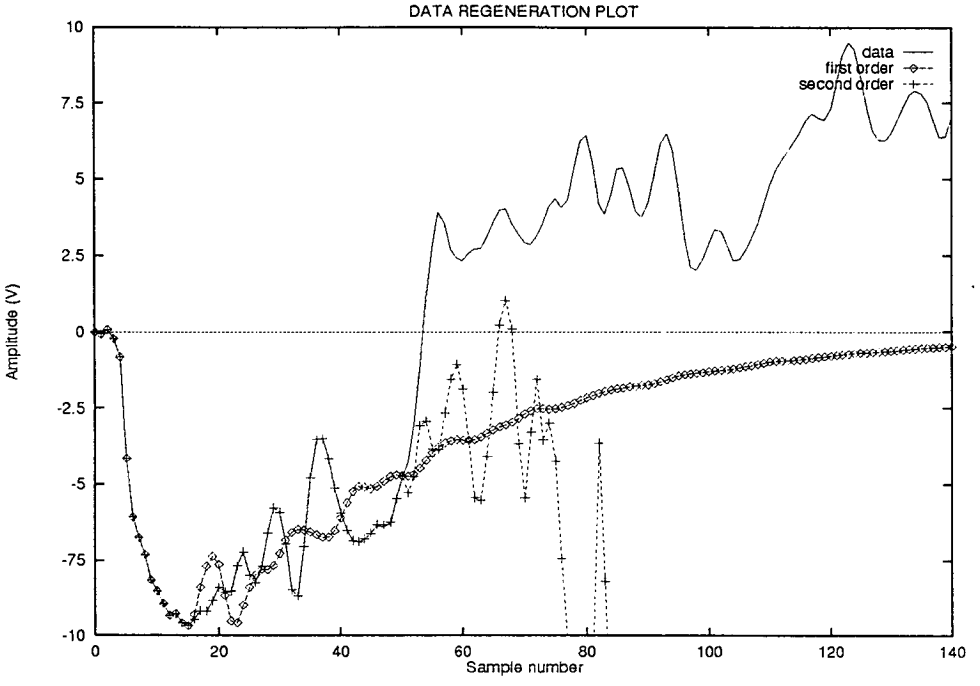


Figure 4.16: *Performance of Volterra filters with embedding dimension 16 trained on data including initial excitation period. Training length 50.*

Figure (4.17) show some similar results for RBF networks. As indicated by the results from the predictive tests, the RBF network rapidly becomes unstable, although still performing - for a short while - better than the linear filter. As a consequence, statistics for RBF networks when used regeneratively are not included in appendix A.

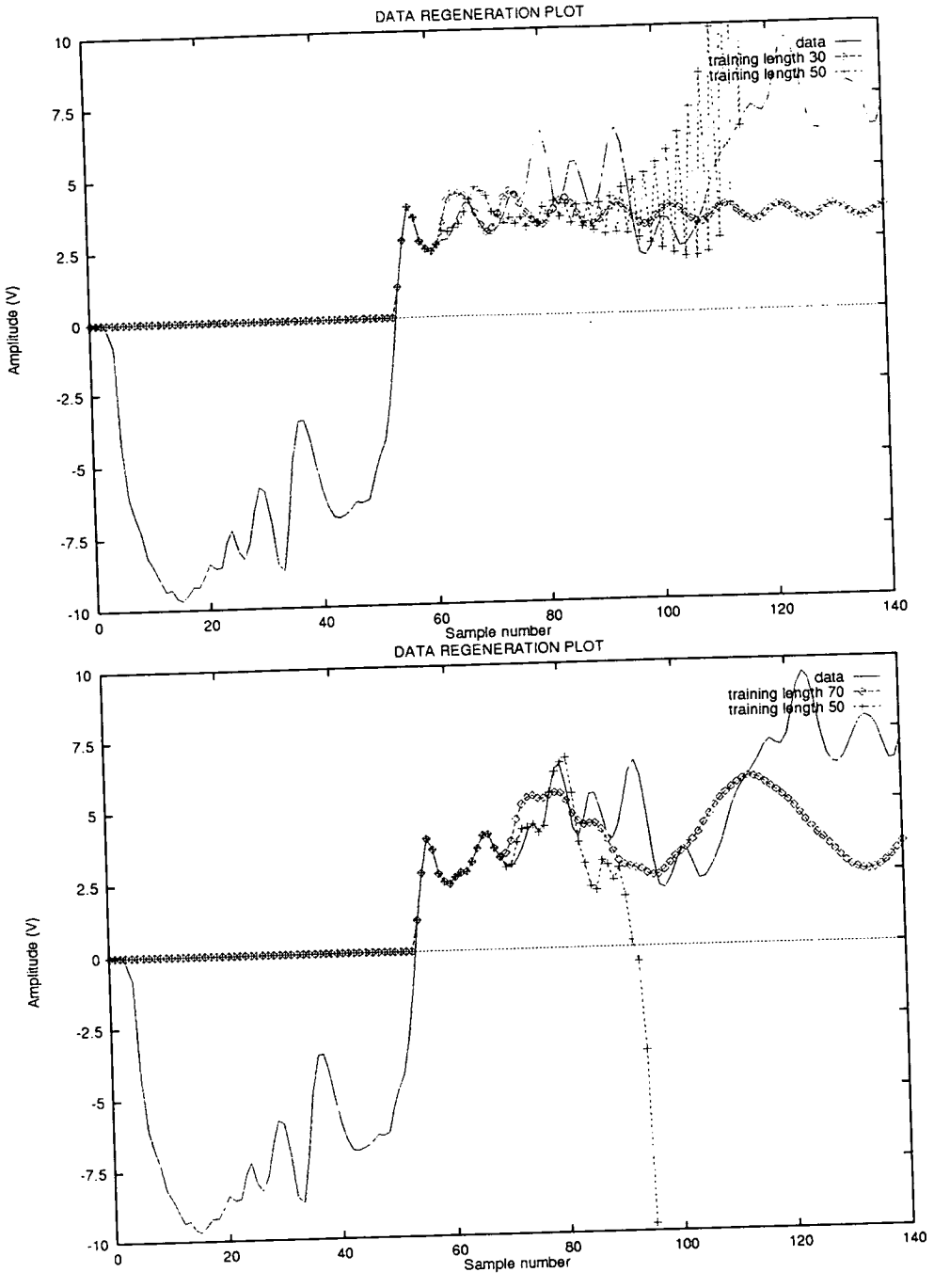


Figure 4.17: *Performance of RBF networks on data without initial excitation period. The top graph shows results for networks with embedding dimension 8 and 8 centres. The bottom one shows results for networks with embedding dimension 16 and 16 centres.*

4.8 Linear AR regeneration

It is established [74,82,83] that the output error from a linear prediction-error filter (i.e. those described by equations (4.2) and (4.3)) becomes white if the filter embedding dimension is large enough. This implies that as the embedding dimension of such a filter is increased, there is a corresponding reduction in correlation between the error, $\epsilon(n)$, of adjacent output samples, until, ultimately, the prediction-error process at the filter output consists of a sequence of uncorrelated samples, resulting in the whitening of the original process. Equations (4.3) and (4.2) can be rewritten to show this:

$$e_w(n) = d(n) - H_\rho^T X_\rho(n) \mid_{\rho \to \infty} \tag{4.37}$$

where H_ρ^T is a filter with embedding dimension ρ, $X_\rho(n)$ is the associated input vector, and $e_w(n)$ represents an uncorrelated white error process. It follows, therefore that the original process may be regenerated by substituting a Gaussian white (GWN) noise process for $e_w(n)$:

$$d(n) = \hat{d}(n) + w(n) \tag{4.38}$$

where $w(n)$ is a (GWN) process with mean power equal to $e(n)$. This type of filter is known as an autoregressive (AR) GWN driven model.

In the previous chapter it was shown that following the percussive attack, the drum sounds appear to become stationary. This indicates that it becomes possible to use long embedding dimensions, and thus such a GWN driven AR model becomes practical [12,84,85]. Figure (4.18) shows the result of using such a model with embedding dimension 200 and training length 10000 after 4100 samples (0.09 seconds) in the data record. The AR model regenerates the general form of the drum sound, although the high frequency information appears to be a little inaccurate. Comparison of the spectral content of the linear model (figure (4.19)) and the original sound (figure (4.20)) shows them

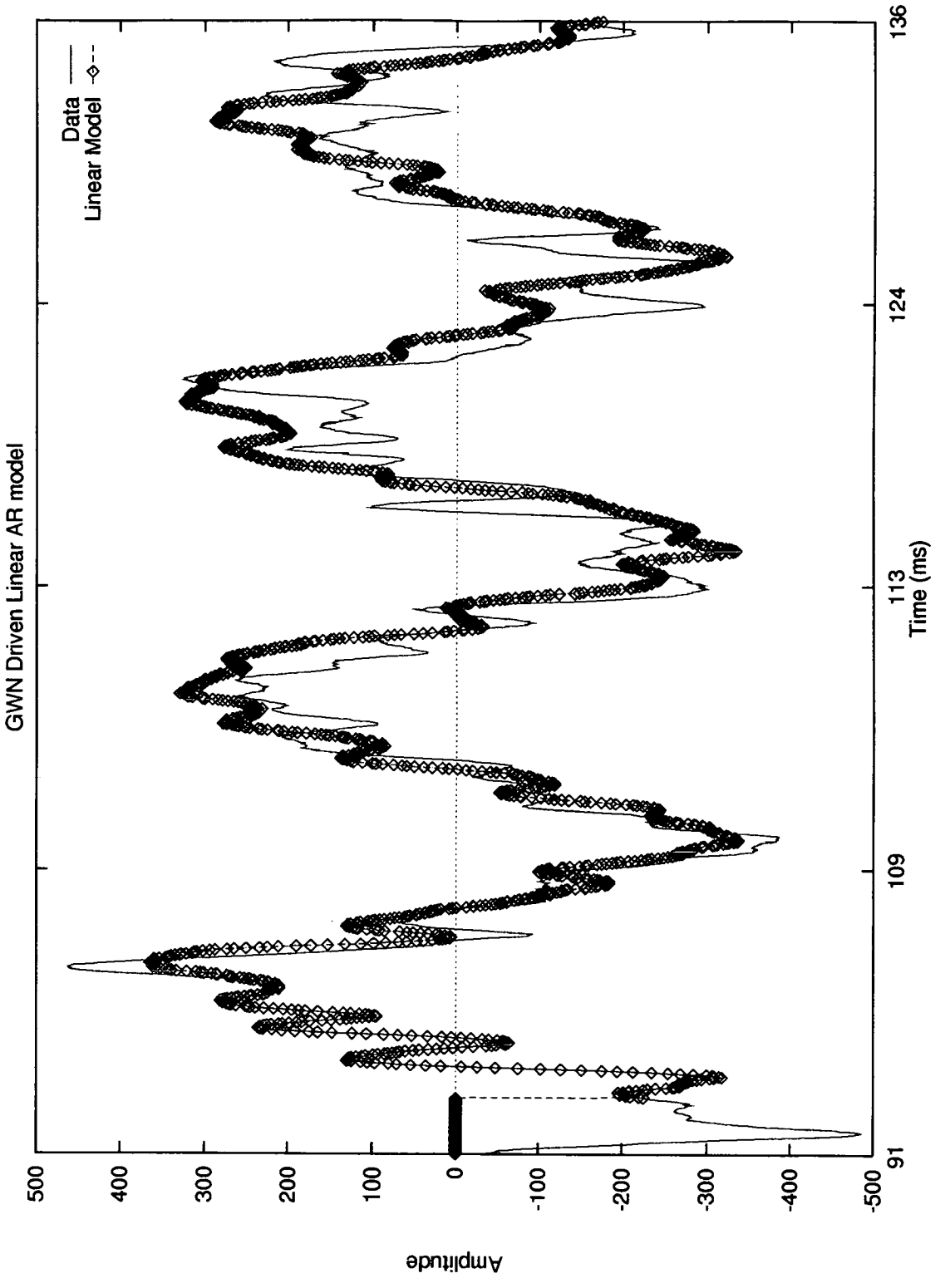


Figure 4.18: *GWN driven AR linear model of the later parts of a typical drum sound.*

