Wavelet Analysis

for

Onset Detection

Crawford Tait

A Thesis submitted to the Faculty of Science at the University of Glasgow, for the degree of Doctor of Philosophy in July 1997.

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 – 1346

Abstract

Many of the auditory perception processes which researchers have sought to automate can be decomposed into stages, the first of which involves segmentation of the input audio. In music this stage equates to locating note onsets, and advances in this task should therefore ease further analyses. There are also many direct applications of onset detection, including synchronisation of audio with other media and location of significant time points in graphical editing of audio.

It is for these reasons that this work focuses on the task of detecting onsets. An onset is considered as a particular type of change in the time-frequency representation of a sound. The modulus plane derived from a semitone-based harmonic wavelet analysis is first transformed, to account for the varying frequency sensitivity and mapping from amplitude to loudness observed in the human auditory system. Vectors are then derived from adjacent regions of the plane, and compared for change using Minkowski's distance measure. Peaks of distance correspond to significant changes, and commencing partials are sought at peak locations to identify onset peaks.

The process of testing the method is considered in some detail, and an experiment is derived in which a test piece is recorded using a wide range of timbres from a MIDI synthesiser. The piece includes a repeated note and a range of intervals, and legato and staccato styles are demonstrated. Separate test cases demonstrate results in the presence of reverberation, dynamic variation, low notes, short notes, vibrato, tremolo and drum sounds (with overlapping cymbals). The main body of tests was conducted using a large number of parameter settings and variations of the analysis method (including different loudness scales and exponents in the distance measure) to achieve optimal results, but the reduction to a single analysis method with one parameter was also considered. The use of a novel technique to compensate for slowly rising onsets is also investigated. Although the domain is restricted to monophonic musical audio, many of the test cases contain overlap and the method is shown to have some potential in the analysis of polyphonic examples.

The results of this experiment are assessed in the context of error tolerances, derived from consideration of a number of typical applications. It is shown that such assessment is not a straightforward matter and, for example, there may be interaction

between the type of timbre and the error tolerance which will apply in a specific application.

In summary, the thesis establishes that onset detection can be accomplished by monitoring a distance measure calculated from a harmonic wavelet analysis; and does this via the design and implementation of a comprehensive experiment.

Acknowledgements

I would like to thank my long-suffering supervisor Bill Findlay, who has consistently been the source of new ideas and constructive criticism, and who has provided many hours of stimulating discussion. His understanding has been central to my progress.
I am also grateful to my second supervisor John Patterson for participating in fruitful discussions at a time when they were required.
Amongst the many others with whom I have exchanged ideas, I wish to thank Stephen Arnold and Graham Hair from the Department of Music, Cordy Hall, and all of the enthusiastic researchers who I have met at numerous ICMCs.
Lastly, I wish to acknowledge the encouragement of my girlfriend Jakki Cunningham, whose patience has been apparently unlimited.
The work was conducted with the financial assistance of a University of Glasgow post-graduate scholarship, and funding for conference attendance from the Department of Computing Science.

Figures

Tables

Contents

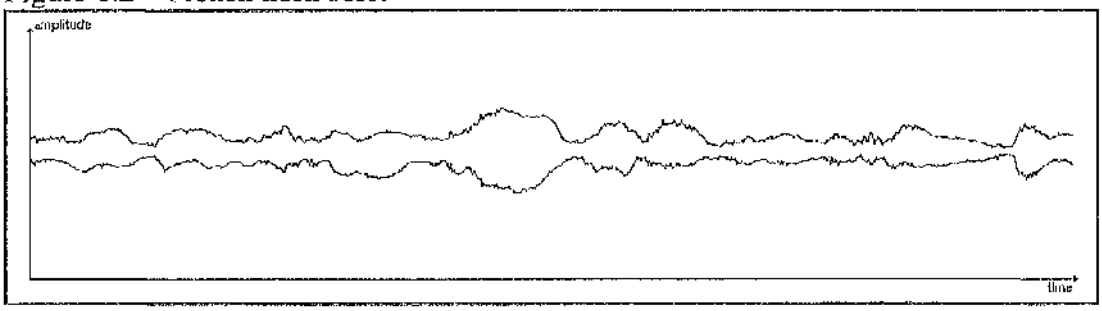
Chapter 1
Introduction

The introduction of computers into various music- and audio-related fields has resulted in their widespread use for the storage, editing, and transformation of digital audio. In addition, computers have long been utilised by certain composers to synthesise sequences of digital audio. The technology required by such applications is continually becoming less expensive, and all of these factors (as well as the introduction of a consumer digital audio format, the compact disc) have resulted in the existence of a large body of (musical) digital audio. However, that digital audio is largely impenetrable with much of the software in current use. The problem is that the dominant way of viewing digital audio is as a sequence of sample values plotted against time, without any further analysis. This representation highlights significant time points only in the most simple cases, and reveals little (if anything) about the audio content.

Figures 1.1 and 1.2 illustrate these points (see Appendix C for data format and graphing conventions). Figure 1.1 is typical of the simplest examples, and shows a repeated rim shot recorded from a drum machine. Location of the individual hits (by increase in amplitude) presents little difficulty in such cases. A more realistic example is shown in figure 1.2, which is sampled from a live performance of a french horn solo [Schubert 94]. It begins during a note which is followed by ten note onsets. Environmental reverberation causes notes to overlap, and this coupled with the slow nature of the attacks means that note onsets are not visible on the amplitude plot.

Figure 1.1 – Repeated rim-shots.

Figure 1.2 – French horn solo.



The above examples are monophonic (containing a single instrument playing one note at a time) and yet, even in such simple cases, the amplitude envelope rarely helps in locating individual notes. Also, the recording environment and style of playing can mean that even a restriction to the monophonic case allows a wide range of possibilities. However, a musically trained human listener would be able to ascertain pitches (where relevant), and overcome expressive timing in the performance, to derive a score (similar to that originally written by the composer). Automation of this task in particular has been the ultimate goal of much previous work in the area, and in the following sections a number of such potential applications are described.

1.1 Automatic Transcription

The starting point for such analyses has most often been pitch detection (as will be discussed in more detail in the next chapter). The idea is that if one can reliably detect pitch, then the emergence of a new pitch indicates a note onset. Whilst this may often be true, the author believes that overlapping notes (and interfering sounds in general) present problems for such an approach, and also that this view of the task limits its extensibility and applicability to other kinds of sound. The following decomposition of the problem, utilising onset detection as a first step, is therefore proposed as an alternative.

- Identify the time of each note onset.
- Find the pitch of each note (guided by the note locations from stage one) and infer a key signature.
- Use this information, perhaps in conjunction with higher level musical knowledge, to derive the rhythmic structure and time signature of the original score.

This is over simplified, since no mention has been made of notated ornaments or dynamics. However, the method is more flexible and (for example) interpretation of the rhythmic structure can be based solely on inter-onset intervals, or can also take into account the detected pitches (previous approaches have utilised both methods). Also, if the onset detection method is versatile enough, the application of this kind of decomposition to non-pitched or non-monophonic examples could be envisaged. Finally, it disentangles the main problems so that, whilst they may interact, their solutions can be considered separately.

This approach (and the critical first stage of onset detection) have not been extensively studied in the past and, whilst these are perhaps the main reasons for choosing to focus on onset detection, it has other applications of its own.

1.2 Digital Editing

In the days when all audio recordings were stored on analog tape, editing of those tapes could only be accomplished with the aid of a razor blade. The tape would be moved back and forward over the play head of a tape machine until the correct point was established, and then the tape was cut. Composite reels could be constructed by splicing sections of tape together.

In many ways, the predominant editing techniques employed with digital audio have progressed little (if any) since then. Editing software will generally display the audio as a strip, or as a waveform (such as those above), that the user may magnify to examine in more detail. Edit points can be found only by audition or by visual inspection of the audio waveform, and 'cut and paste' are still the major editing tools.

In musical examples, a transcription would include all the necessary information to specify edit points in terms of high level musical structure. However, in this area especially, one can imagine many non-musical applications (the editing of speech, sound effects and other sampled sounds). Also, whilst onsets may be the most likely edit points, the location of other significant points could be useful. For example, the location of note offsets is generally not considered as part of the transcription problem, although they may be of interest in some cases.

In summary, the consideration of this application area in particular prompts the generalisation of *onset detection* to the process of finding *points of interest* in the audio and then classifying those as onsets, offsets or some other definable event. Automating the generation of such information would greatly assist many editing tasks.

1.3 MIDI With Digital Audio

The Musical Instrument Digital Interface [IMA 83] is a communications protocol designed to allow electronic music equipment from various manufacturers to exchange control information. Computers are often used as *sequencers*, for storage and editing of streams of MIDI performance data. However, the increased power of personal computers has led to the emergence of MIDI sequencing software that also allows the recording of digital audio tracks onto the computer's hard disc (via special analog to digital conversion hardware).

The problem remaining is that, even though the MIDI and audio tracks may both represent the same piece of music, there is no way for the sequencer to take this into account. For example, it is usually possible to quantise the data in MIDI tracks to a specified time grid. Given the onset locations in a monophonic audio track, one could imagine using these as a time grid for MIDI/audio synchronisation, attaching notes to their closest neighbour in a MIDI track, or even subsequently altering timing within the audio itself. Such facilities are beginning to appear in some commercially available software, but the proprietary nature of the methods makes it difficult to evaluate them in the context of research.

This area provides an interesting special case — there may be times when additional information is available about the audio to be analysed. For example, it might be easier to locate musically significant points in a recording of a piece if a representation of the score of that piece is also available. If this could be accomplished in real time, one potential application would be the construction of MIDI accompaniment programs able to follow a performance and play along. Having said this, the current work focuses on situations where such information is not available (since a solution to the more general problem of locating onsets without prior knowledge would be considerably more useful).

1.4 Multimedia Applications

As well as attempting to reconcile different representations of a piece of music, there are situations where it would be desirable to synchronise graphics or some action with a piece of music. Many applications in the fields of animation, music video and performance can be imagined (for example, given onset locations, it would be considerably easier to synchronise the actions of animated characters to accompanying music). This area could exploit some of the generality introduced in the previous section, since it is likely that events aside from note onsets would be used as synchronisation points.

1.5 Non Musical Applications

Whilst this work ultimately focuses on detecting note onsets in music, similar principles may apply to (for example) locating sound effects in audio soundtracks, or detecting events in other kinds of time series. The possibility of wider application arises from the removal of musical considerations (such as pitch) from the onset detection phase mentioned earlier, but these issues will remain largely unexplored here.

1.6 In Summary...

The observations of various applications presented above have motivated the investigation of an onset detection method, with the following constraints.

- The input will be monophonic, but in order to deal with realistic examples the method should tolerate overlap and a degree of background sound (this should also be a step towards the considerably more difficult case of polyphonic input, on which relatively little progress has been made to date).

- The technique should not depend on musical attributes of the input (to allow its application to other types of example, such as timbres without definite pitch).

- Prior knowledge of any kind should not be required, to retain as much flexibility as possible.

The main issue not yet discussed is how such a method should be tested. A range of examples and some measure of performance is required. The issue of accuracy is not as straightforward as it may at first seem. Given an example with slow onsets, would a range of listeners agree on specific onset times? In addition, the accuracy requirements of the applications described vary: in automatic transcription, the higher level processing must overcome timing deviations from the strict score deliberately introduced by the performer (so that lower level onset detection inaccuracies may be acceptable); however, in some editing applications even small errors could be important and a greater degree of accuracy may be required.

These issues will be dealt with in more detail later in the thesis, the remainder of which is structured as follows. Chapter 2 places the work in context by giving a review of related work in various fields and discussing the different approaches which have been taken. Chapters 3 and 4 describe the method which has been developed:

first, why a particular recently developed time-frequency decomposition has been adopted; second, a description of the technique developed for highlighting change in the audio, and how the output of this is analysed to locate onsets. Chapters 5, 6 and 7 deal with an area which it is felt has been under-reported in previous work: the process of testing and evaluating musical analyses, and onset detection in particular. The design of an experiment and the results obtained are described. Finally, the impact of the work is assessed – how the technique would be applied in the application areas described, the importance of comprehensive evaluation and the ways in which further work could build upon what has been achieved.

Chapter 2
Background

Onset detection (or, more generally, signal segmentation based on some higher level criteria) has been investigated in a number of different contexts. For example, whilst onset detection is not usually the first stage in automatic transcription systems, it is often utilised as a pre-processing phase in speech recognition. In addition, there has been parallel development of techniques in those fields which have subsequently been applied to the problem. In order to best describe the context of the current work, this chapter is split into three main sections: a description of work in related areas of research, a discussion of the techniques used and a note on the methods of testing which are currently in use. The chapter concludes with an explanation of how the current work fits into this framework.

2.1 Related Areas of Research

In this section, the relevance (and current state) of a number of related research areas is addressed.

2.1.1 Automatic Transcription

An overview of this process has already been presented, and much of the fundamental concepts can be traced back to the work described in [Piszczalski & Galler 78]. The problem is divided into the low level analysis of the audio signal (to identify pitched notes) and a higher level analysis of those events to derive the original score. The low level stage is based entirely on a time-frequency analysis of the monophonic input audio (the Short Time Fourier Transform, described later). Whilst the subtleties of pitch perception are noted, pitch detection is central to the analysis (as in most automatic transcription work) and the emergence of an audible pitch, or a sudden change in pitch, marks a note onset. Results are achieved with this method, but it is noted that 'not all transitions are found so easily, and further work remains to be done in this area'. This early paper also made some other important observations that have influenced subsequent work.

- High resolution in both time and frequency cannot be obtained in the same analysis, so that it is difficult to ascertain both the pitch and the timing of a note with a high degree of accuracy. This was tackled by analysing two Fourier transforms: one at a high frequency resolution, and one at a high time resolution. The time resolution of the system still seems rather low (31 msecs windows are used), however this is an application where higher level processing (in this case, user input about the tempo) can help improve results. The ideas of 'multiresolution analysis' and separation of low and high level processing have subsequently been much developed.

- A better model of how the ear processes sound may be useful. For example, pitch is related to frequency (as is loudness to amplitude) on a logarithmic scale, and we are not equally sensitive to all frequencies.

- Access to example waveforms of the individual instruments making up the sound to be analysed may help in locating them in the input.

- The importance of testing, and the fact that 'no system has yet processed a large and varied set of musical sound inputs' are noted.

The influence of such observations will become obvious in the sections which follow (for example, the next important project took place in the early 1980s and is discussed in the methods section). Recent research in automatic transcription as a single problem seems rare – the task has been decomposed, in various ways, into more tractable sub-problems. The techniques utilised in the most relevant of these will be discussed later (the higher level analyses, for example, will not be covered, but are surveyed in [Desain & Honing 94a]).

2.1.2 Auditory Scene Analysis

The problem being addressed in this work is that of disentangling a mixture of two or more sounds. Due to the fact that frequency components of sounds are likely to overlap and interfere, some knowledge of the individual sources is often employed.

Whilst it is likely that any solution to this problem would also include information on the onset times of notes, source separation is perhaps the most difficult problem in the area and is far from solved.

An early investigation of the problem is reported in [Moorer 75], where the goal was to derive the score from a recording of a duet. A number of restrictions are placed on the input, which (along with the key signature) act as prior knowledge to aid in the identification of notes. For example, overlapping notes must not have coinciding frequency components and only notes with a harmonic structure are allowed (in such notes, if the fundamental frequency is F, the Nth frequency component above the fundamental has frequency $(N+1)\times F$). A periodicity detector applied to the amplitude signal guides the use of bandpass filters in the next stage. The output of these filters is examined for partials, and segmented according to a simple thresholding rule. Knowledge of the harmonic structure of the notes can then be used to group partials. When applied to two examples meeting the imposed restrictions, the system performs satisfactorily. It is possible that such an approach could be extended, but the concepts of harmonicity and pitch are of crucial importance in locating notes (and it has been explained why the current work seeks to avoid such assumptions). However, the beginnings of the problem decomposition now prevalent can again be seen.

The process of detecting and grouping partials is common in such work. Whilst Moorer sought to form groups corresponding to notes (with harmonic structure), others have attempted to use more general grouping principles. Such principles are generally derived from psychoacoustic experiments and include (for example) common modulation, onset and offset synchrony. The field is described in detail in [Bregman 94], which explains how two events are more likely to be grouped if their onset times are simultaneous, whereas separation is more likely when a new event occurs during another that is sustained. This hints at the importance of onsets (discussed in more detail in chapter 4), and the way in which music is experienced as a series of discrete units, whose boundaries are indicated by changes in timbre, pitch or loudness is also described. The most important observation for onset detection is that the synchronous commencement of a number of frequency components is a good indication that those components are a part of the same sonic event.

[Cooke 93] describes the application of auditory grouping principles to the output of an auditory model (these will be discussed later), in attempt to actually separate speech from interfering noise. The grouping of partials by common onset time is

perhaps most interesting in the current context, but is not utilised by Cooke (separation is attempted, but onset times are not generated).

Similar work which does utilise common onset, and is aimed at musical input, is described in [Mellinger 91]. Onsets are detected in a range of frequency bands by smoothing the band limited signals, and then cross correlating them with an 'onset kernel'. This is a function designed so that the correlation is high when a period of no energy in a channel is followed by the rise of a signal. At times when a number of partials commence, the sum of the cross correlations is high, and a note onset is signalled. The ideal kernel would correspond in shape to the attack of each partial in each note, so that different instruments would be best detected by a different set of kernel functions. However, whilst a few different time lengths are used, varying the shape is not considered in depth and it is unclear (from the small set of single note examples) how well the method performs in practice. Such methods may be extensible however, and the technique for identifying exponentially rising transients in [Mani & Nawab 95] is based on a similar technique.

The utilisation of prior knowledge (or assumptions about the input) involved in much of the work presented in this section is also evident elsewhere. Automatic accompaniment uses knowledge of the score in tracking other performers, and an example of beat detection using knowledge of particular types of drum will be discussed later. The next section deals with speech recognition, which has received considerably more attention than any of the musical problems mentioned, and almost always utilises knowledge about how speech is produced, as well as higher level grammatical information.

2.1.3 Speech Recognition

There are interesting parallels between the segmentation of continuous speech, and the segmentation of musical signals. Speech can be considered as consisting of phrases (with some high level language structure), each composed of words constructed from individual phonemes. The structuring of music into phrases of notes with some internal structure is obviously analogous. This is perhaps what leads to the adoption of a similar problem decomposition – the segmentation of speech is often attempted prior to recognition, and a similar approach to musical signals has been pursued (and will be promoted herein).

However, the low level techniques applied to speech signals are difficult to generalise, since they are usually based on some model of the signal which is well suited to its method of production (that is, the vocal tract). Changes in this model are then tracked

to segment the signal. An analogous musical segmentation would attempt to model a particular instrument and, therefore, be highly specialised. A number of such approaches to speech segmentation are described in [Andre-Obrecht 88]. In addition, training of the system is generally undertaken prior to any attempt at recognition and the current work seeks to avoid such prior knowledge.

Hidden Markov Models are currently widely used in the segmentation and recognition of speech (an introduction is given in [Rabiner & Juang 86]). An explanation of why such methods are unsuitable for analysis of polyphonic music is given in [Shuttleworth & Wilson 93]. Whilst it might be possible to train a model to recognise notes played on a variety of instruments in the monophonic case, this does not appear to have been investigated (and methods involving such training will not be considered here).

2.1.4 Change Detection

The task of speech segmentation is often included in the more general field of statistical change detection (surveyed in [Basseville 88]). What is perhaps more worthy of note here is that the analysis of musical signals is apparently never included. The most likely explanation for this is that whilst the majority of musical signals which will be considered have some features in common (and this must be exploited to develop a useful technique), they arise from a variety of different physical systems (that is, musical instruments). As a result, the changes to be detected can be sufficiently different that they will defy detection by the kind of system modelling which is generally undertaken in this field.

Whilst the methods may not be generally applicable to musical signals, a later chapter will discuss how observations of the techniques used in this field have influenced the current work.

2.2 Development of Techniques

A parallel development of methods can sometimes be observed across the various areas of research previously described. In this section, the choice of methods in the current work is placed in context by describing the evolution of the signal processing techniques involved.

2.2.1 Amplitude-Based

Some attempts at locating onsets have been made using only the time varying amplitude of the audio signal. In [Schloss 85], for example, a low-pass filtered version

of the audio to be analysed is traversed, and the gradient over some interval is repeatedly calculated. Segments in which a significant increase in gradient occurs are then marked as onsets. The algorithm is parameterised on gradient threshold, time increment and minimum note length, and applied to percussion instruments. Such restrictions are inherent in amplitude-based methods, since only percussive instruments produce detectable amplitude peaks. The method was also utilised in [Chafe et al 85], where a detection rate of better than 95% is reported for the piano.

Recent work has rarely utilised the amplitude signal alone (for reasons which will become apparent), and it is difficult to see how such methods could be extended. An 'onset gate' (used to trigger further processing) is described in [Arcelo 95]. Simple amplitude based methods might be used in this application because speed of processing is of paramount importance for real-time operation. The method described first passes the signal through a high-pass filter, because low frequency components can cause false alarms in the next stage ([Foster et al 82] also used this method, because higher harmonics are shorter lived in time and less likely to overlap). An Automatic Gain Control is then used to force the signal to some average amplitude, thus avoiding the problem of loudness affecting the detector's results. Short- and long-term averages are then computed and when the difference exceeds some threshold, an onset is signalled. This method can be related to the previous, since the difference could be viewed as a gradient measure over some interval. No musical examples are given.

2.2.2 Fourier Analyses

The problems inherent in amplitude-based methods are discussed further in [Foster et al 82] which describes a number of amplitude analysis methods and demonstrates that, while they may be complementary to other techniques, analysis of the amplitude envelope alone is not generally sufficient.

The paper is primarily concerned with automatic transcription, so that analysis of the time varying frequency spectrum (to ascertain pitch) must also be undertaken. It then becomes apparent that this aids segmentation, since a change in pitch generally marks a new note. The paper therefore concludes on a method that tracks pitch in reverse, and assesses each new hypothesis for its fit to the currently established pitch. A threshold is set, below which the new audio ceases to fit the current pitch, causing an onset to be marked. It is unclear how well this method copes with repeated notes, and the preceding chapter has explained why it is desirable not to rely on pitch perception.

However, the separation of low-level from higher-level analyses and the initial attempt to factor out onset detection are instructive.

In seeking to overcome a reliance on detecting pitch, while taking into account evolution in the frequency domain, one possibility is to look for marked changes in individual frequency bands and then coalesce this information to highlight note onsets. Such an approach is utilised in [Pearson & Wilson 90] which also uses a multiresolution time-frequency decomposition. This Multiresolution Fourier Transform (MFT) overcomes some of the problems associated with earlier forms of Fourier analysis (a more detailed discussion is presented later). Onsets are detected on a per frequency band basis and must be detected consistently across a range of resolutions. The level with highest time resolution then locates the onset precisely.

Work with the MFT is presented in more detail in [Pearson 91] and [Scott & Wilson 92]. Pearson's thesis avoids restrictions to strictly monophonic input (noting that reverberation introduces polyphony), and also avoids prior knowledge of musical structure or the instruments in the input. However, notes are modelled as a set of harmonically related partials so that the methods do not apply to percussion instruments or inharmonic sounds. In addition, the difficulty of deriving a model to fit even different notes played on the same instrument is noted. The system is successfully applied to a two note piano example. Two examples of trios show how, whilst partials can be extracted from polyphonic input, the phase information (on which the technique is based) is easily corrupted in these cases.

Methods such as this may encounter problems, since defining an onset in a frequency band is still not straightforward (the possibilities for variation of Mellinger's onset kernel illustrate this). Moreover, it is difficult to set automatically the number of bands in which onsets must be detected before signalling an onset in the complete sound, because instruments present a wide range of harmonic complexity, and higher harmonics may be hard to identify in softer notes. In addition, notes may exhibit vibrato (frequency modulation) or glissando (frequency glide), meaning that they will not be confined to the same set of frequency bands for their duration (and may cause repeated false onsets as they move). Attempts to track such frequency movement are made in [Cooke 91] (but onset times are not generated), whilst [Shahwan 94] investigated extending Mellinger's onset detection method so that movement did not cause repeated onsets to be detected. This involved applying an 'inhibitory kernel' to adjacent channels, so that the detection of an onset in a given channel is dependent not only on an increase in energy in that channel, but also on the non-existence of energy in adjacent channels. This means that tones with frequency movement only trigger an

onset response when they begin. Whilst the system required training, and a similar method could therefore not be employed herein, the effect of frequency movement on any onset detection method must be borne in mind.

Attempts have been made to utilise frequency band onsets without a note model, or use of unreliable phase information. In this context, [Shuttleworth & Wilson 93] describes the summing of a gradient measure across frequency bands in the MFT to give a measure of 'total onset energy'. Dynamic range compression was then applied in an attempt to make the onset measure independent of note loudness. An onset is detected when this signal crosses an adaptive threshold. The paper's two examples illustrate the method, but some spurious peaks are evident and actual detection of peaks (and subsequent location of onsets) is not undertaken. In addition, whilst relying less on low level note modelling, the paper promotes the integration of higher level musical knowledge into the analysis (the current work aims to be free of all such prior knowledge).

2.2.3 Wavelet Analyses

The wavelet transform (discussed in detail later) has recently emerged as a useful tool in audio analysis, because it produces a logarithmic division of the frequency scale (related to our perception of pitch), and has time resolution which varies with frequency in each band, giving potentially improved time localisation over traditional Fourier analysis. Early investigations demonstrated lines of constant phase in the plane of complex wavelet coefficients, leading to points in time at which discontinuities (for example, artificially generated impulses) occurred in the signal [Grossmann et al 87].

While such lines are not so evident in the context of real signals, this observation has been used as the basis for an onset detection method [Solbach et al 95]. The phase plane generated from a wavelet-based auditory model is analysed, and a function calculated that is designed to exhibit peaks at lines of constant phase. An example is given of a piano piece in which the resulting graph contains peaks at the note onsets. However there is still work to be done on the automatic extraction of peaks, and the analysis of a wider variety of examples. Also, as mentioned in the paper, the phase patterns of two sounds occurring at different times can interfere. It seems that further investigation of such issues is required: for example, most live recordings include some degree of reverberation and the resulting interference in the phase pattern would almost certainly cause problems.

2.2.4 Auditory Representations

In [Smith 96], an auditory model is used to produce a set of signals (each corresponding to a frequency band) that are related to those in the auditory nerve. These signals are then analysed with a neural network to highlight onsets and offsets in each band (based also on information in adjacent bands), and segmentation is based on the number of simultaneous onsets or offsets required. An example is given in which the number of onsets required can be set so as to produce a successful segmentation of a short flute piece.

A similar method of generating onset maps is illustrated in [Brown & Cooke 94]. Onsets are highlighted on a per frequency band basis, using a mathematical model of an onset cell observed in physiological studies of the cat. Some of the problems inherent in such per frequency band methods have already been noted. Like Mellinger's onset kernel, the cell model can be tuned. In the paper it is tuned to respond only to rapid onsets, and how other kinds of onset may be detected is not discussed (onset detection is not the primary goal).

When considering auditory representations, it should be remembered that the system to which they act as input (that is, the brain) is hardly understood at all. It is often noted that such factors as experience, association and learning enable humans to detect individual notes or instruments in mixtures, and modelling of such processes is in its infancy. As such, the current work should be viewed in the context of signal processing (rather than artificial perception) and only the most salient features of the auditory system's time frequency representation will be considered.

2.2.5 Model-Based

It was explained in the section on speech recognition why the author believes that model-based segmentation is not flexible enough to tackle a wide range of sound sources. However, in this final section on methods, a number of techniques involving musical or source-specific knowledge are described.

[Foster et al 82] describes an application of autoregressive segmentation, in which an autoregressive model is fitted to the audio data on each side of a time point T. Each model is then run on the data from which the other was derived, and the energy in the residual signals compared. The residual signals give a measure of the models' fit to the data, and the calculated differences effectively show how much the model changed at time T. The hypothesis is that onsets are marked by peaks in the difference between the original residual signal and that computed with the other model. The method is shown to be successful in cases where amplitude analysis alone would fail, however it

is not sensitive to amplitude change and as a result cannot detect repeated notes. In addition, it is stated that the instrument under analysis must be well modelled as an autoregressive process, although neither further explanation of this nor comprehensive examples are given. The paper concludes on a method using pitch detection, which could be viewed as another kind of constraining model (changes in pitch cannot be tracked if the sound is unpitched).

In [Goto & Muraoka 95], frequency band onsets are detected (as in some of the work described previously), but knowledge of the sound source is then employed in an attempt to identify the contributions of bass and snare drums to the sound. The drums are modelled by their likely appearance in the frequency domain. This works well on a restricted corpus of examples, but it is difficult to see how the analysis could be generalised, since most instruments cannot be characterised so simply.

The method was therefore extended by resorting to higher level musical knowledge [Goto & Muraoka 96]. In this version, the time-frequency plane is analysed for chord changes, and beat tracking is then based on the hypotheses that such events are likely to occur at beat positions (and also on certain beats).

2.3 Methods of Testing

At this stage, it is hoped that the reader has an idea of the various techniques which have been applied to the analysis of music, and also to other signals such as speech. An aspect of this work which has not been discussed is the process of testing. It is immediately obvious, when researching the field of signal analysis (and especially audio analysis), that the area of speech processing is by far the most highly developed. This is perhaps because there are many obvious areas of commercial exploitation for a successful speech recognition system, and an important side effect is that the testing procedures in use are much more sophisticated than those reported in the field of musical analysis.

As an example, the paper on speech segmentation mentioned earlier [Andre-Obrecht 88] included tests on five sets of ten sentences by one speaker, 20 numbers spoken by four male and six female speakers, and fifty telephone numbers spoken by a single speaker. Such demonstration of a method's performance is *de rigueur* in the speech processing community, and [Cole & Hirschman et al 92] notes that "the availability of common corpora of speech and text is a critical resource that has been partly responsible for the significant gains made in speech and language processing in recent years". The report also comments that making such databases widely available is a

massive task (this is obvious when one considers the richness of language and the potential influence of different speakers, dialects, languages and so on).

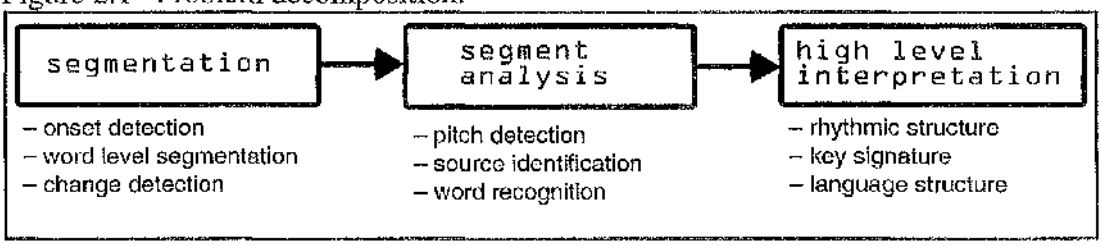
The testing of musical systems is just as important, and will be discussed in detail in its own chapter. However, none of the musical work reported here includes more than a few illustrative phrases, and there has apparently been no attempt to investigate the kind of testing which should be performed (or what would constitute a realistic set of benchmark tests).

2.4 Conclusions

The work which has been presented in this chapter spans a number of different areas of research, but a common decomposition of the various signal analysis problems can be observed. Figure 2.1 illustrates this as a simple three stage process. In the first instance, the data is segmented into sensible units (for example notes or words) based on some common characteristics of the boundaries in the input data. Next, these units are analysed (or matched against some known criteria) for some higher level properties (for example pitch or word content). Lastly, higher level knowledge (for example of music or language) is used to interpret the information from the second stage (for example to derive the score of a piece of music or the structure of an utterance). Of course, there may also be feedback paths, so that the high level interpretation may suggest a missed note or an incorrect word identification.

This decomposition is also flexible enough that many other kinds of analyses can be approached in the same way. For example, [Gustafson et al 78] takes a similar approach to the rhythmic analysis of heartbeats (in this case stage two is very simple, since the timing information from stage one is most important).

Figure 2.1 - Problem decomposition.



That said, the main area of concern in the current work is the analysis of musical signals. It has been explained how early work on automatic transcription and auditory scene analysis attempted to solve the whole problem, whereas there has been a trend towards more detailed investigations of the individual stages in recent years. Pitch

detection and the higher level analysis of events in particular have received much attention, whilst onset detection has rarely been considered in isolation. The benefits of detecting onsets without prior knowledge, or reference to higher level concepts such as pitch, have been explained. It has also been observed that the onset portion of a sound is of critical importance in its identification by a listener [Grey 75], so that the detection of onsets (in as generalised a way as possible) should also be an important stage in the automated recognition of sounds. It is for these reasons that this thesis focuses on the process of onset detection.

So that the analysis remains within the constraints which have been specified above, it is based on the definition of an onset as a set of one or more frequency components (usually referred to as partials) commencing within a short space of time. This allows for sounds of varying complexity, as well as unpitched sounds. The time within which the partials will commence has been studied previously, and this will be discussed further in chapter 4.

In addition, there are no accepted benchmark tests for the evaluation of new musical analysis systems. The development of a set of test cases is therefore also described, in the hope that the proper evaluation of new techniques may promote the kind of progress which has taken place in the speech processing world.

Chapter 3
Time-Frequency Representations

Although it is difficult to generalise, most music consists of a series of events taking place over time, and many of those events will be notes with an associated pitch. Pitch has been described as the quality of sounds which allows a listener to order them on a scale from low to high. It is a perceptual phenomenon, and pitch judgements are therefore subjective. However, the perceived pitch of a periodic waveform can be related to its fundamental frequency. In addition, the timbre of a sound arises from the pattern of its other partials (see, for example [Howard & Angus 96] for discussions of pitch and timbre), and these observations motivate the use of a time-frequency decomposition as a first step in any audio analysis.

Conveniently, a variety of methods exist to find out how the energy of a signal is spread over its constituent frequencies at different times. However, such analyses are always limited, in that high resolution in both frequency and time cannot be simultaneously achieved. This can be intuitively understood, by considering that to detect a frequency, f, a signal must be observed for at least one period — that is for at least $1/f$ seconds. So the accuracy with which a given frequency component can be located in time decreases for lower frequencies. However, different methods deal with this trade-off differently and this chapter discusses the two most common: Fourier and wavelet analysis, as well as a variation of the latter developed specifically for musical input. In addition, it is known that the ear performs a frequency analysis, and many researchers have attempted to model its characteristics. A section is therefore also devoted to auditory modelling, before the chapter concludes on the method which has been adopted in the current work.

3.1 Fourier Methods

The Fourier transform is the oldest and most widely used tool for time-frequency decomposition (partially due to the development of an efficient algorithm for its calculation, the FFT). Only the most intuitive introduction is given here, to enable comparison with other methods, and a more complete treatment (in the context of musical analysis) can be found in [Jaffe 87a & b].

The basis functions of Fourier analysis are sinusoids. An input signal is decomposed into contributions from a set of sinusoids, evenly spaced in frequency up to the highest frequency component in the signal. Each sinusoid is associated with a complex coefficient, from which can be extracted amplitude and phase values, and summing the implied sinusoids gives the original signal. The same amount of data is generated as is given as input, so that analysing a longer audio segment gives more coefficients and hence a better frequency resolution.

The coefficients returned are essentially averaged over the duration of the input. To study time varying characteristics, the input is split into short time segments (the size of which dictates the time, and also the frequency, resolution). Each of these segments is then analysed as above, and the whole procedure is known as the *Short Time Fourier Transform*, or STFT.