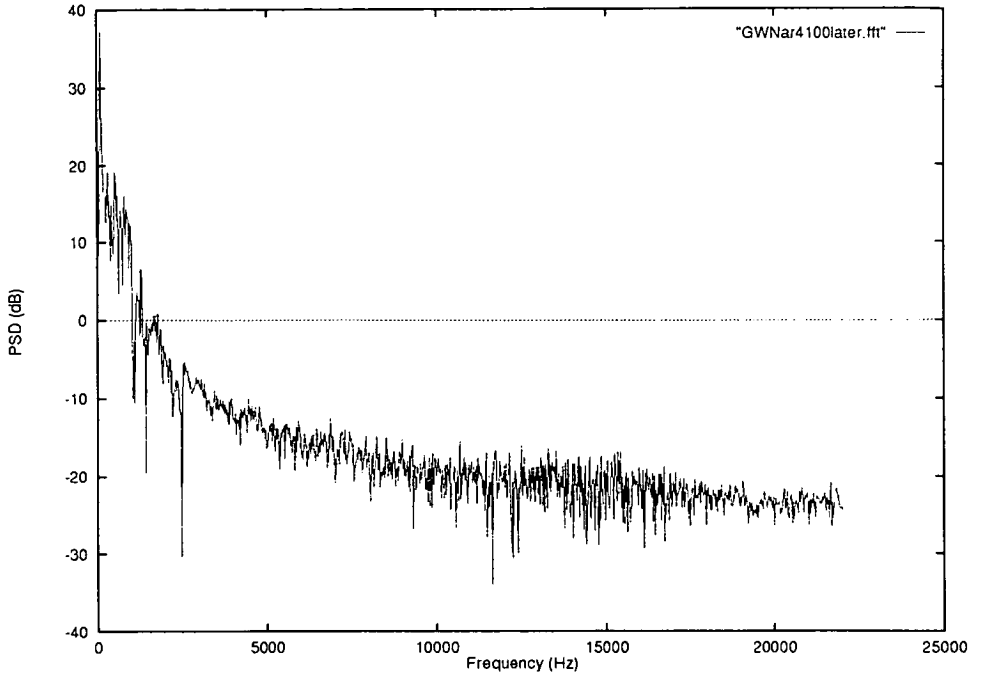


Figure 4.19: *1024 point power spectrum of GWN driven AR model.*

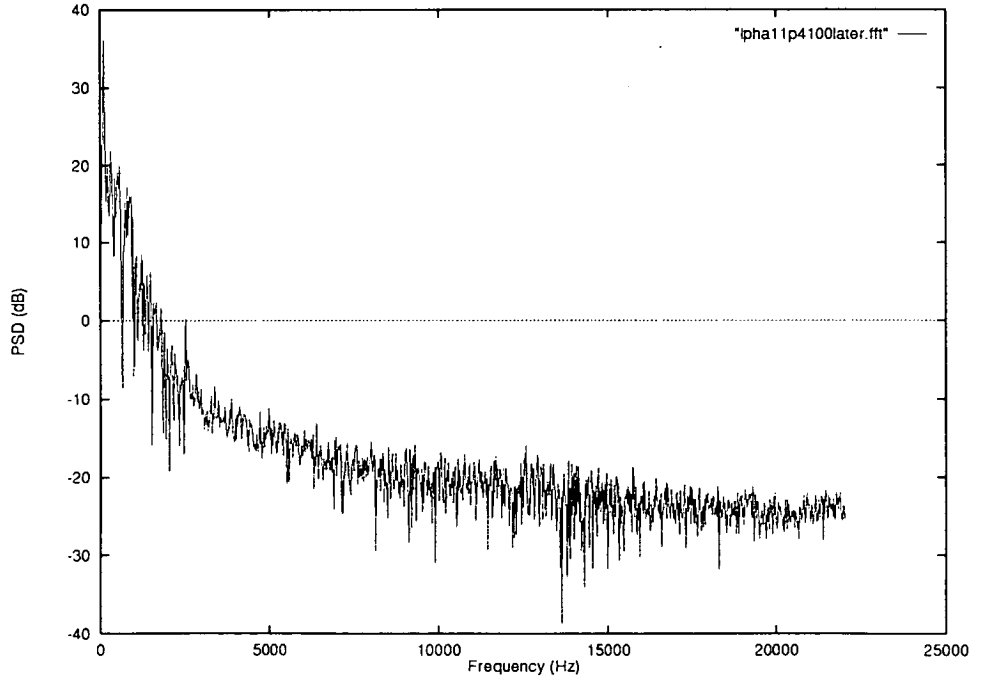


Figure 4.20: *1024 point power spectrum of data.*

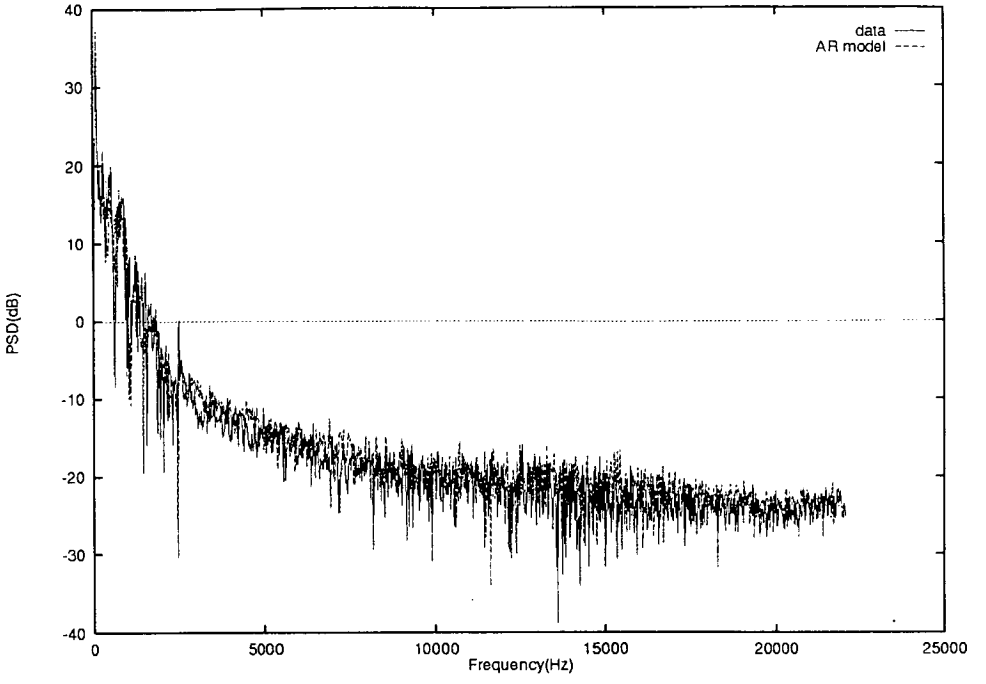


Figure 4.21: *Superposition of both 1024 point power spectra.*

to be almost identical, which is confirmed by superposition of the two plots (figure (4.21)).

4.9 Conclusion

In this chapter, two models, the Volterra filter and the RBF network, which might prove able to reproduce the initial percussive attack of a drum have been examined. It has been shown that each model is capable of being trained using a standard linear least squares technique. In the resulting tests it has become evident that the Volterra filter is far more successful than the RBF network at regenerating the initial percussive attack, but only for short data lengths over which stationarity can be assumed. Once the high degree of nonstationarity associated with the initial percussive attack has subsided, it is possible to model the subsequent drum sound with a simple, large embedding dimension linear GWN driven AR filter.

Both nonlinear models suffer from the problem of redundancy, i.e. they contain a large number of coefficients which do not contribute significantly to the signal output by the model. The next chapter explores the possibilities of removing some of these redundancies using two techniques - orthogonal least squares learning, and least squares noise thresholding.

Chapter 5

Coefficient Reduction Techniques

5.1 Introduction

One of the principal reasons for the unpopularity of nonlinear filters is the factorial increase in the number of filter coefficients as the embedding dimension and data training length of the filter rise. In point of fact, it is most probable that only a small percentage of the filter coefficients actually contribute to the filter output. In addition, when considering RBF networks it has been shown that for transient signals the lack of a coherent centre selection technique results in poor performance as a large number of centre vectors are required in order to cover the input signal space adequately. Both of these problems can, however, be addressed by an *a posteriori* removal of unwanted coefficients (or centres in the case of the RBF network).

This chapter examines two techniques for selecting such a reduced coefficient set; the orthogonal least squares (OLS) learning algorithm [49], and the least squares noise thresholding (LSNT) algorithm. The OLS method for choosing coefficients essentially involves the active selection of a suitable subset of coefficients from a large number of candidates. This contrasts with the LSNT method, in which unnecessary coefficients are discarded. Both techniques are applied to Volterra filters and RBF networks trained upon acoustic percussion data, and the performance of the resulting reduced-set filters is compared to that of the original filters. It is shown that using either technique a significant

reduction in the model size can be achieved without degrading the quality of the regenerated signal.

5.2 The Orthogonal Least Squares Learning Algorithm

5.2.1 Orthogonal Least Squares Coefficient Estimation

In the previous chapter, it was shown that both RBF networks and Volterra Filters could be expressed as vector inner products in a manner analagous to linear models:

$$d(n) = H\theta(n)^T + \epsilon(n) \tag{5.1}$$

where $d(n)$ is the desired filter output, H is the filter coefficient vector, $\theta(n)$ is the filter input vector (i.e. the combination of terms created from the embedding vector),

$$\theta(n) = [\{L(n)\} \text{ or } \{X_{vp}(n)\}] = [\vartheta_1(n), \cdots, \vartheta_M(n)] \tag{5.2}$$

($X_{vp}(n)$ for Volterra filters, and $L(n)$ for RBF networks), and $\epsilon(n)$ is the error between the desired signal and the actual filter output. This equation can be extended to describe an output vector of N samples extending over a period of time:

$$D(n) = \Theta(n)^T H + E(n) \tag{5.3}$$

where

$$D(n) = [d(n), \cdots, d(n+N-1)]^T \tag{5.4}$$

$$\Theta(n) = [\theta(n), \cdots, \theta(n+N-1)]^T \tag{5.5}$$

$$E(n) = [\epsilon(n), \cdots, \epsilon(n+N-1)]^T \tag{5.6}$$

It is also convenient to express the variation in time of the input elements $\vartheta_i(n)$:

$$\vartheta_i = [\vartheta_i(n), \cdots, \vartheta_i(n+N-1)]^T \tag{5.7}$$

where ϑ_i are known as the filter regressors. The OLS method involves the transformation of the set of ϑ_i into a set of orthogonal basis vectors, thus making it possible to calculate the contribution to the desired output energy from each basis vector. This is achieved by decomposing the matrix Θ into two matrices:

$$\Theta = WA \tag{5.8}$$

where A is an M X M triangular matrix with 1's on the main diagonal:

$$A = \begin{bmatrix} 1 & \alpha_{12} & \alpha_{13} & \cdots & \alpha_{1M} \\ 0 & 1 & \alpha_{23} & \cdots & \alpha_{2M} \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 & \alpha_{(M-1)M} \\ 0 & \cdots & 0 & 0 & 1 \end{bmatrix} \tag{5.9}$$

and W is an N X M matrix with orthogonal columns w_i such that

$$W^T W = \Gamma \tag{5.10}$$

where Γ is diagonal with elements

$$\gamma_i = w_i^T w_i \tag{5.11}$$

The signal space spanned by the set of input time vectors ϑ_i is also covered by the set of orthogonal basis vectors comprising the columns w_1 to w_M. Using these, equation (5.3) can be rewritten as

$$D = Wg + E \tag{5.12}$$

where

$$A\hat{H} = \hat{g}. \tag{5.13}$$

The orthogonal LS solution for $\hat{g}$ is found by

$$\hat{g} = H^{-1} W^T D \tag{5.14}$$

as detailed in the previous chapter. From this, the desired set of coefficients, $\hat{H}$ can be found using equation (5.13) which can be derived with the classical or

modified Gram-Schmidt methods [86]. This method can therefore be used to solve for the LS estimate $\hat{H}$. In this case however, a variation on the algorithm, presented in the next section, is to be used in order to select a reduced number of coefficients from the total set.

5.2.2 Subset Selection

Equation (5.14) can be rewritten as

$$\hat{g} = \frac{w_i^T D}{w_i^T w_i}. \tag{5.15}$$

Because w_i and w_j are orthogonal for $w_i \neq w_j$, the energy of $d(n)$ (i.e the sum of squares) can be calculated using equations (5.12) and (5.15) as

$$D^T D = \sum_{i=1}^{M} g_i^2 w_i^T w_i + E^T E. \tag{5.16}$$

If D is the desired output vector after its mean has been removed, then the variance of $d(n)$ is given by

$$\frac{D^T D}{N} = \frac{1}{N} \sum_{i=1}^{M} g_i^2 w_i^T w_i + \frac{E^T E}{N}. \tag{5.17}$$

Clearly it is evident that, of the two parts on the right hand side of this equation, $\frac{1}{N} \sum_{i=1}^{M} g_i^2 w_i^T w_i$ accounts for the component of the desired output variance due to the regressors, and $\frac{E^T E}{N}$ accounts for the remaining unexplained variance. The contribution of each w_i towards the desired output variance can thus be seen to be $\frac{g_i^2 w_i^T w_i}{N}$. A measure of the contribution of each regressor towards the filter output can thus be found by expressing this variance as a fraction of the desired output variance. This is known as the error reduction ratio (ERR):

$$[err]_i = \frac{g_i^2 w_i^T w_i}{D^T D} \tag{5.18}$$

The OLS method of coefficient selection is effectively an algorithm for iteratively choosing regressors with the largest ERR's until the cumulative ERR is greater than some predetermined tolerance ρ where $0 \leq \rho \leq 1$.

5.2.3 Implementation of the OLS Algorithm

The coefficient selection procedure can be summarised as follows:

- The first step is to calculate the variance for all the filter regressors ϑ_i,

$$\begin{aligned} w_1^{(i)} &= \vartheta_i \\ g_1^{(i)} &= \frac{(w_1^{(i)})^T D}{(w_1^{(i)})^T w_1^{(i)}} \\ [err]_1^{(i)} &= \frac{(g_1^{(i)})^2 (w_1^{(i)})^T w_1^{(i)}}{D^T D} \end{aligned}$$

for $1 \leq i \leq M$, and then find the regressor with the largest ERR:

$$[Merr]_1 = \max \left\{ [err]_1^{(i)} \right\}, \qquad 1 \leq i \leq M$$

This regressor becomes the first one selected by the OLS algorithm,

$$w_1 = w_1^{(i_1)} = \vartheta_{i1}.$$

- The remaining vectors are selected by, at the kth step, inspecting each unselected input time vector in turn, and finding the component of that vector orthogonal to all previously selected vectors:

$$\begin{aligned} \alpha_{jk}^{(i)} &= \frac{w_j^T \vartheta_i}{w_j^T w_j} \qquad 1 \leq j < k \\ w_k^{(i)} &= \vartheta_i - \textstyle\sum_{j=1}^{k-1} \alpha_{jk}^{(i)} w_j \end{aligned}$$

for $1 \leq i \leq M, i \neq i_1, \cdots, i_{k-1}$. The ERR of this orthogonal component is then calculated as before,

$$\begin{aligned} g_k^{(i)} &= \frac{(w_k^{(i)})^T D}{(w_k^{(i)})^T w_k^{(i)}} \\ [err]_k^{(i)} &= \frac{(g_k^{(i)})^2 (w_k^{(i)})^T w_k^{(i)}}{D^T D}. \end{aligned}$$

Once again the largest ERR is found,

$$[Merr]_k = \max \left\{ [err]_k^{(i)}, \qquad 1 \leq i \leq M, i \neq i_1, \cdots, i_{k-1} \right\}$$

and the time vector responsible for this ERR is chosen

$$w_k = w_k^{(i_k)} = \vartheta_{ik} - \sum_{j=1}^{k-1} \alpha_{jk}^{(i)} w_j.$$

The procedure is terminated at the M_sth step, when

$$1 - \sum_{j=1}^{M_s} [err]_j < \rho \tag{5.19}$$

Having chosen a suitable subset of regressors, the standard least squares technique is applied in order to calculate the resultant filter coefficients.

Given a required level of unexplained variance, it is evident that the OLS selection procedure will generally be able to produce a smaller filter than the initial model, as those inputs to the filter whose contribution to the output variance is duplicated will not be selected, and filter coefficients for those inputs need not be calculated.

5.3 The Least Squares Noise Thresholding Algorithm

Whilst the OLS algorithm is essentially a "greedy" one, which picks the best possible coefficient at every iteration, the LSNT method works in the opposite fashion, discarding undesirable coefficients in order to produce a reduced model size. Section 4.2 in the last chapter showed how the standard least squares algorithm is used to calculate the filter coefficients h_i. Assuming that a number of the coefficients thus calculated are redundant, as is probably the case using the filter structures under discussion, it would be reasonable to assume that these coefficients contribute very little to the filter output, and are consequentially small in magnitude. This assumption is borne out by examination of table(A.18) in appendix A, which shows the coefficients calculated for

the regenerative Volterra filter shown in figure (4.15). It can be seen that some of the smaller coefficient values are up to two orders of magnitude less than the principal coefficient values. Removal of these coefficients should therefore have little effect on the filter output.

In practice, an additional consideration is the finite numerical precision involved in the implementation of the least squares algorithm. In particular, the technique used in inverting the autocorrelation matrix R_x, singular value decomposition, can magnify numerical roundoff errors caused by the autocorrelation matrix being singular or ill-conditioned [75]. This results in small errors δh_i being associated with each coefficient. Where the coefficient magnitude itself is small, this error may add a significant amount of noise to the output of the associated filter tap, thus degrading the quality of the filter output. Exclusion of filter taps with coefficients falling below an appropriately chosen noise threshold, followed by recalculation of the remaining coefficients, is therefore unlikely to affect filter performance.

The LSNT Thresholding algorithm thus involves three simple steps:

- Calculate the filter coefficients for the full size filter.
- Remove all coefficients with values below a predetermined noise threshold.
- Recalculate values for the remaining coefficients.