This is an oversimplified description, but it indicates the important drawbacks of the STFT.

- The frequency scale is divided linearly, whereas a scheme related to pitch would be more appropriate — whilst pitch perception is by no means a simple matter, a logarithmic division of the frequency scale is central (see [Plomp 76] for the perceptual aspects or [Meddis & Hewitt 91] for an attempt at modelling).

- The basis sinusoids are well localised in frequency, but not in time (they exist for the whole time of the analysis).

Solutions to these problems within the confines of the Fourier transform have been suggested. For example, the logarithmic frequency analysis of [McGee & Merkley 91] gives a closer match to pitch on the frequency scale. Also, the Multiresolution Fourier Transform in [Wilson et al 92] gives varying degrees of time and frequency resolution (although it is computationally intensive and generates a massive amount of data from which that of interest must be isolated).

However, a completely different transform is available – the wavelet transform – which solves these problems and is therefore inherently better suited to the applications we have discussed (although it has been little used in this context).

3.2 Wavelets

Again, only an intuitive introduction to wavelet concepts is given in this section, to highlight the differences when compared with other types of analysis. Further introductory material on wavelets in the context of signal analysis may be found in

[Rioul & Vetterli 91] and [Graps 95] (also [Kronland-Martinet et al 87], [Kronland-Martinet 88] and [De Poli et al 91] deal with music in particular); more comprehensive treatments are available in [Kaiser 94] and [Combes et al 90].

The wavelet transform is based on a single function, often called the 'mother wavelet' or 'analysing wavelet', which satisfies certain well-defined admissibility conditions (these imply that the function has finite energy, and zero mean value). Also, the analysing wavelet is localised in both the frequency and time domains. Different shapes of analysing wavelet give rise to what are known as wavelet families. An input signal is decomposed into contributions from time-shifted and scaled versions of the analysing wavelet — thus this kind of analysis is often known as 'time-scale' rather than 'time-frequency'. However, the concept of scale can be related to frequency: larger scale wavelets exist over longer periods of time and are related to lower frequencies; whereas smaller scale wavelets exist for shorter time periods, corresponding to higher frequencies. In this respect, the analysing wavelet can be thought of as a bandpass filter.

These points are illustrated in Figures 3.1 and 3.2 (where the x and y axes represent time and amplitude, respectively), generated by the author from the Daubechies family of wavelets [Daubechies 88]. Figure 3.1 shows the analysing wavelet at a small scale, shifted to three different points in time. Figure 3.2 shows the same wavelet at a larger scale, again shifted to three different points in time.

Thus, the wavelet transform performs a multiresolution analysis. At large scales better frequency resolution, but worse time resolution is achieved — these wavelets correspond to bandpass filters with low centre frequency and narrow bandwidth, and represent features at larger time scales. Smaller levels of scale give better time resolution, but worse frequency resolution — these wavelets correspond to bandpass filters with high centre frequency and wide bandwidth, and represent transient features existing for only a short period of time.

Figure 3.1 – Small scale DAUB4 wavelets.

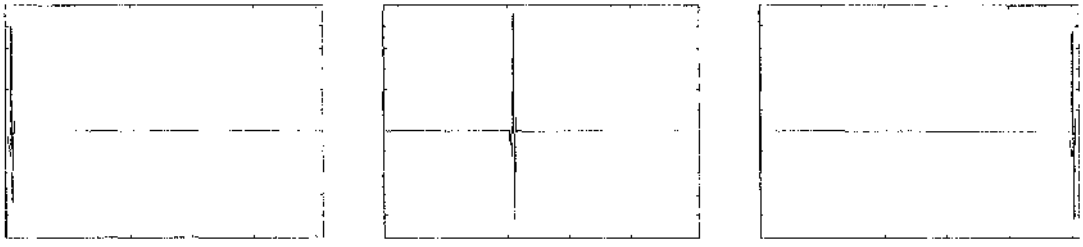
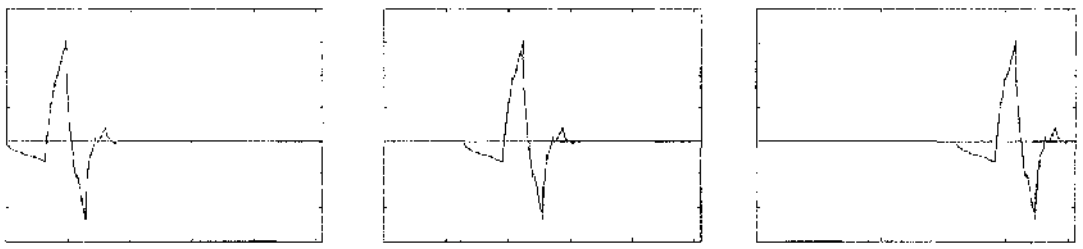


Figure 3.2 – Large scale DAUB4 wavelets.



Typically, the scale of the analysis decreases by a factor of two, so that successive levels are associated with twice as many wavelets, each existing for half the time (and thus corresponding to double the frequency) of those in the previous level. This doubling in frequency results in a so-called constant Q analysis (the ratio of the centre frequency to the bandwidth in each level is constant). In addition, it produces a natural division of the frequency range into octaves, and in that sense the wavelet transform is well suited to analysis of musical signals.

Wavelet analysis generates similar information to the STFT about evolving frequency content, since each wavelet is associated with a complex coefficient from which phase and modulus values may be calculated. In addition, the original signal can be reconstructed from these coefficients (and the analysing wavelet function). However, there are obvious differences, the most important of which are highlighted below.

- Wavelets are localised in time, whereas the sinusoids of Fourier analysis are not. This makes wavelet analysis particularly suitable for detecting sharp changes in signals. The time window chosen in the STFT restricts the type of features that will be highlighted in the analysis.

- The logarithmic division of the frequency scale in wavelet analysis is related to that carried out in the auditory system, and also to the musical scale. In the STFT, the time window dictates a linear frequency scale.

Further comparisons of wavelet analysis with the STFT can be found in [Rioul & Vetterli 91], [Graps 95] and [Strang 93].

The properties of wavelet analysis make it inherently suitable for analysis of musical audio in many ways. Also, the wavelet transforms which will be discussed can all be efficiently implemented. For these reasons, wavelets currently provide interesting

possibilities for research, especially in the musical context (since it has been relatively under-studied). The main drawback is the octave decomposition of many wavelet transforms, which is obviously too coarse for many musical analyses. However, the choice of analysing wavelet is restricted only by the admissibility conditions, which raises the possibility of manipulating its properties for particular applications. This has given rise to a variety of wavelet families, and the next section discusses wavelets for musical analysis.

3.3 Musical Wavelets

One example of musical analysis is described in [Kronland-Martinet 88], where the analysing wavelet has two distinct peaks in its spectrum separated by an octave. Wavelet analysis of an audio segment consisting of two melodic lines played simultaneously then highlights points where two notes separated by an octave were played. Whilst this is an interesting example, a more general transform suited to highlighting various features in musical input is desirable.

Such observations have lead to the recent development of a semitone-based wavelet transform [Newland 93, 94A, 94B & 95] which effectively divides the frequency scale into bands, each encompassing a single note on the (equal tempered) musical scale. Although (due to the complexities of pitch perception and the subtleties of real musical instruments) it might be argued that there is no uniquely favourable division of the frequency scale into semitones, the theoretical development in the above papers is perhaps as close as can be hoped for.

The transform is also convenient in that an efficient implementation using the Fast Fourier Transform exists (but the result should not be confused with the Fourier analyses previously described). The Fourier transform of the input audio is first calculated, then the resulting coefficients are partitioned into semitone bands (subject to rounding errors) and an inverse Fourier Transform is calculated for each band. This results in a series of time varying complex wavelet coefficients for each semitone (a more detailed description of the algorithm and its implementation can be found in appendix B). Investigation of the phase plane has not been attempted in any depth (for the reasons discussed previously), and further analyses utilise only the modulus values calculated from the complex coefficients.

Figures 3.3 and 3.4 illustrate the type of wavelet analysis described above, applied to the audio examples of figures 1.1 and 1.2 (see appendix C for a note on the graphing technique used). The scale axis is divided into semitones as described (with lower frequencies towards the time axis), and the modulus values obtained from the complex

coefficients are mapped onto dot densities. The whole time-frequency plane is shown, although subsequent figures will generally focus on the range of scales that are of interest.

Figure 3.3 – Wavelet analysis of repeated rim-shots.

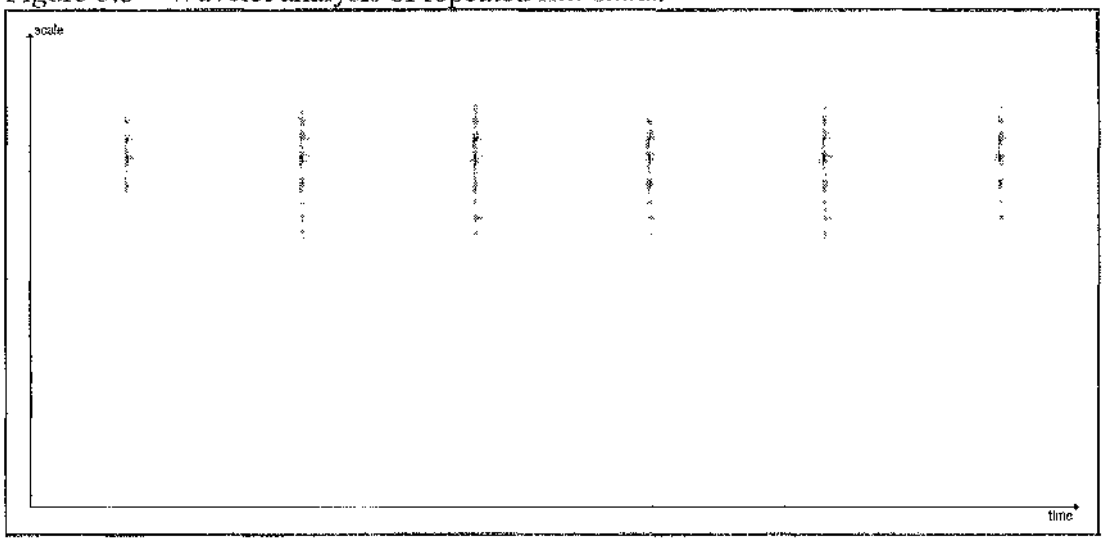


Figure 3.4 – Wavelet analysis of french horn solo.

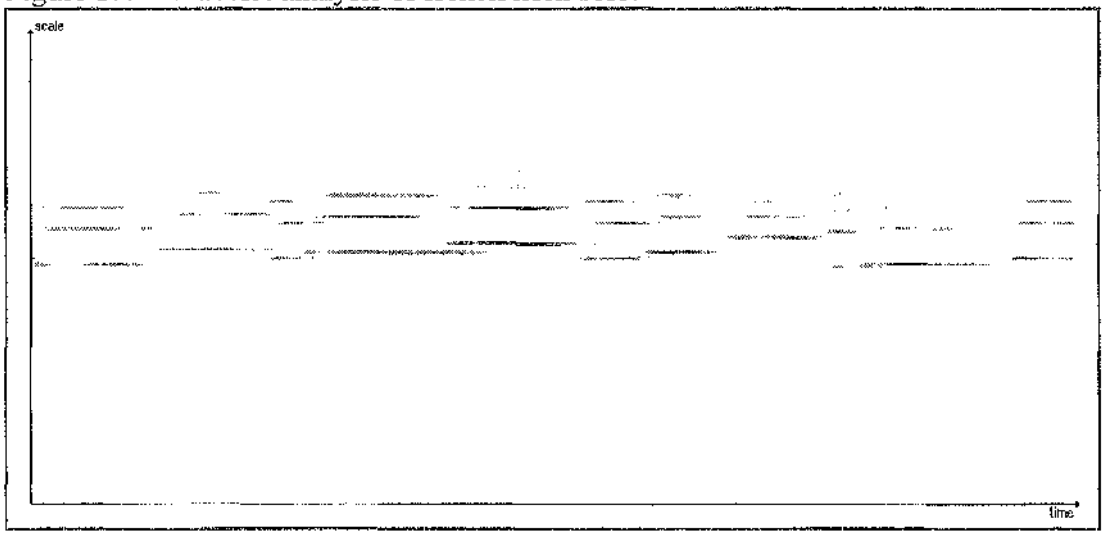


Figure 3.3 is typical of untuned percussion instruments in that the notes are short but cover a wide range of frequencies, producing vertical bands in the transform. Such examples should be among the easiest for any onset detection method. The playing style and reverberant environment in the french horn example, on the other hand,

result in notes that rise slowly and overlap considerably, making location of the note onsets difficult.

As would be expected, the increased frequency resolution of this transform dictates a lower time resolution. If the input signal contains N samples, the highest octave band contains N/4 coefficients and if this is divided into twelve semitone bands they will have an average of N/48 coefficients. Whilst this may be viewed as prohibitive in some applications, it can be seen that using a sample rate of 44.1 kHz gives rise to an average time resolution of approximately 1 msec in the highest octave band (doubling for each successively lower octave). Such resolution in the context of musical analysis is perhaps not so bad when it is considered that, for example, a demi-semi-quaver (an eighth of one beat) at the fast tempo of 160 beats per minute lasts for some 47 msecs. A more detailed discussion of time resolution issues will be undertaken later, in the context of specific examples.

The other known drawback of this type of analysis can be observed in Figure 3.3: whilst the wavelets are strictly limited to their semitone bands in the frequency domain, they are less well localised in the time domain and some spreading of the energy in each hit is observable.

3.4 Auditory Representations

It has been described how several researchers have attempted to model the human auditory system, in the hope of subsequently modelling the higher level aspects of perception. However, the separation of low from high level processes which is promoted herein must be considered. Even if the auditory system could be precisely modelled up to the signal generated on the auditory nerve, interpretation of that signal (as usually carried out by the brain) is still required. It is for this reason that an alternative approach has often been employed. The idea is to use observations of the auditory system's characteristics to transform the time-frequency plane prior to further analysis. This should result in the perceptually salient features being enhanced.

Such an approach is described in [Wang & Shamma 95], which notes that it is uncertain how features are extracted from the time-frequency plane at higher levels. It is also stated that the division of the frequency scale is well modelled by a wavelet transform. This is also pursued in [Brookes et al 96], which explains how the bandwidth of each division on the frequency scale should be proportional to its centre frequency, although the auditory system's division is not so precisely logarithmic (tending towards linearity at lower frequencies). The paper also notes that the auditory

system is not equally sensitive to all frequencies, and some measure of this should be incorporated into the analysis.

3.5 Adopted Method

The modulus values derived from Newland's semitone-based wavelet transform have been adopted as the basis for further analyses. This is primarily because the transform is ideally suited to the current work, but has not been used in any practical applications. The division of the frequency scale approximates that of the auditory system (and can be based on semitones), and it would seem reasonable to apply some of the other observations which have been made on auditory perception.

For example, a weighting could be applied to each semitone level, based on its centre frequency, to emphasise features that occur in frequency ranges where auditory sensitivity is high. Many researchers have attempted to derive such a measure experimentally, so that tones at different frequencies could be weighted such that they would be judged equally loud. [Stevens 72] presents a frequency weighting function derived by combining the results of a large number of such studies, and an approximation of this, extrapolated to the low and high frequency regions not covered in the studies, has been applied to the modulus plane (a detailed description is given in Appendix B). This is similar to the method adopted in [Brookes et al 96], although that only involved emphasising the most important region of the spectrum.

Loudness judgements are complicated by many other factors such as duration, masking, and so on ([Moore 82] explains many such phenomena), making the application of a simple weighting function to complex sounds problematic. However, the aim here is not to exert precise control over the perceived loudness of a complex sound, but merely to apply some measure of the relative sensitivity of the auditory system at different frequencies, so as to improve the psychoacoustic relevance of the transform.

Another factor which could be taken into account is the relationship between amplitude and perceived loudness. This has traditionally been expressed via the decibel scale as

$$decibels = 20\log_{10}\left(\frac{A_1}{A_0}\right)$$

in which A_0 is the amplitude of some reference sound. However [Stevens 55] notes that equal decibel increments do not, in practice, appear to correspond to equal loudness increments. This observation led to the development of the *sone* scale to explain experimental observations. Loudness in sones is expressed as

$$sones = kA^{0.6}$$

where k is a constant dependent on units used. However, Stevens also notes that the form of the spectrum affects the perceived loudness, so that measuring the loudness of a complex sound is not straightforward. Such difficulties persist in this area ([Moore 82] includes a discussion), but in seeking some perceptually relevant transformation to map the modulus values to a scale more closely related to loudness, those above are the best candidates.

Figure 3.5 – Figure 3.3, with frequency weighting and decibel scale applied.

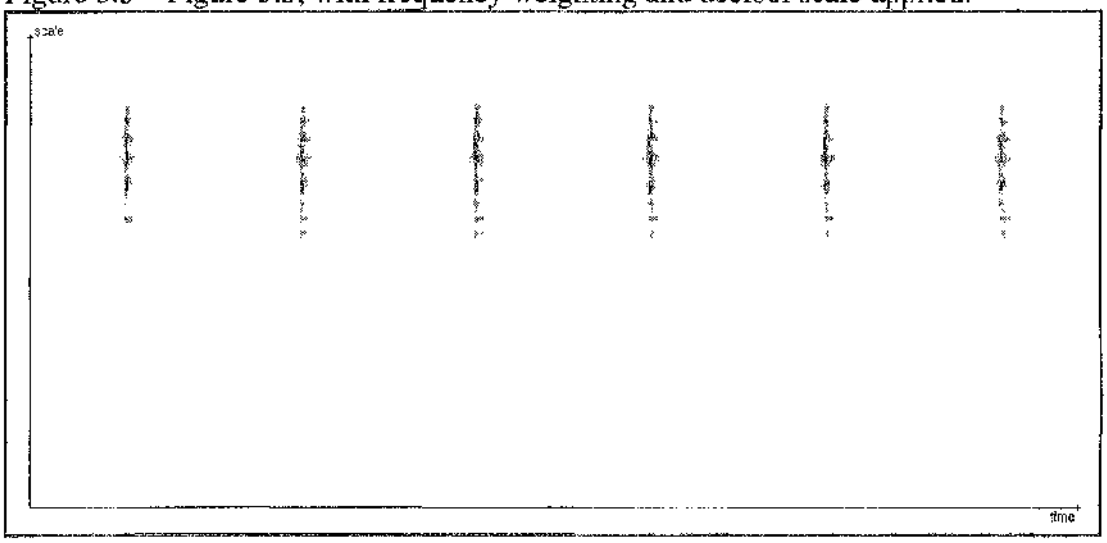
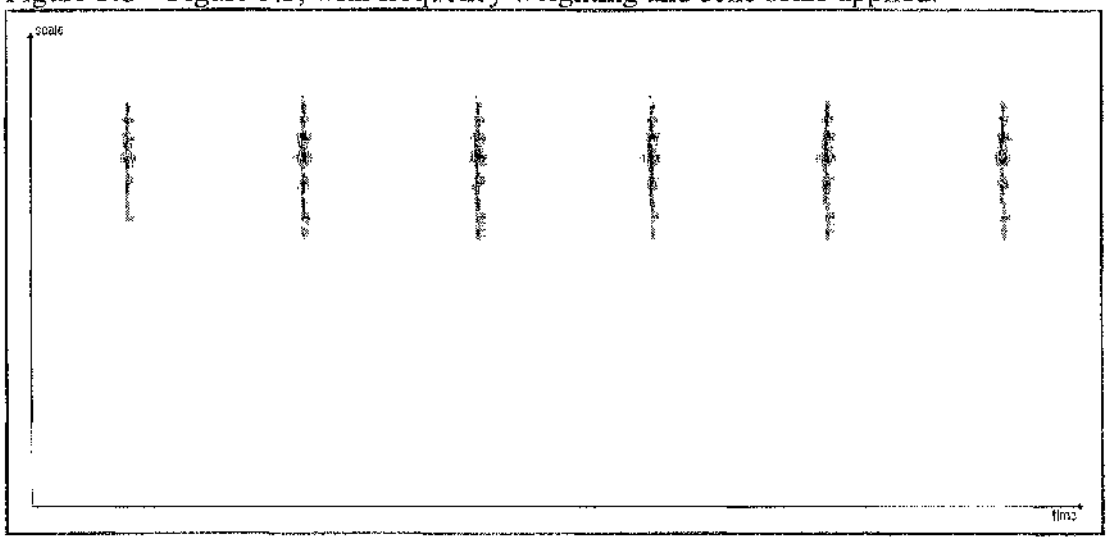


Figure 3.6 – Figure 3.3, with frequency weighting and sone scale applied.



Figures 3.5 to 3.8 show these techniques applied to the rim-shot and french horn examples. Transformations based on both the decibel and sone scales have been implemented, and both are shown for comparison (further detail on the methods used can be found in Appendix B). In all cases, the significant parts of the plane are emphasised and some of the noise evident in the originals disappears (the spreading previously evident in Figure 3.3 is also overcome to some extent). Later examples will show that the auditory transformations applied to the time-frequency plane also improve the results of further analyses.

Figure 3.7 – Figure 3.4, with frequency weighting and decibel scale applied.

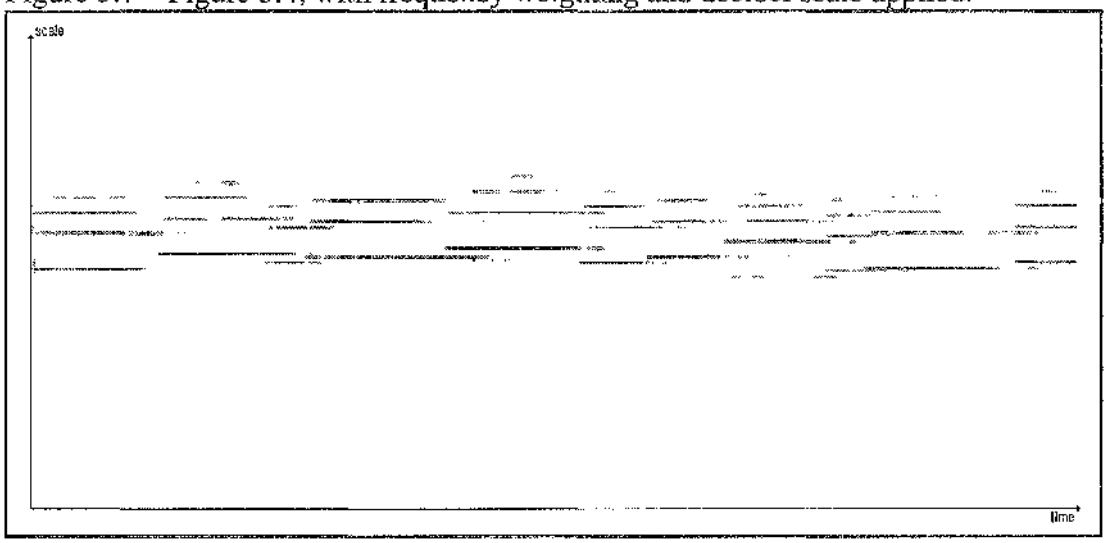
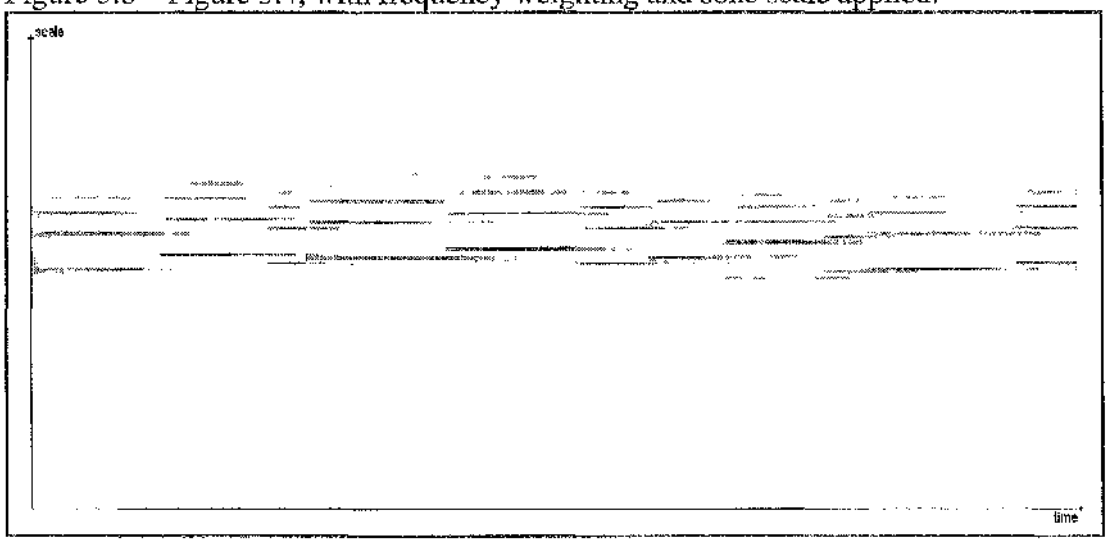


Figure 3.8 – Figure 3.4, with frequency weighting and sone scale applied.



3.6 In Summary...

The poor time localisation and inappropriate division of the frequency scale in Fourier analysis imply that it is not well suited to analysis of music for time-domain structure. The wavelet transform provides localisation in both the time and frequency domains, but traditional methods have involved decomposition into octaves, which is too coarse for musical analyses. Newland's harmonic wavelets overcome this problem, and have been adopted as the most suitable time-frequency decomposition for the current work. In addition, perceptually motivated transformation of the modulus values has been applied to enhance the significant features in the plane. The main drawback in using this transform is its relatively poor time resolution, however the next chapter will show that successful analyses of the time-frequency plane for onsets can be undertaken in spite of this.

Chapter 4
Onset Detection

Having arrived at a suitable time-frequency decomposition, the issue of how to detect the locations of onsets can now be addressed. Rather than seeking to model any particular type of onset, the common characteristics of the various kinds of change to be detected are explained. These suggest a method of change detection which is then described in detail.

4.1 Observing Onsets

The use of time-frequency decompositions has enabled studies of the detailed structure of various types of sound. In [Grey 75], 16 individual notes from assorted woodwind, brass and string instruments were analysed. Observation of the onsets shows that all of the strong partials commence within around 40 msecs of the first. However, it is also reported that the duration of attack can vary with pitch, player and instrument, making it difficult to model the situation with any generality.

[Pollard & Jansson 82] quote a wider range of times for duration of the starting transient before the steady state of the note (5 up to 350 msecs), and because many instruments can exhibit long attacks, it must not be assumed that the onset is necessarily very sudden. This work also notes that it is likely that the brain finds overall feature changes during onsets (rather than interpreting detailed information from a range of frequency bands). This would appear to support earlier comments on the problems associated with automating such per frequency band analyses. In addition, it is mentioned that some comparison between the transient and steady state parts of a note should enable detection of the change.

These are the basic observations which have motivated the adopted change detection method, and the features being described can be seen in the french horn and rim-shot examples of figures 4.1 and 4.2. These figures show identically sized areas of the time-frequency plane (51 semitone bands, 128 msecs duration) on the decibel scale. The percussion example (figure 4.1) has many scale levels increasing rapidly in energy at almost the same instant, and shows that the events to be detected may be

very short and without clear harmonic structure. The french horn example (figure 4.2) exhibits partials which start slowly over some finite period of time, and also includes a small amount of the kind of overlap which must often be overcome. Such ringing may arise due to environmental reverberation or the characteristics of the instrument. Whilst these examples are by no means exhaustive, they illustrate extreme cases of the kind of onsets to be detected.

Figure 4.1 – Rim-shot onset.

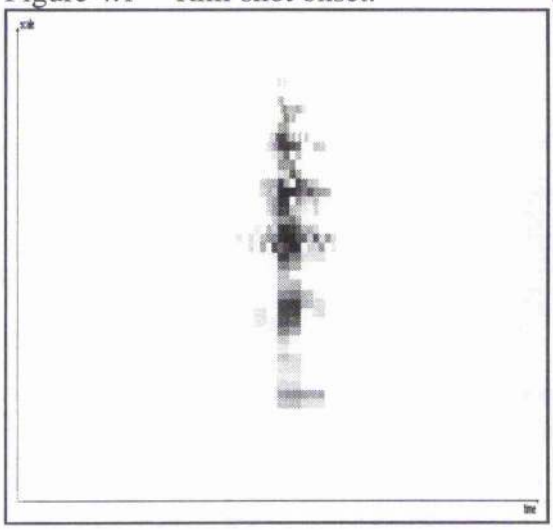


Figure 4.2 – French horn onset.

Although the onsets above are shown in isolation, and only cases with a single melodic line (monophonic) have been discussed, some degree of background noise or other sound may be present. Therefore, in addition to highlighting onsets in the presence of overlapping partials, the measure of change should be as resistant as possible to such continuous interference. Given these observations on the nature of note onsets, the following cases of potential transition types are defined.

- One note sounds at a time, with perceived gaps between notes.
- One note sounds at a time, but with a legato playing style so that as one note ends another seamlessly begins.
- Notes ring out and overlap, so that a number of notes may be sounding at any one time. This begins to approach the complexity of the polyphonic case, the main simplification being that different instruments are not playing the same note at the same time. It will not be true, for example, that note offsets occur in the

same order as onsets — since softer notes are likely to have shorter durations than louder ones of the same nominal length.

Of course, the above classification refers to transitions between two notes – all three could potentially occur in the same piece and in the context of different attack characteristics. Also, environmental factors can be important, and the examples shown previously have illustrated how reverberation obscures gaps between notes in many real situations.

4.2 Highlighting Change

Before describing the adopted method of change detection formally, the concept is explained in the context of the onsets depicted in figures 4.1 and 4.2. Given some point in time (denoted by a vertical line on the time frequency plane), windows of modulus values before and after are compared for any change. Figure 4.3 shows the situation at approximately the onset time of the single rim-shot hit. The thicker central line denotes the point in time in question, whilst the dashed boxes either side represent the windows to be compared. It is evident that a comparison of the two adjacent windows would show a peak in their difference at the onset (if they were to be moved horizontally across the whole plane). Figure 4.4 illustrates how a peak in difference is also likely to occur for less percussive onsets, and also includes a second pair of windows located in the steady state portion of the second note (where it is apparent that a near zero difference should occur). Although not made explicit in the diagrams, it should be obvious that (as long as the measure of difference used is symmetric) many offsets will also be highlighted by such a method. This is not viewed as a deficiency, and will be discussed later.

Implementation of this idea requires both a measure of what is taking place in each of the rectangular windows, and a way of comparing these. The first step is to divide the modulus plane (along the time axis) into a series of vectors, each constituting a slice through the plane at the highest time resolution. A way of comparing areas of the time frequency plane, based on this vector representation, was described by the author in [Tait 95]. That technique is inappropriate in this context, however, as it takes account of detailed timing information when calculating the similarity. Based on the previous discussion of onsets (and the example in figure 4.4), it is evident that there is no need to take account of exactly *when* a partial started – only how prominent it was during the window. Detailed timing information is therefore not used at this stage and an average vector, calculated from those in each window, is sufficient to highlight change.

Figure 4.3 – Detecting the rim-shot onset.

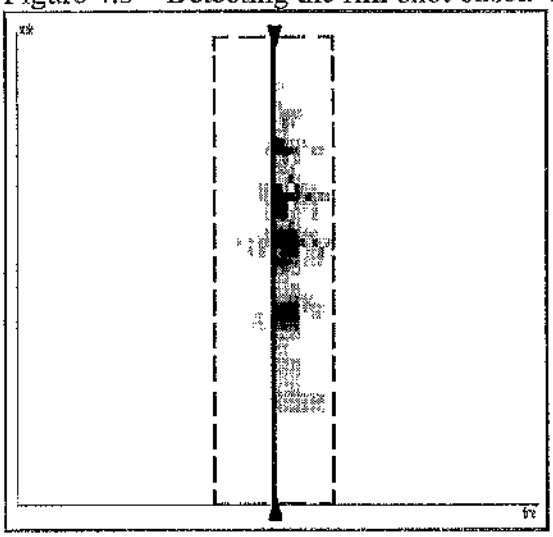
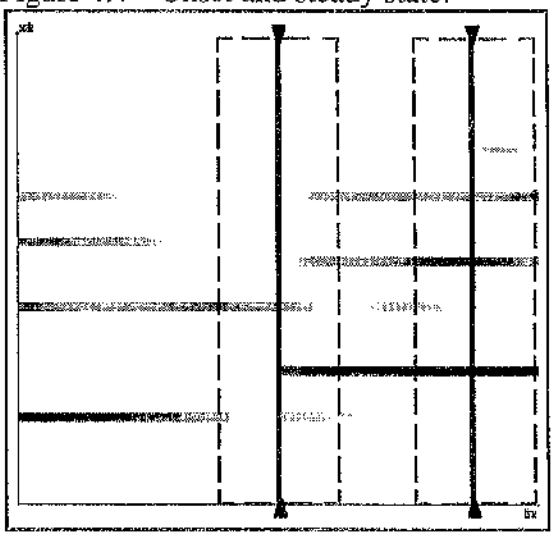


Figure 4.4 – Onset and steady state.



Quantifying the change between two such vectors has been considered by several workers. [Gray & Markel 76] investigate the relative merits of a number of distance measures when applied to different types of spectra, in the context of speech recognition. Meanwhile, [Plomp 76] describes the use of the Euclidean distance to determine the similarity of musical tones and vowel sounds. Before describing this work in detail, we will establish some notational conventions. The plane of modulus values is divided into vectors, each constituting a slice through the plane at the highest time resolution. If semitone bands 1 to N are being considered, and two N element vectors L and R (representing the averages in the left and right windows above) have been calculated, then

$$D = \sqrt{\sum_{k=1}^{N} (R_k - L_k)^2}$$

represents the Euclidean distance between them. Plomp used spectra obtained from a perceptually-inspired third octave filter bank (with a decibel scale), producing 15- or 18-dimensional vectors, and then attempted to map the sounds into a timbral space of reduced dimensionality. It was noted that the dissimilarities predicted by the Euclidean distance were in close correspondence with results obtained from listening tests.

Plomp also explains that the Euclidean distance is a special case of Minkowski's measure which, in the notation above, would be defined as

$$Minkowski(L,R,p) = \sqrt[p]{\sum_{k=1}^{N} |R_k - L_k|^p}$$

This raises the question of whether there is a more suitable value for p. As Plomp notes, setting $p > 1$ implies that the greatest difference amongst the spectral components is most important (since it will be emphasised); whereas setting $p = 1$ implies that the number of components with significant differences is more important. Although the results most closely corresponding to the listening tests were obtained with some unspecified $p > 2$, it is noted that the value was not very critical and $p = 2$ was chosen.

The derivation of a suitable similarity measure would allow the retrieval of sounds from a database by simply providing some existing sound resembling that required. The above technique was investigated in this context in [Feiten & Gunzel 93], in which listening test results were again compared with those obtained using the above similarity measure. However, the experiment also measured the effect of using different loudness scales (amplitude, decibel and sone) and various values of p. Interestingly, the results most closely corresponded to those in the listening tests when the sone scale and a value of $p = 5$ were used.

Before considering some concrete examples, a function which would implement the onset detection method described above (that of comparing adjacent windows) is defined. Minkowski's measure is used to compare the average vectors, and the kth element of the average vector obtained from scale levels 1 to N between times t_1 and t_2 (inclusive) is defined as

$$avg(t_1,t_2)_k = \frac{1}{t_2 - t_1 + 1} \sum_{t=t_1}^{t_2} M_{tk}$$

Although semitone levels in the wavelet analysis have different time resolutions, a rectangular grid (with time units at the highest resolution) is superimposed to ease the expression of calculations, and M_{tk} is the modulus value at time t in semitone level k of this grid. In implementation, the redundancy introduced by decreasing time resolution can be exploited to improve the efficiency of some algorithms. Now, the *vector distance* at some time t, calculated from windows to the left and right of sizes l and r time units, with an increment of i time units between the end of the left window and the start of the right, and using exponent p in Minkowski's measure, is expressed as

$$vector\ distance(t,l,r,i,p) = Minkowski(avg(t-l,t), avg(t+i,t+i+r), p)$$

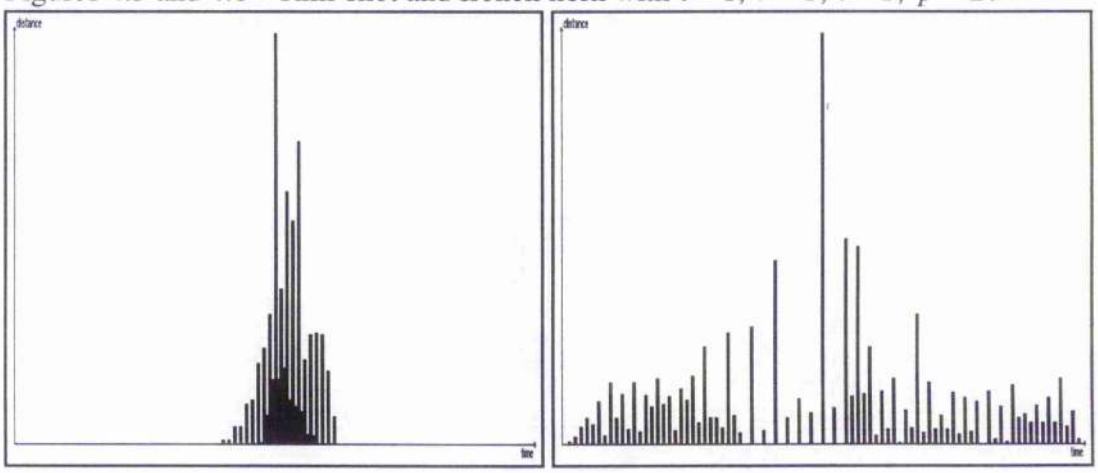
This equation is introduced as it generalises the idea described previously, and provides a common notation in which the various methods that were investigated can be presented.

4.3 Derivation of Method

In this section, the derivation of a measure of change based on the above observations concerning onsets and audio similarity measures is described. The various methods which were tested are shown, as are the effects of using different loudness scales and different values of p.

Figures 4.5 and 4.6 show the results of calculating the Euclidean distance between adjacent vectors in Figures 4.1 and 4.2 (see appendix C for a note on the graphing technique). In other words: the vector distance is calculated (for each possible t) with $l=1$ and $r=1$ (single vector windows), $i=1$ (adjacent windows), and $p=2$ (Euclidean distance).

Figures 4.5 and 4.6 – Rim-shot and french horn with $l=1$, $r=1$, $i=1$, $p=2$.



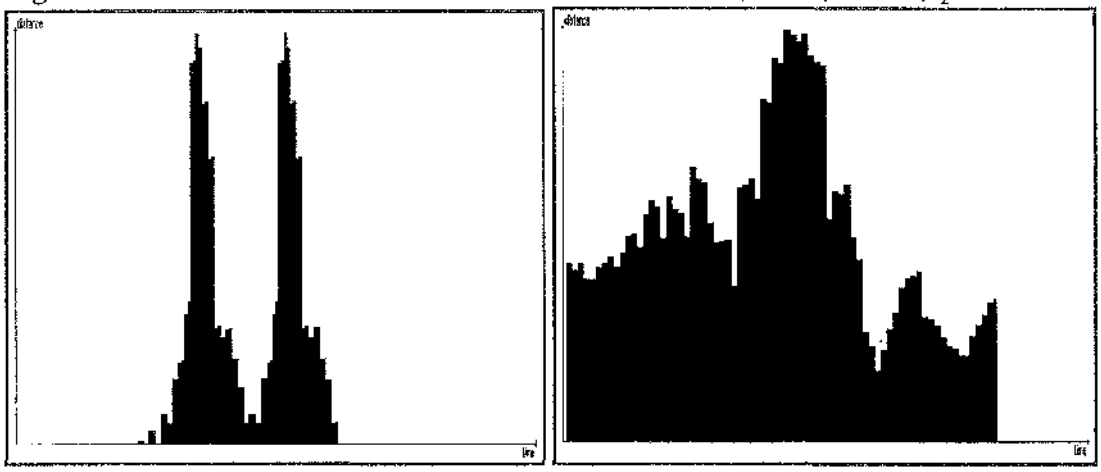
The interspersed zeros occur because there is sometimes no energy in the signal at the levels of highest time resolution, so that there is no difference between adjacent vectors. Although the onsets do produce peaks, the anticipated offset peaks are hard to distinguish. In general, this measure is too sensitive to fleeting change and results in too many spurious peaks.

To overcome this, larger values of i were investigated. These give a measure of change over larger time scales and result in fewer spurious peaks (whilst occasionally highlighting onsets which would be missed by the previous method). Figures 4.7 and

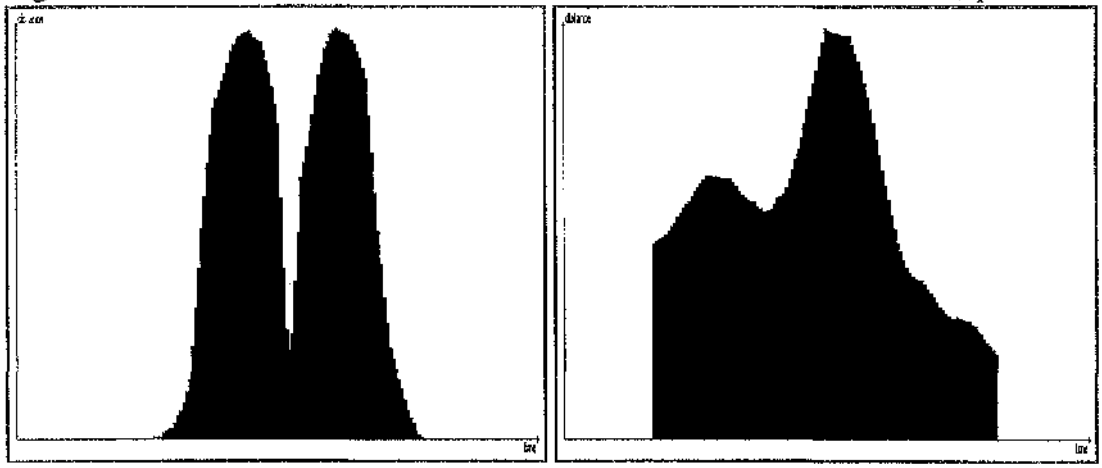
4.8 show the results of increasing i to approximately 30 msecs in the above examples. The onset and offset of the rim-shot are now clear, but the french horn example would still be difficult to interpret.

The remaining problem is that of detecting more gradual change, such as occurs when notes with a relatively long attack time are played in a reverberant environment. These changes are not registered by the methods already described because a single vector (i.e. a time slice of 0.7 msec) does not adequately represent the more extended character of a note — to capture such information, longer time periods must be considered.

Figures 4.7 and 4.8 – Rim-shot and french horn with $l = 1$, $r = 1$, $i = 30$, $p = 2$.



Figures 4.9 and 4.10 – Rim-shot and french horn with $l = 30$, $r = 30$, $i = 1$, $p = 2$.

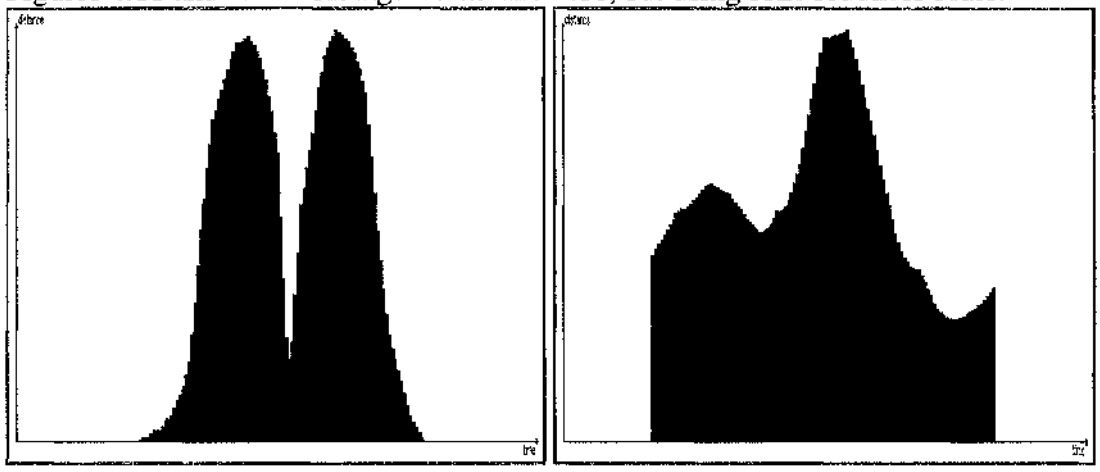


Figures 4.9 and 4.10 show the results obtained by using adjacent windows of 22 msecs. The longer windows are representative of the state of the time-frequency plane over some period of time, so that more gradual change is noticed. The french horn

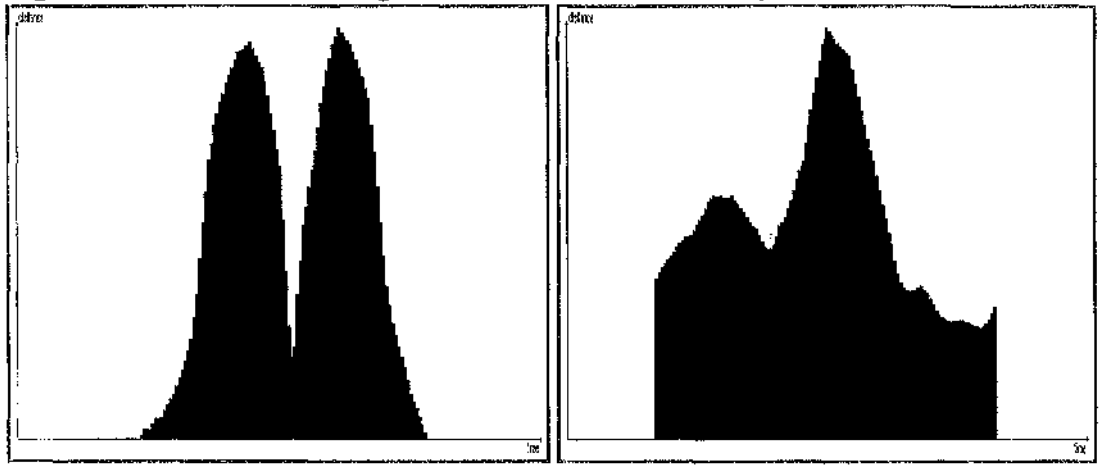
example now exhibits a clear peak at the onset position, with a lesser one at approximately the location of the offset of the previous note. From figure 4.9, however, it is clear that peaks can be widened and somewhat smoothed by this technique — as it happens, this simplifies their automatic detection (as described in the next section). However, it will become apparent that there is a trade off between window size and detection accuracy – shorter windows can detect events more precisely, but the resulting peaks may be much harder to detect, and there will be more spurious peaks.

The use of longer windows with a gap between them (that is, a higher value of i) was not found to be useful, since the results are not greatly improved and there is the added problem of precisely locating the onset in the gap.

Figures 4.11 and 4.12 – As figures 4.9 and 4.10, but using sone loudness scale.



Figures 4.13 and 4.14 – As figures 4.11 and 4.12, but with $p = 5$.



For comparison, Figures 4.11 and 4.12 involved the same parameters as the previous two figures, but with modulus values first mapped to the sone (rather than the decibel) scale. It can be seen that the peaks are slightly more defined, though there is not a vast difference (the results presented in §6.3.1.2 and §E.1.2 illustrate this further).
Finally, the similarity method found to be best in [Feiten & Gunzel 93] was used to compare adjacent windows in Figures 4.13 and 4.14. It is evident that the higher value of p emphasises peaks even more; although, again, the effect is not marked.
A question which has not yet been tackled is that of repeated notes. These will generally be accompanied by a new attack (that is, an increase in energy which will be detected as a change), or will be preceded by a short gap (which will also be detected). Further examples will be given in a later chapter, and can also be found in [Tait & Findlay 95 & 96].

4.4 Peak Detection

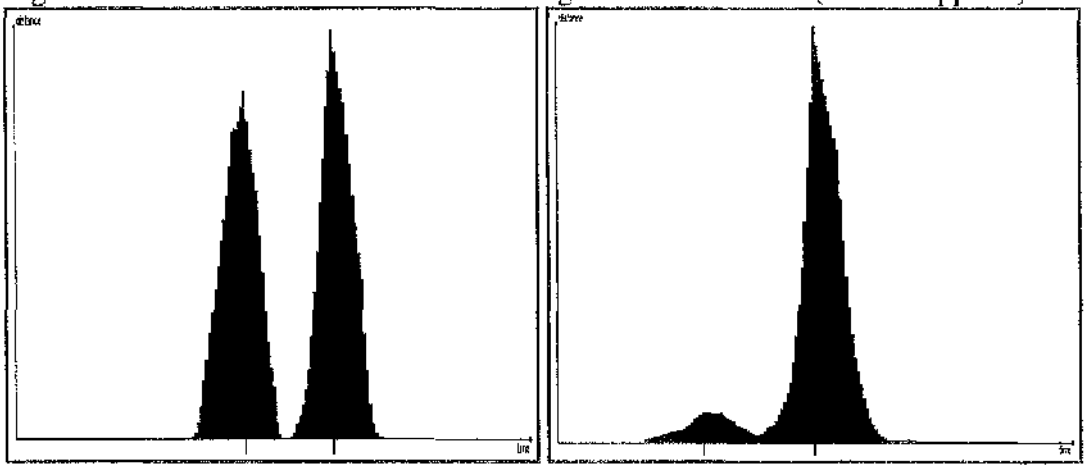
As in [Basseville 88], a 2 stage process is employed: the generation of a change indicating signal (like those above) is separated from the monitoring of that signal to select changes which are of interest. This means that, for example, although the focus here is on detecting onsets, inspection of the results of the change detection for other types of transition could be envisaged. The remaining task, then, is the design of rules that can be used to analyse the kind of graphs shown in the previous section. This process is further decomposed into 2 stages: the detection of peaks and the selection of those peaks which correspond to onsets.
Whilst it might be expected that there would be much existing work on peak detection, that seems not to be the case. A discussion can be found in [Kristo & Enke 89], which considers the real-time detection of peaks in mass spectrograms. As explained therein, many methods are proprietary and unavailable for evaluation. It also emerges that the best method is often highly application-specific (for example, what distinguishes spurious peaks, how peaks are compared and how thresholds are set will vary).
As it happens, it has been unnecessary to resort to elaborate peak detection methods, because (as may have been observed in the previous section) the slowly-moving time windows, in combination with the averaging process utilised in calculating the vector distance, result in peaks that rise and fall smoothly. Such peaks are easily detected by simply locating points at which a series of successive increases is followed by a series of successive decreases. In addition, the relevant peaks generally dominate and spurious peaks are easily disregarded.

The method involves 3 parameters: *peak_thresh*, *min_len* and *peak_reqd*. If calculation of the vector distance for all possible points in time results in a function V mapping the points in time $1..N$ to their respective distances, then V is traversed and all points in time t_i found such that the following conditions are met.

- $V(t_i) * 100 > peak\_thresh$ (the distance is above a percentage threshold).
- If peaks are found at times labelled t_1 to t_P, then $t_i - t_{i-1} \geq min\_len$ (for all i such that $1 < i \leq P$). In other words, peaks must be separated by at least *min_len* (which corresponds to the minimum note length in onset detection). Clusters containing a series of peaks, each separated by less than *min_len*, are resolved by selecting the highest amplitude peak.
- $(V(t_{i\ j-1}) \leq V(t_{i-j})) \& (V(t_{i+j}) \geq V(t_{i+j+1}))$, for all j such that $1 < j < peak\_reqd$, $t - j > 1$, and $t + j < N$. This means that a peak must be preceded by sufficiently many successive increases, and followed by sufficiently many successive decreases.

Peak detection is aided by plotting the distance without the root applied in Minkowski's measure. This is illustrated in Figures 4.15 and 4.16, which show the same results as Figures 4.13 and 4.14, without the root applied. Peaks are greatly emphasised, and the dashes on the x axis indicate peak positions found by setting *peak_thresh*=3, *min_len*=10, and *peak_reqd*=10. Appropriate parameter settings for more realistic examples will be discussed in due course.

Figures 4.15 and 4.16 -- Peak detection in figures 4.13 and 4.14 (no root applied).



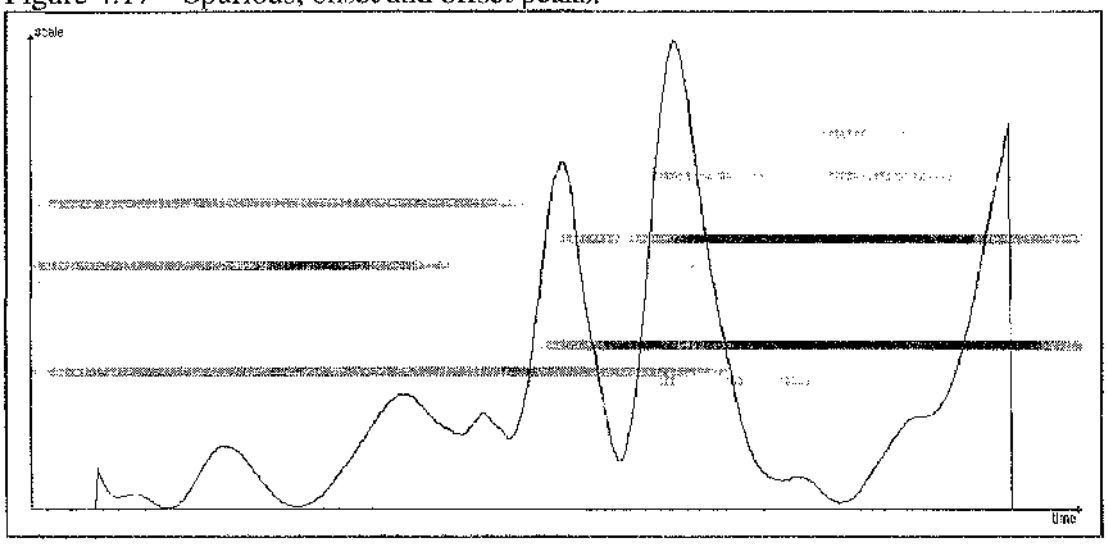
Finally, peak detection was aided in the cases presented later by smoothing the vector distance function. This was accomplished by averaging a number of adjacent values at each point in the function. The number of adjacent values averaged is referred to as the *smoothing window*.

4.5 Categorising Change

Given a peak in the distance function, there are several possibilities: it may correspond to a note onset, an offset, or may have arisen from some variation occurring within a note or another interfering source. As shown in figure 4.15, offsets can produce significant peaks, thus eliminating the possibility of simply considering a peak's height. Rather, the time-frequency plane must be analysed in the vicinity of the peak position, to decide whether or not the highlighted change was indeed an onset.

For example, figure 4.17 shows another part of the modulus plane of the french horn piece from which the single onset in figure 4.2 was taken (lasting 348 msecs). The central onset peak is followed by a large peak corresponding to the offset of the previous note, and is preceded by several spurious peaks. Other possibilities are onsets after silence, offsets before silence and coincident onsets/offsets.

Figure 4.17 – Spurious, onset and offset peaks.



Verification of onsets is accomplished by verifying that a subset of semitone bands exhibited a significant (and sustained) increase in energy, during the time bounded by the analysis windows from which the vector distance was calculated. Two methods will be considered: for each semitone band, either the point in time as indicated by the

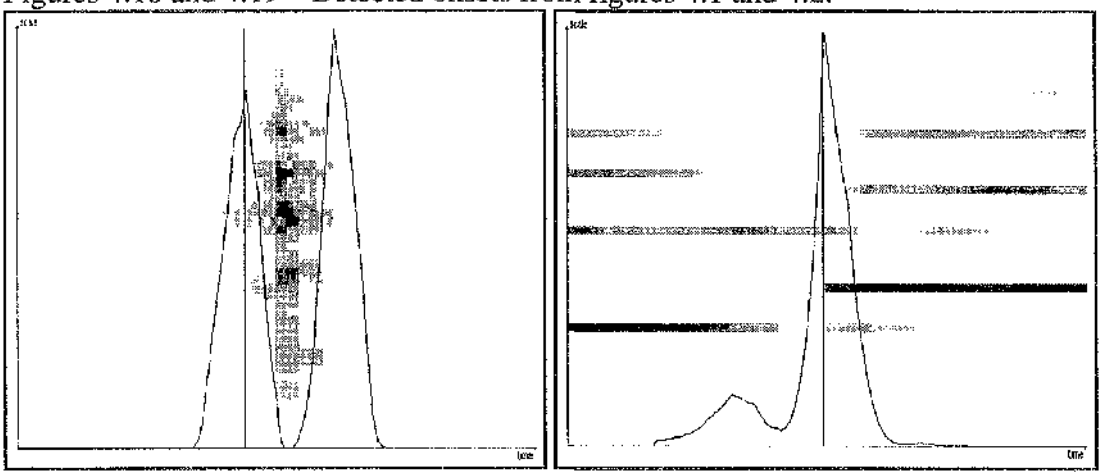
peak location, or the point in time at which the difference between an earlier and a later average is maximised can be found. The latter technique was investigated since it was observed that partials did not, in general, commence simultaneously. Given the semitone band averages to the left and right of the chosen point in the time-frequency plane (*left_avg* and *right_avg*), a partial onset is noted if the later average and the percentage change are both above some thresholds (*band_thresh* and *change_thresh*). That is, the band is deemed a commencing partial if the following conditions are met.

- $right\_avg > band\_thresh$
- $\frac{right\_avg - left\_avg}{left\_avg} * 100 > change\_thresh$

(if *left_avg*=0, then the change is assumed above threshold as long as the first condition has been met)

The peak is then categorised as a note onset if sufficiently many partials commence in its vicinity (thus a third parameter, *num_partials* is introduced).
If partial onset times have been calculated for each level, a set of different onset times will result. In this case, the note onset time is taken to be the time of maximum difference associated with the band containing most energy (that is, with the greatest value of *right_avg*). This was used, after an average of the partial onset times was found to be too skewed by late-starting partials. On the other hand, if peak position is used throughout, that is used as the note onset time.

Figures 4.18 and 4.19 – Detected onsets from figures 4.1 and 4.2.



Again, demonstration of this technique applied to a large number of examples is deferred until a later chapter. For now, figures 4.18 and 4.19 show the onset positions (as vertical lines) located from figures 4.1 and 4.2, using the vector distance as calculated in figures 4.15 and 4.16 (overlaid on the time-frequency plane). These results were both achieved with parameter settings of *change_thresh*=80%, *band_thresh* =0.1, and *num_partials* =1. These settings imply that a small number of relatively rapidly increasing partials are required, and the effect of other settings will be investigated more fully later.

4.6 What Next?

This chapter has described a change detecting measure based on vector distance, which has been applied to the time-frequency plane derived from the wavelet transform. It has been shown how this can be used to highlight onsets, in conjunction with a method of categorising the kind of event taking place at each peak of change.

Having arrived at this technique, several crucial issues must be addressed. For example, which parameter settings and loudness scale give the best results for various types of input? Can these be set so as to provide acceptable results over a range of inputs, or does each example have its own best parameters? An intermediate case may be that the possible inputs can be partitioned into families, each of which is associated with a single parameter set. A user would then only be required to select which family an input should be associated with, rather than adjust individual parameters. This is important, since any application of the method will be most useful if it performs well without requiring a large number of parameters to be adjusted.

These issues can only be resolved by the observation of the method when applied to a suitable range of inputs, and the design of such an experiment is the subject of the next chapter.

Chapter 5
Experimental Design

It was commented in chapter 2 that there are no standard benchmark tests for any of the various types of musical analyses that researchers have sought to automate. This is in contrast to the field of speech recognition, for example, where there are a number of large databases of pre-recorded speech. This means that new techniques can be evaluated in the context of various speakers and types of utterance, and their performance against other published methods considered.

The different goals of onset detection, pitch detection, automatic transcription, auditory scene analysis, and so on almost certainly imply different experimental designs. Whilst this chapter will focus on the testing of onset detection methods, it is hoped that the approach suggested could also be applied in other areas.

5.1 Assembling Test Cases

This section considers the design of an (idealised) set of test cases for audio analysis, before discussing the implementation of a concrete experiment.

In seeking to apply principles of experimental design, we must consider the range of possible inputs, and the potential sources of variability in those inputs. Musical and non-musical examples will be considered separately, as will the monophonic and polyphonic cases. Monophonic musical examples have been the focus of this and much other work, and the structure inherent in such examples aids in the design of a suitable set of test cases. The extension to other more complex types of input can then be considered.

5.1.1 Monophonic Musical Examples

The inputs in this category have only a single instrument playing one note at any given time. It is unrealistic to disallow reverberation, so notes may overlap, but not more than one note should be played at a given instant. Thus, environmental (or instrumental) reverberation may be present, but there will be no chords (for example).

The experiment should include a range of instruments and, to facilitate comparison, a performance of the same piece should be recorded for each one. The instrument can

be viewed as one variable (for which the selection of values will be discussed shortly), and the piece must be composed and performed so as to also include a range of possibilities for each of the following.

- Pitch – an appropriate sampling of each instrument's range (including extremely low and high notes) should be included. Of course, this will vary for each instrument – implying that some elements of the piece may be instrument-dependent.

- Interval – both ascending and descending should be considered: the most important are likely to be the octave, the semitone (microtones are not considered), and repeated notes.

- Note length – runs of extremely short notes are likely to pose greatest difficulty, but a variety should be tested.

- Attack time – many instruments can be played so as to produce a wide range of attack times. Since slower onsets are likely to be harder to detect, these should be included in any experiment.

- Offset time – this is harder to specify, and the form of note offsets will generally also be dependent on the degree of reverberation.

- Degree of overlap – given that reverberation cannot be avoided, the effect of the resulting overlaps must be investigated.

- Dynamics – it is possible that a quiet note following a louder one may be drowned out. To investigate such issues, the dynamics of the piece should indicate a range of such cases.

- Modulation – some instruments can be played so as to produce frequency or amplitude modulation within notes. This is common in music, and must be considered.

- Transition type – finally, a number of possibilities may exist for the type of transition between two notes. The simplest are likely to be with a gap, or following immediately, but glissandi (with pitch gliding smoothly from that of the first to that of the second note) and others may be possible.

Ideally, a range of possible values should be investigated for each variable (some of which are suggested above), such that performance is demonstrated for a range of possibilities – including extreme cases.

The variable with perhaps the greatest degree of freedom is probably the instrument itself. A range must be chosen which includes sufficiently different kinds of sounds, and is far-reaching enough so that it allows any results to be properly assessed in the greater musical context. Instruments which utilise different mechanisms for producing notes tend to have distinct sounds, and a categorisation based on method of sound production has long been used by musicologists [Diagram 76]. This could serve as a guide to the selection of an appropriate set of test instruments.

The classification contains five families, as described below (these are also sub-divided, and the most important distinctions are indicated).

- Chordophones – sound is produced by vibrating strings, which can be bowed, plucked or struck (for example: violin, guitar and piano).

- Aerophones – vibrating air produces their sound, and this category includes a wide range of instruments such as the flute, whistle, clarinet, oboe, trumpet, mouth organ and bagpipes.

- Idiophones – these instruments are made of sonorous material, and are mainly percussion instruments such as bells, gongs, rattles and so on.

- Membranophones – sound from these instruments is made by a vibrating membrane, and they are mostly drums.

- Electrical/Mechanical – this is somewhat of a modern catch-all, and is much less cohesive (including electric guitars, synthesisers, and so on).

With the possible exception of the final category, it can be seen how a sampling of the instruments in the various families could be achieved. This should cover all categories, whilst also representing the variety of instruments within each group. The last category contains a potentially limitless variety of sounds, and what constitute relevant examples may depend on the application. This does not mean, however, that potentially problematic synthetic timbres should be ignored.

Finally, it should be noted that not all of the variables specified above may be relevant to a given instrument – this is particularly obvious in the case of drums, for example.

5.1.2 Monophonic Non-Musical Examples

Having defined the monophonic musical case as above, the remaining types of monophonic example must now be considered.

There are various kinds of analyses which have been applied to speech, but it is evident that a set of test cases could not be defined in the above terms. However, what is known about its structure and method of production do allow the considered design of experiments involving a range of speakers and utterances (for example). Thus, it becomes apparent that in order to design test cases which span some type of input, we must have some prior knowledge about the way in which that input arises and its likely internal structure. This allows the specification of variables to be investigated, and defines the kind of inputs which must be considered. Such specification is also what allows the design of analytical algorithms – observation of onsets motivated the detection method described in the previous chapter, and speech processing generally exploits the restriction to a limited domain.

Given the necessity of characterisation, only monophonic inputs which can be expressed in terms of the classification in the previous section will be considered herein. This does not mean that all inputs must necessarily be 'musical' in any traditional sense. As in the case of drums, there may be no notion of pitch or interval, and there is no requirement for any rhythmic structure. The examples which will be included can be thought of as containing a series of events, whose onset times are to be found.

5.1.3 Polyphonic Examples

This category can be thought of as containing sounds that are mixtures of at least 2 others. If an experiment is to be designed involving polyphonic inputs, detailed classification must again be considered.

One such example arises in [Cooke 91], where the aim was to specify the kind of intrusive sources to be dealt with in a system to separate speech from other sounds. It has already been described how test cases for speech may be derived. The interfering sounds contained both monophonic and polyphonic examples, and were characterised on the degree to which they were wide- or narrow-band, continuous or interrupted, and structured or unstructured. This allowed the construction of a set of test cases which were sufficiently wide-ranging, according to the classification used.

A similarly constructed set of test cases appears in [Arcelo 95], which investigated mixtures of sinusoids and white noise. The well-defined nature of the inputs allowed precise control over sinusoid frequency, signal to noise ratio, and so on. This was carried out in the context of different parameter sets, thus allowing the effect of various parameters to be quantified.

More general mixtures may consist of an arbitrary number of monophonic lines – in any case, the components must be specified in some detailed way (as above) before an experiment can be designed. Of course, the mixture itself will have some properties which must also be considered. These are likely to depend on the exact application and technique under consideration, however the following may arise in a musical context.

- If melodic lines are to be followed, do they cross (in pitch), contain notes starting at the same instant, or involve unisons?
- To what extent do the harmonics of overlapping notes coincide?
- Are non-harmonic timbres, such as drums, to be considered?

An example of such specification (in [Moorer 75]) has already been encountered – duets of 2 monophonic lines were considered, with restrictions on timbre and degree of harmonic overlap.

5.1.4 Derivation of Experiment

§5.1.1 described the form which, ideally, a set of test cases for an onset detection method should take, and we must now consider how a practical set of test cases is to be constructed. These should be manageable, whilst convincingly spanning a range of input types (in the context of the identified variables).

The composition of a single piece which could both be played on many instruments and include a range of possibilities for each of the variables specified would be very difficult. Even if this could be arranged, the change of a single variable whilst holding the others fixed would be difficult to arrange in many cases, as well as requiring a large number of recordings to be made. Finally, what constitutes the desired onset times would be difficult to control and specify, in the context of real performances.

The author has previously addressed these issues in [Tait & Findlay 95], which presented tests using an earlier incarnation of the onset detection method of chapter 4. The set of variables to be investigated was reduced to *interval*, *transition type*, *harmonic complexity*, *attack time*, and combined *offset time/degree of overlap*.

Further, the possible values for each of the variables was reduced. For example, the test piece was very simple, containing notes of the same length, a small number of intervals (including repeated notes) and two transition types (following immediately or with a gap). Two waveforms were used (square and sinusoidal), giving two extremes of harmonic complexity. These were recorded from a synthesiser, which allowed attack time to be varied precisely – nine combinations of three attack times (immediate, medium and very long) and three offset types (ending immediately, or overlapping by approximately one or two note lengths) were considered for each waveform. The notes were triggered from a MIDI sequencer, which would allow the desired onset times to be specified and compared with the results.

That experiment showed that detectable peaks could be produced, even in the presence of slow attacks and overlapping notes. However, it was lacking in its coverage of variables and, more importantly, in its timbral diversity. A set of instrument timbres has therefore been derived which spans the categorisation in the previous chapter. The sounds were again recorded from a synthesiser (so that actual onset times could be ascertained easily), and table 5.1 lists those used, along with their classification.

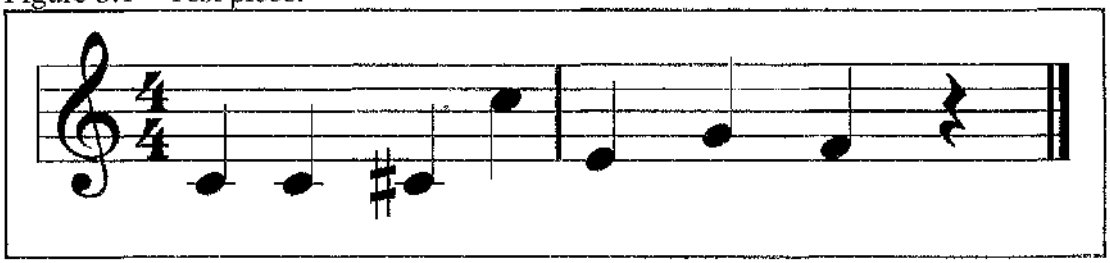
Table 5.1 – Test case timbres.

Instrument	Classification
Piano	Chordophone, struck
Cello	Chordophone, bowed
Muted Guitar	Chordophone, plucked
Pipe Organ	Aerophone
Flute	Aerophone, blow hole
Saxophone	Aerophone, single reed
Oboe	Aerophone, double reed
French Horn	Aerophone, cup mouthpiece
Marimba	Idiophone
Steel Drum	Idiophone
Timpani	Membranophone
Distorted Guitar	Electrical/Mechanical

A test piece was then designed, which included a range of different intervals (this is shown in Figure 5.1). Larger intervals are easily detected by the vector distance

method, so that the important ones to include are a repeated note and a single semitone (also, the method's symmetry means that identical ascending and descending intervals need not be included). Renditions of the piece were included for the drums in the above table, but a pattern consisting of more usual drum sounds was also used.
To investigate the effect of transition type, the piece was recorded in both staccato and legato style for each instrument (except the two idiophones, which did not produce long enough notes). These styles were imitated by making the notes half of their full length, and sustaining until the start of the following note, respectively.

Figure 5.1 – Test piece.



In order to limit the number of test cases, some of the variables were investigated by factoring them out from the main experiment. For example, note length is not of critical importance except when very short notes are considered. Therefore, rather than burden the experiment with different note lengths for each interval, transition style and instrument, a test was conducted to verify that the effect would be as expected for one instrument. This technique was also used to demonstrate the effect of frequency modulation (vibrato), amplitude modulation (tremolo), extremely low notes, dynamic variation, glissandi and reverberation.

5.1.5 Testing Other Analyses

Although we have focused on onset detection thus far, the principles of experimental design which have been derived ought to extend to other types of musical analyses (and other kinds of audio input). That said, the restrictions placed on the inputs will vary depending on the application. For example: in pitch recognition, whilst the same variables should almost certainly be investigated, there is little point in considering unpitched inputs.