5.4 Application

A second order Volterra filter with embedding dimension 16 was trained on the first 50 points of a typical drum sound, as detailed in the last chapter. The two coefficient reduction techniques were then implemented, and the number of coefficients reduced in steps of 5. Values were measured for the filter output MSE after 32 points, and these results can be seen in figure (5.1). Both techniques

are able to select a much smaller subset of coefficients with very little loss in quality. This is demonstrated graphically in figure (5.2), which shows the performance of reduced-set filters with only a third of the original coefficients. Another feature that can be seen in both figures (5.1) and (5.2) is the distinct

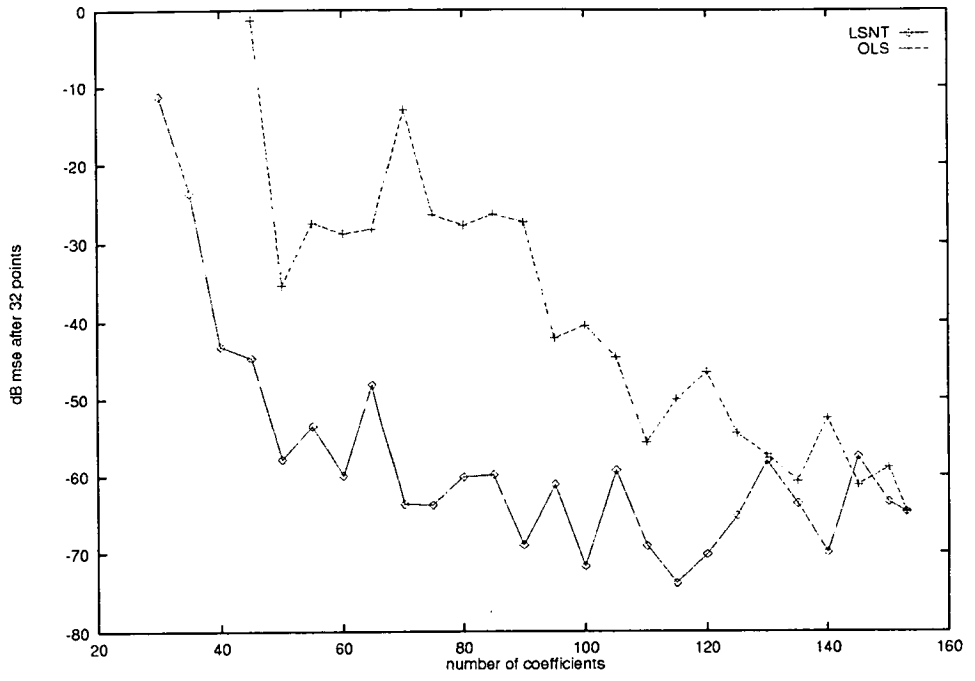


Figure 5.1: *Coefficient reduction: Change in prediction error of a Volterra filter when regenerating a typical drum sound*

performance advantage of the LSNT algorithm when compared to the OLS one. This evidence was corroborated by repeating the same set of tests for a further 14 data records, and averaging the results for the filters that converged. For the sake of comparison, the tests were also run using an RBF network of embedding dimension 16. The results of these tests are demonstrated in tabular form in appendix A, and graphically in figure (5.3).

Once again, it can be seen that both algorithms can produce a marked reduction in the number of filter coefficients whilst avoiding a commensurate increase in the output MSE. Sadly, the selection of a superior subset of centres produced no improvement in the ability of RBF networks to regenerate the

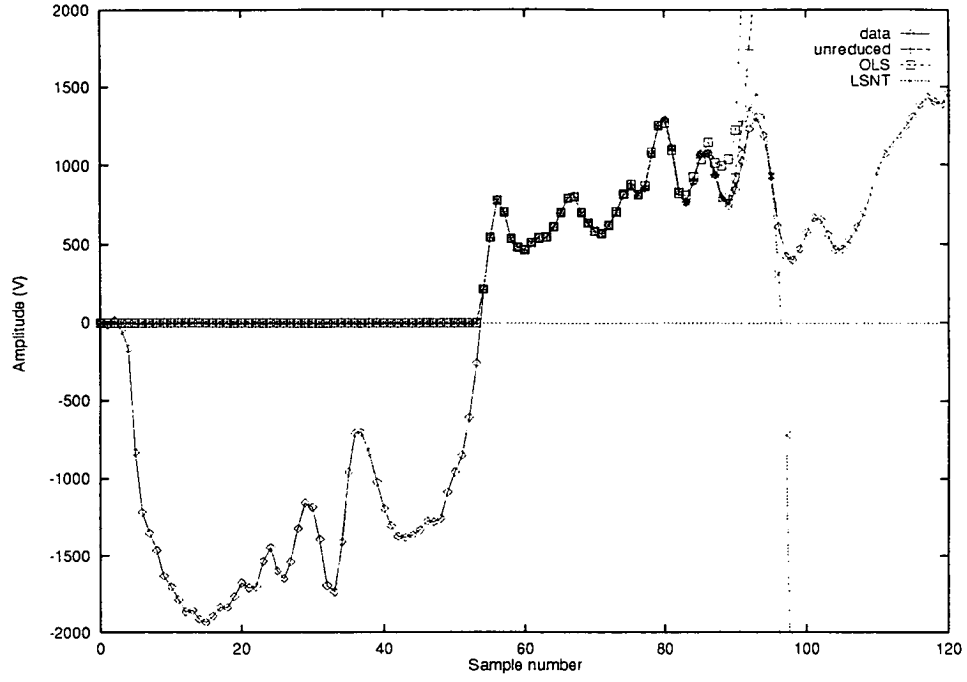


Figure 5.2: *Performance of Volterra filters when regenerating a typical drum sound using a reduced set of only 50 coefficients*

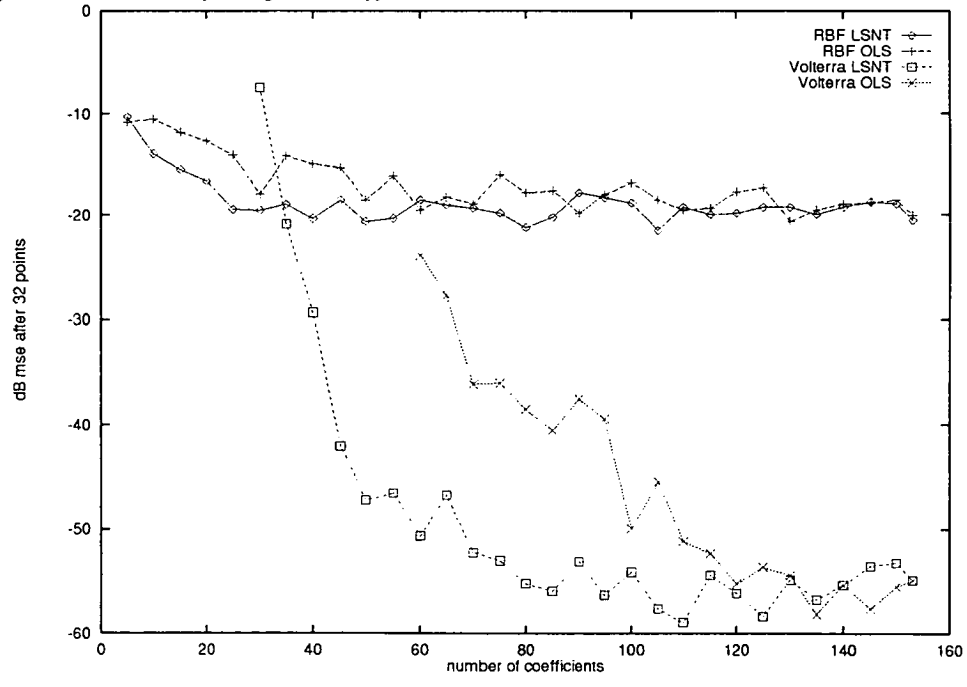


Figure 5.3: *Average prediction errors for coefficient reduction techniques implemented on RBF networks and Volterra filters*

data records successfully, and it must be concluded that Volterra filters provide the only viable basis for a model using the approach proposed in this thesis.

5.5 Conclusion

In terms of implementation, there are two main differences between the OLS algorithm, and the LSNT algorithm. The most desirable feature of the OLS algorithm is its pre-selection of coefficients. The SVD matrix inversion routine is therefore only used once, on a reduced size autocorrelation matrix. The LSNT method on the other hand, requires the SVD routine to be run twice, as the coefficients must be calculated in order to eliminate those below the noise threshold. By avoiding the employment of the SVD matrix inversion routine on the unreduced autocorrelation matrix, the OLS algorithm achieves a large reduction in computational overhead - especially as the size of the unreduced filter (and thus the autocorrelation matrix) increases. However, this advantage in terms of speed is offset by a shortfall in performance relative to the LSNT algorithm, especially when implementing large reductions in the number of coefficients. Chng *et al.* [87] have demonstrated an improved version of the OLS algorithm, although this improvement comes at the expense of speed due to computational complexity.

The second clear difference between the two algorithms is in the method of coefficient selection. The LSNT algorithm removes unwanted coefficients in one pass, whereas the OLS method , as noted earlier, selects the best possible coefficient at each iteration. As a result, the final set of coefficients chosen by the OLS algorithm tends to have a slightly wider spread of amplitudes than those chosen by the LSNT technique. A typical example of this can be found in tables(A.20) and (A.21) in appendix A, which list filter coefficient values for the reduced coefficient Volterra filters whose properties were compared in figures (5.1) and (5.2).

Despite the success of both algorithms in removing redundancy from the Volterra filter structure, it must be stressed that the number of terms remaining in the final filter is typically only marginally smaller than the number of points in the data record that the filter is able to reproduce. Using the filter model as a method of data compression is therefore impractical.

Chapter 6

Conclusions and further work

6.1 Conclusions

The ultimate purpose of this work was to produce an accurate model of a typical drum. It was hoped that this model would be able to reproduce the range and variety of sounds of a real drum, while retaining the essential qualities of drum sounds that are so instantly recognisable to listeners. It was also hoped that, given initial data, the model would be flexible enough to emulate a number of different drums, and different tunings of the same drum.

Although this objective was not fully achieved, significiant steps were taken towards the creation of such a model. The first of these steps was the establishment of a database of recorded sounds, sampled in an anechoic chamber, and taken from two different drums using a variety of sampling pathways. Given the transient nature of drum sounds, this large data base was necessary in order to obtain statistically meaningful results for any given section of the sounds duration through the use of ensemble averaging.

Once the database had been formed, it was possible to proceed with the second step of the work, namely the analysis of the drum sounds. It was decided that the principle target of this analysis would be the drum's initial percussive attack, as it was evident that this characterised the sound and, as shown by previous research, had also proven very difficult to model. Bispectral analysis was used to detect the presence of phase coupling between different frequency components in the signal, and it was observed that this shape was

indicative of nonlinear modulation between different frequency components. It was shown that in theory, such modulation was independent of any artificial signal shaping due to the data acquisition process. Since shaping is an integral part of modern music recording techniques, any information present within the signal which was affected by such shaping could not be involved in the characterisation of the drum sound. By contrasting bispectra for data records gathered using acquisition paths with different phase and frequency characteristics, the transparency of quadratic phase coupling to the signal path was confirmed in practice.

Examination of the bicoherence spectra provided further evidence of significant nonlinearities present during the sound's initial percussive attack, as it was evident that low frequency components of the sound were oscillating independently from one another, whilst the high frequency components of the sound were dependent on a combination of these independent low frequencies. The most likely explanation for this is that the high frequency oscillations are caused by modulation between the lower frequencies.

The conclusion drawn from these results is that during the percussive attack, drum sounds depart significantly from the traditional Helmholz model used to describe them. As the Helmholz equations are only valid for small peturbations to the membrane, the probable reason for this is the large displacement of the membrane due to the initial excitation caused by the drum stick. Once the oscillation of the membrane had attenuated (i.e. after the initial percussive attack) examination of the bispectrum and bicoherence showed that the nonlinear components of the signal had vanished almost completely, and the system was now characterisable in linear terms. With the attenuation of nonlinear behaviour, the modulation that gave rise to the high frequency terms would also vanish, leading to the rapid attenuation of the high frequency terms and the very short duration which is typical of the initial percussive attack.

The final task undertaken was to develop a model which incorporated the nonlinear information that had been shown to be present within the drum

sound. With this end in mind, two nonlinear filter structures were examined, the radial basis function (RBF) network and the Volterra filter. Although these two filters had substantially different internal architectures, it was shown that they shared the ability to be configured using the established linear least squares algorithm. Given the transient nature of the signal to be modelled, the ability to be trained in a non-adaptive fashion was a prerequisite for any potential model. Intensive testing on all the data records gathered indicated that the data records could be regenerated to a high degree of accuracy by using second order Volterra filters with embedding dimensions of 12 or 16. Furthermore, it was observed that these models were unable to regenerate any points outwith the training data, and that the accuracy of the reproduced signal was maximised by using small training data lengths. Both of these observations demonstrated once again the extreme nonstationarity of the initial percussive attack. In confirmation of the conclusions noted in the preceeding paragraph, it was then shown that the sound following this part of the signal could be modelled successfully using a simple linear autoregressive filter with high embedding dimension.

A major drawback in the use of the nonlinear filtering techniques used was the large number of coefficients employed by such models. The result of this was that the size of the model was substantially bigger than the data it was used to regenerate. One hypothesis for this disparity was the presence of a great deal of redundancy within the model, and two methods of removing any such redundancy were examined. Using these two methods, the orthogonal least squares (OLS) learning algorithm and the least squares noise thresholding (LSNT) coefficient reduction algorithm, it was found that the size of the model could be reduced to approximately the same size as the data regenerated without any degredation in the quality of the output signal. It was also seen that there was a tradeoff between the speed advantage of the OLS algorithm, and the better performance of the LSNT algorithm.

6.2 Further Work

A number of key areas in which the work presented in this thesis could be extended have been identified:

- Development of a complete model
- Ensemble feature extraction
- Extended tracking capability

Development of a complete model

Although the thesis showed that each section of a drum sound could be successfully reproduced, a model implementing the techniques discussed in this thesis has not yet been constructed. Such a model could be developed using the look-up table technique [8] of filter coefficient selection to regenerate the signal at different times.

Ensemble feature extraction

One of the aims in developing a model based system of reproduction was to incorporate the flexibility and variation inherent in a real drum. In order to do this, it would be necessary to train the model on an ensemble of drum sounds in order to extract the features common to all the data records. Such an ensemble-trained model should then be able to reproduce a variety of sounds by initialising the model with the relevant data. Preliminary tests indicated that the amount of samples regenerated successfully in each data record seemed to be inversely proportional to the number of data records the model was trained upon. An improvement on this is clearly necessary for any useful model.

Extend tracking capability

One of the major drawbacks of the Volterra filter model is its failure to regenerate more than a short section of the sound for any given set of coefficients. One of the factors contributing to this failure was the disparity between the high amplitude, slowly attenuating low frequency terms and the low amplitude, rapidly attenuating high frequencies of the percussive attack. In chapter 4, by comparing figures (4.6) and (4.5) it could be seen that the Volterra filter was far less successful at regenerating low-frequency signals. Using a linear AR model to regenerate the fundamental frequency in parallel with a second order Volterra filter to regenerate the high frequency transients might well produce a significant performance improvement.

Appendix A

Data Prediction and Regeneration: Tables of Results

A.1 Introduction

In this appendix, records for Volterra filters and RBF networks operating in a variety of different configurations are shown. The appendix is divided into four sections. Section one deals with results obtained using the algorithms predictively. In the second section, the performance of the different filters when regenerating data is documented. The third section presents a tabular record of the effectiveness of the two weight reduction techniques, and the appendix closes with lists of coefficients calculated for a Volterra series trained on a typical drum sound with and without the different coefficient reduction techniques.

A.2 Predictive results

Tables (A.1) to (A.11) show MSE values for Volterra filters and RBF networks after one hundred points were tracked, and after two thousand points had been tracked. In each case, the result displayed was obtained by averaging results taken from the full ensemble of data records.

Filter	Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
First order	8	1000	-23.3dB	-28.7dB
Second order	8	1000	-11.0dB	-11.3dB
First order	8	200	-23.7dB	-28.6dB
Second order	8	200	-12.1dB	-11.0dB
First order	8	50	-23.7dB	-27.4dB
Second order	8	50	-2.4dB	+1.0dB
First order	12	1000	-23.7dB	-29.0dB
Second order	12	1000	-10.9dB	-9.4dB
First order	12	200	-24.1dB	-28.6dB
Second order	12	200	-8.42dB	-12.4dB
First order	12	50	-23.6dB	-27.2dB
Second order	12	50	-1.0dB	+3.3dB
First order	16	1000	-24.7dB	-29.4dB
Second order	16	1000	-10.1dB	-8.3dB
First order	16	200	-23.9dB	-28.6dB
Second order	16	200	-14.0dB	-5.9dB
First order	16	50	-22.3dB	-25.3dB
Second order	16	50	-0.4dB	+1.4dB

Table A.1: *Mean performance statistics for Volterra filters trained on 44 kHz Low Tom data (Butterworth).*

Filter	Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
First order	8	1000	-30.3dB	-32.1dB
Second order	8	1000	-16.2dB	-12.4dB
First order	8	200	-31.0dB	-31.6dB
Second order	8	200	-21.7dB	-11.6dB
First order	8	50	-29.9dB	-31.2dB
Second order	8	50	-7.5dB	+1.2dB
First order	12	1000	-30.8dB	-32.3dB
Second order	12	1000	-15.6dB	-10.5dB
First order	12	200	-31.2dB	-31.6dB
Second order	12	200	-20.4dB	-9.1dB
First order	12	50	-29.5dB	-29.2dB
Second order	12	50	-1.4dB	+3.3dB
First order	16	1000	-29.1dB	-32.4dB
Second order	16	1000	-14.8dB	-9.6dB
First order	16	200	-31.6dB	-30.1dB
Second order	16	200	-16.6dB	-2.1dB
First order	16	50	-28.7dB	-28.0dB
Second order	16	50	-1.7dB	+3.4dB

Table A.2: *Mean performance statistics for Volterra filters trained on 44 kHz Low Tom data (Butterworth) without initial excitation.*

Filter	Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
First order	8	1000	-32.3dB	-32.9dB
Second order	8	1000	-16.9dB	-13.34dB
First order	8	200	-32.4dB	-32.3dB
Second order	8	200	-21.7dB	-11.9dB
First order	8	50	-31.4dB	-28.4dB
Second order	8	50	-10.2dB	+0.3dB
First order	12	1000	-32.6dB	-33.1dB
Second order	12	1000	-16.8dB	-11.3dB
First order	12	200	-32.9dB	-32.3dB
Second order	12	200	-20.1dB	-9.5dB
First order	12	50	-32.2dB	-30.5dB
Second order	12	50	-5.0dB	+2.5dB
First order	16	1000	-32.9dB	-33.1dB
Second order	16	1000	-15.9dB	-11.2dB
First order	16	200	-33.4dB	-32.2dB
Second order	16	200	-16.6dB	-2.3dB
First order	16	50	-31.3dB	-29.5dB
Second order	16	50	-2.9dB	+2.7dB

Table A.3: *Mean performance statistics for Volterra filters trained on 44 kHz Low Tom data (Bessel) without initial excitation.*

Filter	Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
First order	8	1000	-26.8dB	-30.1dB
Second order	8	1000	-11.5dB	-11.5dB
First order	8	200	-25.5dB	-29.9dB
Second order	8	200	-15.7dB	-7.7dB
First order	8	50	-28.2dB	-26.7dB
Second order	8	50	-9.1dB	0.0dB
First order	12	1000	-27.3dB	-30.4dB
Second order	12	1000	-10.7dB	-10.1dB
First order	12	200	-27.5dB	-29.9dB
Second order	12	200	-15dB	-0.8dB
First order	12	50	-25.1dB	-26.7dB
Second order	12	50	-3.5dB	+3.0dB
First order	16	1000	-27.6dB	-30.5dB
Second order	16	1000	-10.7dB	-9.4dB
First order	16	200	-27.9dB	-30.0dB
Second order	16	200	-16.7dB	+0.7dB
First order	16	50	-23.7dB	-24.5dB
Second order	16	50	-2.3dB	+3.7dB

Table A.4: *Mean performance statistics for Volterra filters trained on 44 kHz High Tom data (Butterworth) without initial excitation.*

Filter	Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
First order	8	1000	-25.3dB	-28.5dB
Second order	8	1000	-13.4dB	-12.2dB
First order	8	200	-25.6dB	-27.7dB
Second order	8	200	-15.5dB	-10.0dB
First order	8	50	-25.6dB	-26.0dB
Second order	8	50	-9.8dB	0.5dB
First order	12	1000	-25.7dB	-28.7dB
Second order	12	1000	-14.5dB	-11.7dB
First order	12	200	-26.5dB	-28.0dB
Second order	12	200	-16.2dB	-8.2dB
First order	12	50	-25.7dB	-25.9dB
Second order	12	50	-6.6dB	+1.8dB
First order	16	1000	-26.2dB	-28.9dB
Second order	16	1000	-14.4dB	-10.5dB
First order	16	200	-26.9dB	-28.3dB
Second order	16	200	-14.9dB	-4.3dB
First order	16	50	-25.2dB	-24.7dB
Second order	16	50	-6.1dB	+2.1dB

Table A.5: *Mean performance statistics for Volterra filters trained on 33 kHz Low Tom data (12 kHz Butterworth) without initial excitation.*

Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
8	1000	-18.1dB	-24.1dB
8	200	-20.0dB	-23.1dB
8	50	-12.0dB	-8.6dB
12	1000	-21.4dB	-23.2dB
12	200	-20.6dB	-23.4dB
12	50	-9.9dB	-11dB
16	1000	-19.4dB	-23.8dB
16	200	-21.7	-22.7dB
16	50	-4.1dB	+3.05dB

Table A.6: *Mean performance statistics for RBF Networks trained on 44 kHz Low Tom data (Butterworth) using thin plate spline nonlinearity.*

Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
8	1000	-16.9dB	-17.9dB
8	200	-22.3dB	-15.5dB
8	50	-13.7dB	-6.1dB
12	1000	-16.9dB	-18.9dB
12	200	-22.3dB	-14.8dB
12	50	-12.1dB	-4.1dB
16	1000	-16.2dB	-18.5dB
16	200	-22.0dB	-13.9dB
16	50	-12.6dB	+0.2dB

Table A.7: *Mean performance statistics for RBF Networks trained on 44 kHz Low Tom data (Butterworth) without initial excitation.*

Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
8	1000	-15.0dB	-17.0dB
8	200	-16.0dB	-14.6dB
8	50	-9.14dB	-4.6dB
12	1000	-13.9dB	-17.7dB
12	200	-16.4dB	-13.2dB
12	50	-6.6dB	+3.0dB
16	1000	-12.7dB	-17.1dB
16	200	-16.6	-12.6dB
16	50	-1.6dB	+5.0dB

Table A.8: *Mean performance statistics for RBF Networks trained on 44 kHz Low Tom data (Butterworth).*

Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
8	1000	-17.0dB	-16.7dB
8	200	-22.9dB	-15.4dB
8	50	-17.2dB	-6.1dB
12	1000	-16.7dB	-16.6dB
12	200	-22.8dB	-15.3dB
12	50	-14.1dB	-5.0dB
16	1000	-16.7dB	-17.1dB
16	200	-21.7dB	-14.9dB
16	50	-12.3dB	-5.4dB

Table A.9: *Mean performance statistics for RBF Networks trained on 44 kHz Low Tom data (Bessel) without initial excitation.*

Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
8	1000	-17.0dB	-16.7dB
8	200	-22.9dB	-15.4dB
8	50	-17.2dB	-6.1dB
12	1000	-16.7dB	-16.6dB
12	200	-22.8dB	-15.3dB
12	50	-14.1dB	-5.0dB
16	1000	-16.7dB	-17.1dB
16	200	-21.7dB	-14.9dB
16	50	-12.3dB	-5.4dB

Table A.10: *Mean performance statistics for RBF Networks trained on 44 kHz High Tom data (Butterworth) without initial excitation.*

Embedding dimension	Training length	av. MSE (100 pts)	av. MSE (2000 pts)
8	1000	-13.2dB	-16.4dB
8	200	-16.9dB	-15.1dB
8	50	-14.6dB	+0.9dB
12	1000	-14.7dB	-16.4dB
12	200	-18.5dB	-13.9dB
12	50	-14.5dB	-2.8dB
16	1000	-14.3dB	-16.4dB
16	200	-18.3dB	-14.5dB
16	50	-12.5dB	-2.3dB

Table A.11: *Mean performance statistics for RBF Networks trained on 33 kHz Low Tom data (12 kHz Butterworth) without initial excitation.*

A.3 Regenerative Results

Tables (A.12) to (A.16) show average prediction error values for Volterra filters operating regeneratively. Measurements were recorded for the MSE after 15 points and then 30 points had been regenerated. In the event of the filter output becoming unstable for a particular data record, then the results for the remaining filter records were averaged, and a note was entered indicating that the averages were not calculated over the whole set of data records. Where the filters became unstable every time, the entry has been left blank. Results for RBF networks are not shown as these were so unstable that it was impossible to obtain a meaningful set of results whilst using equivalent parameters.

Filter	Embedding dimension	Training length	av. MSE (15 pts)	av. MSE (30 pts)
First order	8	70	-11.1	-8.0dB
Second order	8	70	-	-
First order	8	50	-10.8dB	-9.0dB
Second order	8	50	-	-
First order	8	30	-21.0dB	-14dB
Second order	8	30	(-30.4dB over 14)	-
First order	12	70	-18.2dB	-10.0dB
Second order	12	70	-	-
First order	12	50	-20.3dB	-13.0dB
Second order	12	50	(-24.2dB over 14)	(-27dB over 3)
First order	12	30	-22.1dB	-1.1dB
Second order	12	30	-65.8dB	(0dB over 6)
First order	16	70	-12.0dB	-9.0dB
Second order	16	70	(-25.7dB over 9)	-
First order	16	50	-16.0dB	-13.0dB
Second order	16	50	-63.4dB	(-55dB over 17)
First order	16	30	-31.7dB	+14.7dB
Second order	16	30	-39.4dB	(+4.2dB over 11)

Table A.12: *Mean performance statistics for Volterra filters regenerating 44 kHz Low Tom data (Butterworth). Parenthesised values indicate that the average was not calculated over the whole data set.*

Filter	Embedding dimension	Training length	av. MSE (15 pts)	av. MSE (30 pts)
First order	8	70	-10.4dB	-6.5dB
Second order	8	70	-	-
First order	8	50	-9.8dB	-8.2dB
Second order	8	50	-	-
First order	8	30	-12.3dB	-9.4dB
Second order	8	30	(-33.0dB over 8)	-
First order	12	70	-11.4dB	-8.8dB
Second order	12	70	(-6.0dB over 2)	-
First order	12	50	-10.0dB	-8.4dB
Second order	12	50	(-26.0dB over 11)	(+3.0dB over 1)
First order	12	30	-20.1dB	-10.8dB
Second order	12	30	-73.0dB	(-10.0dB over 3)
First order	16	70	-9.9dB	-8.4dB
Second order	16	70	(-27.3dB over 12)	(-26.0dB over 1)
First order	16	50	-11.2dB	-10.4dB
Second order	16	50	-46.7dB	(-20.0dB over 9)
First order	16	30	-31.7dB	-0.6dB
Second order	16	30	-32.3dB	(+5.0dB over 4)

Table A.13: *Mean performance statistics for Volterra filters regenerating 44 kHz Low Tom data (Butterworth) without initial excitation. Parenthesised values indicate that the average was not calculated over the whole data set.*

Filter	Embedding dimension	Training length	av. MSE (15 pts)	av. MSE (30 pts)
First order	8	70	-8.0dB	-6.1dB
Second order	8	70	-	-
First order	8	50	-7.8dB	-5.7dB
Second order	8	50	(-1.0dB over 1)	-
First order	8	30	-9.3dB	-8.1dB
Second order	8	30	(-16.5dB over 5)	-
First order	12	70	-8.7dB	-5.8dB
Second order	12	70	(-4dB over 1)	-
First order	12	50	-9.6dB	-6.7dB
Second order	12	50	(-31.0dB over 7)	(-21.5dB over 3)
First order	12	30	-15.6dB	-12.3dB
Second order	12	30	-53.8dB	(-11.5dB over 3)
First order	16	70	-13.1dB	-9.1dB
Second order	16	70	(-13.7dB 0ver 12)	-
First order	16	50	-13.2dB	-10.7dB
Second order	16	50	-42.3dB	(-40.2dB over 9)
First order	16	30	-28.2dB	(-1.7dB over 17)
Second order	16	30	-36.1dB	(-17.0dB over 4)

Table A.14: *Mean performance statistics for Volterra filters regenerating 44 kHz Low Tom data (Bessel) without initial excitation. Parenthesised values indicate that the average was not calculated over the whole data set.*

Filter	Embedding dimension	Training length	av. MSE (15 pts)	av. MSE (30 pts)
First order	8	70	-9.4dB	-6.2dB
Second order	8	70	(-3.0dB over 1)	-
First order	8	50	-6.8dB	-5.6dB
Second order	8	50	-	-
First order	8	30	-10.1dB	-8.7dB
Second order	8	30	(-3dB over 3)	-
First order	12	70	-9.9dB	-7.7dB
Second order	12	70	(+8dB over 1)	-
First order	12	50	-10.9dB	-8.9dB
Second order	12	50	(-17.9dB over 11)	(-14.0dB over 3)
First order	12	30	-18.1dB	-7.9dB
Second order	12	30	-50.1dB	(-11.7dB over 8)
First order	16	70	-12.1dB	-8.4dB
Second order	16	70	(-35.6dB over 12)	(-11.1dB over 1)
First order	16	50	-15.2dB	-11.3dB
Second order	16	50	-48.7dB	(-35.9dB over 17)
First order	16	30	-36.6dB	+1.9dB
Second order	16	30	-30.8dB	(+0.9dB over 11)

Table A.15: *Mean performance statistics for Volterra filters regenerating 44 kHz High Tom data (Butterworth) without initial excitation. Parenthesised values indicate that the average was not calculated over the whole data set.*

Filter	Embedding dimension	Training length	av. MSE (15 pts)	av. MSE (30 pts)
First order	8	70	-6.7dB	-4.9dB
Second order	8	70	-	-
First order	8	50	-8dB	-6.7dB
Second order	8	50	-	-
First order	8	30	(-8.4dB over 19)	(-6.3dB over 19)
Second order	8	30	(-9.7dB over 5)	-
First order	12	70	-7.1dB	-7.4dB
Second order	12	70	(-13dB over 1)	(-11dB over 1)
First order	12	50	-9.5dB	-6.7dB
Second order	12	50	(-29.9dB over 13)	(-8.5dB over 3)
First order	12	30	-15.4dB	-7.5dB
Second order	12	30	-60.7dB	(-9.8dB over 7)
First order	16	70	-8.6dB	-7.1dB
Second order	16	70	(-31.1dB over 19)	(-23.0dB over 8)
First order	16	50	-10.5dB	-9.3dB
Second order	16	50	-56.7dB	(-21.3dB over 17)
First order	16	30	-29.2dB	-5.4dB
Second order	16	30	-27.2dB	(-5.8dB over 4)

Table A.16: *Mean performance statistics for Volterra filters regenerating 33 kHz Low Tom data (12 kHz Butterworth) without initial excitation. Parenthesised values indicate that the average was not calculated over the whole data set.*

A.4 Coefficient Reduction techniques

Table (A.17) shows average prediction errors after the regeneration of 32 points for 153-centre RBF networks and second order Volterra filters (containing 153 taps in total) with embedding dimension 16 when the two reduction algorithms are implemented to remove increasing numbers of coefficients. Where the filter failed to converge for one or more data records, the entry is left blank.

Tables (A.18) to (A.21) show coefficient values for a second order Volterra filter with embedding dimension 16 trained on a typical data record, and compare values for the full set of coefficients with those of the reduced-set filters created when using the OLS and LSNT algorithms to select a subset of 50 coefficients.