5.2 Assessing the Results

Given a set of test cases, how the results are to be quantified and compared must be specified in advance. This involves establishing what the desired results are at the outset, so that the actual results may be assessed.

5.2.1 Evaluating an Onset Detection Method

Even for onset detection, the task of assessing results is not as simple as it might at first seem. For each input, the onset times which are subsequently to be automatically detected must be known. However, these may depend on the application into which the automated procedure is to be introduced. For example, a user of a graphical editing system will employ a combination of visual feedback, audition and experience in trimming some sound to the start of a note (for example). However, this may not coincide with the exact time at which a sound is *perceived* to commence. [Vos & Rasch 81] have investigated this idea, and it is also discussed in [Gordon 84]. Basically, there is a delay between the arrival of the first vibrations of a note, and its perception as a note onset. The quantification of this delay, and the derivation of an onset detection method based on its observation, is discussed in the above works. Such observations imply that, if the technique under consideration is intended to mimic human performance, then criteria must be used which allow fair comparison (taking any such factors into account).

It has been stated that precise replication of psychological phenomena is not the aim of the current work, so that the results of listening tests are not required for evaluation of the method. However, what will be viewed as the correct onset times must be established, as must some way of assessing the actual results. This is achieved by triggering the notes of the test piece from a MIDI sequencer, so that their onset times can be calculated given the tempo and observed time of the first onset (observing the first onset overcomes any consistent delay which may be present in the system). The individual and average differences between the expected and resulting onset times can then be derived.

5.2.2 Evaluating Other Analyses

Analogous complications will arise when assessing the results of other types of analyses. Pitch, for example, is very much a perceived phenomenon, so that knowledge of the pitch which would be perceived by a human listener must be employed when pitch recognition results are being evaluated. In addition, there are a number of examples of sounds which appear to have a pitch, but without the usual

harmonic structure. These were utilised in [Meddis & Hewitt 91], which describes a pitch recogniser producing the same results as a human listener would for several such cases.

5.3 Conclusions

It is felt that little consideration has thus far been given to the process of testing the various kinds of musical analyses which researchers have sought to automate. Whilst every possibility cannot be considered, there is sufficient structure in the domain of music that a set of test cases can be methodically constructed. The type of inputs spanned by the test cases can then be demonstrated, and the broader relevance of the results assessed.

This chapter has focused on how this process can be applied (in the context of onset detection) to the domain of monophonic musical inputs, and given indications of how it may be applied to evaluating other kinds of analyses.

The next chapter presents the results obtained using the onset detection method of chapter 4 on the test data developed in §5.1.1.

Chapter 6
Results

This chapter presents the results of applying the onset detection method of chapter 4 to the test cases described in §5.1.1. It should be noted that all of the test cases used are included on a CD which accompanies this thesis (Appendix F gives a track listing).

6.1 The Experimental Timbres

The test piece of Figure 5.1 was recorded at a tempo of 122 beats per minute, using voices from a Yamaha SY22 synthesiser, as listed in table 6.1. Where suitable presets were not available, custom voices were designed, the parameters of which are given in Appendix A. The results are presented in alphabetical order, in pairs of legato and staccato styles (where applicable). The staccato style involved notes of half a beat length, with gaps of the same length between, whereas the legato style notes were a full beat long. Some of these cases were unnaturally difficult for the analysis, since real instruments often require some kind of release before each new note and such seamless transitions would not be possible (especially in the case of repeated notes).

Table 6.1 – Test case voices.

Instrument	SY22 Voice
Cello	I5.1
Distorted Guitar	P1.4
Flute	P4.8
French Horn	P4.4
Marimba	P6.7
Muted Guitar	P5.4
Oboe	P5.1
Piano	P3.1
Pipe Organ	P4.1
Saxophone	P5.2
Steel Drum	C4.2
Timpani	I8.8

In addition, the other factors mentioned in the previous chapter were investigated using the test cases in table 6.2. Where possible, these were based on timbres from the main body (for ease of comparison).

Table 6.2 – Additional test cases.

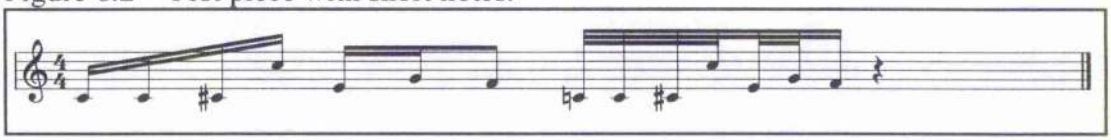
Test case	Voice	Comment
Reverberation	P3.1 (piano)	Inbuilt Rev Hall, Dep=6
Dynamic variation	P3.1	See Figure 6.1
Low notes	P3.1	2 and 3 octaves below original
Vibrato	P4.4 (horn)	
Tremolo	P4.8 (flute)	
Short notes	P3.1	See Figure 6.2
Drum pattern	P8.8	See Figure 6.3
Glissando	square wave	See Figure 6.4

Figure 6.1 shows how the MIDI velocities of the notes varied for the example of dynamic variation (the amplitude envelope is shown later). This was intended to be difficult, as the quieter notes following louder ones could easily be drowned out.

Figure 6.1 – Velocities for dynamic variation.

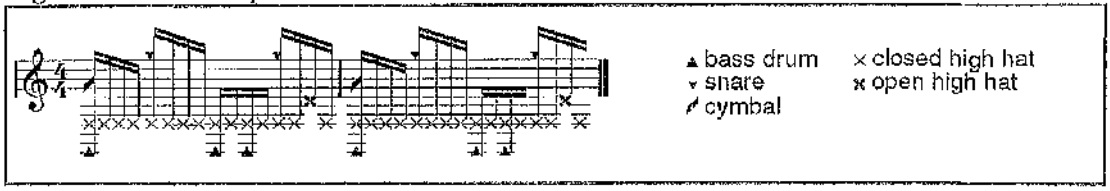
The score for the test case including shorter notes is given in figure 6.2. The window sizes and minimum length parameter in the analysis dictate the length of the shortest note which can be detected, and this test was intended to show that the results were as expected.

Figure 6.2 – Test piece with short notes.



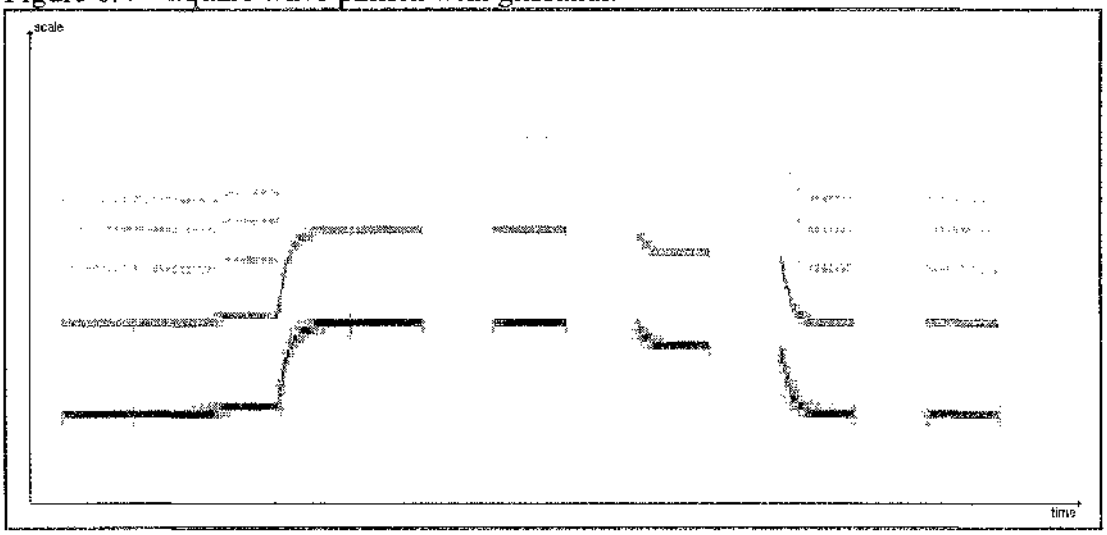
A drum machine pattern was also tested. Percussive timbres ought to be easily detected, but the example is not strictly monophonic and contains coincident onsets as well as cymbals which overlap several of the following notes.

Figure 6.3 – Drum pattern.



The synthesiser used for the other examples was not capable of introducing pitch glides between notes, so to investigate this a Novation BassStation was used with portamento added to a square wave timbre. As square waves were used previously in [Tait & Findlay 95], the test piece from that work was employed for comparison (without the pitch glides, the onsets can be successfully detected). Figure 6.4 shows a portion of the resulting modulus plane, which includes small and large glides between notes, as well as notes starting with glides. Although such large glissandi are not often encountered in music and the method under test in the current work is not particularly suited to detecting such transitions, any comprehensive programme of tests should include this type of example.

Figure 6.4 – Square wave pattern with glissandi.



In each case, what were to be regarded as the correct onset times had to be established so that the calculated onset times could be evaluated. This is not as straightforward as it may at first seem. In graphical editing, an amplitude waveform is used as a guide to zoom in to the onset location. However, this is virtually impossible unless a note is preceded by silence. In other situations, repeated audition of the slowed down audio is used to help pinpoint onset locations. In all such scenarios, the outcome is highly

subjective, and different degrees of precision are likely to be employed in different applications.

These problems were overcome by observing the location of the first onset (which is obviously preceded by silence), using the zoom facility in the application developed to carry out the tests (see Appendix D for a description). The point chosen was the first at which the amplitude began to rise from zero. Subsequent onset times were calculated from the first, using the tempo and note lengths (the examples were recorded using a MIDI sequencer so that timing was well known).

6.2 The Methods Tested

In chapter 3, a number of transformations of the modulus plane were introduced, and in chapter 4 a parameterised onset detection method was described. In order to assess the effect of the variables involved, each test timbre was analysed using a wide ranging set of different transformations and parameter settings. Before describing this process in detail, another technique applied to the modulus plane must be introduced.

6.2.1 Adaptive Normalisation

When conducting the tests described in [Tait & Findlay 95], it became apparent that slowly rising amplitude envelopes produced much smaller peaks of vector distance than rapidly rising ones. Although this is to be expected (and peaks were still produced), the subsequent classification of those peaks as onsets was also more difficult. In attempt to overcome this problem, a technique which we will call adaptive normalisation was developed.

The idea is to compensate for slowly rising envelopes by scanning the modulus plane along the time axis, and scaling the modulus values to ensure that at least one is always equal to the maximum possible (in other words, there is some modulus value which will be plotted as black at each point in time). Of course, periods of silence must be skipped so a small threshold is employed.

A detailed description of the algorithm is given in appendix B, but the results are illustrated here. Figure 6.5 shows the modulus plane of a sine wave with a rise time of approximately 1 second (it was the test case with the poorest result in [Tait & Findlay 95]). The vector distance function (calculated with i=1, l=r=100, p=2 and no root applied) is overlaid, and has negligible amplitude at onsets.

Figure 6.6 illustrates the effect of applying adaptive normalisation to the modulus plane of figure 6.5. Easily detectable peaks are now present at onsets, although the very low amplitude at each note start means that a gap is left between adjacent notes, resulting in pairs of offset–onset peaks

Figure 6.5 – Slowly rising square wave and vector distance.

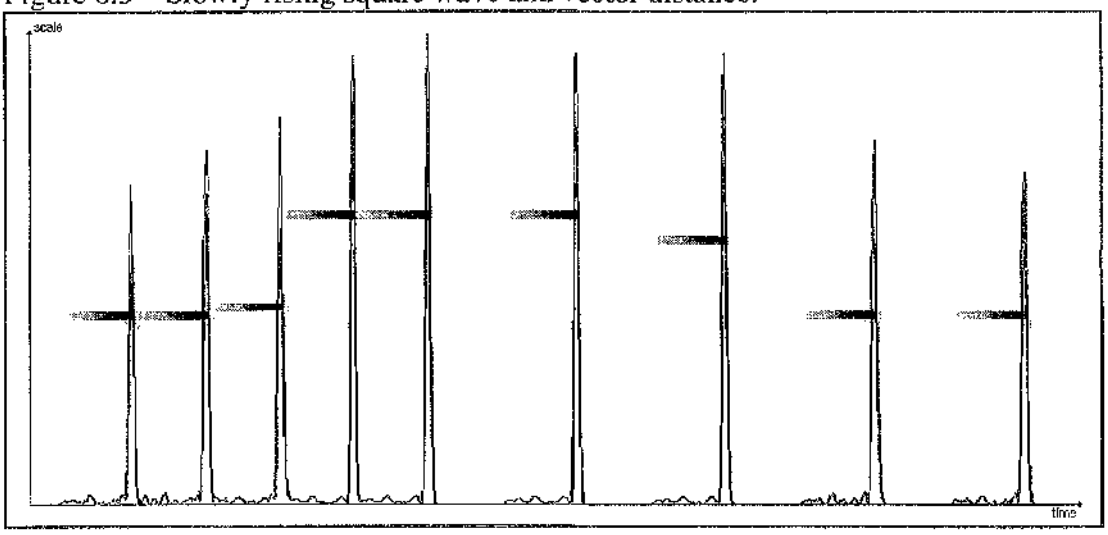
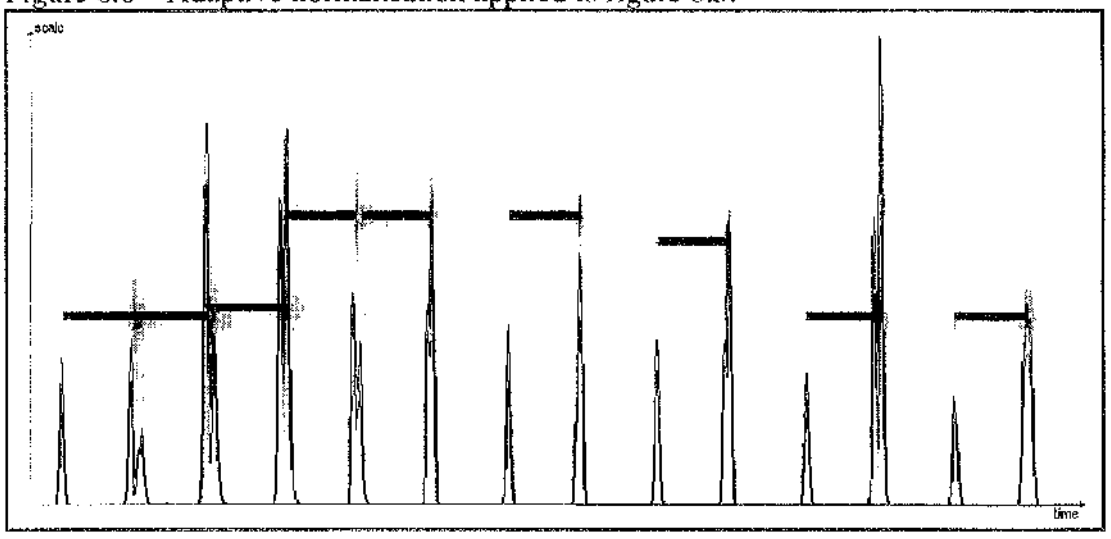


Figure 6.6 – Adaptive normalisation applied to figure 6.5.



It was thought that this process might improve results for instruments with slowly rising envelopes, so it was included as a part of the experiment. It is worth noting, however, that adaptive normalisation is not suitable for application to examples involving overlapping notes. In such cases, the overlap can dominate the new onset and the technique was not designed to cope with this type of example.

6.2.2 Method Parameters

A total of 20 analysis methods were tried on each input. The variables investigated were as follows.

- Exponent – the value of p used in Minkowski's measure ranged between 1 and 5.
- Loudness scale – decibel and sone scales were compared.
- Adaptive normalisation – tests were conducted with and without this transformation applied.

All analyses included the application of the equal loudness weighting.
Thus, for each input sound file, 20 different versions of the vector distance function were calculated. Each of those was smoothed and a range of different parameter settings for the onset detection phase was attempted as follows.

- *change_thresh* – 50 to 95 in steps of 5
- *band_thresh* – 0 to 50 in steps of 5
- *num_partials* – 0 to 6

The vector distance calculation parameters were fixed at $l = 100$, $r = 100$, $i = 1$, and a smoothing window of 20 points was used. The peak detection parameters were fixed at *peak_thresh*=1, *min_len*=140, and *peak_reqd*=10. The value of *min_len* corresponds to a minimum note length of approximately 100 msecs, and the window sizes are therefore approximately 70 msecs.
The above parameters were used for all but a few of the additional test timbres, which it was expected would require alternative settings – these will be discussed when the results are presented.

6.3 Results

The results are presented in two sections: the main body (the staccato and legato versions of the timbres listed in table 6.1), and the additional tests of more unusual inputs (as listed in table 6.2).
In comparing results for a particular input, a set of detected onsets were judged to be better than another if: they included more onsets, or had the same number of onsets with fewer spurious detections, or had the same number of onsets with the same number of spurious detections and had a smaller average error when compared to the actual onset times.

6.3.1 Main Body of Tests

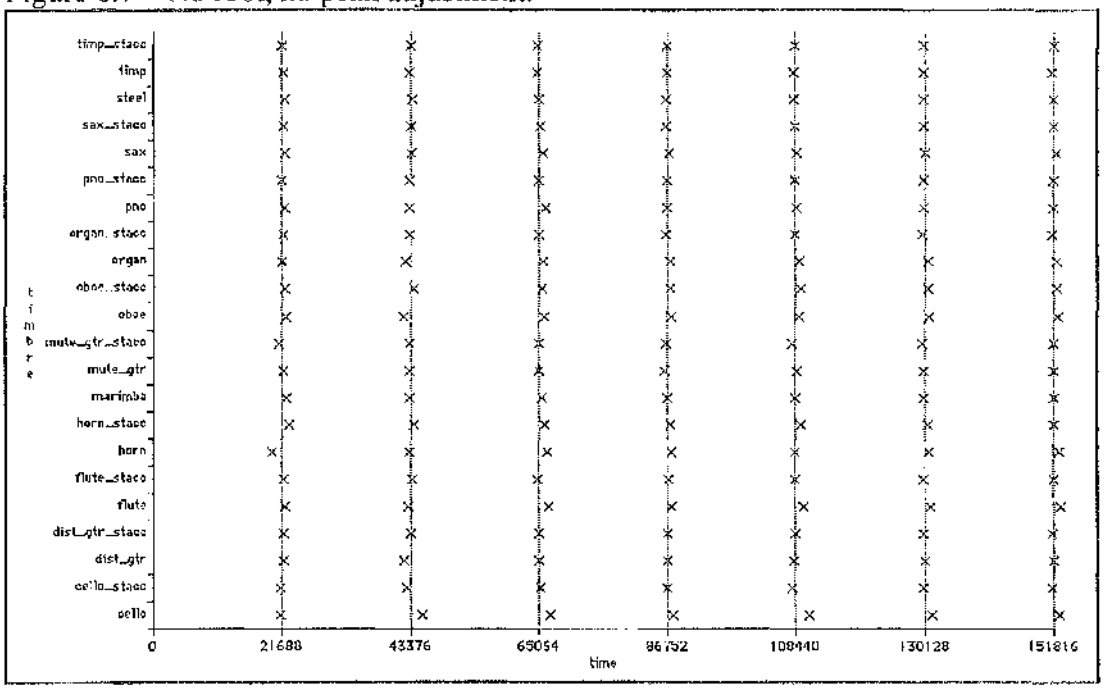
The main body of tests includes a range of different timbres and transitions types, but they are intended to be typical. In other words, highly unusual cases which might

skew the results are not included at this stage. This also allows us to consider how each of the analysis methods performed in the context of a set of typical examples.

6.3.1.1 Results by Timbre

Given the experiment described thus far, there are two issues still to be resolved. First, we would like to know the effect of applying the root in Minkowski's measure (previously, the root had not been applied, in order to emphasise peaks). Also, the effect of the adjustment to the peak positions (described in §4.5) should be examined. This was intended to compensate for the different starting times of harmonics, and it should be verified that better results are indeed obtained.

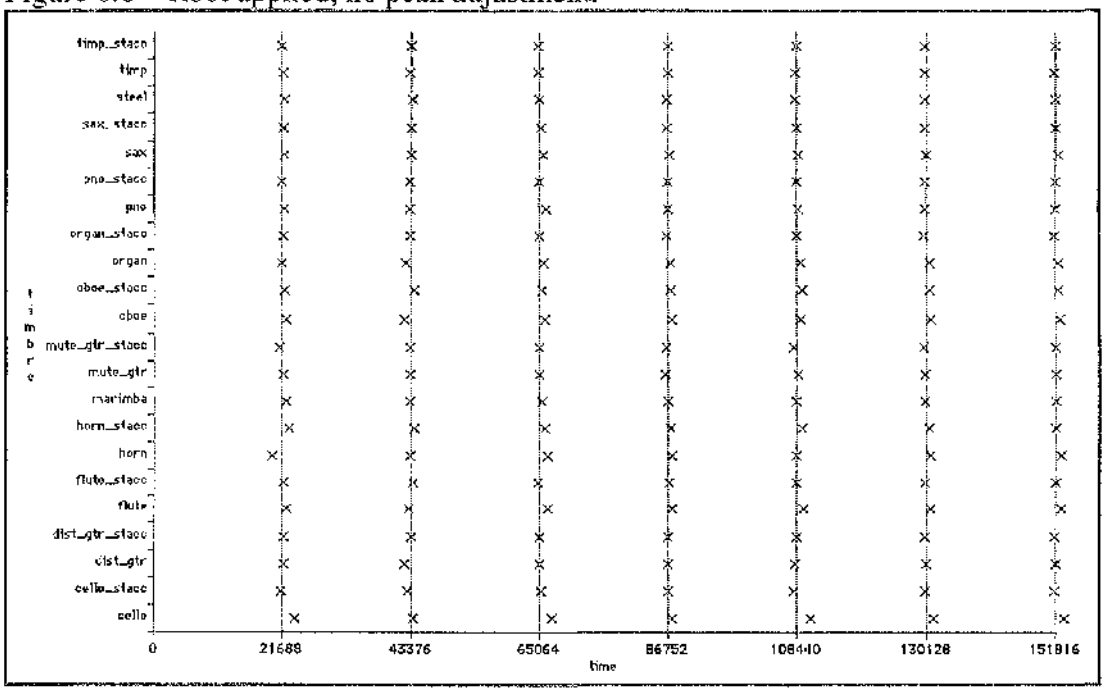
Figure 6.7 – No root, no peak adjustment.



In the first run, the root was not applied and peak adjustment was not used. Figure 6.7 shows a graph of the results (more detailed numerical summaries of the graphs in this section, including analysis parameters for each timbre, are given in appendix E). The vertical lines represent the actual onset times (of quarter notes at 122 bpm), and the horizontal rows of crosses represent the best results (across all 20 analyses and various onset detection parameters) calculated from the input sound files named on the left. These correspond to the timbres in table 6.1 (the suffix *_stacc* denotes a staccato example) and are ordered alphabetically from the x axis up. It can be seen that there are no omissions (a more detailed assessment of the accuracy is deferred until the next chapter).

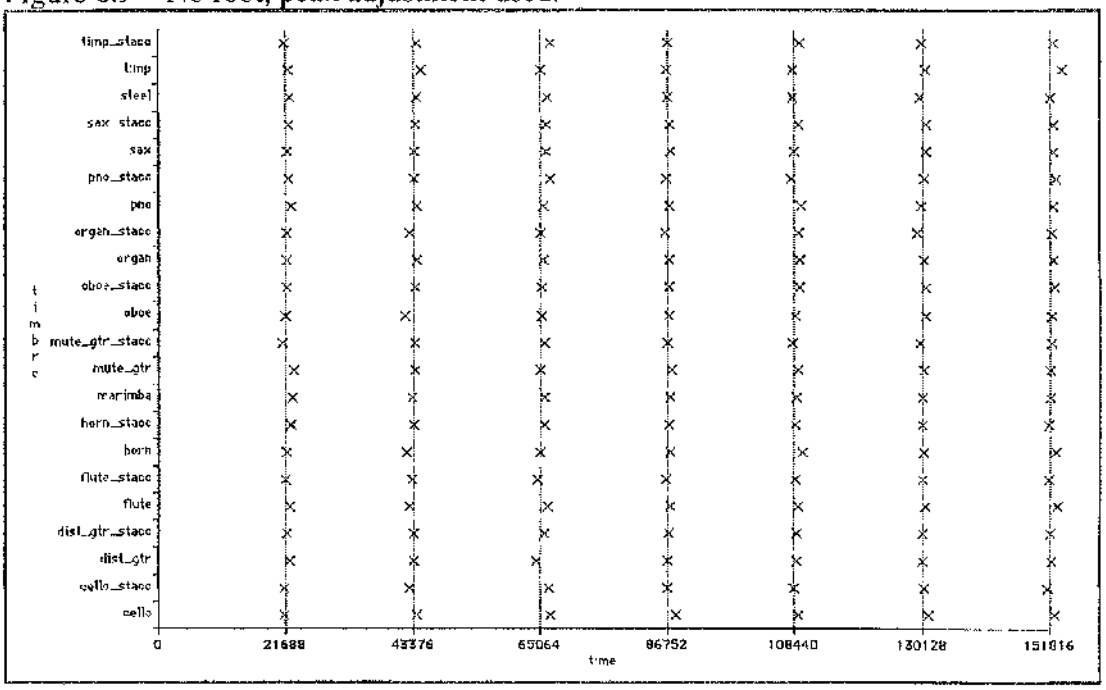
Figure 6.8 shows the results of the same experiment, but with the root applied in Minkowski's measure. Even a cursory visual inspection indicates that the pictures are almost identical and, in fact, the average difference (between actual and detected onset times) was only affected in two cases. The cello result was made approximately 1 msec worse, whereas the flute result was improved by around 3 msecs. Given that these were two of the poorest results already, these differences are not so significant and it was decided that the root would not be used. Of course, this also reduces the computation involved in the analysis.

Figure 6.8 – Root applied, no peak adjustment.



The whole experiment was then run again, with peak adjustment used, and the results are graphed in figure 6.9. Surprisingly, this did not improve the results, and the overall average difference was increased by around 2 msecs. However, there was no overall trend – some individual results were improved whilst others were made worse. Also, the same results were now obtained in a number of different analyses (whereas before there was always one best method). It was apparent that the use of peak adjustment dominates the analysis, so that the vector distance function now only points to a relevant vicinity in the modulus plane. In addition, the variable effect of applying peak adjustment seemed to indicate that the exact method used may need to be tailored to the pattern of harmonics at the note onset, for a given instrument. Further investigation of an onset detection method incorporating such knowledge was beyond the scope of this work, and the original method (with no root applied and no peak adjustment) was adopted for subsequent tests.

Figure 6.9 – No root, peak adjustment used.



Before considering the performance of each method, we give illustrations of the results obtained for each timbre (in the experiment of figure 6.7, with the methods employed as indicated in table E.1) . These consist of the transformed modulus plane with the vector distance function overlaid and vertical lines indicating the detected onset positions. The amplitude envelopes are also shown below, marked with the detected onset times. Detailed evaluation of these results will be undertaken in the next chapter.

The amplitude plots do not in general correspond to the same length of time as the modulus planes, due to zero padding and subsequent zooming-in on the modulus planes, so that corresponding vertical lines in the graph pairs are not always in alignment.

Also, due to the non–linear method used in plotting the amplitude envelopes (see appendix C), the highlighted points in time may occasionally correspond with observable onsets less well than they do when plotted on the modulus plane.

Figure 6.10 – Cello.

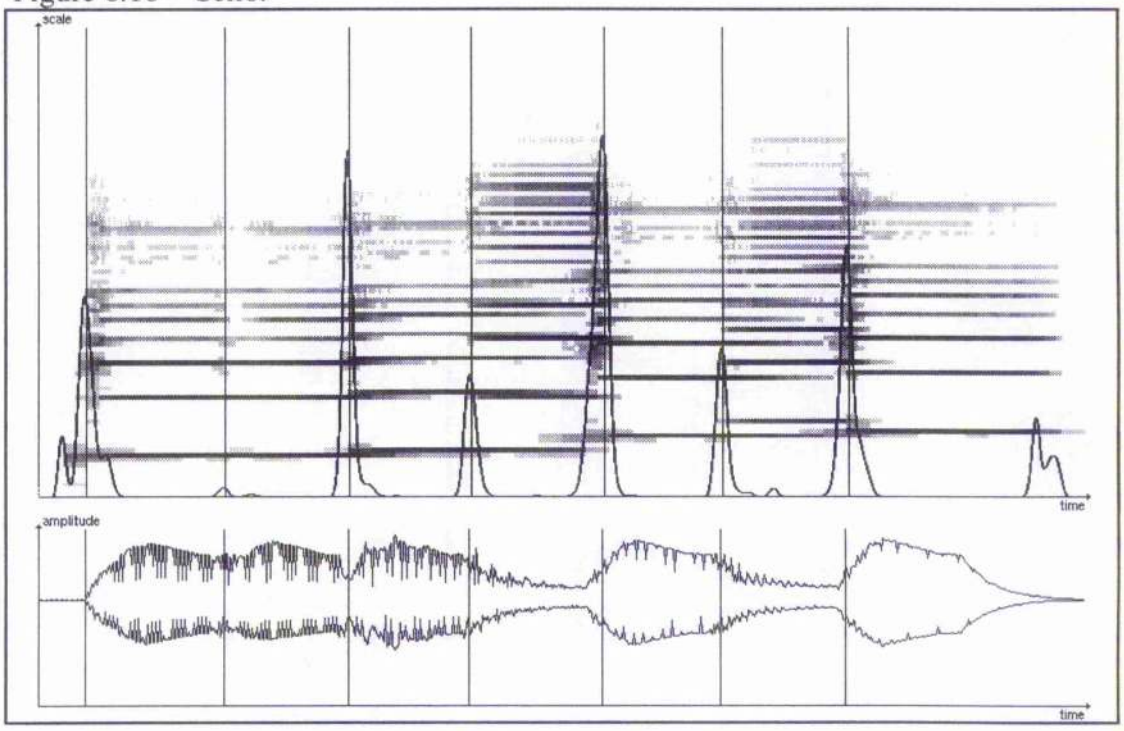


Figure 6.11 – Staccato cello.

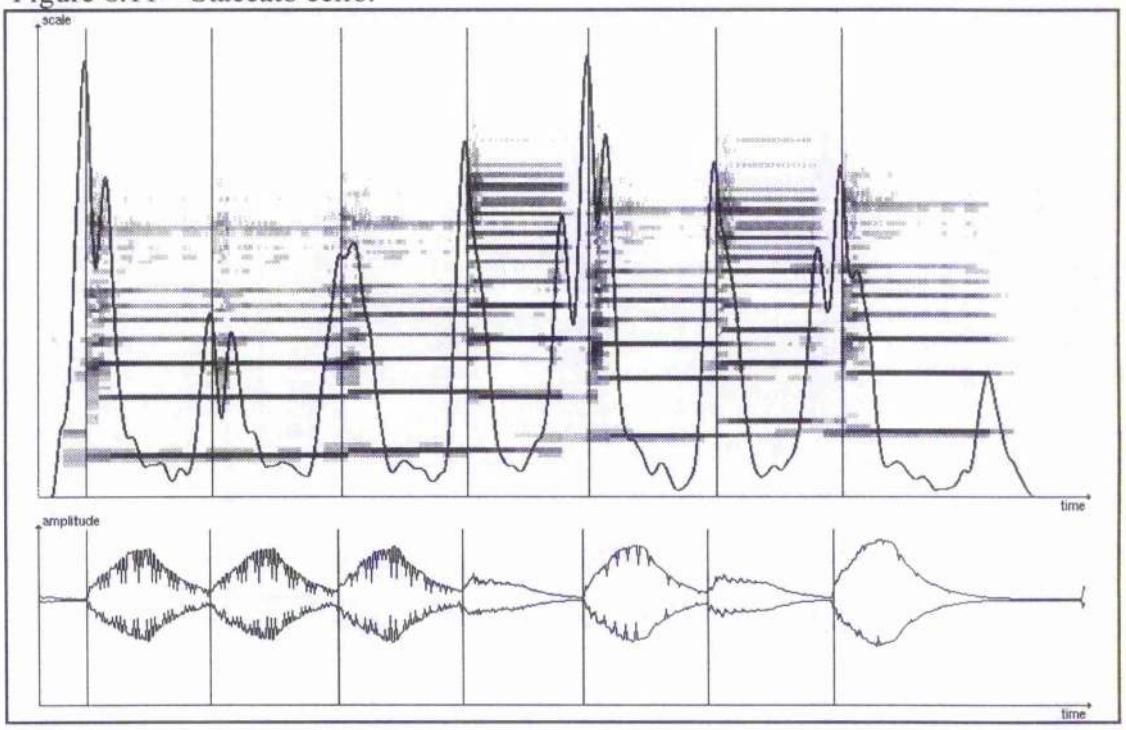


Figure 6.12 – Distorted guitar.

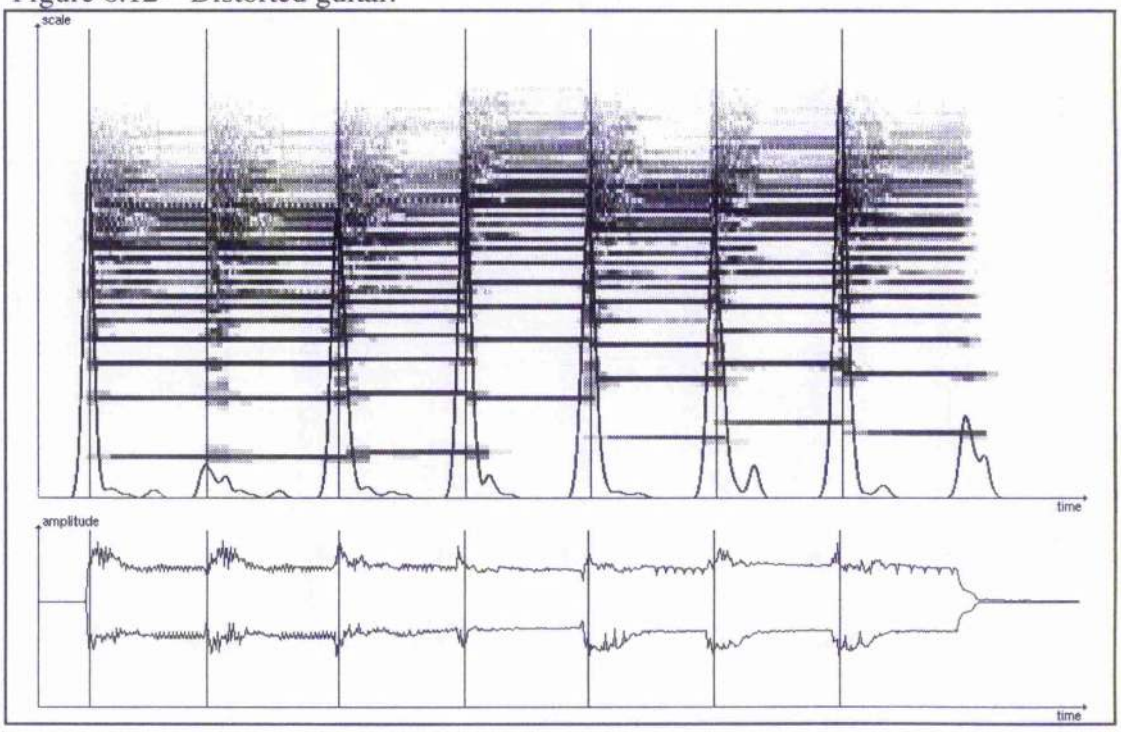


Figure 6.13 – Staccato distorted guitar.

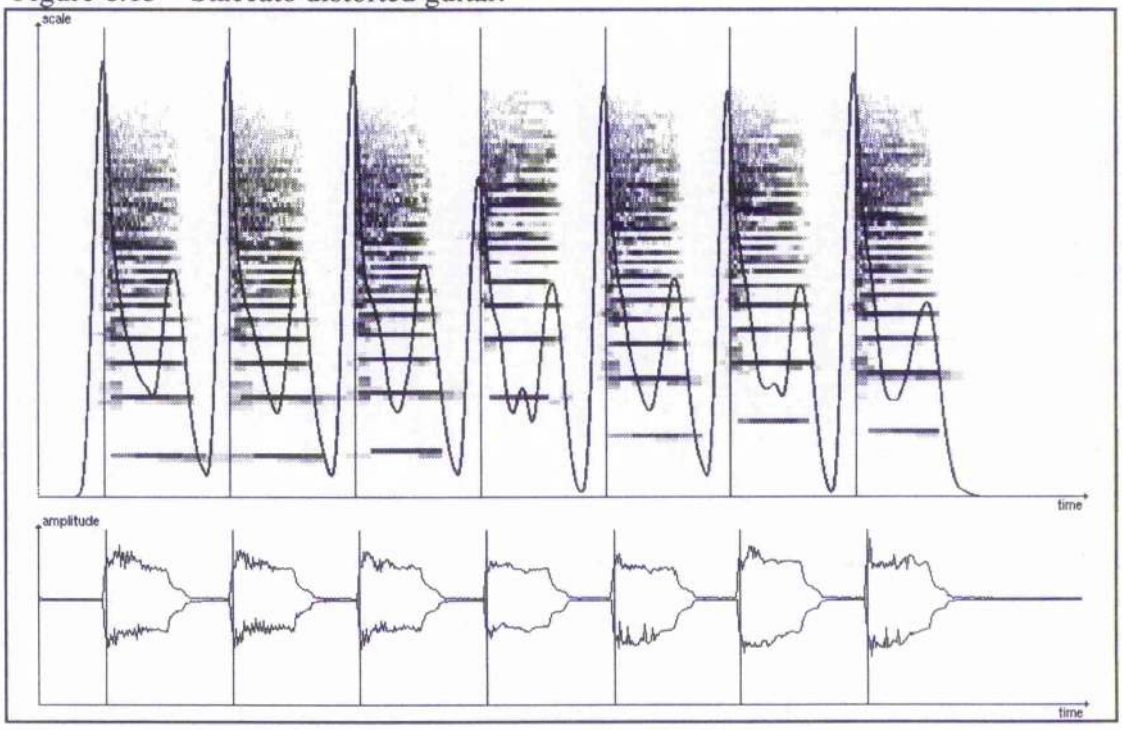


Figure 6.14 – Flute.

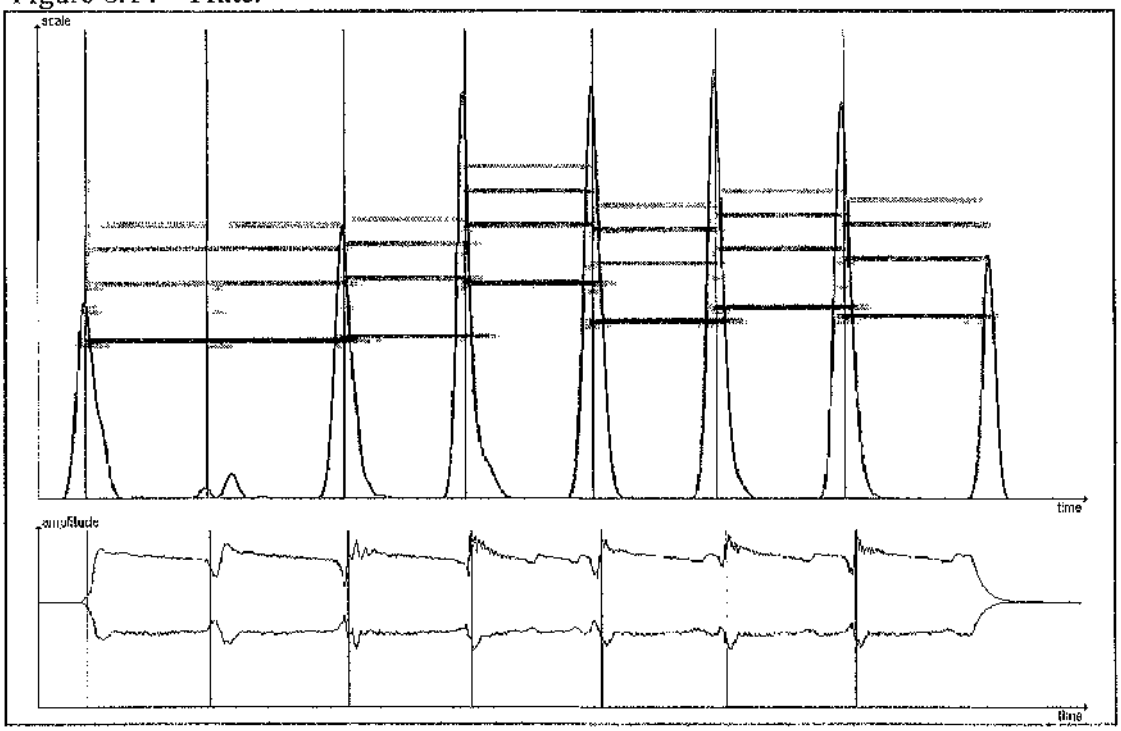


Figure 6.15 – Staccato flute.

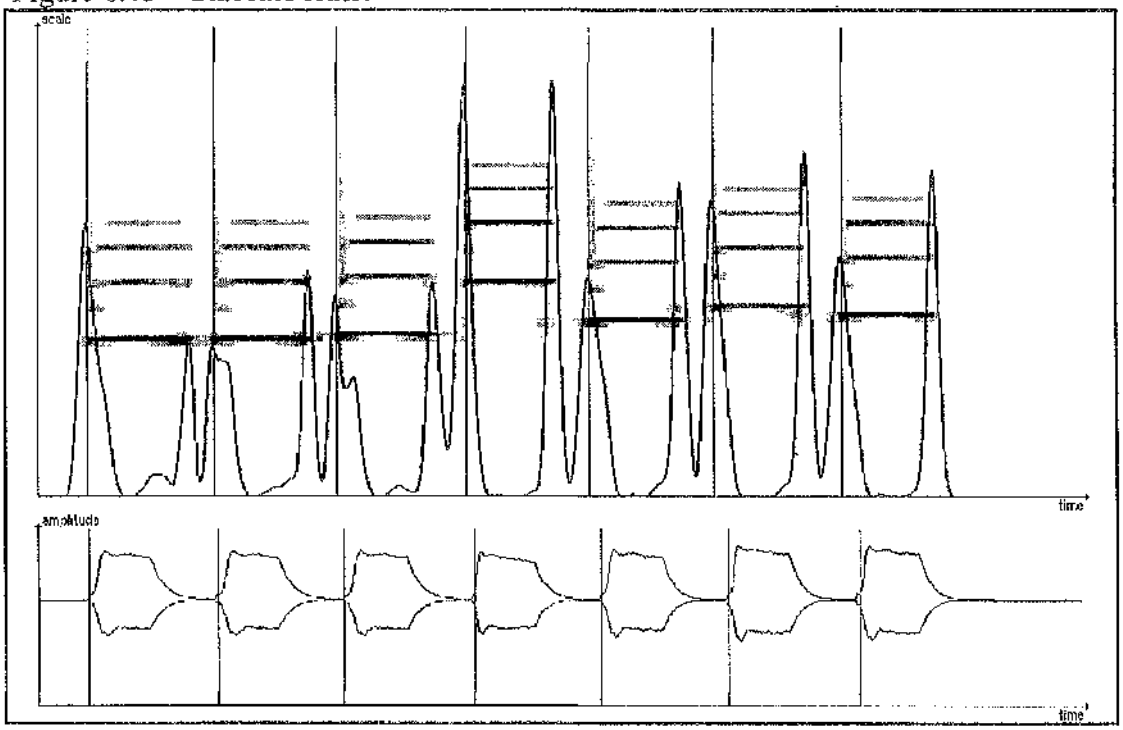


Figure 6.16 – French horn.

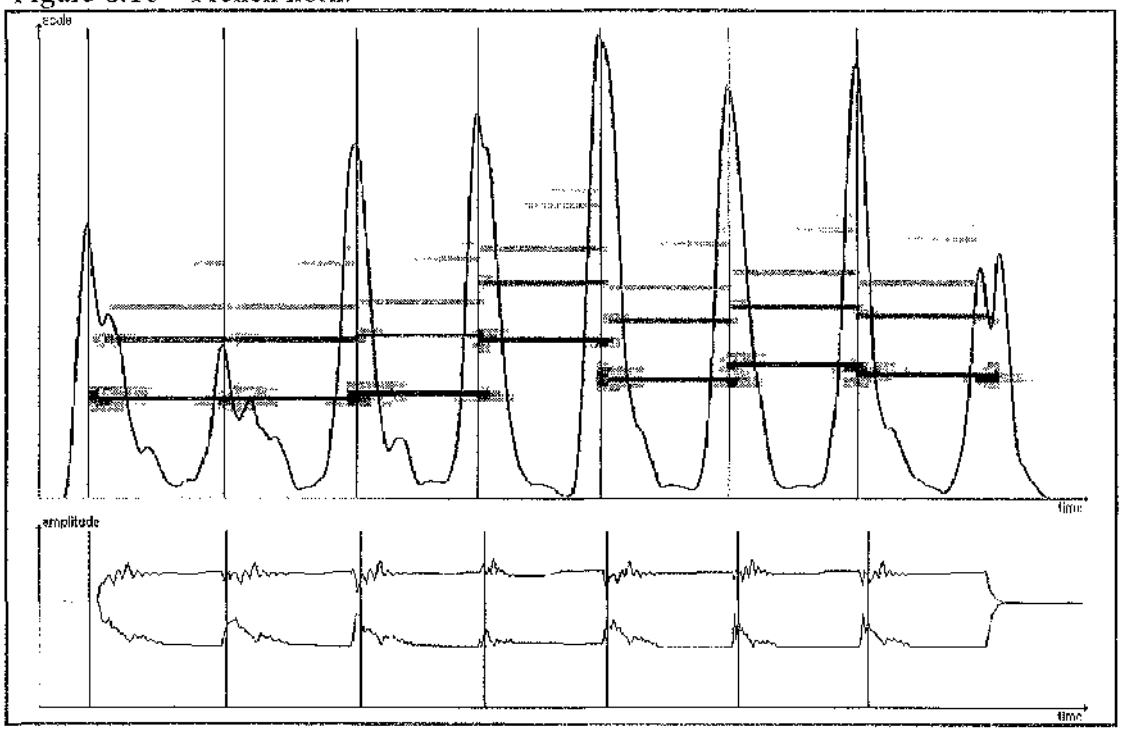


Figure 6.17 – Staccato french horn.

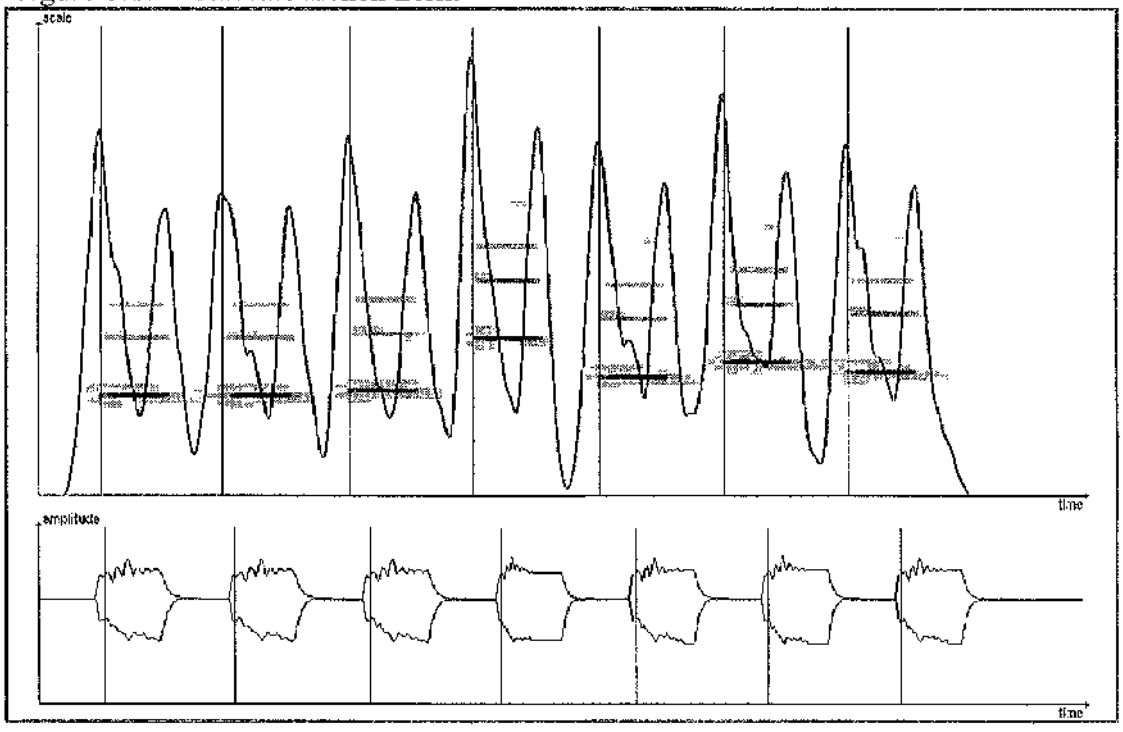


Figure 6.18 – Marimba.

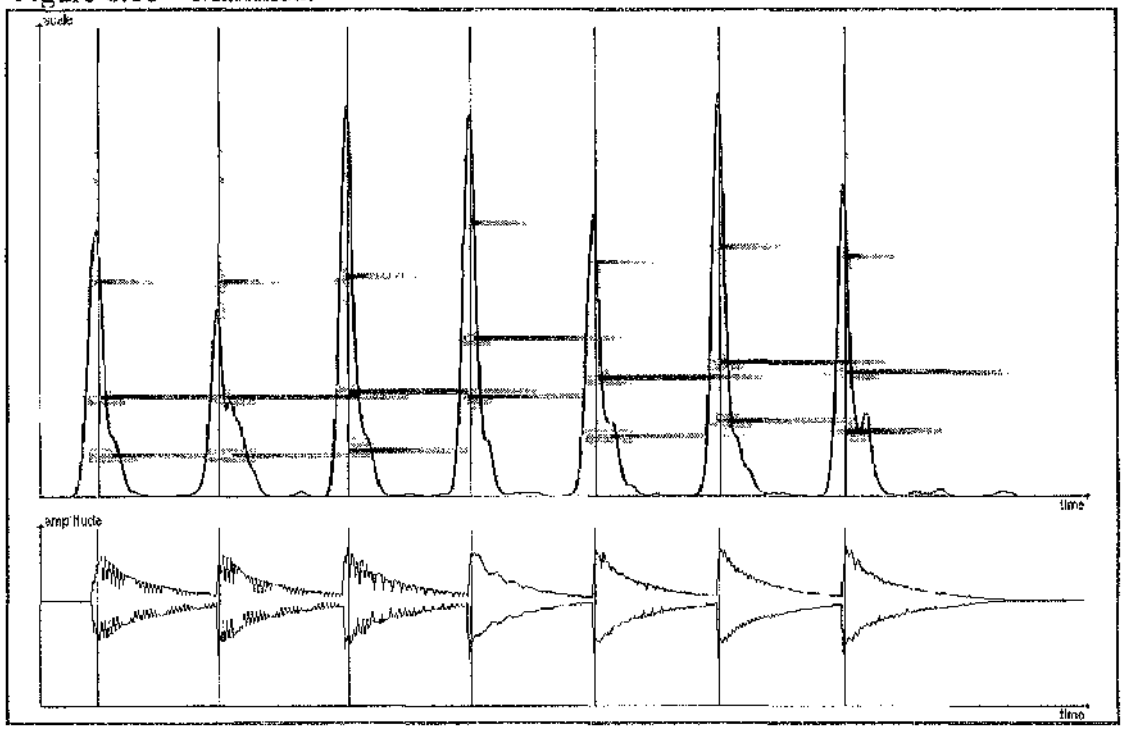


Figure 6.19 – Muted guitar.

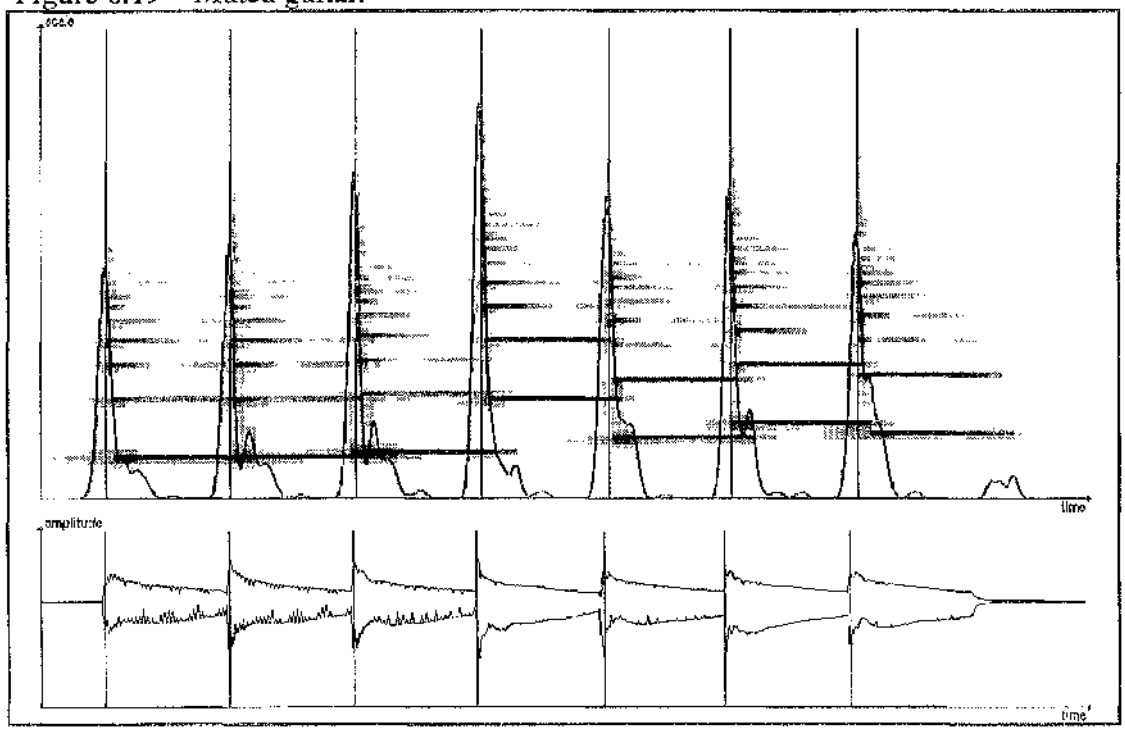


Figure 6.20 – Staccato muted guitar.

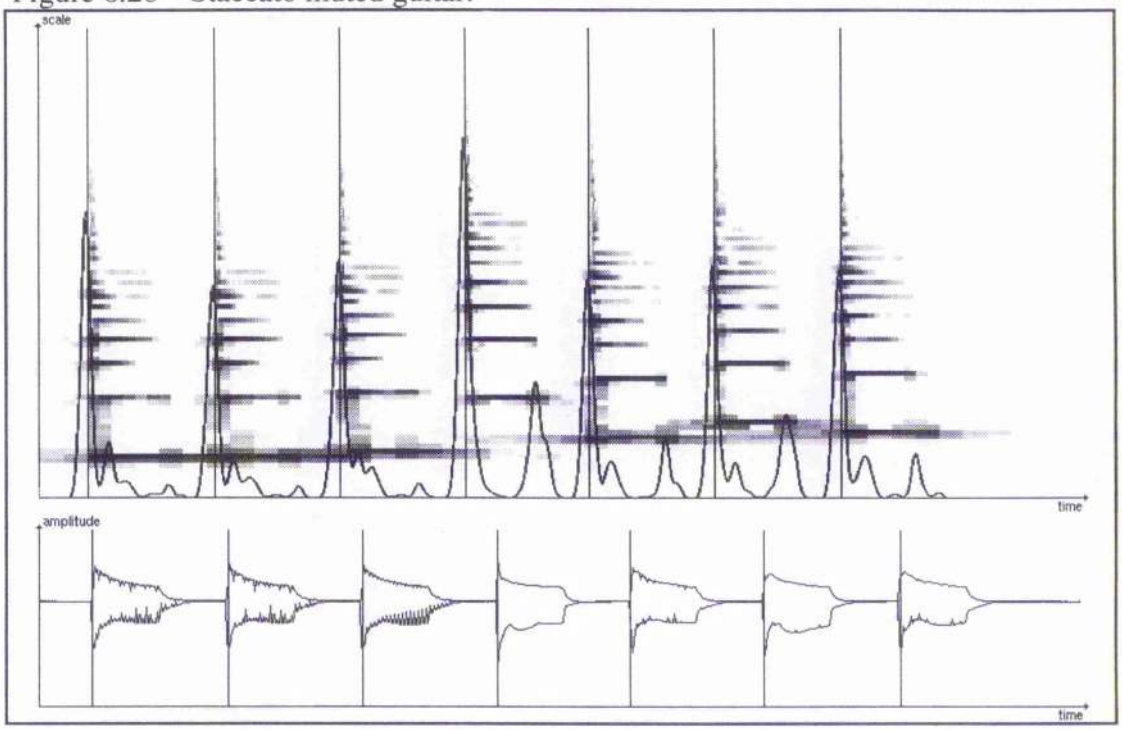


Figure 6.21 – Oboe.

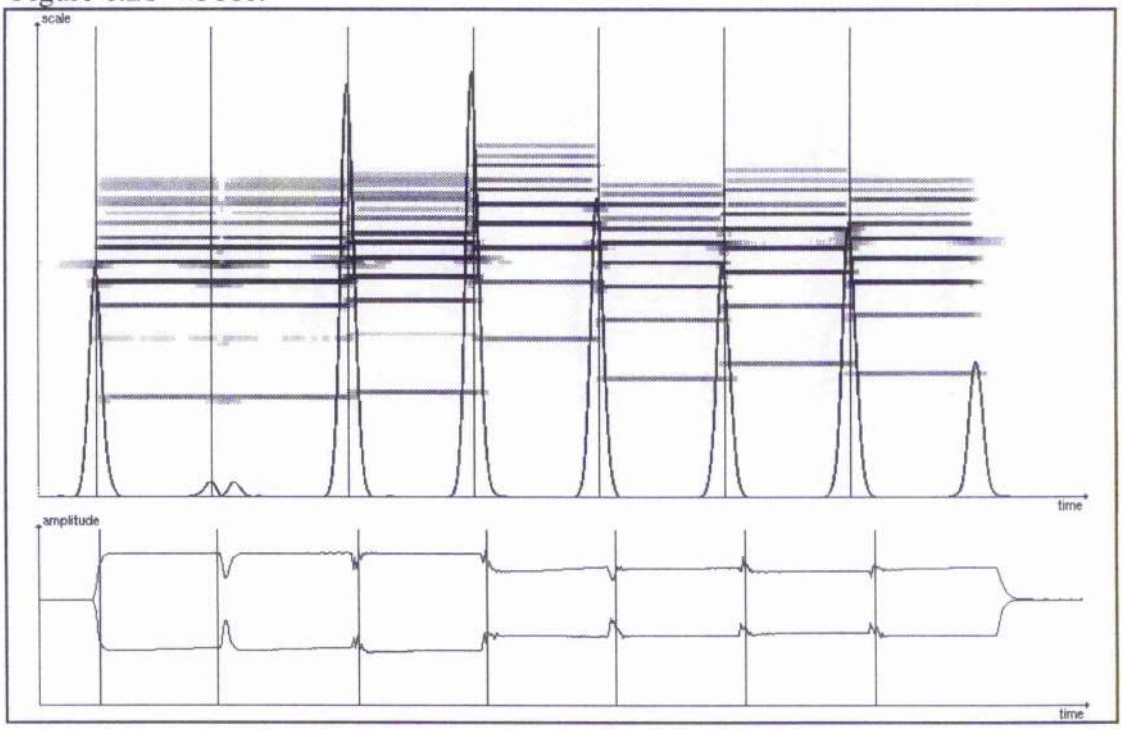


Figure 6.22 – Staccato oboe.

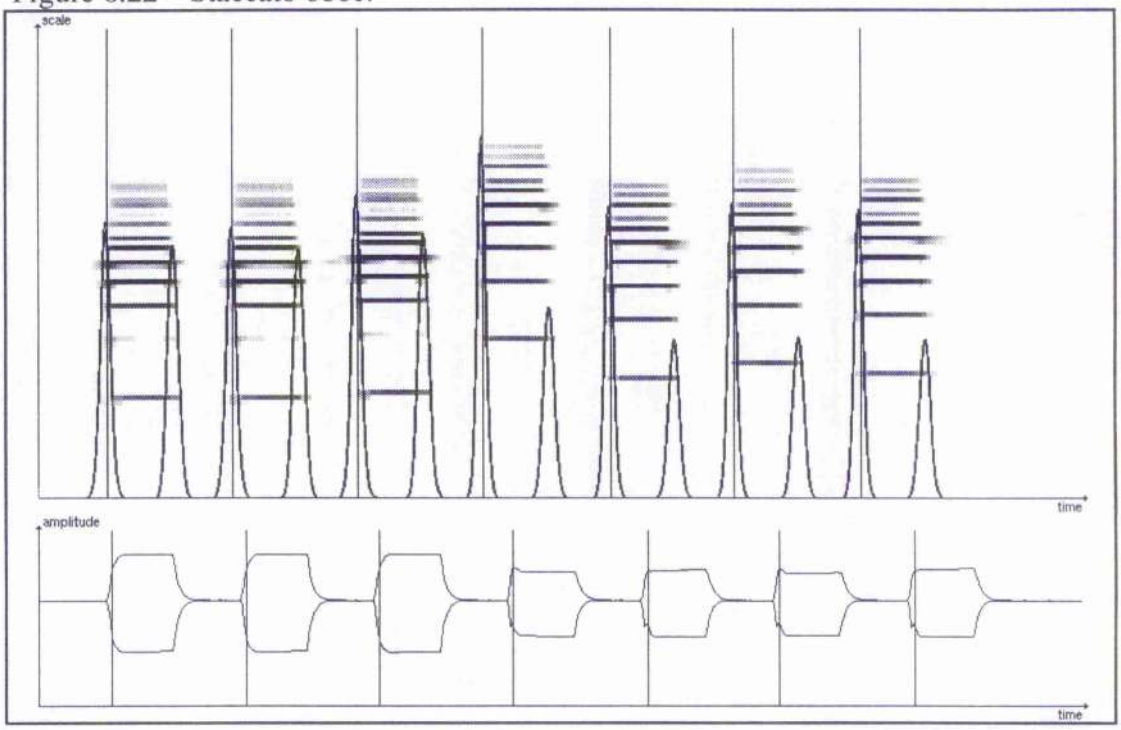


Figure 6.23 – Organ.

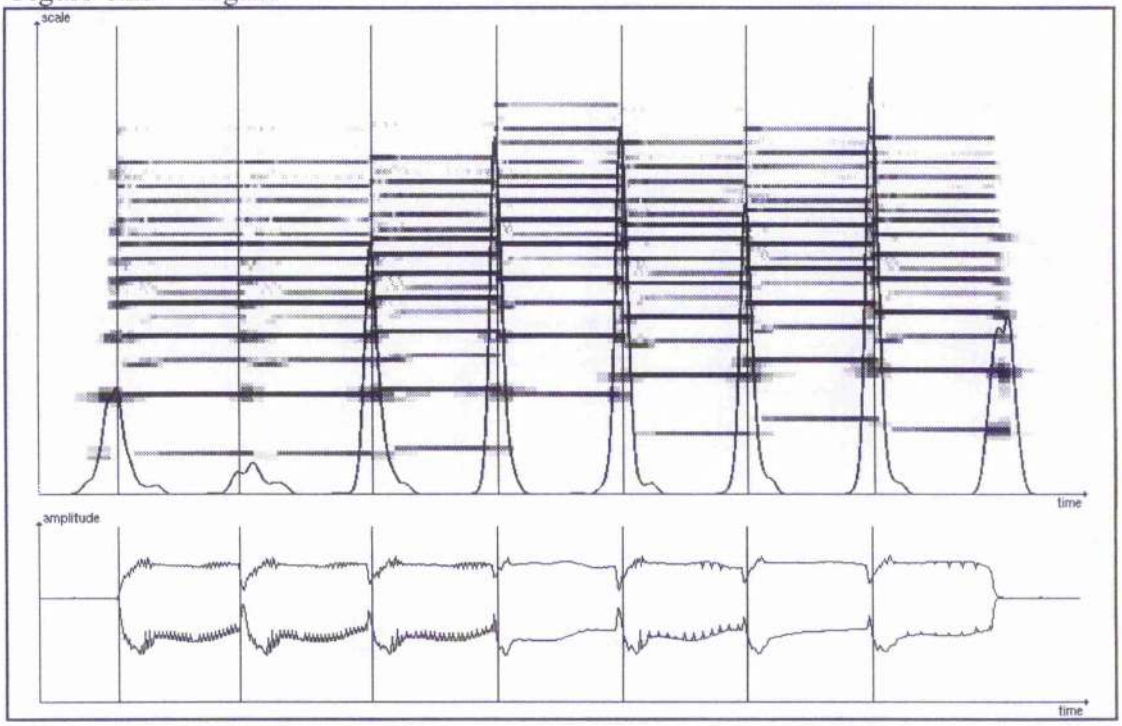


Figure 6.24 – Staccato organ.

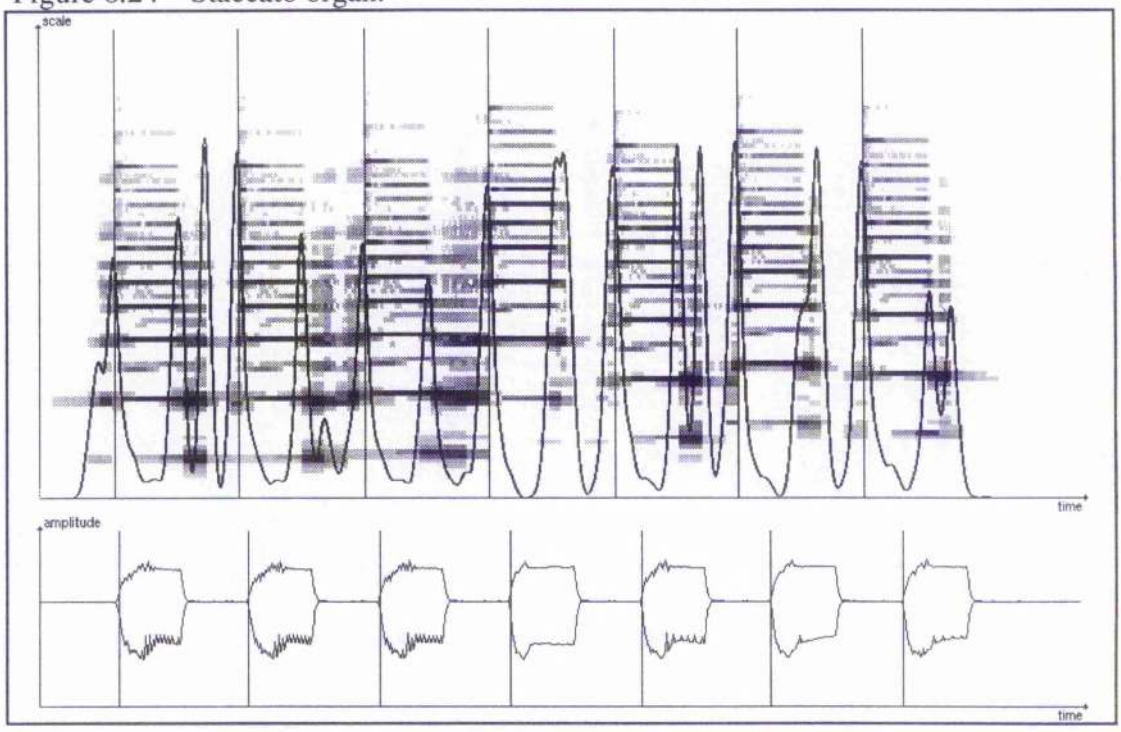


Figure 6.25 – Piano.

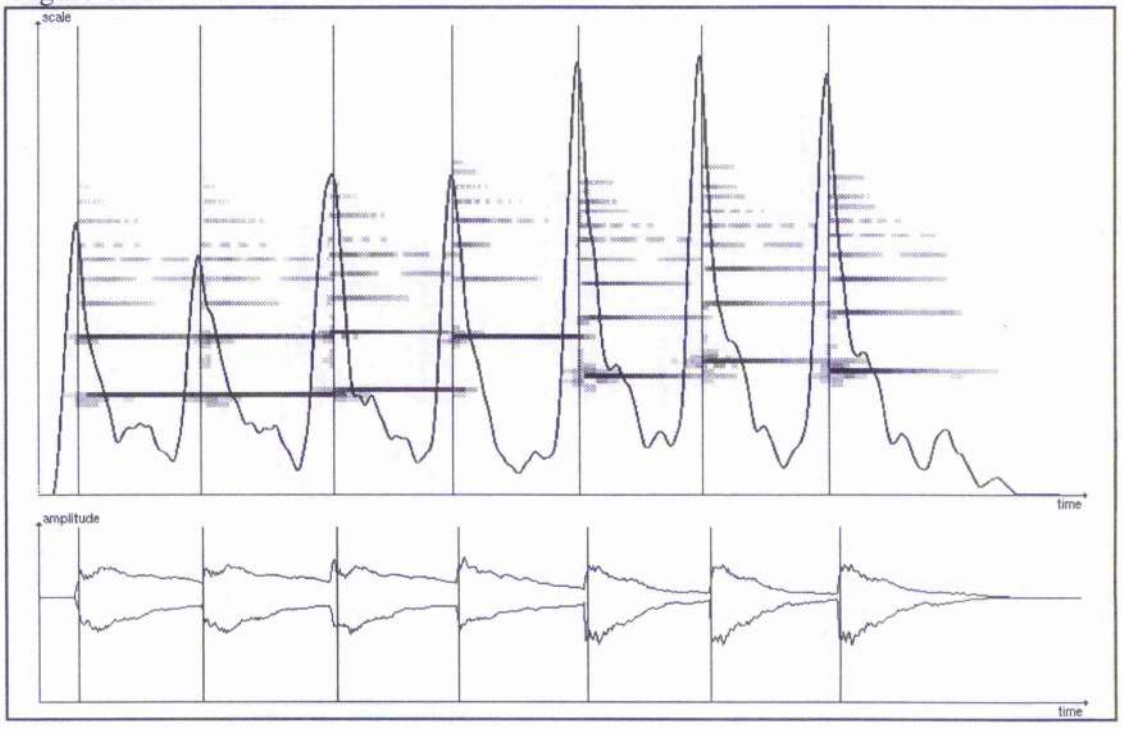


Figure 6.26 – Staccato piano.