# Coeffs.	RBF		Volterra		# Coeffs	RBF		Volterra	
	OLS	LSNT	OLS	LSNT		OLS	LSNT	OLS	LSNT
153:	-20.0	-20.5	-54.9	-54.9	75:	-16.0	-19.8	-36.0	-53.0
150:	-18.5	-18.9	-55.5	-53.2	70:	-18.9	-19.3	-36.1	-52.2
145:	-18.8	-18.7	-57.6	-53.5	65:	-18.2	-19.0	-27.7	-46.7
140:	-18.9	-19.2	-55.3	-55.3	60:	-19.5	-18.5	-23.9	-50.6
135:	-19.5	-19.9	-58.1	-56.7	55:	-16.1	-20.3	-	-46.5
130:	-20.6	-19.2	-54.4	-54.8	50:	-18.5	-20.6	-	-47.2
125:	-17.3	-19.2	-53.6	-58.3	45:	-15.3	-18.5	-	-42.0
120:	-1.7	-19.8	-55.2	-56.1	40:	-14.9	-20.3	-	-29.3
115:	-19.3	-19.9	-52.3	-54.4	35:	-14.1	-18.9	-	-20.8
110:	-19.5	-19.2	-51.1	-58.9	30:	-17.9	-19.5	-	-7.4
105:	-18.5	-21.5	-45.4	-57.6	25:	-14.0	-19.4	-	
100:	-16.8	-18.8	-49.9	-54.1	20:	-12.7	-16.6	-	
95:	-18.0	-18.3	-39.4	-56.3	15:	-11.8	-15.5	-	
90:	-19.8	-17.8	-37.5	-53.1	10:	-10.5	-13.9	-	
85:	-17.6	-20.2	-40.5	-55.9	5:	-10.8	-10.3	-	
80:	-17.8	-21.2	-38.5	-55.2					

Table A.17: *Average prediction errors for coefficient reduction techniques implemented on 153-centre RBF networks and second order Volterra filters with embedding dimension 16. A dash indicates that the filter in question had failed to converge for one or more data records.*

8.592216e-06	-9.123354e-05	7.823308e-05	-1.328363e-04
-4.320728e-04	2.379142e-05	-5.253504e-04	-9.726590e-05
3.158574e-04	9.097143e-06	-3.319085e-04	-3.788487e-04
-3.034216e-04	-5.069021e-04	9.754655e-06	5.961479e-06
-4.093240e-04	-1.273479e-04	-1.065566e-04	3.878812e-04
9.911430e-06	-7.203077e-05	-1.513719e-04	-1.166978e-04
3.591408e-04	5.104278e-04	1.020444e-04	1.024943e-05
3.973500e-04	-2.164762e-04	-3.569768e-04	-2.313668e-04
8.797140e-05	-2.565305e-04	-7.652525e-04	1.135454e-05
3.812987e-04	-2.531908e-04	-4.556534e-04	-1.857276e-04
3.435746e-04	-2.156921e-04	-3.863524e-04	-2.124119e-04
1.230678e-05	9.576096e-04	3.325938e-04	-4.640969e-04
-9.188011e-05	4.051914e-04	2.202030e-04	1.839587e-04
4.435021e-04	-2.586223e-04	1.179166e-05	7.364467e-04
5.863860e-04	-3.141648e-04	1.176615e-04	-3.517178e-04
-3.250439e-04	1.414449e-04	2.937593e-04	-1.479114e-04
-6.542564e-04	1.081158e-05	1.511016e-04	6.729133e-04
3.658947e-04	3.334723e-04	-6.803925e-04	-6.149666e-04
1.084484e-04	4.105526e-04	-8.537497e-05	-3.505717e-04

Table A.18: *Coefficient values for a typical second order Volterra filter with embedding dimension 16 (part 1).*

-8.344833e-04	1.116944e-05	-2.541548e-05	4.602092e-04
5.377988e-04	2.639458e-04	-4.485330e-04	-6.231609e-04
1.706031e-04	4.337268e-04	5.530453e-04	5.277566e-04
-3.498011e-04	2.632383e-04	1.071537e-05	-7.658395e-04
1.208513e-04	5.298278e-04	1.058413e-04	5.500713e-05
2.853365e-04	-8.022210e-05	6.248217e-05	7.525973e-04
5.378633e-04	4.712703e-04	3.575966e-04	-7.205130e-04
1.126260e-05	-8.037298e-04	-1.439881e-04	3.995226e-04
2.341572e-04	6.420936e-04	4.194829e-04	7.997850e-05
-6.047631e-04	-4.059567e-04	-3.260986e-04	-2.938496e-04
1.308320e-04	-4.856125e-04	5.347060e-05	1.133959e-05
-1.734440e-04	-2.689715e-05	-1.687929e-04	-1.790001e-05
5.368585e-04	5.200112e-04	-3.553279e-04	-5.300513e-04
-2.813460e-04	-5.438217e-04	3.596194e-05	6.541359e-05
-2.233397e-04	-1.543126e-04	-1.206861e-04	1.131515e-05
7.073525e-04	3.954922e-04	1.476402e-05	-2.081928e-04
2.572903e-04	6.036542e-04	4.973259e-04	6.650532e-04
9.069373e-04	-1.236312e-04	-2.360088e-04	-1.877977e-05
-2.347229e-04	5.816862e-04	7.081753e-04	-8.355975e-04

Table A.19: *Coefficient values for a typical second order Volterra filter with embedding dimension 16 (part 2).*

4.049131e-04	-6.460002e-03	7.332056e-03	-5.981240e-04
4.740708e-04	-5.084458e-05	-4.586559e-03	6.213292e-05
1.905263e-03	1.239552e-03	-1.140784e-03	-4.845918e-04
-4.489268e-04	5.786942e-04	-4.139840e-03	6.125333e-04
1.036485e-03	-4.082513e-04	-4.097064e-03	-7.991514e-04
3.786071e-03	3.971811e-03	1.231004e-03	-5.847604e-03
1.076843e-02	-9.467473e-03	-5.790780e-03	1.234505e-02
5.206791e-04	4.042855e-04	1.239383e-03	-2.395758e-03
4.914776e-03	-1.031683e-03	1.200067e-03	-1.822518e-03
-6.498973e-05	2.965726e-03	-8.739639e-03	2.056154e-03
3.549346e-05	9.214471e-04	5.618778e-04	3.383463e-03
2.815339e-03	4.721489e-03	-4.637315e-03	-2.761280e-03
-2.593706e-03	1.032989e-03		

Table A.20: *Coefficient reduction: values for a subset of 50 coefficients selected using the OLS algorithm.*

-6.142851e-04	-1.991372e-03	-3.168995e-04	6.188466e-04
1.156712e-04	5.259690e-04	-7.054743e-04	-7.371039e-04
1.018108e-03	2.373202e-04	7.793341e-04	-3.509098e-04
2.921002e-04	-1.197148e-03	9.016732e-04	-7.350617e-04
-3.256441e-05	1.324323e-04	1.318340e-03	-1.807564e-03
1.703456e-03	-2.368318e-04	-1.311882e-03	1.199111e-03
-1.361366e-03	9.334768e-04	2.047862e-03	-8.494636e-04
2.025451e-03	-6.673865e-04	-1.843798e-03	-6.109932e-04
-5.138334e-04	8.689020e-04	6.011152e-04	1.526359e-03
-2.077485e-03	1.712213e-03	-1.703128e-05	-1.926568e-03
1.233998e-03	-1.494274e-03	1.697243e-04	8.688469e-04
4.852389e-04	1.171503e-03	-1.758425e-03	1.956591e-03
9.661912e-04	-6.558400e-04		

Table A.21: *Coefficient reduction: values for a subset of 50 coefficients selected using the LSNT algorithm.*

Bibliography

[1] L. E. Kinsler, A. R. Frey, A. B. Coppens, and J. V. Sanders, *Fundamentals of Acoustics*. J. Wiley, 1982.

[2] Brüjel & Kaïer, "B&K half inch condenser microphone cartridges type 4133-4134," reference manual, Brüjel & Kaïer, 1966.

[3] C. Everard, *The Home Recording Handbook*. Virgin Books, 1985.

[4] E. Lindeman, "Routes to chaos in a non linear musical instrument model," in *A.E.S. 84th Convention*, (Paris), 1988.

[5] J. L. Meillier and A. Chaigne, "AR modeling of musical transients," in *ICASSP '91*, pp. 3649 — 3652, IEEE, May 1991.

[6] J. Laroche, "A new analysis/synthesis system of musical signals using Prony's method. Application to heavily damped percussive sounds.," in *ICASSP'89*, pp. 2053 — 2056, IEEE, 1989.

[7] M. Heal, "Buyers bible," *Making Music*, p. 46, Dec. 1992.

[8] I. Guyard and A. Chaigne, "Musical sound synthesis by finite differences method: P-C implementation and new developments.," in *A.E.S. 84th Convention*, (Paris), 1988.

[9] M. Karjalainen and V. Välimäki, "Implementation of real-time synthesis of string and wind instruments on a floating-point signal processor," in *Eusipco '92*, vol. 3, (Brussels), pp. 1717 — 1720, Aug. 1992.

[10] J. M. Adrien, R. Causse, and E. Ducasse, "Sound synthesis by physical models, application to strings," in *A.E.S. 84th Convention*, (Paris), 1988.

[11] A. Chaigne, "Vibration of viscoelastic guitar strings," *Journal of Sound and Vibration*, 1992.

[12] M. B. Sandler, "New synthesis techniques for high order all-pole filters," in *Proc. IEE part G*, vol. 139, pp. 9 — 16, IEE, Feb. 1992.

[13] G. B. Giannakis, "Cumulants: A powerful tool in signal processing," *Proc. IEEE*, vol. 75, pp. 1333 — 1334, Sept. 1987.

[14] T. Matsuoka and T. J. Ulrych, "Phase estimation using the bispectrum," *Proc. IEEE*, vol. 72, pp. 1403 — 1411, Oct. 1984.

[15] C. L. Nikias, "Higher order spectra in signal processing," in *Signal Processing V: Theories and Applications* (L. Torres, E. Masgrau, and M. A. Lagunas, eds.), pp. 35 — 41, Elsevier Science, 1990.

[16] C. L. Nikias, "ARMA bispectrum approach to nonminimum phase system identification," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-36, pp. 513 — 524, Apr. 1988.

[17] R. Pan and C. L. Nikias, "The complex cepstrum of higher order cumulants and nonminimum phase system identification," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-36, pp. 186 — 205, Feb. 1988.

[18] A. Polydoros and A. T. Fam, "The differential cepstrum: Definition and properties," in *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 77 — 80, Apr. 1981.

[19] A. Swami and J. M. Mendel, "ARMA parameter estimation using only output cumulants," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-38, pp. 1257 — 1265, July 1990.

[20] B. Friedlander and B. Porat, "Asymptotically optimal estimation of MA and ARMA parameters of non-Gaussian processes from high-order moments," in *Workshop on Higher Order Spectral Analysis*, (Vail, Colorado), pp. 245 — 250, June 28 — 30 1989.

[21] A. V. Oppenheim and J. S. Lim, "The importance of phase in signals," *Proc. IEEE*, vol. 69, pp. 529 — 541, May 1981.

[22] P. Huber, B. Kleiner, and T. Gasser, "Statistical methods for investigating phase in stationary processes," *IEEE Trans. Audio and Electroacoustics*, vol. AU-19, pp. 78 — 86, Mar. 1971.

[23] A. V. Oppenheim, J. S. Lim, and S. Curtis, "Signal synthesis and reconstruction from partial Fourier domain information," *Journal of the Optical Society of America*, vol. 73, pp. 1413 — 1420, nov 1983.

[24] R. C. Cox and D. M. Robinson, "Some notes on phase in speech signals," in *ICASSP '80*, pp. 150 — 153, IEEE, 1980.

[25] N. Rozario and A. Papoulis, "The identification of certain nonlinear systems by only observing the output," in *Workshop on Higher Order Spectral Analysis*, (Vail, Colorado), pp. 78 — 82, June 28-30 1989.

[26] C. Pezeshki, S. Elgar, R. C. Krishna, and T. Burtin, "Auto and cross bispectral analysis of a system of two coupled oscillators with quadratic nonlinearities posessing chaotic motion," *ASME Journal of Applied Mechanics*, vol. 59, pp. 657 — 663, Sept. 1992.

[27] C. Pezeshki, S. Elgar, and R. C. Krishna, "Bispectral analysis of possessing chaotic motion," *Journal of Sound and Vibration*, vol. 137, pp. 357–368, 1990.

[28] J. A. Cadzow, "Spectral estimation: an overdetermined rational model equation approach," *Proc. IEEE*, vol. 70(2), pp. 907 — 939, Sept. 1982.

[29] S. Billings, "Identification of nonlinear systems - a survey," in *Proc. IEE*, vol. 127, pp. 272 — 285, IEE, Nov. 1980.

[30] A. Lapedes and R. Farber, "Nonlinear signal processing using neural networks: Prediction and system modelling," prop. Proc. IEEE Report, July 1987.

[31] E. Van den Eijnde and J. Schoukens, "Steady-state analysis of a periodically excited nonlinear system," *IEEE Trans. Circuits and Systems*, vol. CIRC-37, pp. 232 — 242, Feb. 1990.

[32] O. Macchi and M. Jaidane-Saidane, "Adaptive IIR filtering and chaotic dynamics: Application to audiofrequency coding," *IEEE Trans. Circuits and Systems*, vol. CIRC-36, pp. 591 — 599, Apr. 1989.

[33] P. Duvaut, "A unifying and general approach to adaptive linear-quadratic discrete time Volterra filtering," in *ICASSP '89*, pp. 1166 — 1170, IEEE, 1989.

[34] C. Davila, A. Welch, and H. Rylander, "A second-order adaptive Volterra filter with rapid convergence," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-35, pp. 1259 — 1263, Sept. 1987.

[35] J. M. Vesin, "An alternative scheme for adaptive nonlinear filters using radial basis functions," in *Eusipco '92*, vol. 2, pp. 1069 – 1072, Aug. 1992.

[36] J. Chen and J. Vandewalle, "Study of adaptive nonlinear echo canceller with Volterra expansion," in *ICASSP '89*, pp. 1376 — 1379, IEEE, 1989.

[37] S. Chen, G. Gibson, C. Cowan, and P. M. Grant, "Reconstruction of binary signals using an adaptive radial basis function equaliser," *Signal Processing*, vol. 22, pp. 77 — 92, Jan. 1991.

[38] V. Mathews and J. Lee, "A fast recursive least-squares second order Volterra filter," in *ICASSP '88*, pp. 1383 — 1386, IEEE, 1988.

[39] K. Kim and E. Powers, "Fast RLS algorithm for a second-order Volterra filter," in *29th IEEE Conference on Decision and Control*, pp. 3520 — 3521, IEEE, 1990.

[40] M. Dokic and P. Clarkson, "Performance characteristics of a third order adaptive Volterra filter," in *IEEE International Symposium on Circuits and systems*, pp. 2785 — 2788, IEEE, 1992.

[41] B. Picinbono and P. Duvaut, "Optimal linear-quadratic systems for detection and estimation," *IEEE Trans. Information Theory*, vol. IT-34, pp. 304 — 311, Mar. 1988.

[42] B. Picinbono, "Higher-order statistical signal processing with Volterra filters," in *Workshop on Higher Order Spectral Analysis*, (Vail, Colorado), pp. 62 — 67, June 28-30 1989.

[43] K. Kim and E. Powers, "A digital method of modeling quadratically nonlinear systems with a general random input," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-36, pp. 1758 — 1769, Nov. 1988.

[44] T. Koh and E. Powers, "Second-order Volterra filtering and its application to nonlinear system identification," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-33, pp. 1445 — 1455, Dec. 1985.

[45] M. Powell, "Radial basis functions for multivariable interpolaton: A review," in *Algorithms for Approximation* (J. Mason and M. Cox, eds.), pp. 143 — 168, Oxford Science, 1987.

[46] T. Shepherd and D. S. Broomhead, "Nonlinear signal processing using radial basis functions," in *Advanced Signal Processing Algorithms and Architectures SPIE Conf. Proc. no 1348*, July 1990.

[47] K. Nisbet, B. Mulgrew, and S. McLaughlin, "Nonlinear prediction and the Wiener process," in *Advanced Signal Processing - SPIE Conf Proc no 1565 q*, pp. 244 - 254, July 1991.

[48] B. Mulgrew, K. Nisbet, and S. McLaughlin, "Degeneracy in non-linear predictors," in *Eusipco '92*, vol. 2, pp. 791 – 794, Aug. 1992.

[49] S.Chen, C. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Trans. Neural Networks*, vol. 2, pp. 302 — 309, Mar. 1991.

[50] M. Campbell and C. Greated, *The Musician's Guide to Acoustics*. J. M. Dent & Sons Ltd., 1989.

[51] Brüjel & Kaïer, "B&K measuring amplifiers types 2608, 2610 and 2616," reference manual, Brüjel & Kaïer, 1967.