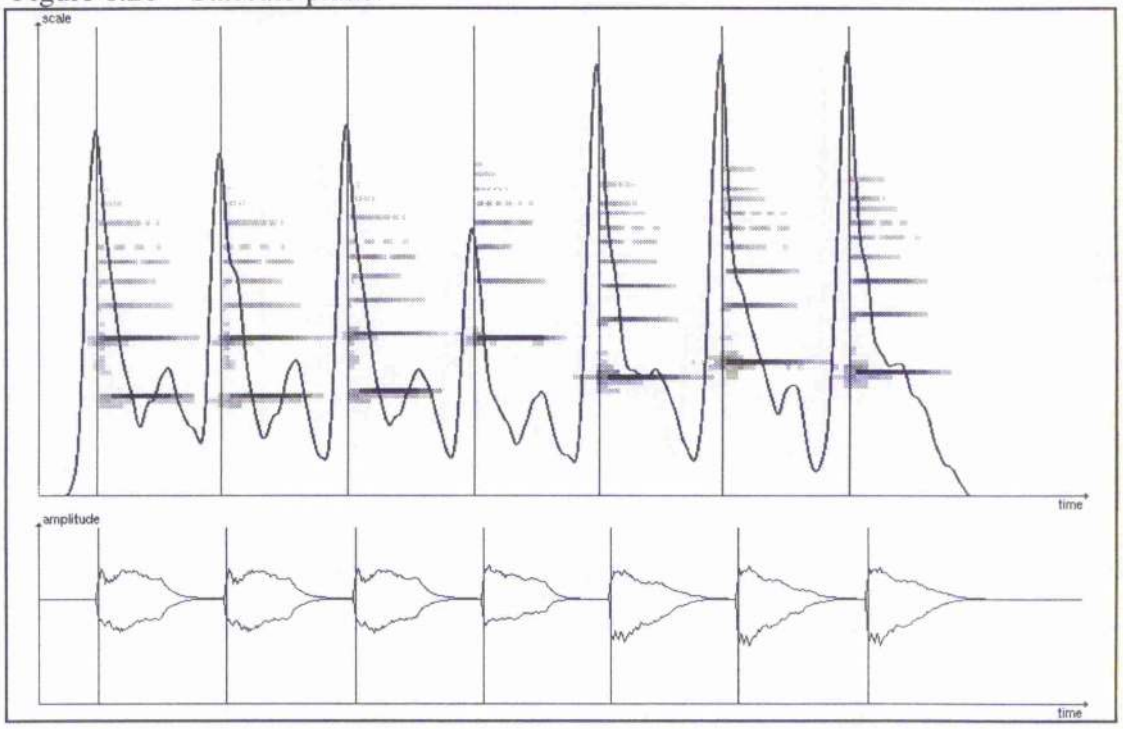


Figure 6.27 – Saxophone.

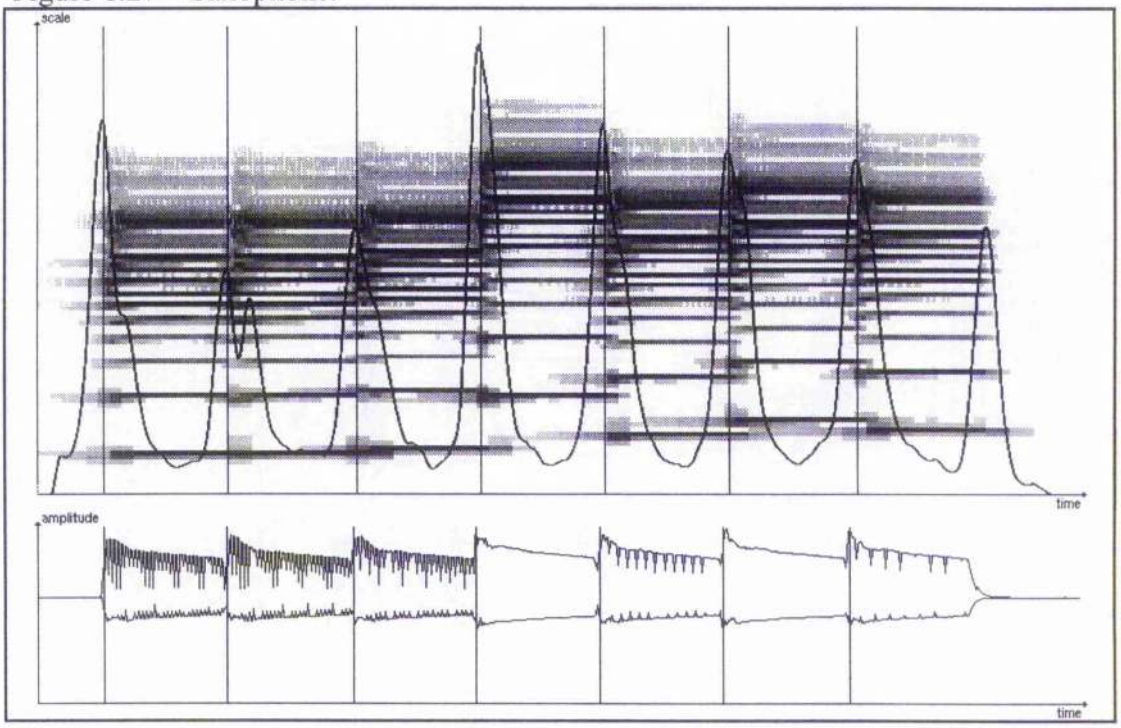


Figure 6.28 – Staccato saxophone.

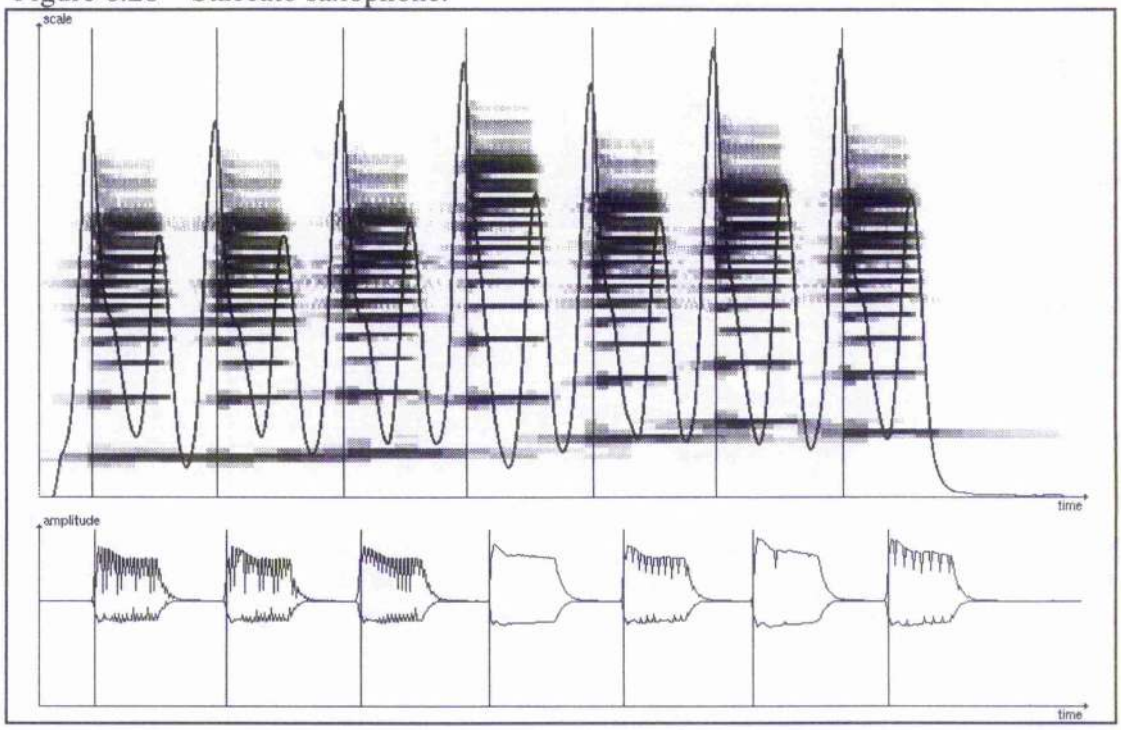


Figure 6.29 – Steel drum.

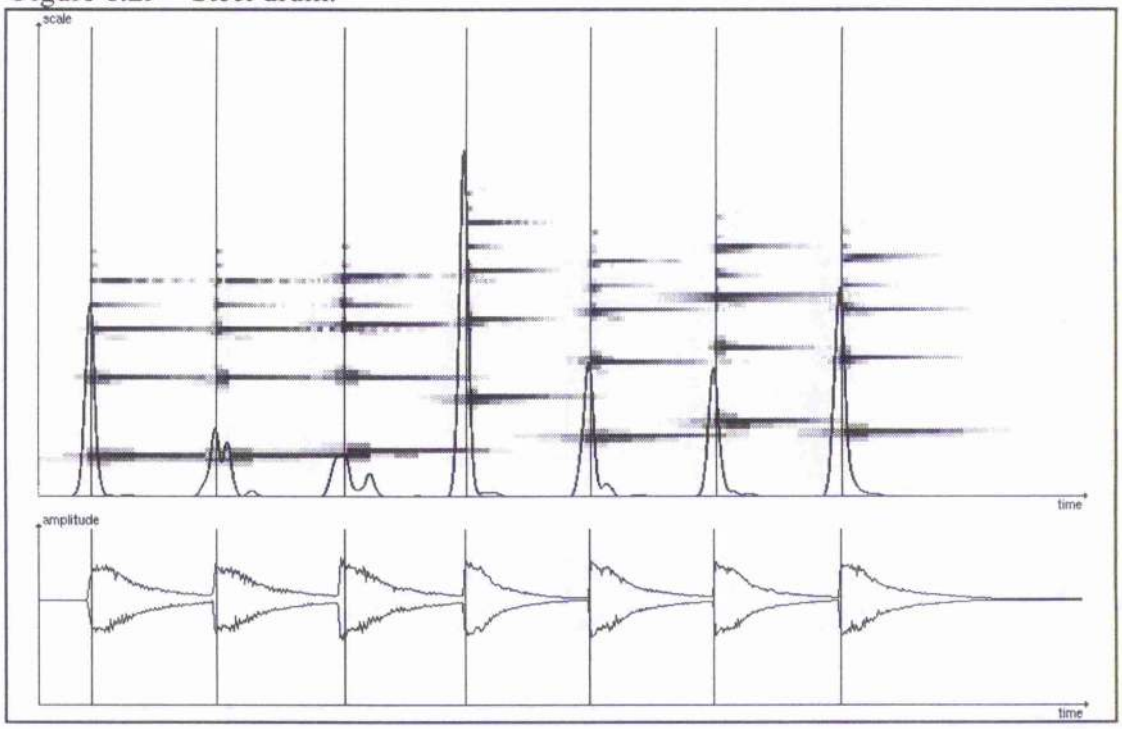


Figure 6.30 – Timpani.

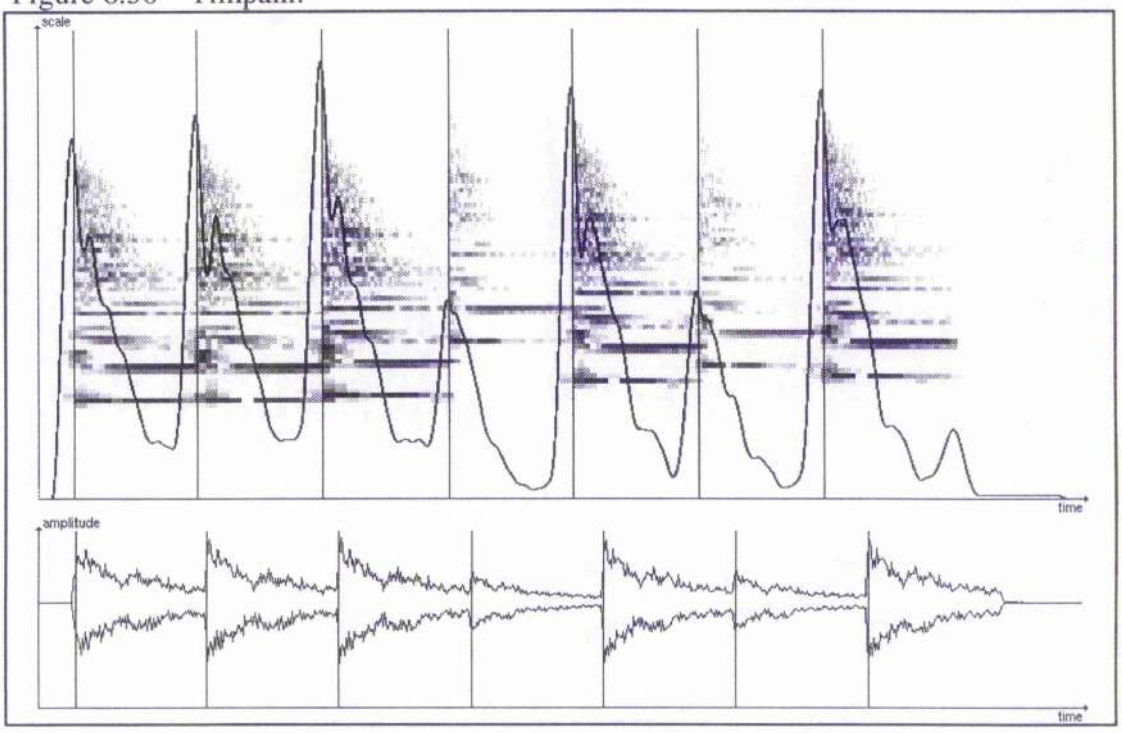
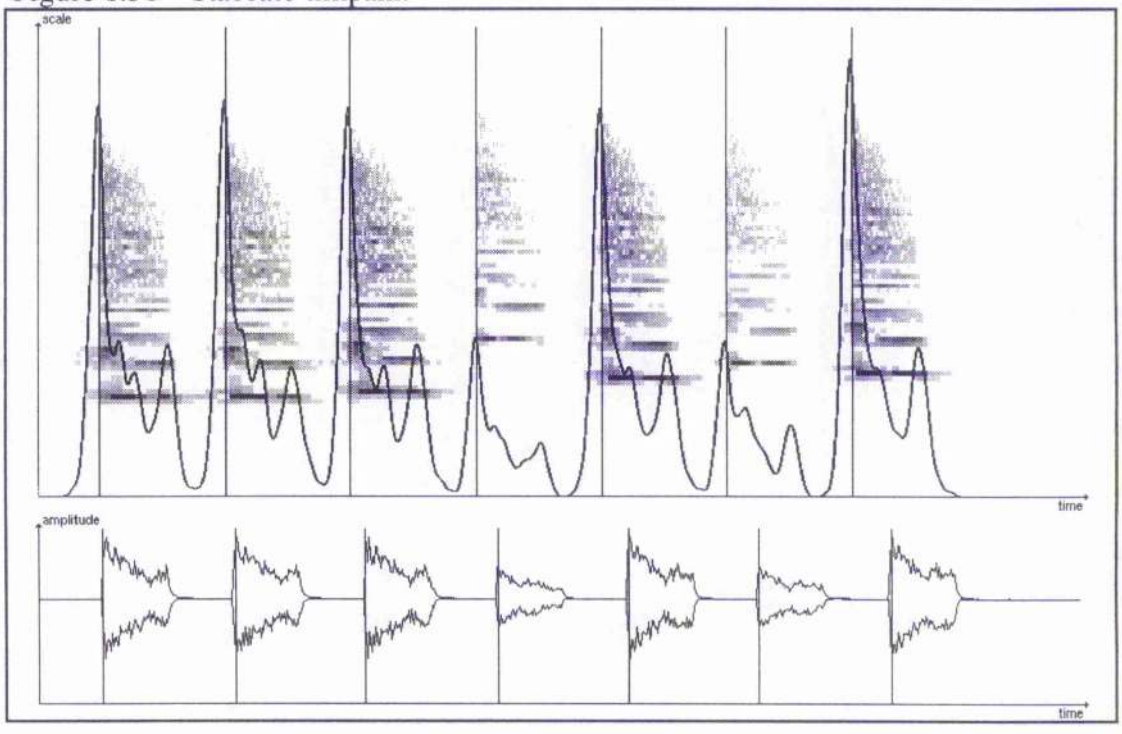


Figure 6.31 – Staccato timpani.



6.3.1.2 Results by Method

As the results in the previous section gave no clear indication of a single best method (this is evident in table E.1), we now consider the experiment in terms of how each analysis method performed.

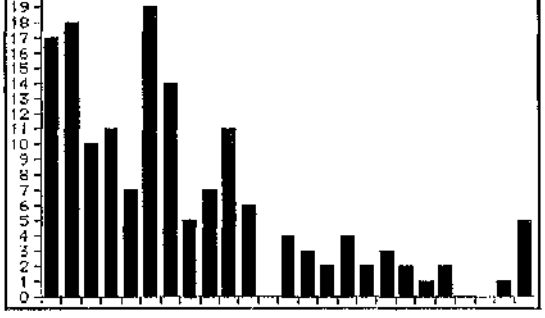
For each combination of exponent, loudness scale and application or non-application of adaptive normalisation, the distribution of the absolute values of the errors for all onsets was plotted. Numerical summaries of this information are given in tabular form in tables E.4 and E.5: including the average, minimum and maximum errors, the standard deviation of the errors, the number of onsets detected and the number of spurious detections.

The most striking feature of these results is that the application of adaptive normalisation does not, in general, improve the onset detection accuracy -- in fact, all of the results obtained using this technique are considerably worse, with many large errors. Therefore, although we have seen that it can be usefully applied in certain cases, adaptive normalisation (in its present form) should not become an integral part of the analysis (again, this is considered in more detail in the next chapter).

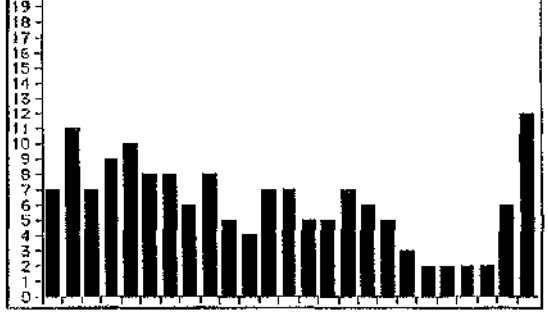
When the results obtained with each loudness scale are examined, it can be seen that there is not a marked difference.

Figure 6.32 -- Error distributions for each analysis method (the bins on the x axis cover 2.27 msecs each, except the last which includes all errors >54.42 msecs).

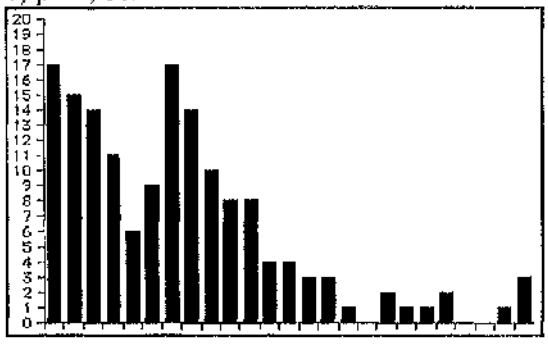
a) p=1, dB scale.



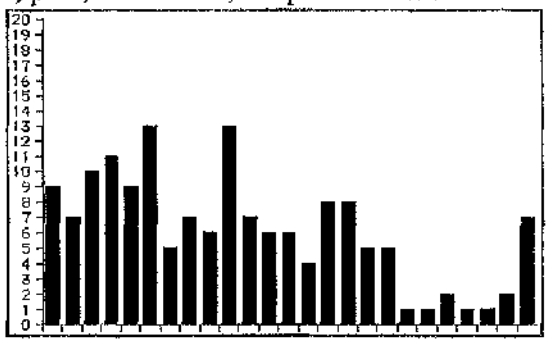
b) p=1, dB scale, adaptive normalisation.



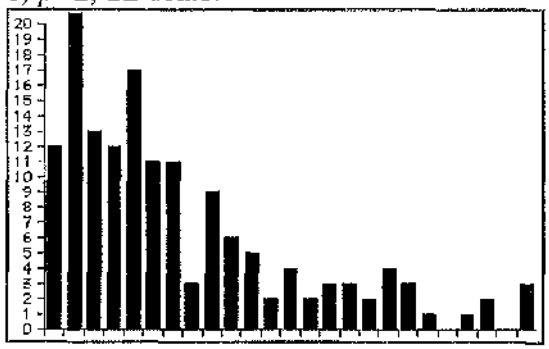
c) p=1, sone scale.



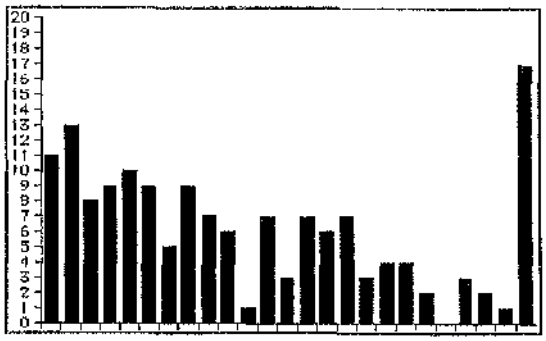
d) p=1, sone scale, adaptive normalisation.



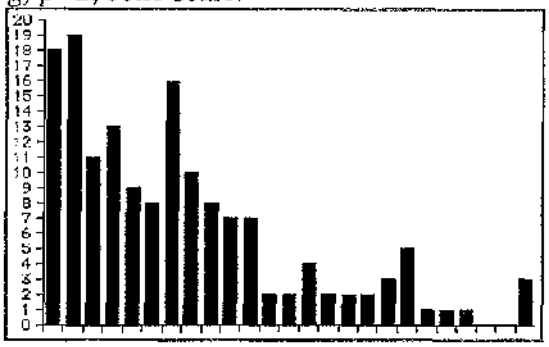
e) p=2, dB scale.



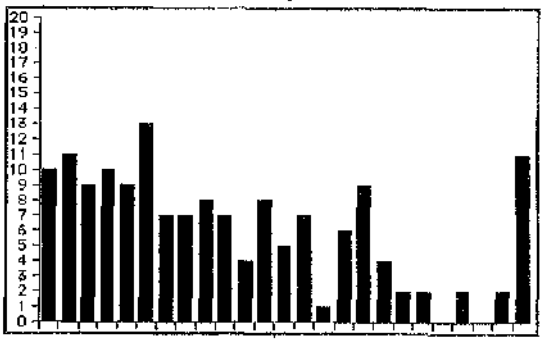
f) p=2, dB scale, adaptive normalisation.



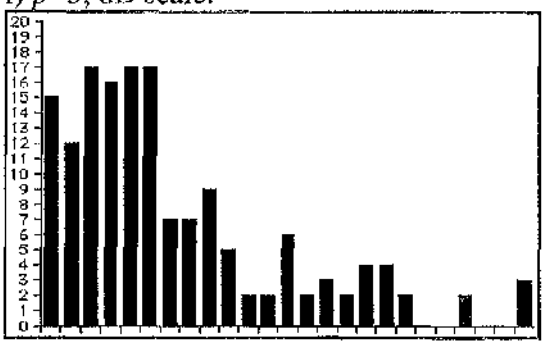
g) p=2, sone scale.



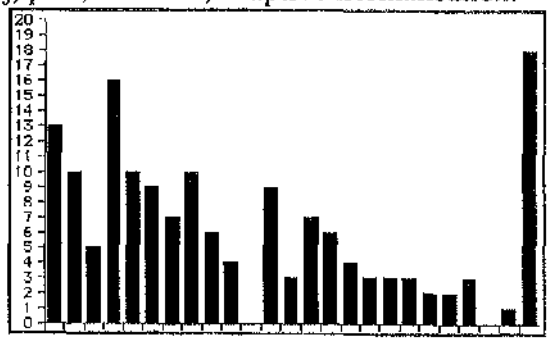
h) p=2, sone scale, adaptive normalisation.



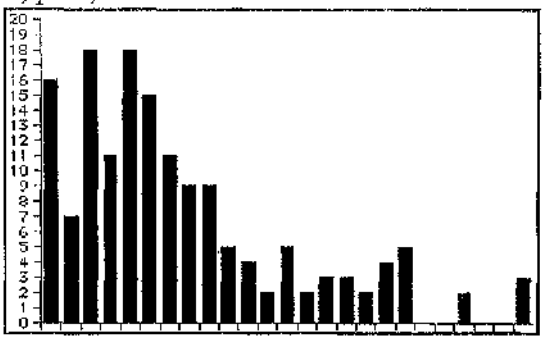
i) p=3, dB scale.



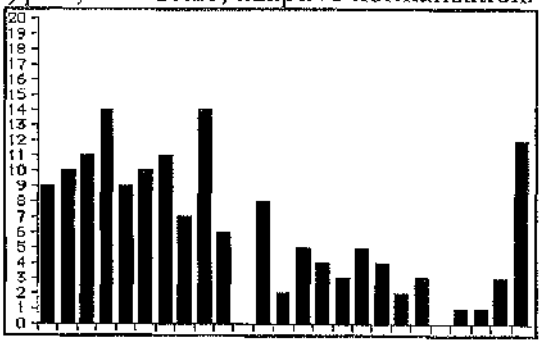
j) p=3, dB scale, adaptive normalisation.



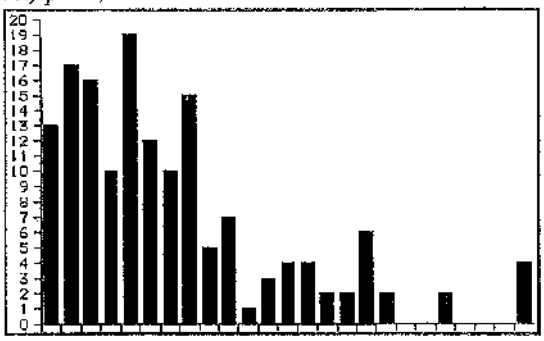
k) p=3, sone scale.



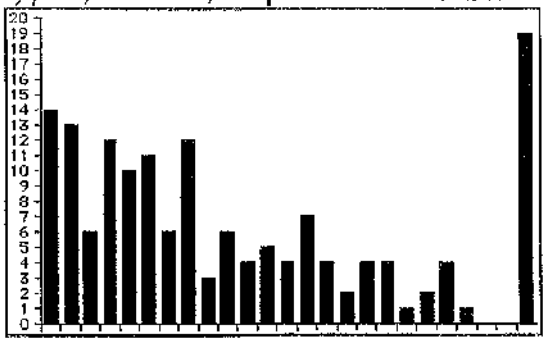
l) p=3, sone scale, adaptive normalisation.



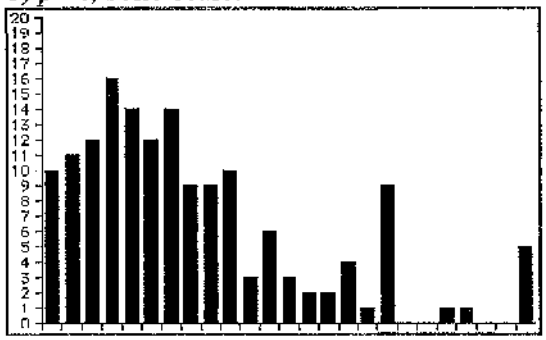
m) p=4, dB scale.



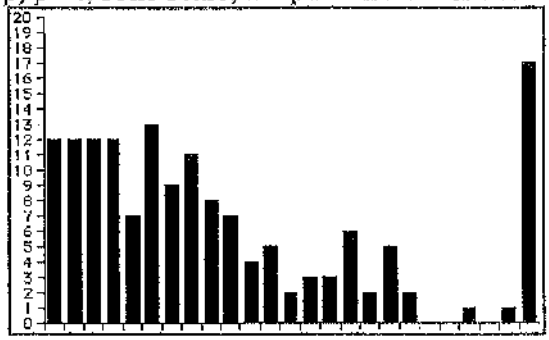
n) p=4, dB scale, adaptive normalisation.



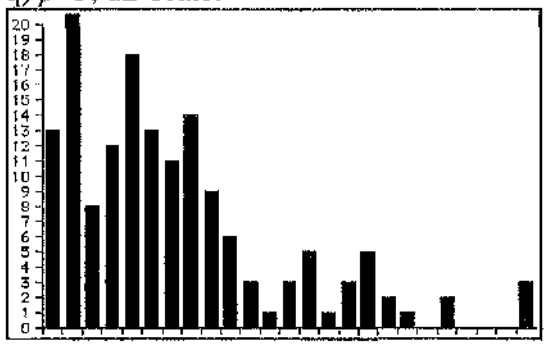
o) p=4, sone scale.



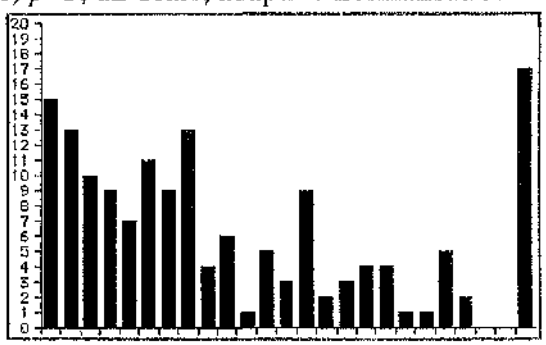
p) p=4, sone scale, adaptive normalisation.



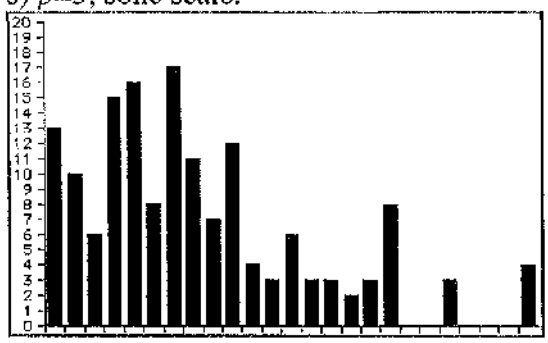
q) p=5, dB scale.



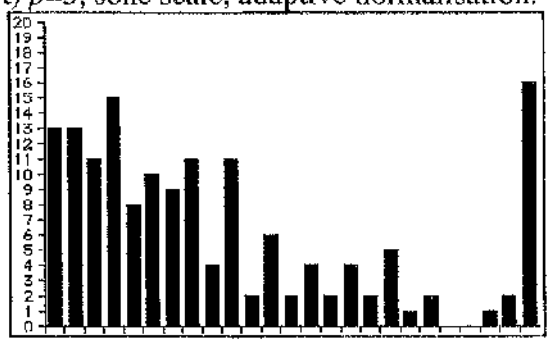
r) p=5, dB scale, adaptive normalisation.



s) p=5, sone scale.



t) p=5, sone scale, adaptive normalisation.



6.3.2 Additional Test Cases

The examples in this section were used to investigate the behaviour of the onset detection method when applied in a variety of more difficult situations, and with types of timbre not included in the main body of test cases. The tests were conducted in the same fashion as the main body (unless a specific parameter set was required, which will be indicated), and detailed results are again given in appendix E (table E.6).

Figure 6.33 – Piano with reverberation.

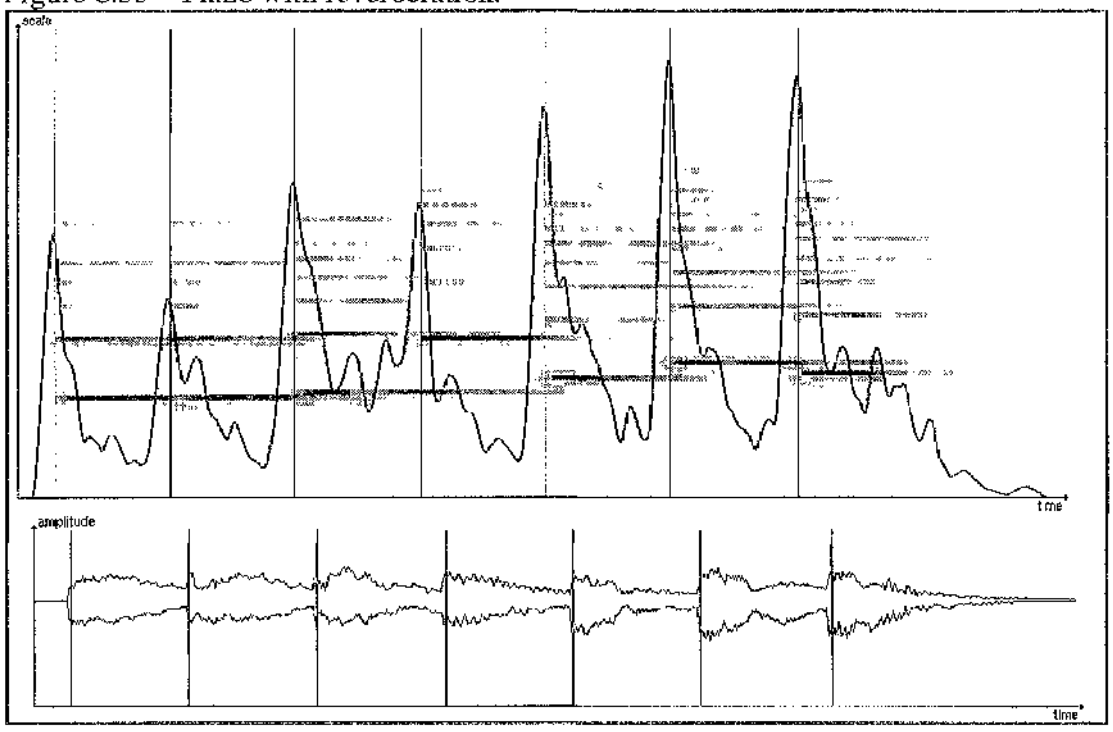
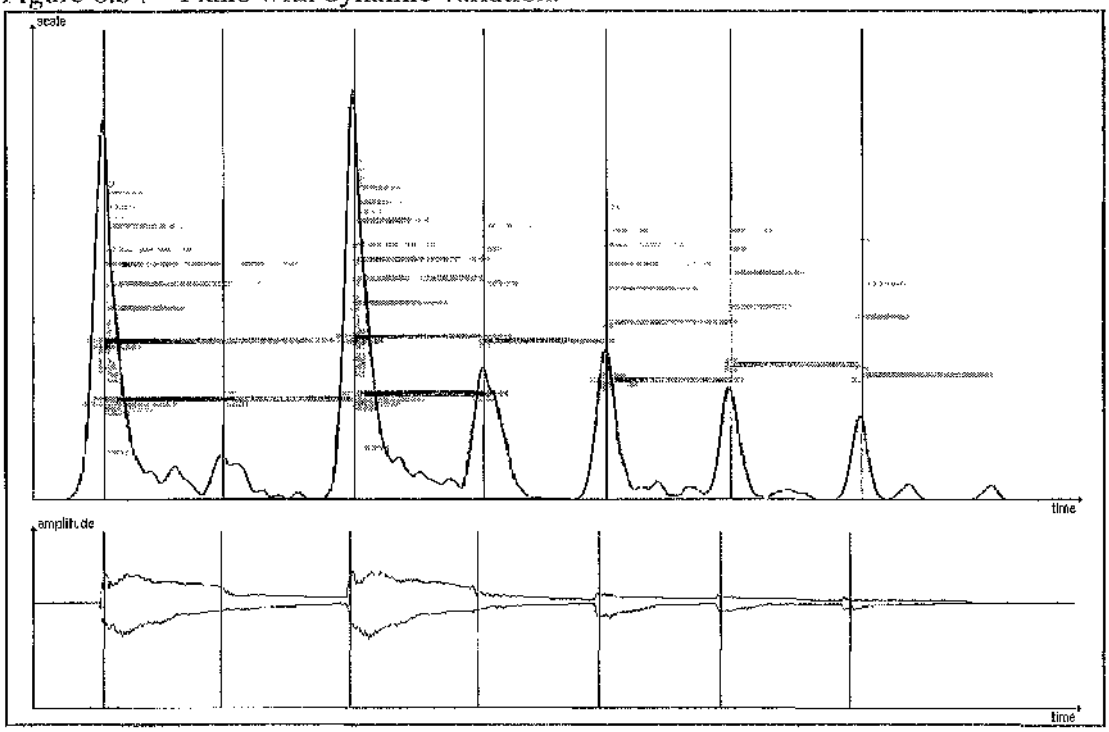


Figure 6.33 shows the piano example of figure 6.25 with reverberation applied. This causes notes to ring out and overlap considerably, which would be problematic for many other methods, but should not significantly effect the vector distance results.
It can be seen that all onsets are still located, although the numerical results show that the accuracy has deteriorated somewhat, when compared with the original piano example. The onset detection parameters are similar to the original, although the chosen method has changed and the detections are now almost all early.
The test piece with dynamic variation (see figure 6.1) was generated using the piano timbre, and the results of the onset detection process are shown in figure 6.34. It was expected that this example would cause problems (especially the quiet repeated note), but the results are surprisingly good (with accuracy not drastically affected). The exponent used has changed and, as would be expected, the band threshold and number of partials required for an onset have been reduced.

Figure 6.34 – Piano with dynamic variation.



The next two examples illustrate the effect of very low notes. There is an inherent decrease in time resolution at lower scale levels, and it might be expected that this would lead to less accurate results (a transposition of two octaves, as in the example of figure 6.35, corresponds to an increase in time resolution by a factor of four in the modulus plane).

In fact, the average difference is only slightly worse than the original, and the same analysis method (with similar onset detection parameters) is chosen. It is likely that the presence of higher harmonics allows the accuracy of the analysis to be maintained.

In order to investigate the behaviour of the method at even lower frequencies, the piano piece was transposed down by another octave. Figure 6.36 shows the modulus plane and onset detection results obtained.

The same analysis method is used, except that adaptive normalisation is applied. However the onset detection parameters are much different, indicating that a small number of strong (but not rapidly increasing) partials are being sought – this reflects a move to identifying the higher partials. In addition, the accuracy has worsened, but the average difference is not greater than twice that obtained when the same piece one octave higher was analysed.

Figure 6.35 – Piano two octaves lower than original.

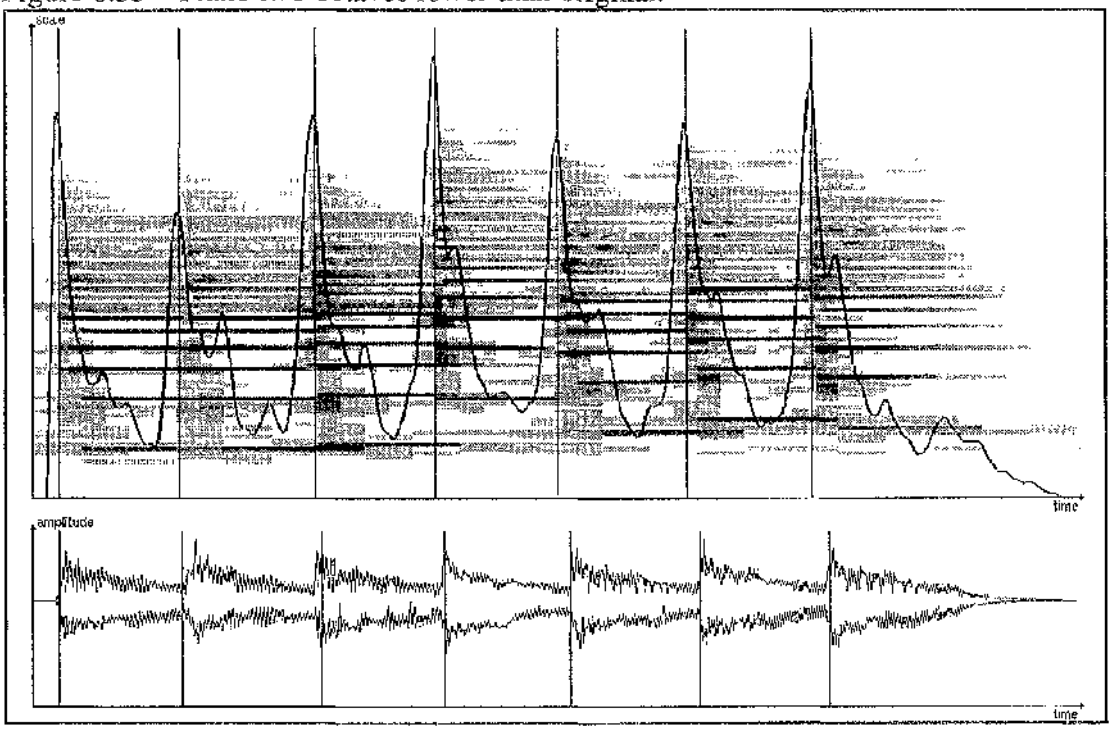
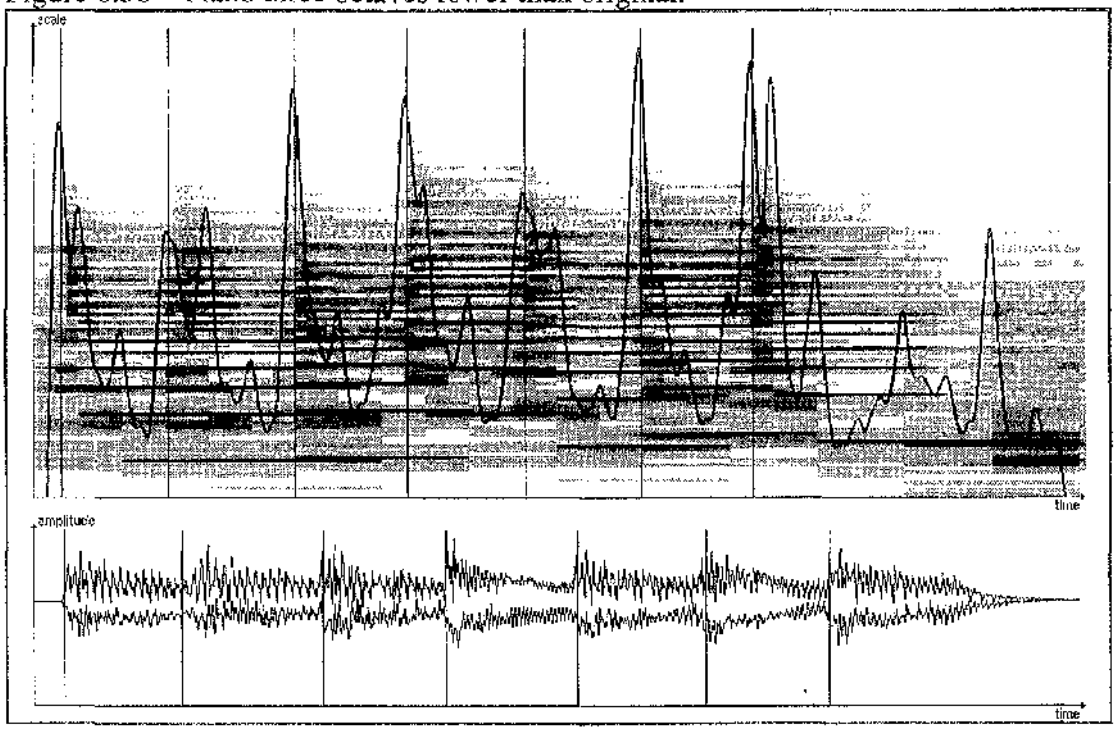


Figure 6.36 – Piano three octaves lower than original.



An unusual effect is visible towards the end of this example. The smearing of the lower partials in the last note, due to decreased time resolution, results in a series of non-negligible modulus values after the end of the piece. When adaptive

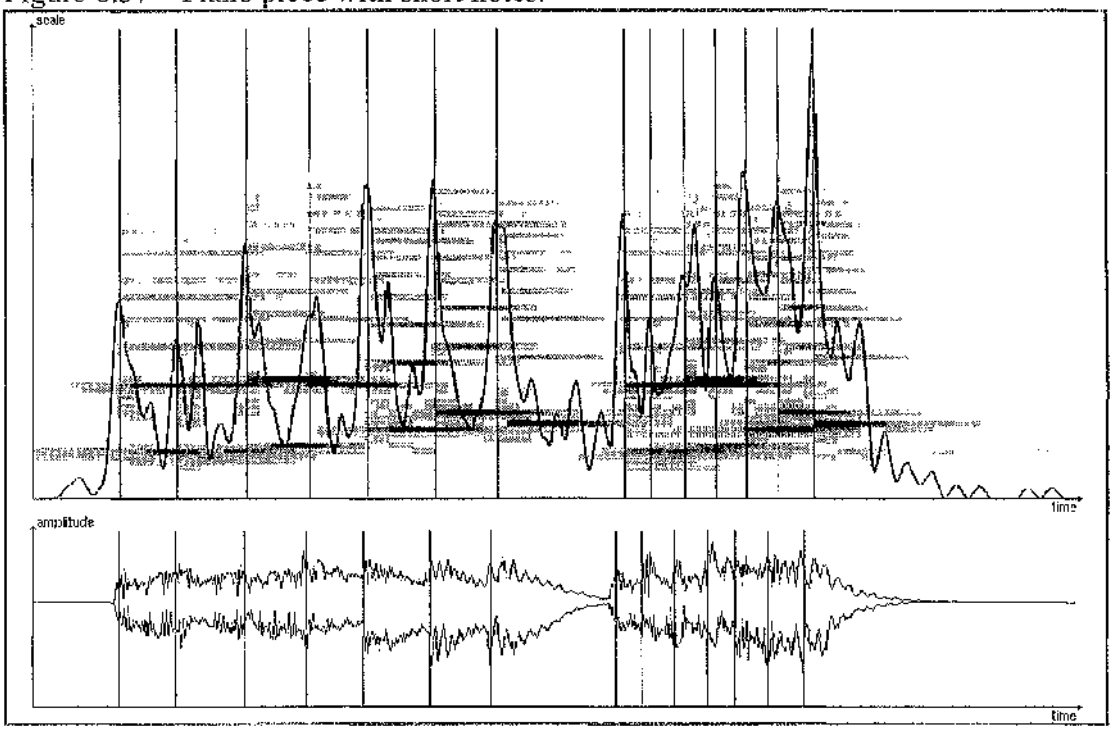
normalisation is applied, these are treated as part of a note and all of the modulus values around that time artificially increased as a result. It can be seen that the spurious peaks of vector distance which result occur on the boundaries of low resolution modulus values (and are rejected as onsets, in any case).

The next test case involves much shorter notes than those already described (see figure 6.2). The analysis (as used until now) would fail in this case, because the averaging windows are longer than the shortest notes present. In order that peaks are as clearly defined as possible, it is desirable that the window size is less than half the length of the shortest note in the input. This means that there will be a point in time towards the middle of each note at which neither window extends beyond the boundaries of the note, and when the difference should therefore be small (assuming that a reasonably steady state exists during notes). If the technique behaves as expected, reducing the length of the windows should enable all onsets to be successfully detected. Therefore, all of the time related parameters were reduced to approximately a third of their previous values as follows: l=r=30, *min_len*=40, *peak_reqd*=3, and the smoothing window was seven points long.

The results are shown in figure 6.37. The analysis method chosen was the same as the original piano piece, but the number of partial onsets required to signal a note onset increased significantly. This would enable the large number of spurious peaks (a side effect of the shorter windows) to be rejected. In addition, accuracy was improved over the original test – this is also due to the use of shorter windows, and highlights the trade-offs explained in chapter 4. It is worth noting that shorter windows were not used on the other cases because, although more accurate results can be achieved in some cases, in others onsets are missed or misidentifications occur. For example, if a timbre has a long rise time, shorter windows would not produce satisfactory results (however, short notes with such long rise times would not ordinarily be expected).

The following two examples illustrate the effect of modulation on the vector distance technique. It would be expected that any kind of modulation would result in spurious peaks, whilst also potentially obscuring transitions between notes. Two cases are considered – amplitude modulation (tremolo in music), and pitch modulation (vibrato). The legato versions of timbres from the original test cases are used, so that the effect on the detection of both the first note and subsequent transitions can be observed.

Figure 6.37 – Piano piece with short notes.

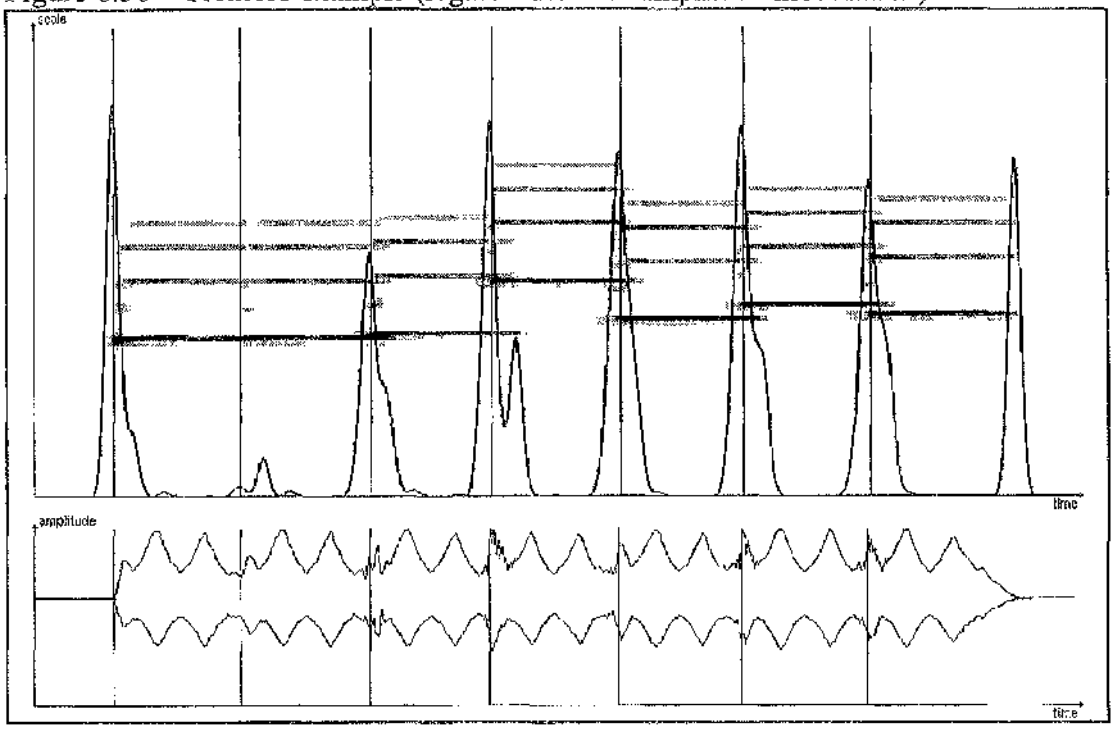


The flute was arbitrarily chosen as an example of an instrument which can be played with tremolo added by the performer. The synthesiser voice was restricted to amplitude modulation, and the modulation wheel moved to its highest position. The same range of experimental methods and parameters were used as before, and the resulting modulus plane and amplitude envelope are shown in figure 6.38.

The same analysis method and similar onset detection parameters were chosen as in the original legato flute example, however accuracy was markedly better. This is surprising, and one possible explanation is that the restarting of the modulation on each new note introduces discontinuities which help to punctuate transitions (although not shown here, the average error for the staccato flute with amplitude modulation applied was only reduced by 40%).

The french horn was chosen as the vibrato example. Similarly, the voice was restricted to pitch modulation, and the modulation wheel moved to its highest position. We would expect that pitch modulation would cause the energy previously confined to individual semitone bands to spread across a number of adjacent levels. This would have the effect of smoothing transitions (especially those involving small intervals), whilst making partial onsets harder to identify and introducing spurious peaks as a result of the modulation. In summary, because there is a greater effect on the modulus plane, there will be a greater effect on vector distance.

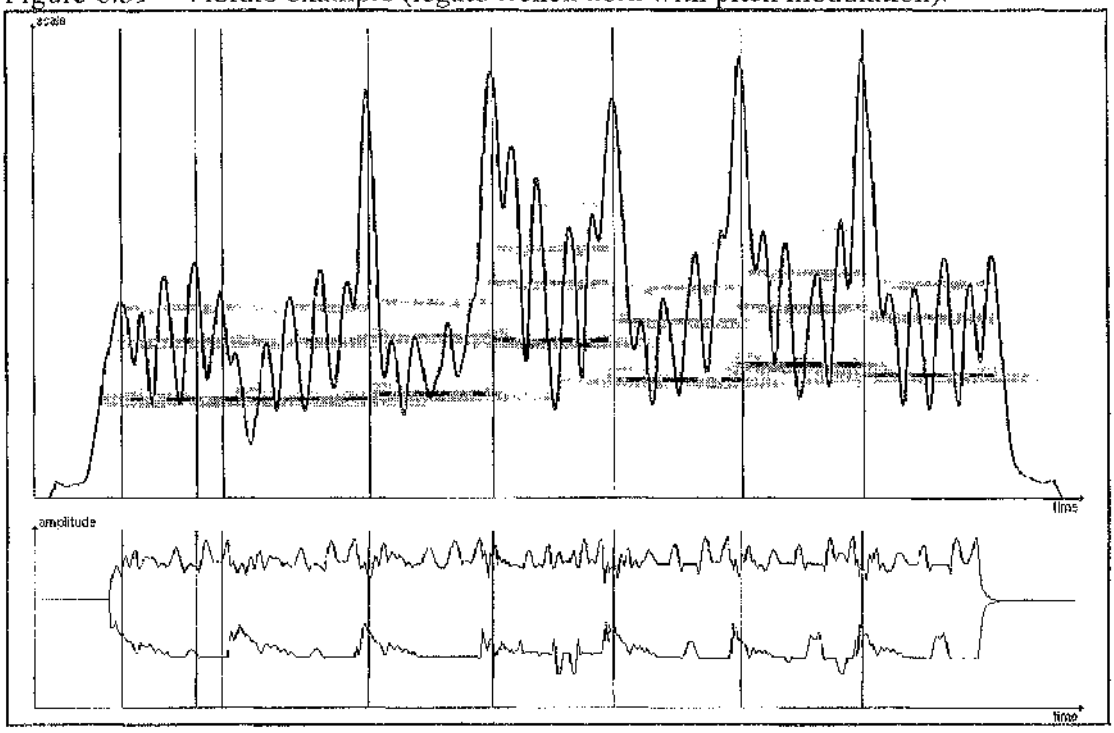
Figure 6.38 – Tremolo example (legate flute with amplitude modulation).



In fact, none of the analysis methods or onset detection parameters resulted in the successful detection of all onsets. On studying the modulus plane, it became apparent that both loudness scales were de-emphasising many of the scale levels with only a small amount of energy too much. The raw modulus plane (with only equal loudness weighting applied) was therefore used. The *band_thresh* parameter was then set at zero, so that even scale levels with only negligible energy would be considered when identifying onset peaks. A high relative change (*change_thresh*=200) could then be sought in those low energy bands.

Figure 6.39 shows the results of this process. The first onset is highlighted, the repeated note is obscured most, but thereafter the dominant peaks all correspond to onsets. Therefore, although peaks are still generated, the vibrato makes identification of those peaks as onsets considerably harder.

Figure 6.39 – Vibrato example (legato french horn with pitch modulation).



Although some percussive timbres were included in the experiment, drums have thus far not been considered in any depth. This is because percussive timbres have been investigated in some detail previously (see, for example, [Schloss 85]), and also because such cases should present least difficulty for the method described herein. However, in order to illustrate the method's flexibility, an analysis of the polyphonic drum pattern of figure 6.3 (reproduced in figure 6.40) is now described.

The example includes two different cymbal crashes, which overlap up to six of the following onsets, and can be seen to mask the high hats on the modulus plane (figure 6.41). Equal loudness weighting and the decibel scale were used, with the vector distance calculated using a value of p=2 and window sizes of l=r=10 (since very short events and percussive onsets were involved). The other parameters were set at *min_len*=128, *peak_reqd*=1 (since peaks were narrow), *band_thresh*=10, *change_thresh*=65, *num_partials*=6, and a smoothing window of two points was used. It can be seen that all onset times are highlighted (even those masked by cymbals), although coincident onsets are of course not distinguished.

The reader will recall that the amplitude plots do not in general correspond to the same length of time as the modulus planes, due to zero padding and subsequent zooming-in on the modulus planes, so that corresponding vertical lines in the graph pairs are not in alignment (particularly in this case).

Figure 6.40 – Drum pattern score.

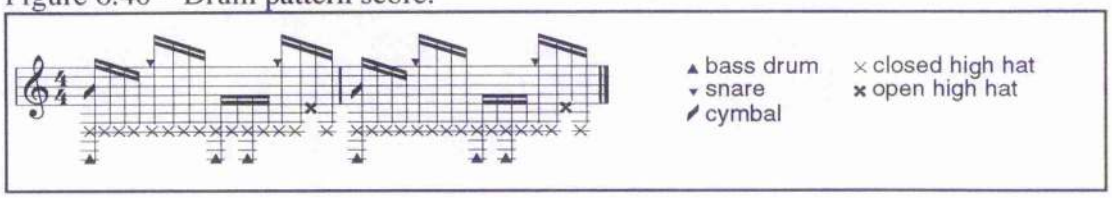
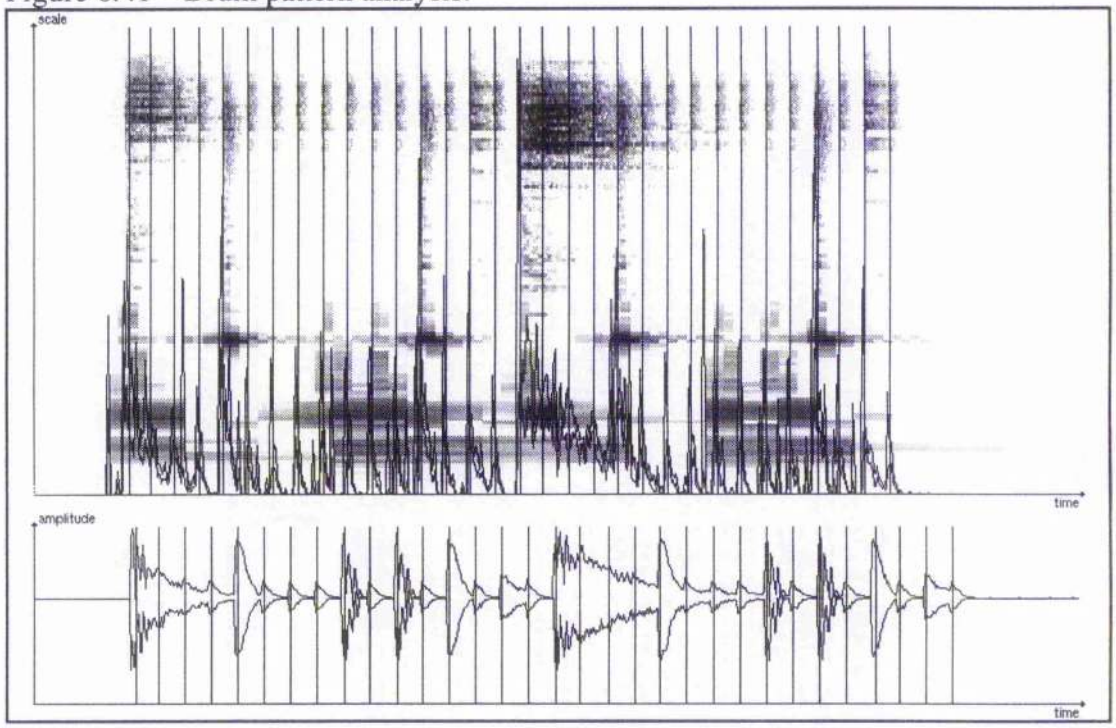


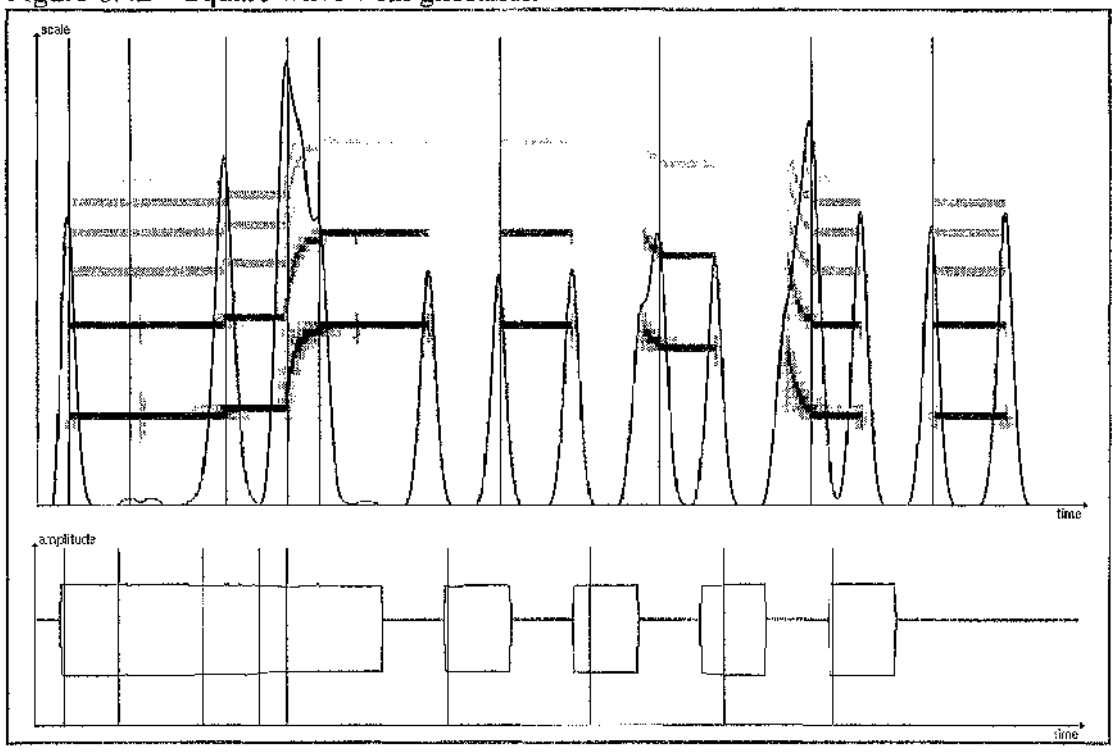
Figure 6.41 – Drum pattern analysis.



Lastly, we consider an example on which the vector distance method would be expected to fail, but that is nonetheless interesting. For the purposes of illustration, the decibel scale and a value of p=2 were applied to the glissandi example of figure 6.4. Figure 6.42 shows the result of applying the onset detection method with the parameters set as follows: *band_thresh*=5, *change_thresh*=50 and *num_partials*=0 (others were as for the main body of test cases).

The results obtained are as much as could be hoped for using the vector distance method. The end points of steady state regions give rise to peaks, and the appropriate ones are identified as onsets. Adjusting the analysis parameters to locate smaller events gives rise to complex peaks during the glissandi, and repeated detection of onsets (as predicted in [Shahwan 94]). Later, we will consider why this occurs, and how the method may be altered to detect such events.

Figure 6.42 – Square wave with glissandi.



6.4 Conclusions

This chapter has presented the results of the experiment developed in chapter 5, and (it is hoped) convinced the reader of a number of points.

First of all, the onset detection method has performed satisfactorily, in that all onsets in the main body of tests were highlighted, even in the most difficult of cases. Of course, the accuracy was variable, and discussion of the detailed results is continued in the following chapter. In addition, almost all of the additional test cases were successfully analysed, with only the vibrato and glissando examples causing problems.

Secondly (and no less importantly) we have seen why a carefully designed experiment, involving a wide range of test timbres, is necessary. It is fair to say that no small subset of the above results would have provided as clear a picture, and there are certainly many more types of input which could be studied.

Chapter 7 focuses on the detailed results and issues arising from them. For example: what makes the results from certain timbres less accurate, can the analysis be standardised into a method which provides reasonable performance across many inputs, and how do the results compare with the performance criteria implied by a range of applications?

Chapter 7
Assessment of Results

This chapter is devoted to further discussion and interpretation of the results presented in the previous chapter and in appendix E.

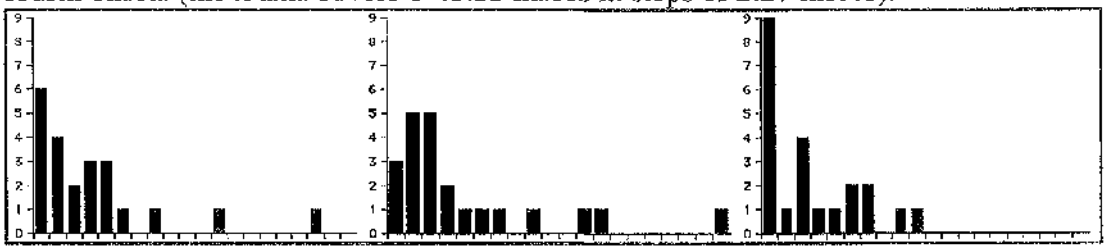
Although some indications of the technique's performance have been given, we would like to know more about which timbres were problematic, which methods performed best and whether the various parameters can usefully be fixed. We should also consider the results in relation to the theoretical accuracy achievable, the accuracy of previously published techniques, and the accuracy demanded by the applications identified earlier. The discussion is organised similarly to the previous chapter, in that the results are discussed by method and by timbre.

The additional test cases were included primarily to illustrate the flexibility of the method, and a commentary has already been given in the previous chapter on these examples, therefore they are not discussed further here.

7.1 Main Body of Tests

For a clearer picture of the errors occurring in the onset detection process, the distribution of the absolute values of the errors was plotted for the first note (preceded by silence), the second note (which was a repetition of the first), and the fourth note (which is separated from the previous note by 11 semitones). The resulting graphs are shown in figure 7.1.

Figure 7.1 – Absolute value error distributions from figure 6.7 for the first, second and fourth onsets (the x axis covers 0-45.35 msecs in steps of 2.27 msecs).



The results are as expected in that the jump of 11 semitones ought to be most easily detected by the vector distance method, followed by the onset preceded by silence, with the repeated note being most difficult. This is partly because, whereas the type of onset alone dictates how easily it is detected after silence, coincident offsets can often make location of transitions easier in the legato style examples. However, although the shapes of the distributions vary, the sample size here is relatively small and (for example) the number of instruments with errors of less than 9.07 msecs (400 samples) is constant in all three cases.

Tables 7.1a–c show the remainder – those examples in which the error was more than 400 samples for each of the three kinds of onset. Negative errors denote early detections, positive errors late ones (note that errors are given in samples here).

Tables 7.1 a, b and c – Absolute errors > 400 samples in figure 6.10 (first, second and fourth onsets).

timbre	error
horn	-1750
horn_stacc	1160
oboe	720
marimba	576
mute_gtr_stacc	-478
oboe_stacc	448
cello_stacc	-436

timbre	error
cello	1988
dist_gtr	-1270
oboe	-1128
organ	-811
horn_stacc	624
cello_stacc	-588
oboe_stacc	488

timbre	error
cello	980
flute	854
oboe	680
horn	674
horn_stacc	544
organ	517
oboe_stacc	440

The french horn accounts for both of the greatest errors for the first note (see figures 6.16 and 6.17). The partials start gradually (over a period of time), and adaptive normalisation was employed. However, over-compensation is apparent in this case, as the detected onset was rather early. The results for the legato style example improve considerably for the second and fourth onsets, as a result of a more sudden offset helping to punctuate transitions. The staccato version is generally a little better – adaptive normalisation was not used here, but only a weak fundamental is apparent at the onset and the peak of vector distance tends to occur a little later, when more partials have commenced (in fact, every detection was late in this example). Similarly, when the results for the legato example without adaptive normalisation applied were examined, every onset detection was late. In summary, the type of onset encountered

here seemed to require some compensation for consistently late detections; but, although results were sometimes improved, adaptive normalisation was not suitable.

At this point it is worth considering why the staccato results should be much different for transitions when compared with those obtained from the first note. In cases where there is silence between notes, the results are fairly consistent (see the oboe in figure 6.22, for example). However, when the modulus plane of some of the other examples is examined (the staccato cello in figure 6.11 is one such case), it can be seen that there is often a degree of ringing inherent in the synthesiser voice (imitating that which would be present in the instrument itself). This means that, for example, in the staccato cello example there is actually overlap between adjacent notes. Adaptive normalisation was again used on both cello pieces, in attempt to overcome the very slow rise and consistent late detection when it was not used. Although the first note is improved, the others appear to suffer from the overlap (a discussion of why adaptive normalisation should not be applied to overlapping notes has already been given). The cello pieces are also unusual in that the higher notes were produced by the synthesiser in such a way that their spectra were more complex than the other notes (this is evident in figure 6.11 and 6.12). This made them easier to pinpoint, and accounted for the staccato cello having the lowest error on the fourth note. It can be seen from this discussion that there was an interaction of many factors in these examples which made them most difficult to interpret.

The appearance of the marimba in the above table is surprising – examination of the modulus plane (figure 6.18) shows that each note contains only a few strong partials, and the (late) detection is again skewed towards a time at which all are present. This was the maximum error for the timbre, and the average was considerably lower.

The repeated note results are generally poorer, and the error for the distorted guitar in this case was approximately four times its average error. Figure 6.12 shows that the timbre was both harmonically complex and did not fade during the notes. These factors particularly obscured the repeated note, which gave rise to only a small peak in vector distance. The organ's error was also worse than average for this note, and figure 6.23 shows that the cause was similar.

Whilst the fourth onset is generally detected more accurately, the flute makes a sudden appearance high in the table of errors for this note. Again, adaptive normalisation was used in this case and detected onset times were repeatedly late without it. There is some evidence that such late detections were not resolved by application of adaptive normalisation in the legato style examples (see the cello, horn, flute and organ in table E.1). Inspection of the flute example again shows unexpected overlap, and the

adaptive normalisation causes the end of the previous note to dominate the beginning of the next, which suggests an explanation for this phenomenon.

7.1.1 Results by Timbre

The standard deviation of all the errors in table E.1 is 446, and there are 3 examples which contain errors greater than 2 standard deviations from the average. These were the legato cello, flute and horn examples which contained 4, 3 and 2 such errors respectively.

These examples have already featured in the above discussion of results for different transition types. The results indicate that, whilst some method of compensating for slow attacks may be required, adaptive normalisation has not proved entirely effective. It was used in 5 out of the 6 test cases involving the worst 3 timbres (showing that it was improving results in those cases), but the fact remains that these timbres gave the greatest errors. As an aside, the only other timbre to have adaptive normalisation applied was the organ. Here, when compared with results when it was not applied, the legato case was only slightly better; however, the average error in the staccato example was reduced from 706 to 167 samples. Therefore, although this particular technique may not be applied in a blanket fashion, we can see that results can be dramatically improved by such a method.

It was hoped that the results might demonstrate a grouping of the inputs – for example, by analysis method or onset detection parameters. This is evident when adaptive normalisation is considered – in 3 out of 4 of the timbres to which it was applied, it was applied to both staccato and legato examples, and we have seen how these timbres share other attributes. There is some evidence for such grouping by exponent: five of the ten timbres which included staccato-legato pairs had the same exponent chosen for both test cases. However, there is not enough evidence to support the hypothesis that identical (or close) exponents would be applied to perceptually similar timbres.