[52] Brüjel & Kaïer, "B&K power amplifier type 2706," reference manual, Brüjel & Kaïer, 1966.

[53] Blue Chip Technology, *ADC input/output card ADC1014*. Hawarden Industrial Park, Manor Lane, Deeside, Clwyd, 1989.

[54] L. Cohen, "Time frequency distributions - a review," *Proc. IEEE*, vol. 77, no. 7, pp. 941 — 981, 1989.

[55] A. Swami and J. M. Mendel, "Cumulant based approach to the harmonic retrieval problem," in *ICASSP'88*, pp. 2264 — 2267, IEEE, 1988.

[56] S. Nam, S. Kim, and E. Powers, "Utilization of digital polyspectral analysis to estimate transfer functions of cubically nonlinear systems with non Gaussian inputs," in *ICASSP '89*, pp. 2306 — 2309, IEEE, 1989.

[57] C. K. Papadopolous and C. L. Nikias, "Bispectrum estimation of transient signals," in *ICASSP '88*, pp. 2404 — 2407, IEEE, 1988.

[58] C. K. Papadopolous and C. L. Nikias, "Transient signal estimation with higher order statistics," in *Workshop on Higher Order Spectral Analysis*, (Vail, Colorado), pp. 89 — 93, June 28-30 1989.

[59] C. L. Nikias, "Higher order spectra in signal processing," in *Workshop on Higher Order Spectral Analysis*, (Vail, Colorado), June 28-30 1989.

[60] C. L. Nikias and M. R. Raghuveer, "Bispectrum estimation: A digital signal processing framework," *Proc. IEEE*, vol. 75, pp. 869 — 891, July 1987.

[61] M. R. Raghuveer and C. L. Nikias, "Bispectrum estimation : a parametric approach," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-33, pp. 1213 — 1230, Oct. 1985.

[62] J. M. Mendel, "Use of higher-order statistics in signal processing and system theory: An update," in *SPIE Conf. on Advanced Algorithms and Architectures for Signal Processing III*, (San Diego, CA), Jan. 1988.

[63] D. R. Brillinger, "The identification of polynomial systems by means of higher order spectra," *Journal of Sound and Vibration*, vol. 12, pp. 301 — 313, 1970.

[64] D. R. Brillinger and M. Rosenblatt, "Asymptotic theory of estimates of k-th order spectra," in *Spectral Analysis of Time Series* (B. Harris, ed.), pp. 153 — 188, Wiley, 1967.

[65] D. R. Brillinger and M. Rosenblatt, "Computation and interpretation of k-th order spectra," in *Spectral Analysis of Time Series* (B. Harris, ed.), pp. 189 — 232, Wiley, 1967.

[66] K. Hasselmann, W. Munk, and G. MacDonald, "Bispectra of ocean waves," in *Time Series Analysis* (M. Rosenblatt, ed.), pp. 125 — 139, Wiley, 1963.

[67] H. Cramér, *Random Variables and Probability Distributions*. Cambridge University Press, 1970.

[68] H. Cramér, *Mathematical Methods of Statistics*. Princeton University Press, 1946.

[69] J. Doob, *Stochastic Processes.* Wiley, 1953.

[70] H. Cramér, *The Elements of Probability Theory.* Wiley, 1955.

[71] S. Alshebeili, B. Mertzios, and A. Venetsanopoulos, "Efficient implementation of nonlinear Volterra systems via Laguerre orthogonal functions," in *Eusipco '92*, vol. 2, pp. 685 - 688, Aug. 1992.

[72] J. Barret, "Formula for output autocorrelation and spectrum of a Volterra system with stationary gaussian input," in *Proc. IEE*, vol. 127, pp. 286 — 289, IEE, Nov. 1980.

[73] S. Haykin, *Introduction to Adaptive Filters.* McMillan, 1984.

[74] S. Haykin, *Adaptive Filter Theory.* Prentice-Hall international, 1991.

[75] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes in C.* Cambridge University Press, 1988.

[76] M. Schetzen, "Nonlinear system modeling based on the Wiener theory," *Proc. IEEE*, vol. 69, pp. 1557 — 1573, Dec. 1981.

[77] B. Picinbono and P. Duvaut, "Geometrical properties of optimal Volterra filters for signal detection," *IEEE Trans. Information Theory*, vol. IT-36, pp. 1061 — 1068, Sept. 1990.

[78] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Sytems.* Wiley, 1980.

[79] F. Yassa, "Identification of the Volterra kernels of nonlinear systems," in *Proc. IEE*, vol. 127, pp. 296 — 304, IEE, Nov. 1980.

[80] B. Everitt, *Cluster Analysis.* Wiley, 1980.

[81] T. Kohonen, "The self-organising map," in *Proc. IEEE*, vol. 78, pp. 1464 — 1480, Sept. 1990.

[82] S. Kay, *Modern Spectral Estimation, Theory and Application.* Prentice-Hall, 1988.

[83] L. Marple, *Digital Spectral Analysis With Applications.* Prentice-Hall, 1987.

[84] M. B. Sandler, "Implementation strategy for high order recursive filters," *Electronics Letters*, vol. 25, pp. 1058 — 1060, Aug. 1989.

[85] M. B. Sandler, "Algorithm for high precision root finding from high order lpc models," in *IEE part I*, vol. 138, pp. 596 — 602, IEE, Dec. 1991.

[86] A. Björck, "Solving linear least squares problems by Gram-Schmidt orthogonalisation," *Nordisk Tidskr. Informations-Behandling*, vol. 7, pp. 1 — 21, 1967.

[87] E. Chng, S. Chen, and B. Mulgrew, "Backtracking OLS algorithm for model selection," in *Mathematical Aspects of Digital Signal Processing*, IEE, Feb. 1994.

Publications

- S. Hovell, B. Mulgrew. "Analysis of drum sounds using higher order spectra", in *Eusipco '92*, vol. 3, pp 1701 – 1704, Aug. 1992.

- S. Hovell, B. Mulgrew. "Nonlinear prediction and regeneration of acoustic percussion sounds", in *Adaptive Methods and Emergent Techniques for Signal Processing and Communications*, pp 369 – 375, June, 1993.

- S. Hovell, B. Mulgrew. "Application of Quadratic Phase Coupling to Acoustic Signal Characterisation", submitted to *The Journal of the Audio Engineering Society*, February, 1994.

Analysis of Drum Sounds using Higher Order Spectra

Simon A. Hovell and Bernard Mulgrew

Department of Electrical Engineering, University of Edinburgh,
King's Buildings, Edinburgh EH9 3JL, Scotland, UK
Telephone +44 31 650 5655 Fax +44 31 650 6554

This paper examines the application of higher order spectral analysis to the analysis and modelling of drum sounds. Bispectral analyses of a drum sound are presented, and it is shown that there is information present in the higher order spectra of drum sounds which is not present in the power spectrum. This information could be used to design a more accurate model of the sound.

1. Introduction

Currently, the most accurate method of reproducing drum sounds has proved to be through sampling. The disadvantage of this is that any particular sample is unchanging. In reality, the sound produced by a particular drum varies from beat to beat depending on how hard it is struck, where on the drum skin the stick hits, and the rate at which the skin is hit. In order to reproduce this using samples, it would be necessary to store a vast number of samples for any specific drum. Even if this were accomplished, it would be almost impossible to simulate the variety shown by a real drummer.

Although models have been synthesised which are spectrally almost identical to real drums, these models have always *sounded* artificial. it has been shown [1] that in many audio and visual signals, the phase information present must be used in order to create an accurate reconstruction of the signal. It is probable that a drum sound is one such signal. The purpose behind this work is to develop a model that is capable of accurately reproducing the sound produced by real drums. Once designed, such a system could be adapted to model different drums, and different tunings of the same drum. It would therefore be far more flexible than a sample based system, and also more economical.

2. Higher Order Spectral Analysis

Analysis of the power spectral density (PSD) of discrete time and stochastic processes has been a fundamental tool in digital signal processing for some time. The PSD is obtained by taking a Fourier transform of the autocorrelation function of the process; this autocorrelation function can be regarded as the second order term in an infinite cumulant series pertaining to the process. By examining higher order terms in this series, and also their Fourier transforms, more information about processes can be revealed.

Given a set of n real random variables $\{ x_1, x_2, \cdots, x_n \}$, their joint cumulants of order $r = k_1 + k_2 + \cdots + k_n$ are defined [2,3] as

$$c_{k_1 \cdots k_n} \equiv (-j)^r \left. \frac{\partial^r \ln \phi(\omega_1, \omega_2, \cdots, \omega_n)}{\partial \omega_1^{k1} \partial \omega_2^{k2} \cdots \partial \omega_n^{kn}} \right|_{\omega_1 = \omega_2 = \cdots = \omega_n = 0} \tag{2.1}$$

where

$$\phi(\omega_1, \omega_2, \cdots, \omega_n) = E\left\{ e^{j(\omega_1 x_1 + \omega_2 x_2 + \cdots + \omega_n x_n)} \right\} \tag{2.2}$$

is their joint characteristic function.

The Nth order spectrum $C(\omega_1, \omega_2, \cdots \omega_{N-1})$ of the process $\{X(k)\}$ is defined to be the Fourier transform of its Nth order cumulant sequence [2,3]:

$$C(\omega_1) = \sum_{\tau_1 = -\infty}^{+\infty} c_2(\tau_1) e^{-j(\omega_1 \tau_1)}$$

(N=2: Power spectrum)

$$C(\omega_1, \omega_2) = \sum_{\tau_1 = -\infty}^{+\infty} \sum_{\tau_2 = -\infty}^{+\infty} c_3(\tau_1, \tau_2) e^{-j(\omega_1 \tau_1 + \omega_2 \tau_2)}$$

(N=3: Bispectrum)

$$C(\omega_1, \omega_2, \omega_3) =$$

$$\sum_{\tau_1 = -\infty}^{+\infty} \sum_{\tau_2 = -\infty}^{+\infty} \sum_{\tau_3 = -\infty}^{+\infty} c_4(\tau_1, \tau_2, \tau_3) e^{-j(\omega_1 \tau_1 + \omega_2 \tau_2 + \omega_3 \tau_3)}$$

(N=4: Trispectrum)

(2.3)

One of the most important properties of higher order spectra is that they do not suppress all phase information. The third order spectrum (bispectrum) contains information about quadratic phase coupling amongst harmonic components.

Consider the signal

$$X(k) = \cos(\lambda_1 k + \theta_1) + \cos(\lambda_2 k + \theta_2) + \cos(\lambda_3 k + \theta_3) \tag{2.4}$$

where $\lambda_1 + \lambda_2 = \lambda_3$. It is evident that if θ_3 is an independent random variable, then the three cosines are independent. On the other hand, if $\theta_3 = \theta_1 + \theta_2$, then the third cosine is a result of phase coupling between the other two. The power spectra for both these cases are identical, but in the second case, the bispectrum would reveal the quadratic phase coupling. In a similar fashion, the trispectrum detects cubic phase coupling.

3. Acoustic Theory

A vibrating circular membrane, radius a, fixed at the rim can be represented by the Helmholz equation [4],

$$\frac{\Theta \partial^2 R}{\partial r^2} + \frac{\Theta}{r}\frac{\partial R}{\partial r} + \frac{R}{r^2}\frac{\partial^2 \Theta}{\partial \theta^2} + k^2 R\theta = 0 \tag{3.1}$$

assuming a standard solution, of the form

$$y(r, \theta, t) = \Theta(\theta) R(r) e^{j\omega t} \tag{3.2}$$

where $k = \frac{\omega}{c}$ and c is a constant dictated by the membrane.

The particular solution to the equation is

$$\Theta(\theta) = \cos(m\theta + \gamma) \tag{3.3.1}$$

$$R(r) = AJ_m(K_r) + BY_m(K_r) \tag{3.3.2}$$

where $J_m(K_r)$ and $Y_m(K_r)$ are first and second Bessel functions of order m. Physically, the result of this is standing wave vibrations at frequencies

$$f_{mn} = \frac{1}{2\pi}\frac{j_{mn} c}{a} \tag{1.4}$$

It is interesting to note that none of the overtones is at a frequencies which is a harmonics of the fundamental (0,1) mode of vibration. In practice, when a drum is struck, there is a damping effect due to the air [4]. As well as hastening the decay of the oscillations, this also *slows* the vibrations, and thus the overtones are actually at lower frequencies than those predicted by theory. In addition, the damping effect of the air is more pronounced on higher frequency oscillations and radial oscillations (i.e. those with nodal circles), leading to a preponderance of $(n,1)$ modes. The principal contributors to the sound of a drum are therefore the (1,1), (2,1) and (3,1) modes, although many more are present in the initial percussive attack.

4. Results

Data was generated using a "Pearl Low Tom" drum sampled in an anechoic chamber. The sampling was carried out at 33.3 kHz, using a B&K 4133 microphone, and twelve samples were collected on an IBM pc clone. The resulting three second long time series were then split into 49 sections, each 2048 points long, for bispectral and power spectral analysis. Data was windowed with a Hamming window, and fast Fourier transformed. The bispectrum and PSD for each section was then calculated, and then ensemble averaged over the twelve samples. As the signal was non-stationary, it was impossible to average the power spectra and bispectra over time, and as a consequence of this figures 1-3 display results for the first 0.06 second slice of the signal, as this contains transient high frequency components of the sound which are not present in subsequent time slices. Figure 1 shows the power spectrum for the initial slice, and figures 2 and 3 show its corresponding bispectrum and phase spectrum. The degree of phase coupling at some frequency $\omega_n = \omega_1 + \omega_2$ is shown at point (ω_1,ω_2). As the bispectrum is symmetrical in ω_1 and ω_2, and meaningless at points where $\omega_1 + \omega_2 >$ Fs/2, values have only been displayed for one quadrant, where $\omega_1 <= \omega_2$ and $\omega_1 + \omega_2 <=$ Fs/2. The data used to create the bispectrum plot was also frequency averaged by a factor 4 in order to reduce computation time. It is apparent from figure 2 that there is a significant degree of quadratic phase coupling shown in the bispectrum of the drum sound, particularly between higher frequencies and the fundamental (0,1) mode (which can be seen from figure 1 to be at approximately 244 Hz) and also the (2,1) mode (472 Hz). Varying degrees of phase coupling can be seen in bands up the frequency spectrum as far as 11.5 kHz (bin 180). Examination of the power spectrum and bispectrum of subsequent time slices showed that, in accordance with the theory of section 3, almost all the higher frequency components of the sound disappear within 0.18 seconds, with a commensurate reduction in the amplitude of the higher frequencies of the bispectrum.

5. Conclusions

A drum was sampled and the resultant data was analysed using the power spectrum and the bispectrum. The results can be seen in figures 1 to 3. It has been established that phase information can be of crucial importance in audio and visual signals, so it is probable that the omission of phase information has been a contributory factor in the poor quality of previous models. In order to synthesise a more accurate reproduction of the original sound, it would be necessary to incorporate phase information. Unfortunately, as is amply demonstrated in figure 3, the phase spectrum of a simple drum is extremely complex, with no apparent structure.

Developing a model of this would seem to be a most complex task. The bispectrum of the sound, however, shows a high degree of form and indicates that there is a significant amount of phase coupling between frequencies, especially in the initial percussive attack. It is therefore evident that much of the information that characterises a drum sound can be found, in a comprehensible form, in its higher order spectra.

References

1. A. V. Oppenheim and J. S. Lim, "The Importance of Phase in Signals," *Proc. IEEE*, 69, pp. 529 - 541 (May 1981).
2. C. L. Nikias and M. R. Raghuveer, "Bispectrum Estimation: A Digital Signal Processing Framework," *Proc. IEEE*, 75, pp. 869 - 891 (July 1987).
3. J. M. Mendel, "Use of Higher-Order Statistics in Signal Processing and System Theory: An Update," *SPIE Conf. on Advanced Algorithms and Architectures for Signal Processing III*, San Diego, CA (August 1988).
4. L. E. Kinsler, A. R. Frey, A. B. Coppens, and J .V. Sanders, "Fundamentals of Acoustics," J. Wiley (1982).

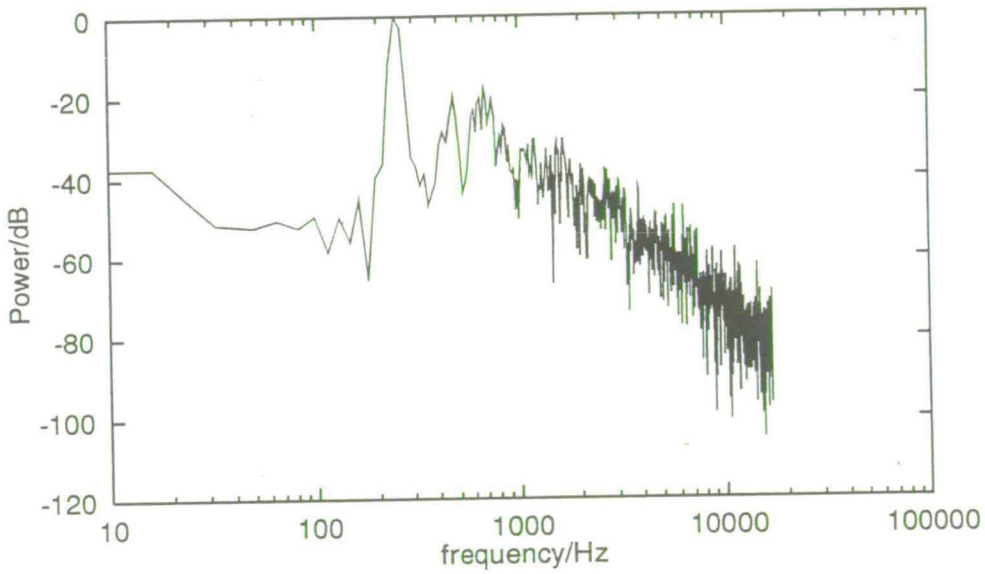


Figure (1): Initial power spectrum of Drum Sound

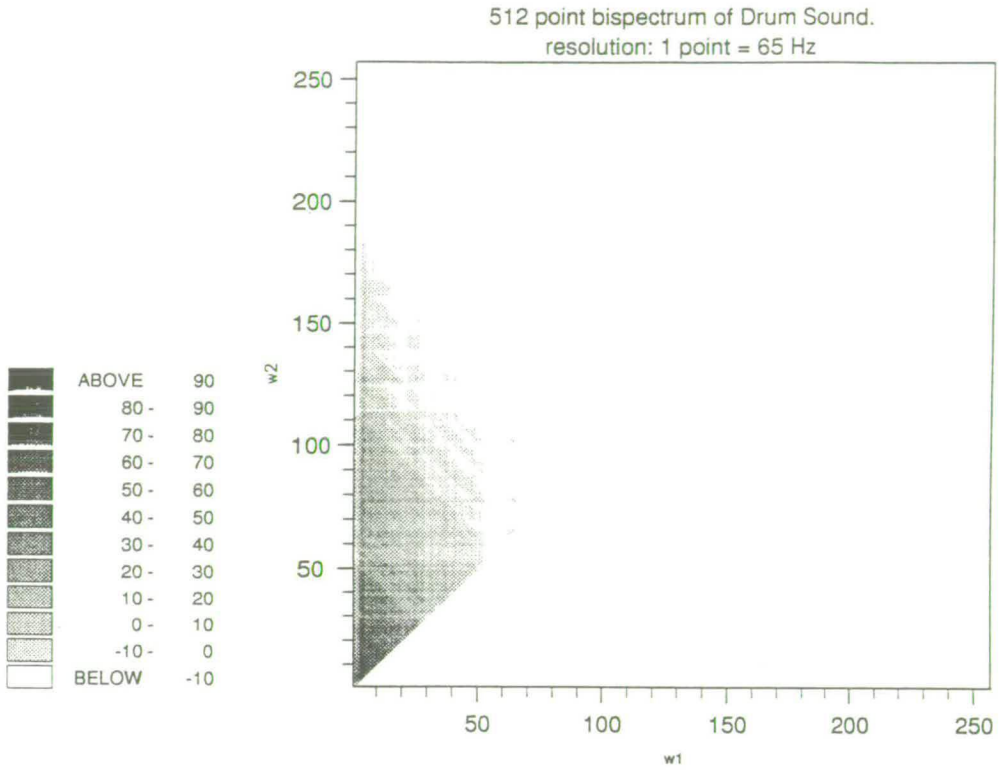


Figure (2): Initial bispectrum of Drum Sound.
ω_1 and ω_2 are shown, while the bispectrum,
$C(\omega_1, \omega_2)$, is implied.
Amplitude of the sum frequency is plotted on a logarithmic scale.

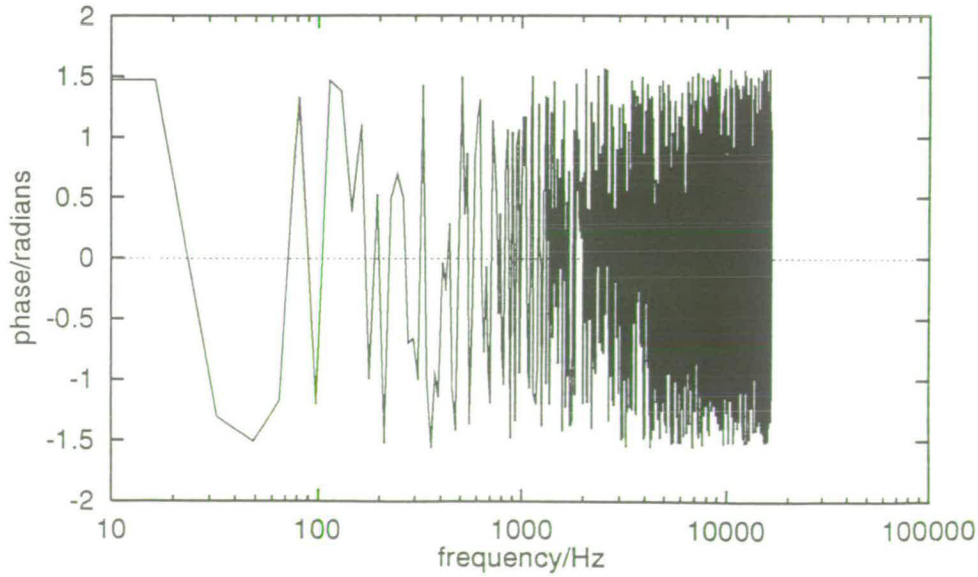


Figure (3): Initial phase spectrum of Drum Sound

Nonlinear Prediction and Regeneration of Acoustic Percussion Sounds

S. Hovell, B. Mulgrew

Department of Electrical Engineering, University of Edinburgh, Mayfield Road, Edinburgh EH9 3JL, Scotland, U.K.

Tel: (+44) 31 650 5655, Fax: (+44) 31 650 5580, Email: sah@ee.ed.ac.uk

1 Introduction

Although many models of drum sounds have been made in the past, despite being spectrally almost identical to real drums, these models have always *sounded* artificial. It is apparent that the most crucial part of a drum sound, the percussive attack, is highly nonlinear, and thus not easily modelled using conventional techniques. As a result, the most accurate, and popular method of reproducing drum sounds has been through sampling. The fundamental disadvantage of this is that any particular sample is unchanging. In reality, the sound produced by a particular drum varies from beat to beat depending on how hard it is struck, where on the drum skin the stick hits, and the rate at which the skin is hit. In order to reproduce this using samples, it would be necessary to store a vast number of samples for any specific drum. Even if this were accomplished, it would be almost impossible to simulate the variety shown by a real drummer.

The purpose behind this work is to develop a model that is capable of accurately reproducing the sound produced by real drums. Once designed, such a system could be adapted to model different drums, and different tunings of the same drum. It would therefore be far more flexible than a sample based system, and also more economical.

2 Acoustic Theory

A vibrating circular membrane, radius a, fixed at the rim can be represented by the Helmholz equation [1]. Physically, the result of this is standing wave vibrations at frequencies

$$f_{mn} = \frac{1}{2\pi}\frac{j_{mn}c}{a} \tag{1}$$

where c is a constant dictated by the membrane. This can be seen in fig(1). It is interesting to note that none of the overtones is at a frequency which is a harmonic of the fundamental (0,1) mode of vibration. In practice, when a drum is struck, there is a damping effect due to the air [1]. As well as hastening the decay of the oscillations, this also *slows* the vibrations, and thus the overtones are actually at lower frequencies than those predicted by theory. In addition, the damping effect of the air is more pronounced on higher frequency oscillations and radial oscillations (i.e. those with nodal circles), leading to a preponderance of $(n,1)$ modes. The principal contributors to the sound of a drum are therefore the (1,1), (2,1) and (3,1) modes. As the drum is struck, and the membrane undergoes its initial displacement, however, many more modes are present, and it would be reasonable to assume that there is a degree of nonlinear interaction between these modes. This assumption is borne out by bispectral analysis of the signal.

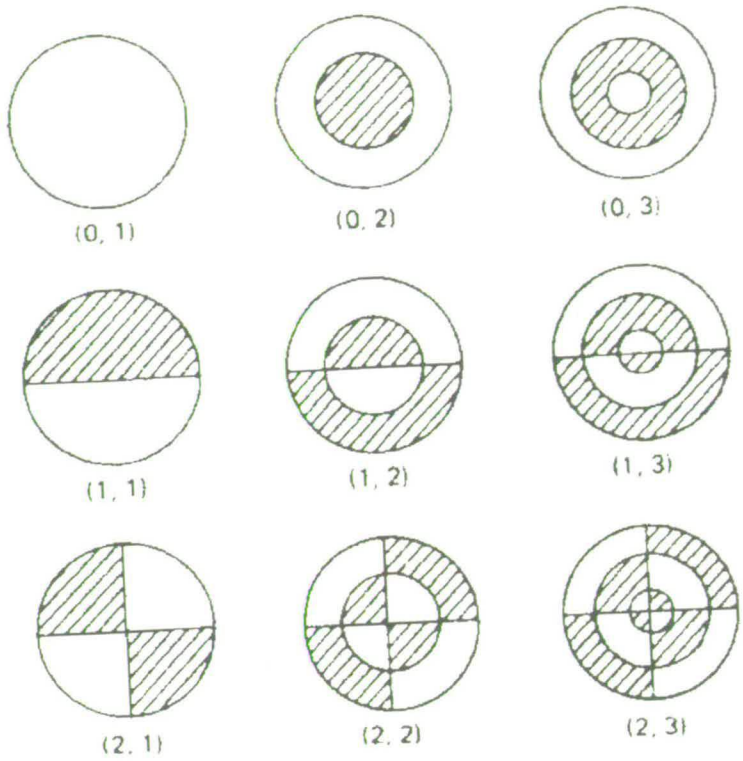


Figure 1: Modes of vibration for a circular membrane

The bispectrum of a signal $x(t)$ can be defined as the Fourier transform of its third order cumulant sequence [2]:

$$B(\omega_1, \omega_2) = \sum_{\tau_1=-\infty}^{+\infty} \sum_{\tau_2=-\infty}^{+\infty} c_3(\tau_1, \tau_2) e^{-j(\omega_1 \tau_1 + \omega_2 \tau_2)} \tag{2}$$

where

$$c_3(\tau_1, \tau_2) = E\left[x(t)\, x(t+\tau_1)\, x(t+\tau_2)\right]. \tag{3}$$

One of the most important properties of the bispectrum is that it does not suppress all phase information. By averaging the signal bispectrum across a set of data samples, information about quadratic phase coupling between different frequencies is revealed. Consider the signal

$$nx(k) = \cos(\lambda_1 k + \theta_1) + \cos(\lambda_2 k + \theta_2) + \cos(\lambda_3 k + \theta_3) \tag{4}$$

where $\lambda_1 + \lambda_2 = \lambda_3$. It is evident that if θ_3 is an independent random variable, then the three cosines are independent. On the other hand, if θ_3 is dependent on θ_1 and θ_2, i.e. $\theta_3 = k\theta_1 + l\theta_2$, where k and l are constants, then the third cosine is a result of phase coupling between the other two. The power spectra for both these cases are identical. The bispectrum for the former is, however, zero, whilst in the second case a peak would be present at (λ_1, λ_2), revealing the phase coupling. Figure (2) shows the averaged bispectrum for the first 0.02s of a typical drum sound. Clearly a significant degree of phase coupling is present. The transient nature of the percussive attack is readily demonstrated by looking at the bispectrum for the subsequent 0.02s (fig (3)), where it can be seen that the amount of phase coupling has reduced dramatically. Models based on linear filter structures cannot incorporate this information, so it becomes necessary to look at nonlinear techniques.

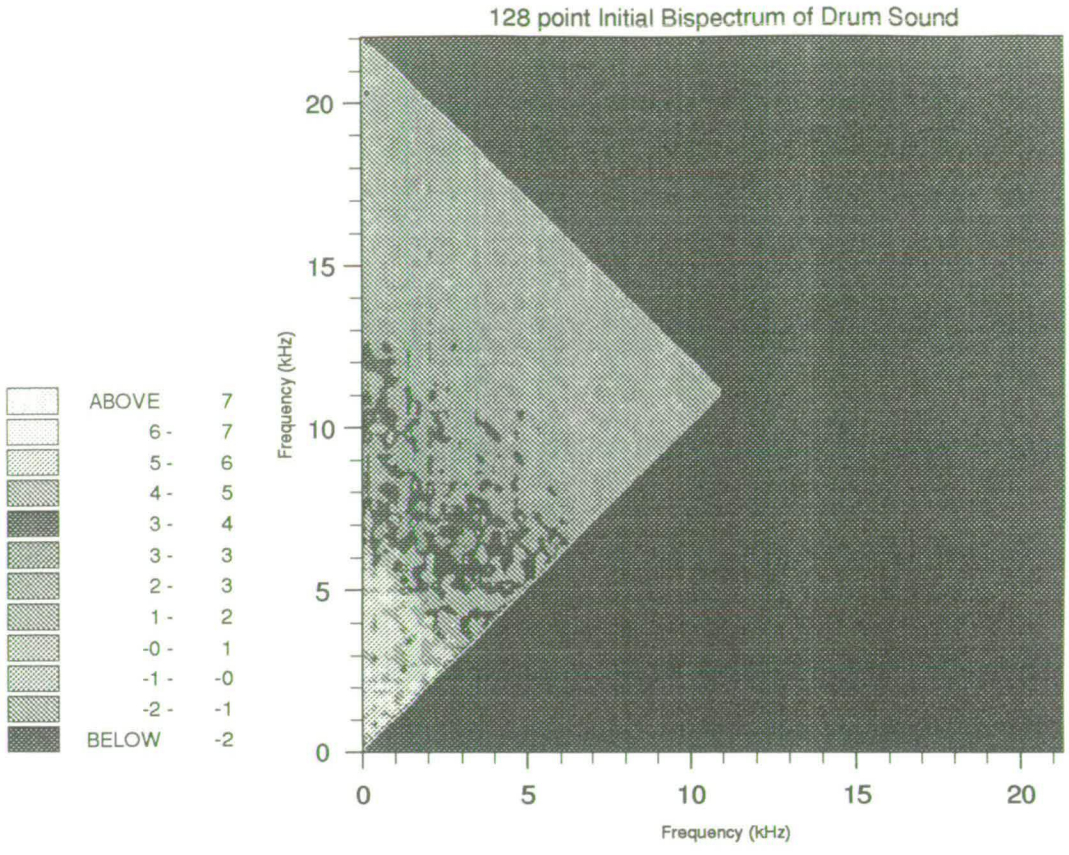


Figure 2: Bispectrum of the first 0.02 seconds of sound generated by striking a drum

3 Regeneration

3.1 Application

The nature of drum sounds necessitates some interesting considerations when attempting to regenerate them. Conventional models for nontransient, stationary signals are often white noise driven, and can be trained using large data sets. The nonstationary nature of the drum sound, however, precludes the use of large training sets, especially during the initial percussive attack. Driving the filter from a constant source is unfeasible due to the transient nature of the sound. One possible solution is to design a multistep predictive filter, i.e. one which uses each predicted output as the next input:

$$\tilde{x}(t+1) = f(X_t) \tag{5}$$

for some embedding vector $X_t^T = (\tilde{x}(t), \tilde{x}(t-1), \cdots, \tilde{x}(t-n))$ at time t.

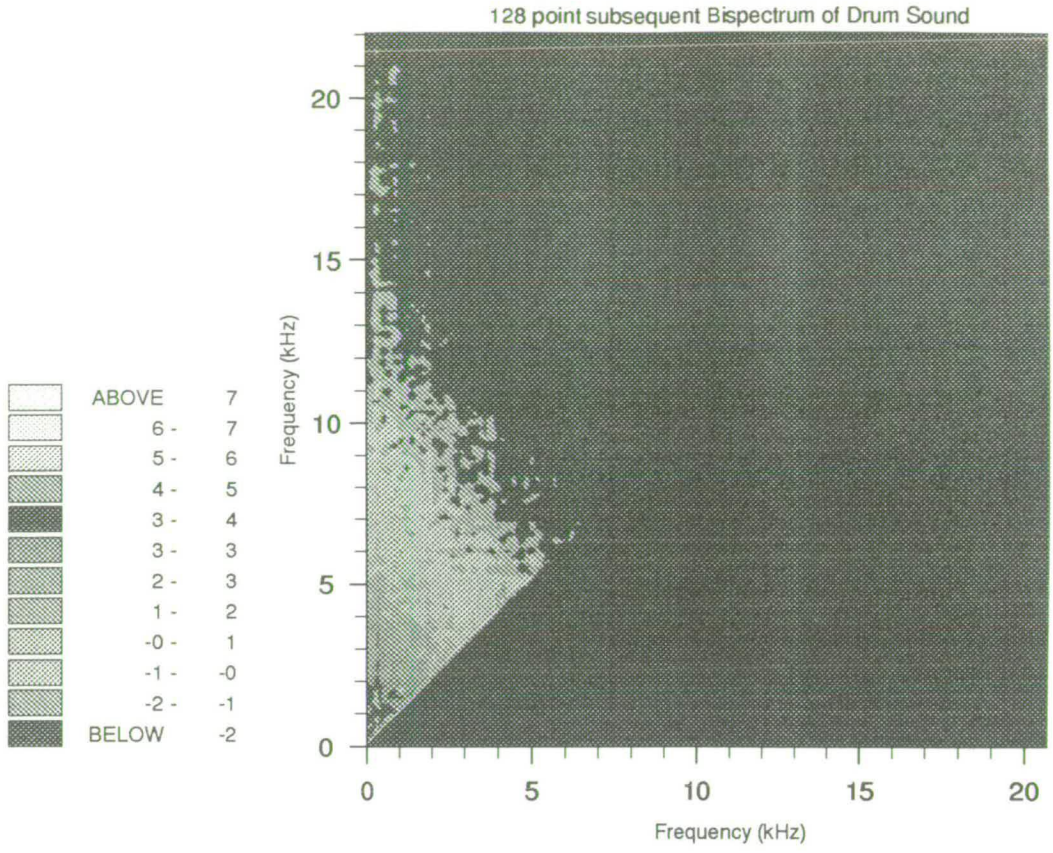


Figure 3: Bispectrum of the second 0.02 seconds of sound generated by striking a drum

3.2 Volterra filters

A Volterra filter of order p (where p is a positive integer) is a nonlinear system with an input-output relationship that can be described by a finite Volterra expansion[3]:

$$y(t) = h_0 + H_1[x(t)] + H_2[x(t)] + \cdots + H_p[x(t)] + \cdots \tag{6}$$

where

$$H_n[x(t)] = \sum_{-\infty}^{\infty} \cdots \sum_{-\infty}^{\infty} h_n(\tau_1, \ldots, \tau_n) x(t-\tau_1) \cdots (t-\tau_n)\, d\tau_1 \cdots d\tau_n \tag{7}$$

where $x(n)$ and $y(n)$ are the filter input and output respectively, and $H_k(\tau_1^k, \ldots, \tau_k^k)$ denotes the k-dimensional filter weight sequence associated with the kth Volterra kernel. The constant h_0 corresponds to $k = 0$ and is often omitted, or set to zero [4]. When this is the case, it can be seen that by setting $p = 1$, the expression obtained is a standard convolution, characterising a linear filter. It is more convenient to write (6) as a vector inner product, $y(n) = H_{vp}^T X_{vp}$, where H_{vp} contains the coefficients of the p^{th} order filter, and X_{vp} contains the corresponding combinations of X_t. From this, it is evident that H_{vp} can be found using standard least squares techniques [5, 4].

3.3 Radial basis function analysis

A radial basis function can be defined as follows [6]:

$$y(n) = \mu_0 + \sum_{i=0}^{N-1} \mu_i s\left(||x_k - C_i||\right) \tag{8}$$

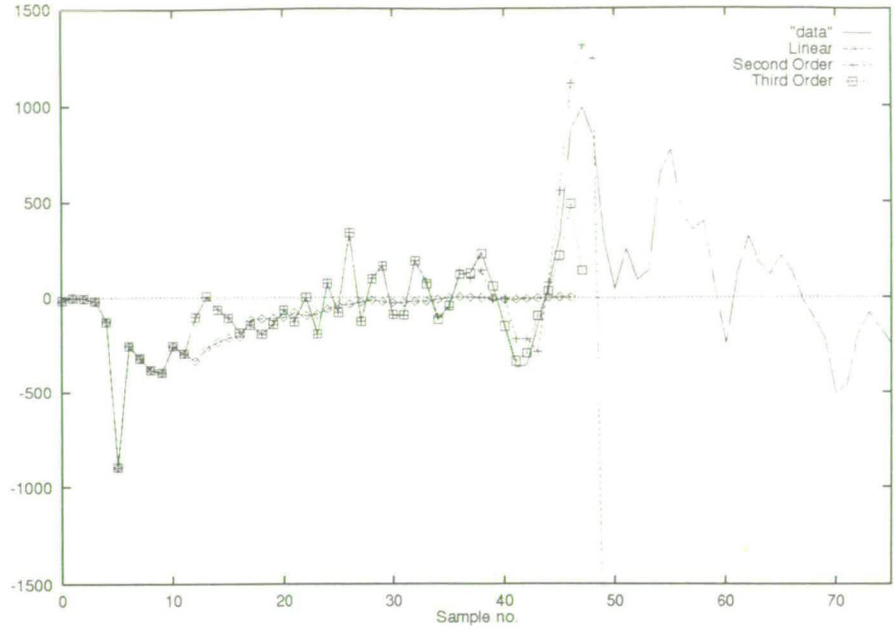


Figure 4: Comparison of tracking capability for different order Volterra filters: training data length 70

where s is an appropriately chosen function, for example the thin plate spline $s(x) = x^2 \log(x)$ or the Gaussian function $s = \exp(-x^2/\beta^2)$.The centres, c_i, are typically chosen at random to cover the space generated by the time series [7, 6]. Once again, a least squares approach can be used to calculate the u_i. The inner product in this case is $y(n) = U^T X_r$ where U contains the coefficients and $X_{ri} = s\left(||x_k - C_i||\right)$.

4 Results

Figures (4) and (5) show the results for a regenerative Volterra filter with an embedding dimension of 12. Clearly, the filter with the *smaller* training set performs more successfully, and neither filter performs well beyond the training data. This is further evidence of the extreme nonstationarity of the sound. Furthermore, the third order Volterra filter offers no significant improvement over the second order filter, implying that the phase coupling detected in the drum bispectrum is successfully reproduced by the second order Volterra filter terms, whilst the third order terms contribute little to the final output. Radial basis functions (figures (6) and (7)) however, do not perform as successfully, although the RBF is more inherently stable.

Both techniques generate large number of terms from the embedding vector - the radial basis function method produces an arbitrary number of centres, each of which is as long as the embedding vector, and the Volterra series produces $\left[\prod_{i=0}^{i=P-1}(n+i)\right]/p!$ terms for a pth order filter with embedding dimension n. This is computationally extravagant, and there is scope for removing possible redundancy. Preliminary trials using two methods, orthogonal least squares weight selection [7] and the somewhat less elegant LS weight reduction technique, in which weights with values below a noise threshold are removed and the LS algorithm reapplied, have met with some success, especially with the Volterra series. Reduced Volterra series with size approaching that of the data successfully regenerated have operated with no significant increase in output error.

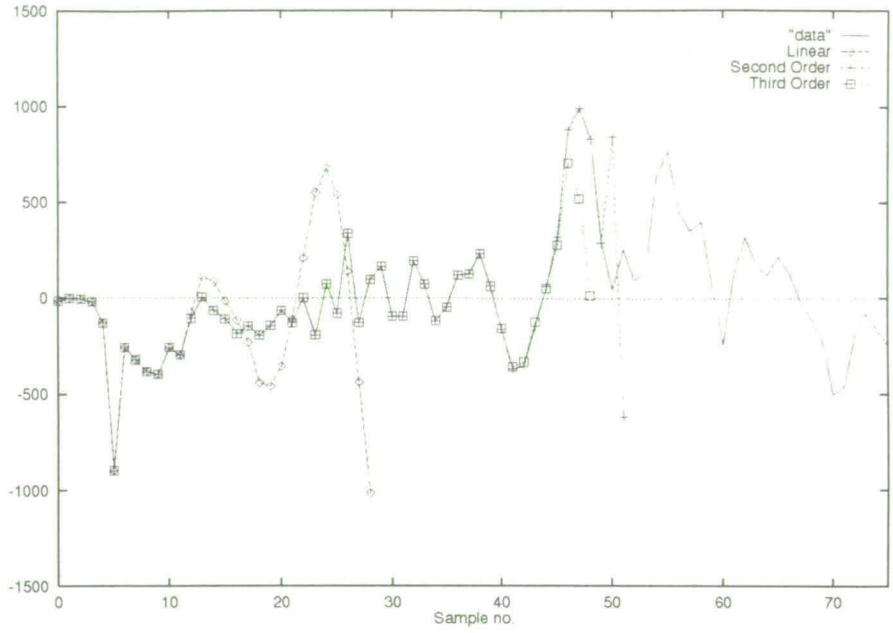


Figure 5: Comparison of tracking capability for different order Volterra filters: training data length 50

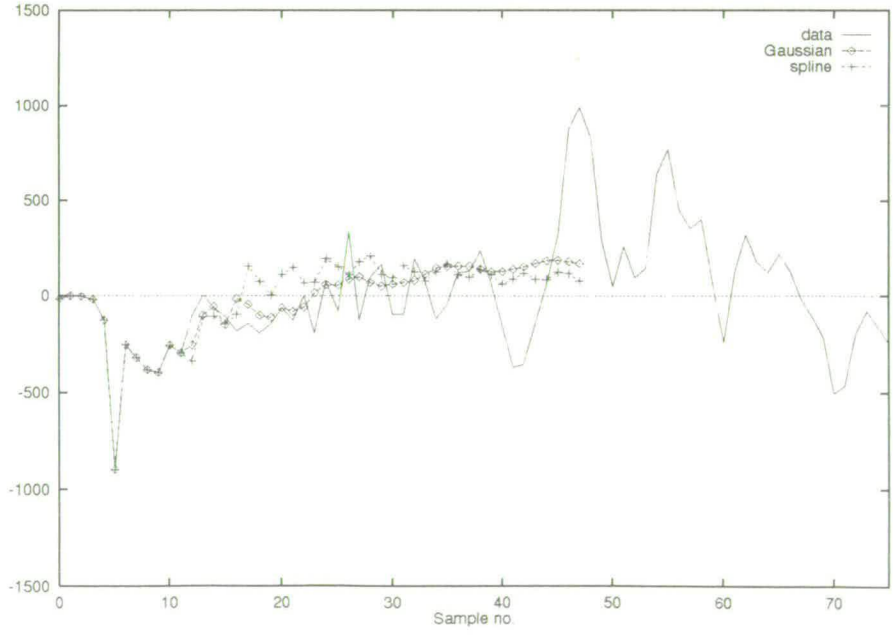


Figure 6: Comparison of tracking capability for RBF's using thin plate spline and Gaussian nonlinearities: training data length 70

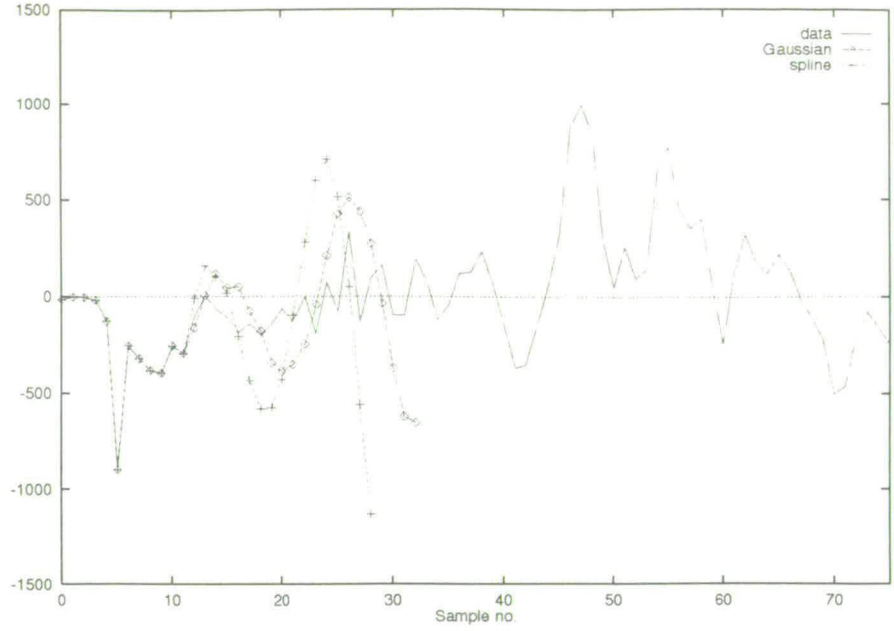


Figure 7: Comparison of tracking capability for RBF's using thin plate spline and Gaussian nonlinearities: training data length 70

5 Further Work

The techniques detailed above have one principle drawback. Each model is unique to the drum sound that it is trained upon. Work is currently being undertaken to train a model on an ensemble of sounds. It is hoped that this model will be able to regenerate the initial percussive attack for a variety of different samples.

References

[1] L. E. Kinsler, A. R. Frey, A. B. Coppens, and J .V. Sanders. *Fundamentals of Acoustics*. J. Wiley, 1982.

[2] C. L. Nikias and M. R. Raghuveer. Bispectrum estimation: A digital signal processing framework. *Proc. IEEE*, 75:869 — 891, July 1987.

[3] M. Schetzen. *The Volterra and Wiener Theories of Nonlinear Sytems*. Wiley, 1980.

[4] B. Picinbono and P. Duvaut. Geometrical properties of optimal volterra filters for signal detection. *IEEE Trans. Information Theory*, IT-36:1061 — 1068, September 1990.

[5] K. Nisbet, B. Mulgrew, and S. McLaughlin. Nonlinear prediction and the wiener process. In *Proc. S.P.I.E.*, 1991.

[6] M. Powell. Radial basis functions for multivariable interpolaton: A review. In J. Mason and M. Cox, editors, *Algorithms for Approximation*, pages 143 — 168. Oxford Science, 1987.

[7] S.Chen, C. Cowan, and P. Grant. Orthogonal least squares learning algorithm for radial basis function networks. *IEE Trans. Neural Networks*, 2:302 — 309, March 1991.