When the onset detection parameters are considered, seven out of ten of the paired examples involve similar parameters for both legato and staccato cases (this is more evident for the *change_thresh* and *num_partials* parameters in table E.1). Closer inspection shows that, for example, many of the timbres have a value of *change_thresh* close to 75 – this seems to be the result of many parameter sets giving the same result, as the figures in the table are averages, and 75 is close to the average of all the values considered for this parameter. We will not dwell on this here, as it is debatable whether onset detection parameters can be compared across different

analysis methods. In the next section, we shall restrict our attention to a single method and comparisons will be more valid then.

7.1.2 Results by Method

If a single analysis method is to be chosen, it should minimise the average error and also have a low maximum error. Of course, it is likely that a high detection rate would be required, and a lowering of accuracy may be acceptable if many more onsets are detected by the method. In addition, we may refer to the error distributions of figure 6.32.

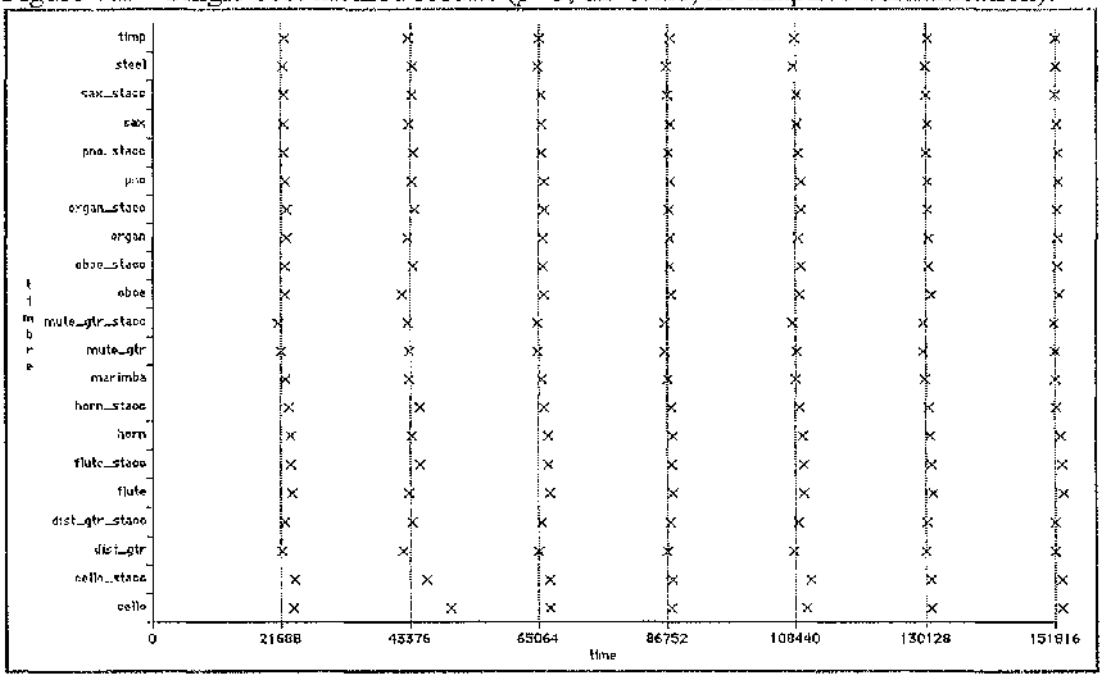
In the experiment reported here, setting p=5 and using the decibel loudness scale gives the lowest average error. Interestingly, this is in partial agreement with the work reported in [Feiten & Gunzel 93], which concluded that results obtained with a similarity measure employing the sone scale and a value of p=5 corresponded most closely to listening tests. The problem with accepting this method in our context is that, whilst the detection rates were all very high, three onsets were missed using that method. In order to achieve a 100% success rate, an increase of only a few milliseconds in average error need be introduced. This is achieved with the decibel scale and a value of p=3. Although the average error is very slightly increased using this method, the maximum error actually decreases by a few milliseconds (the cases with greatest errors will be discussed shortly). It should also be noted that all values of p between one and three give 100% success rates, with higher powers introducing omissions.

It is not surprising that the change of context from measuring similarity to detecting onsets (by measuring *dis*similarity) involves a corresponding decrease of the most suitable exponent in Minkowski's measure. The higher values of the exponent emphasise marked differences, and are inherently less sensitive to small differences between similar vectors. An example of this effect can be seen in figure 6.10 – a value of p=5 was chosen for the cello and only a relatively small peak in distance is evident at the transition between the repeated notes. If the aim was to measure similarity, this would be the desired effect – that a negligible distance results from a repeated event. However, in detecting onsets, such small differences in largely similar vectors can be all that signal a new note (especially a repeated one). For example, figure 6.25 shows how the value of p=1, adopted for the piano, gives rise to similar sized peaks for the first and the repeated note onsets.

We do not consider the particular loudness scales involved further, as implementations can vary (see appendix B), and we have seen that their effect was not so striking.

Having settled on a single method, we should now ascertain how it compares with the optimal results presented in the previous chapter. Figure 7.2 shows the results in the same format as figure 6.7 (they are also presented in tabular form in table E.6). The average error has increased from 411 to 698 samples, however the maximum error has changed from 2396 to 6916 samples. A glance at the graph shows that the greatest error arises from the repeated note in the legato cello example. The second largest is much lower (2804 samples) and occurs on the same note in the staccato cello example. In fact, all of the errors over 1800 samples are introduced by the cello timbre. The cello, flute and horn can be seen to consistently introduce late detections, and between them account for all of the errors greater than 1300 samples.

Figure 7.2 – Single best method results (p=3, dB scale, no adaptive normalisation).



Now that we are restricting our attention to a single analysis method, it makes more sense to consider the range of onset detection parameters employed for various timbres. Recall that the lowest error may be obtained with a range of onset detection parameters, and an average of these for each test case was calculated for reference. The number of successful parameter sets can then give some insight into the nature of

a test timbre. For example, the muted guitar gave the lowest error for this method, and the same result was obtained for many parameter sets. This shows that the onsets were relatively easy to detect and the precise parameter settings were fairly unimportant in their detection. However, the staccato timpani example also had a low error (the fourth lowest), but this result was only obtained with relatively few of the possible parameter settings. This indicates that some aspect of those particular parameter settings was crucial in achieving that result. Similarly, high errors with a small number of parameter sets can be found – for example, the legato flute had the fourth largest error, and this was obtained with only a small number of parameter sets. This would seem to indicate that the onsets were difficult to locate precisely with this method, but that there was still some important aspect of a small subset of the parameters which made them particularly suitable for the timbre.

There was no such clear case of a high error obtained with a large number of parameter sets, the closest being the staccato flute (which had third greatest error, but this was only achieved with around half of the parameter sets). If a large error resulted from many parameter sets, this could be interpreted as indicating that there was no parameter set particularly suited to the timbre, and that there was some basic incompatibility between the method and the test case. It is therefore encouraging that no such case was found amongst the range of timbres investigated.

The average parameters over all test cases were as follows: *change_thresh*=73, *num_partials*=3 and *band_thresh*=20. How the parameters varied across the test cases is now discussed, in order to assess their effect.

The average value of *change_thresh* was again close to the mean of all values tested, and the average settings over all the test cases only covered a range of 72 to 77. This was due to the fact that, even in cases where only a small number of parameter sets gave the best result, many different settings of this parameter appeared. Therefore, its value does not seem crucial in the onset detection process. This is not to say that the parameter is redundant: some threshold on the change indicating a partial onset must be set, and the example of figure 6.39 showed how extreme settings could be useful in some unusual cases.

Although the average of all settings for the *num_partials* parameter is close to the mean of those tested, it takes on values between one and five (only values between zero and six were tested). The distorted guitar and organ both took on the value of five, and observation of the modulus planes in figure 6.23 and 6.12 reveals the reason. The notes are harmonically rich, and are clearly identified by a large number of partial

onsets. Again, although the averages in the staccato cases are slightly lower, successful results are more often achieved with higher settings of this parameter.

At the other end of the scale, those timbres with the greatest errors can be found amongst those with the lowest values of this parameter (the cello, flute and horn cases all had settings for *num_partials* no greater than two). This can be attributed to either one of the following reasons, or a combination of both: the voice included some degree of ringing and subsequently, notes overlapped; or the timbre had few strong harmonics. At this point it is worth stressing that the parameters were static for each timbre, so that the repeated note had to be identified as an onset with the same parameters as all the others. For that note in particular, it would often only be possible to distinguish one or two partial onsets in the above cases. This does not imply that better results could be achieved for the other onsets with different parameters (as peaks cannot be located more precisely, and all were correctly identified), merely that the presence of a repeated note will have a greater effect on the average onset detection parameters for those timbres.

Without labouring over more examples, it is clear that the setting of this parameter is linked to the kind of timbre under analysis. We have seen how different aspects of the timbre influence its optimal setting, and it is unlikely that a single value suitable for many different types of input could be derived.

The last parameter places a threshold on the energy which must be present in a scale level before it can be considered as an onset. The average value for individual test cases ranged from two to 32.

Timbres which took values towards the top of this range were the saxophone, muted guitar and steel drum. These were timbres which either had many strong partials, or were percussive so that there was considerable energy in most partials at the note onset.

The average value of this parameter frequently increased for the staccato cases, when compared with that for the legato version of the same timbre. This was again because of the necessity of detecting the repeated note with the same parameters as the others. In the legato cases, this onset was often only signalled by the re-emergence of weaker high harmonics or onset transients, thus a lower threshold is required. No timbre consistently took a low value for this parameter, and the average was skewed towards the highest value.

As previously, there was no clear correlation between perceptually similar timbres and similar onset detection parameters. Although it has been described how these parameters are related to various perceptually salient aspects of the timbres, the test

cases were intended to cover a broad range of possibilities and did not include many similar timbres. Therefore, it would have been difficult to justify any such observation from the experiment which was carried out.

It will have been obvious from the above discussion that there can be no fixed set of onset detection parameters that will give the same results as were obtained when the most suitable parameters could be chosen. If the parameters are all set at the lowest values encountered, spurious detections will result in some cases; whereas if the parameters are set at the averages, some onsets are inevitably missed. There is also the possibility that the wrong peaks may be chosen, increasing the error in some cases.

The original experiment had shown that the *change_thresh* parameter was relatively unimportant, and if the *band_thresh* parameter was set at the lowest value encountered in the original test cases, then all necessary scale levels should be considered. The *num_partials* parameter was most important in distinguishing different kinds of onset, and so an experiment was run with *change_thresh* =72, *band_thresh* =2 and *num_partials* allowed to vary between zero and ten.

Increasing the range of this parameter allowed it to be set high for the percussive timbres, thus rejecting spurious peaks. The result of this was that only the legato oboe's error increased, and the overall average difference was only increased slightly. The range of parameter settings which gave rise to the best result for each timbre are shown in table 7.2.

Table 7.2 – *num_partials* only varying.

Instrument	Legato	Staccato
Cello	5–6	4–7
Distorted Guitar	5–10	5–10
Flute	1–2	0–5
French Horn	0–2	0–5
Marimba	0–5	n/a
Muted Guitar	2–10	5–10
Oboe	0–4	0–9
Piano	1–7	2–8
Pipe Organ	8–9	5–10
Saxophone	0–10	5–10
Steel Drum	4–10	n/a
Timpani	7–10	7–10

This shows the impossibility of fixing this parameter over all the test cases – the best that could be achieved would be obtaining the lowest error for 16 out of 22 timbres, with a setting of *num_partials*=5.

7.2 Contexts for Assessment

Until now, errors have been given in samples and not viewed in the context of any other measures of accuracy. Whilst this has given an unambiguous and readily understood view of the results, a number of issues remain unresolved.

For example, we have already discussed the difference between perceptual and physical attack times. Without a set of test timbres for which perceptual attack times were well known, there was no way that the results could be assessed for their degree of correspondence with perceived onset times. Further, the notion of physical onset time is not well defined, and (although we have been consistent) other ways of determining the *actual* onset time could be equally as valid. Lastly, what constitutes the correct onset time can be dependent on the particular application, and highly subjective.

The results are therefore evaluated in a number of different ways – in terms of the theoretical accuracy achievable, the accuracy achieved in previously published work, and the application contexts defined earlier.

7.2.1 Theoretically Achievable Accuracy

We first qualify further discussions by considering the limit on the accuracy which can be achieved in general, and via the harmonic wavelet analysis. Chapter 3 included a discussion of the impossibility of locating a frequency component of F Hz with an accuracy greater than $\frac{1}{F}$ secs, using any method of time-frequency analysis. Now, the fundamental frequencies of the notes in the test piece span a range of 261 to 523 Hz, implying a range of theoretical accuracies (when detecting the fundamental) of around 3.8 to 1.9 msecs (169 to 84 samples). It has already been seen how higher harmonics (which could be detected with higher accuracy) can either aid or confound the onset detection process, depending on their time of onset in relation to the physical onset time. However, the introduction of such timbre-specific knowledge is not considered here and we will use the fundamental frequency as a guide to achievable accuracy.

Of course, this theoretically possible accuracy does not translate directly into practical applications. For example, [Depalle & Tromp 96] states that it is normal to require a

signal at least four periods long to identify a particular frequency, but also presents a method whereby this may be reduced to one and a half periods.

In the current work, the notes in the test piece are split between two time resolutions in the wavelet analysis – the two highest notes' fundamentals are at scale levels in which each coefficient represents 2048 samples (46 milliseconds); whereas the fundamentals of the other notes occur in levels with time resolutions of 4096 samples (93 milliseconds) per coefficient. Results with considerably greater accuracy than this have already been presented, which illustrates the effect of partials at higher scale levels.

7.2.2 Previous Results

It is relatively rare to find precisely quantified results in the published work on onset detection. This is perhaps because, whilst it may be relatively easy to establish criteria for success in other areas (pitch detection for example); it is harder to do so for onset detection, due to the problems of perceptual attack time and the differing demands of various applications.

For example, using the multiresolution Fourier transform in [Scott & Wilson 92] means that time resolution can be chosen (although a trade-off with frequency resolution is obviously in effect). A transform with a time resolution of 11.2 msecs is selected, although precise onset detection results are not given.

Based on the same transform, the thesis of [Pearson 91] includes some work on onset detection. Here, the onset times of two piano notes are detected to within 20 msecs of their actual onset times.

The analysis of a percussion piece acts as the main example in [Schloss 85]. The onset detection part of the analysis is said to be 95% accurate, in that only three out of 55 notes are missed. However, no statement of actual onset times is given and no assessment of the accuracy is possible. The analysis method goes on to utilise high level knowledge in recovering a score for the piece.

In summary, it has not been possible to find any comparable study which includes precise results for a range of timbres.

7.2.3 Application Contexts

The requirement of each of the applications can be considered as placing the actual onset within some time frame, based on the detected onset time. For example, in graphical editing, the actual onset must be placed on screen so that the user can proceed to find the desired point in time. This implies that, if the detected onset is

placed in the centre of the time window, the actual onset should be no more than half a window length away to be visible. Adopting this view of the task means that we need only decide on the length of the time frame in the application to enable precise judgements to be made for each detected onset.

However, this view cannot be utilised directly in all of the applications under consideration. For example, in synchronising an event in an animation with an onset, it might be thought the time length of a single frame in the animation could be used in the same way as the window length in the previous case. This is not the case, since frames cannot be arbitrarily centred on detected onset times, without affecting the position of all other frames. This implies that, for example, if an actual onset time is close to a frame boundary, even a small error in its estimated position could be enough to place the detected onset in a different frame. However, the frame length clearly represents a period of time which is below some threshold of visual acuity (since the animation is perceived as continuous motion). Therefore, we shall use the same metric, and adopt the view that an error of less than half a frame length is not detectable in this application.

The synchronisation case is then easiest to define, as there is an established standard of 25 frames per second (for PAL video). This means that the time frame involved is 1764 samples long (at a sampling rate of 44.1 kHz), and the half window is 882 samples long.

In the case of graphical editing, the screen size and resolution, in combination with the number of samples mapped to a single pixel, dictate the length of the time window. For example, assuming that the audio is displayed in a relatively modest sized window of 600 samples, and that the waveform is plotted with one pixel per sample, half of the window's extent would obviously be 300 samples. However, in the author's experience, mapping only one sample to each pixel often presents too detailed a picture to be useful in locating points of interest. A survey of research on auditory temporal acuity in [Gordon 84] concludes that in some cases, the smallest interval between two almost simultaneous sounds which still allows their order to be ascertained is 2 msecs. This implies that a somewhat coarser viewing resolution could be used, without introducing any noticeable inaccuracies. For example, mapping each pixel to a millisecond of audio would extend the half window above to cover 13230 samples. These issues should be borne in mind when considering this application.

The application of onset detection to automatic transcription inevitably involves some higher level processing of the detected onset times. If the desired output is the notated score, timing variations introduced by the performer must be overcome to maintain a

picture of the underlying rhythmic structure. For example, it has been observed that note durations can differ from their scored values by up to 50% [Desain & Honing 89]. However, although there has been much work in this area, it seems that we are still some way from solving such problems [Desain & Honing 94b], and it is not possible to place a figure on the deviations from strict timing which can currently be overcome. We shall therefore restrict our attention to the task of quantisation, noting that the introduction of higher level knowledge should improve the performance of any such system.

Large variations from strict timing can only be overcome by resorting to knowledge of rhythmic structuring principles. A lower tolerance will apply if the detected onset times are simply quantised to a grid defined by the tempo and some musically significant time period (for example, a sixteenth note). This procedure might be adopted in an application combining MIDI with digital audio. There are two tasks which may be required of such a system: the assignment of each event in the audio track to its nearest note position in a MIDI track, and the generation of a metrical grid from an audio track (which it is assumed contains a strictly timed performance). In each case, the user specifies the smallest subdivision of a note allowed and this, in combination with the tempo and time signature, can be mapped to a period of time.

Given a strictly timed performance, and a pre-existing quantisation grid defined by a MIDI track, the first task above equates to simply attaching each note in the audio track to its nearest possible position in the MIDI track. The onset detection process must not introduce a large enough error that a note is assigned to the wrong position. The half-window concept is therefore still applicable since, as long as the error is no greater than half the smallest distance between adjacent positions in the MIDI track, each note will still be assigned the correct position. The test piece was recorded at 122 beats per minute, and specifying a fine quantisation grid by dividing each beat into eight parts gives a half window of 1356 samples.

The second task has been defined somewhat abstractly, and we shall consider the concrete example of extracting the tempo from a series of equally spaced events. If the current notion of the tempo is derived from the inverse of the time between adjacent events, the worst case would occur when an early and a late detection took place on adjacent notes. Depending on the order, this would give either an unusually long or an unusually short beat period. If we assume that the tempo is rounded to the nearest beat per minute then, in order that the derived tempo is not changed from the correct value in such a situation, the sum of the errors must not be greater than half a beat period. This is guaranteed if no single error is greater than a quarter of a single beat period at

the desired tempo. At the tempo of the test pieces, this gives a high error threshold of over 5000 samples. If we consider what would constitute the most difficult of such examples, detecting the time between events in a series of sixteenth notes at a tempo of 160 bpm gives a quarter period of 1034 samples.

Table 7.3 shows the success rate achieved in each of these application contexts, in terms of the number of onsets with errors less than the identified tolerance in each of the following analyses: the optimal method and parameter settings, the best method with 3 variable parameters (selected optimally) and the same method with only the optimal exponent chosen.

Table 7.3 – Success by application context.

	tolerance	optimal	3 params	1 param
PAL synchronisation	882	87%	75%	75%
graphical editing (fine resolution)	300	58%	29%	29%
graphical editing (coarse resolution)	13230	100%	100%	100%
MIDI quantisation	1356	96%	86%	85%
bpm tracking (slow)	5422	100%	99%	99%
bpm tracking (fast)	1034	90%	81%	81%

The worst result by far is obtained in the case of graphical editing by viewing one sample per pixel in a window of 600 pixels. Although this scenario is possible, it is certainly an extreme case. Part of the audio preceding the onset would be required to identify it as such, and the studies quoted in chapter 3 found the starting transient in notes (before the steady state) to vary between five and 350 msecs. Now, given that the window under consideration is only 14 msecs long, it almost certainly provides too close a view to be useful in many cases. Also, a single frame in the synchronisation case lasts for 40 msecs. Given that the starting transient of a note may be longer than this (and that the largest errors generally arise from the timbres with long attack times), some of the errors may be acceptable.

Before concluding, therefore, we add some perspective by considering ways in which the tolerances in the worst cases above might be increased. If we insist on viewing at one sample per pixel, but allow an upgraded monitor (giving a 1000 pixel wide window), the second row changes to: 71%, 50%, 50%. If we use the original window size, but map 10 samples (0.2 msecs) to each pixel, it changes to: 100%, 99%, 99%. In the synchronisation case, if we allow an onset to be placed into an adjacent frame

(increasing the tolerance by a factor of three), the first row changes to: 100%, 99%, 98%.

7.3 Tests With Live Recordings

The analysis of live solo performances was undertaken previously by the author in [Tait & Findlay 95]. These have not yet been considered herein, due to the inherent difficulty in establishing actual onset times. This means that assessment of the onset detection accuracy was restricted to a visual presentation of the modulus plane, with detected onset times marked.

In attempt to overcome this problem, the generation of a click track representing the results of the onset detection process was investigated. At each calculated onset time, a click (such as that in figure 4.1) is placed in an otherwise silent audio track. A stereo audio file can then be generated, with the click track in one channel and the original input audio in the other. Listening to this enables a subjective assessment of the timing accuracy of the onset detection process.

Analysis of suitable recordings would also enable the investigation of whether the optimal parameters identified in the experiment could be applied to similar actual timbres. For example, figure 7.3 shows the results of analysing a portion of a guitar solo recorded from a CD [Jones 91]. The timbre was most similar to the legato muted guitar in the experiment, and the transformations and onset detection parameters from that test case gave the results shown. This example is also interesting in that several of the notes towards the middle are hard to distinguish, however the onset detection appears to give an interpretation which corresponds to that perceived on listening.

The next example shows the application of experimental parameters and click track generation for a polyphonic example, previously discussed in [Tait & Findlay 95]. This is a relatively simple piano piece from [Satie 92], involving coincident and overlapping notes. When the transformations and parameters from the experimental piano timbres are used, only one onset is missed. The omission is a quiet note that occurs whilst the previous note is still ringing. It transpires that reducing the *band_thresh* parameter from 14 to 11 allows the soft note to be detected, and figure 7.4 shows this result (note that the eighth onset was originally missed). It is worth recalling that the optimal parameter settings from the experiment were averages, and the adjusted parameter would have also given the optimal result for the experimental timbre.

Figure 7.3 – Real guitar recording.

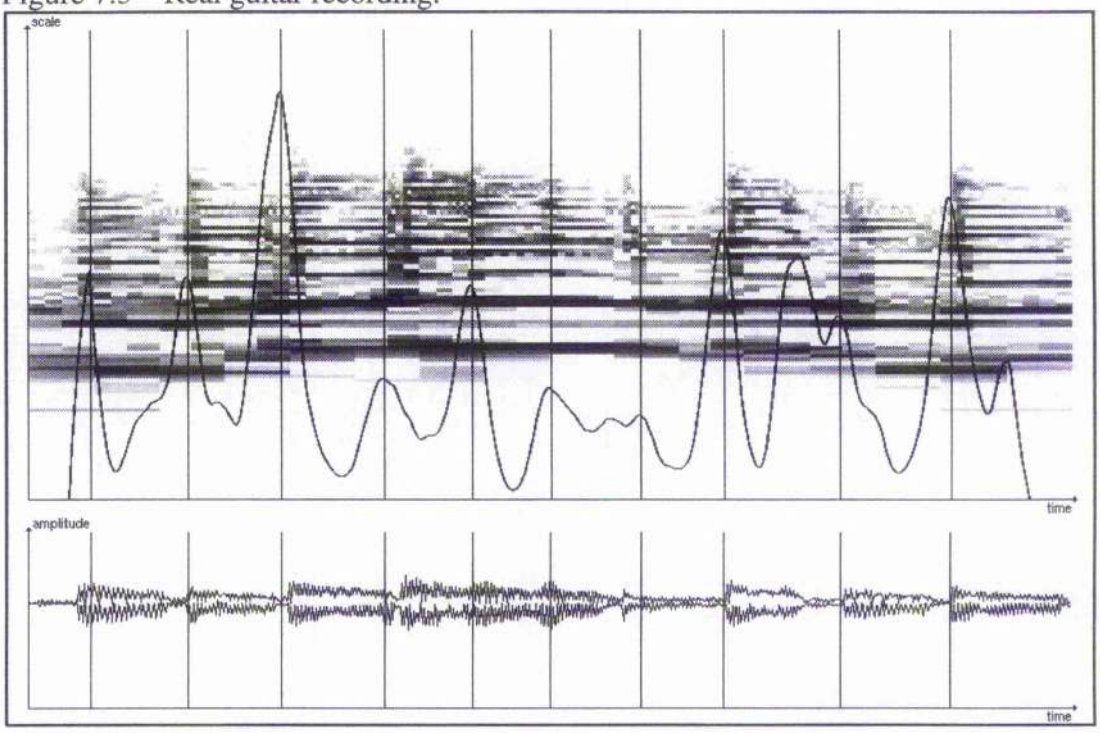
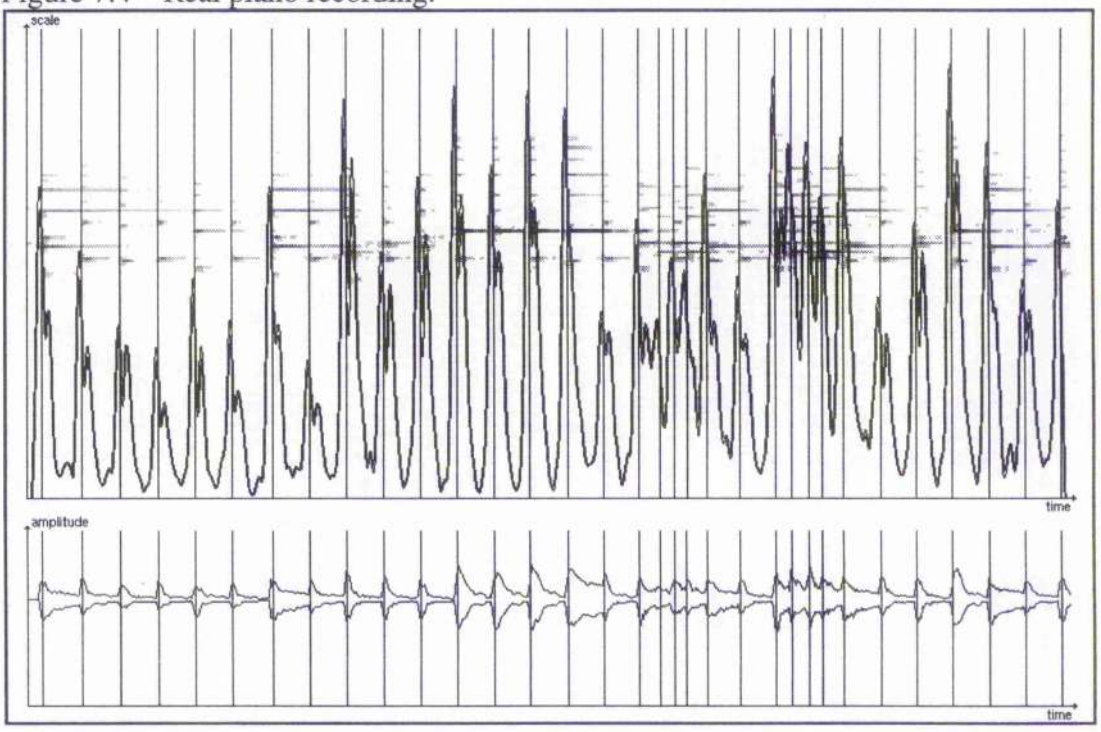
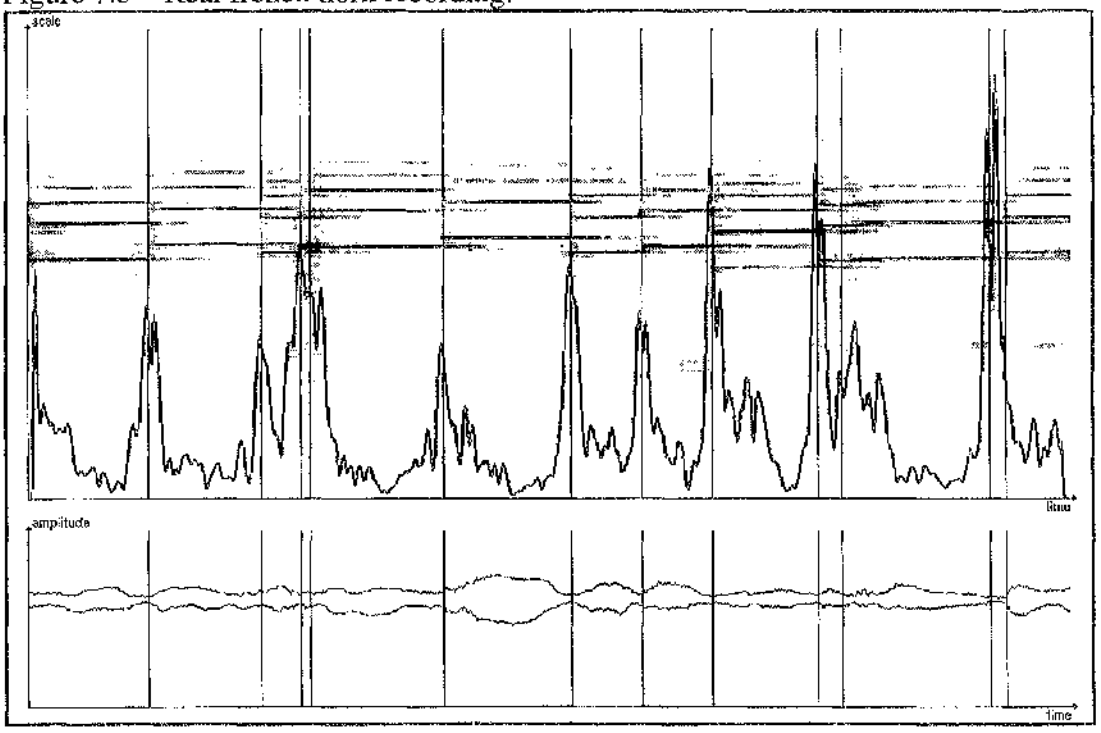


Figure 7.4 – Real piano recording.



Finally, we consider an example of a recording which could not be analysed quite as successfully. Figure 7.5 shows the results obtained when the experimental parameters for the french horn timbre were applied to a recording of an actual french horn passage (this was used for illustration previously, and the original modulus values are shown in figure 3.4). The third and last onsets are accompanied by spurious detections, both caused by artefacts from overlapping previous notes. The french horn timbre in the experiment gave rise to some of the greatest errors, and it is interesting to see that an actual recording also poses problems. A similar difficulty arises, in that some way of overcoming the slow rise times seems to be required, but the application of adaptive normalisation does not quite have the desired effect (especially in the presence of so much reverberation). In summary, the example contains several complicating factors: the style in which the piece is played, the manner in which it was recorded (giving rise to considerable environmental reverberation), and also the presence of audience noise.

Figure 7.5 – Real french horn recording.

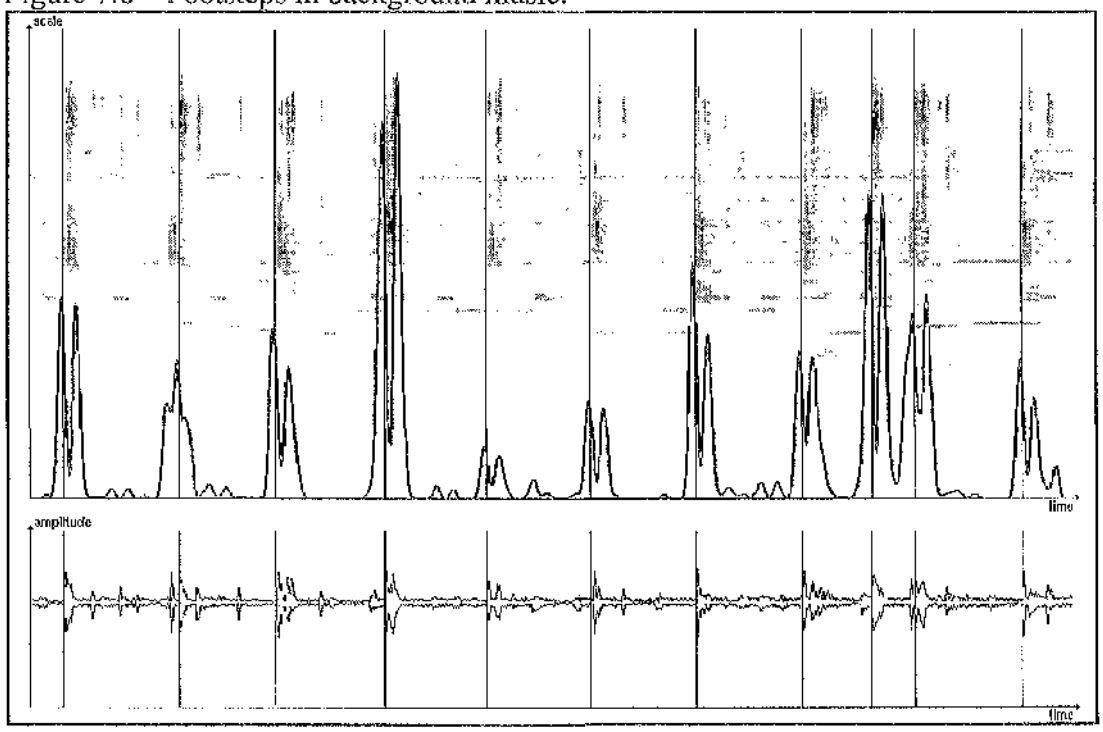


By way of a final example to illustrate the technique's flexibility, a test case previously employed in [Tait & Findlay 96] is used. The input sound is taken from [JTQ 88] and incorporates footsteps in the presence of background music. The

problem of generating timings for each of the footsteps is considered. Figure 7.6 shows the results of applying the analysis method which was identified as overall best, with parameters set as follows: *change_thresh*=72 (the average), *band_thresh*=20 (since there is background sound to be disregarded), and *num_partials*=6 (since each footstep involves many partials).

The location of each footstep is highlighted, the regular rhythm being interrupted by the scrape of a moving foot between steps eight and nine. This example illustrates the benefits of not relying on pitch perception, and shows that the method can be applied to sounds in background noise.

Figure 7.6 – Footsteps in background music.



7.4 Possible Sources of Error

An issue not yet covered in this chapter involves the quantification of any systematic errors which may be present in the experiment. The possibility that the onset times used as the actual ones in the experiment differed from the physical onset times in the audio files must be considered (the reader may recall that onset times were calculated from the observed time of the first onset). Possible sources of such error include the following.

- Timing inaccuracies introduced by the MIDI sequencer.
- Unpredictable delay in the production of sound by the synthesiser after receiving a MIDI message.
- Slight alterations in speed introduced by the process of recording onto tape before digitising.

Due to the lack of precise specifications, and the impossibility of observing every onset in the audio waveforms, it is not possible to quantify the effect of such errors. Direct observation of the first onset time, from which the others were calculated, should overcome any consistent delays.

Observation of the last onset time in several of the staccato examples gave discrepancies from the calculated onset times used in the experiment of between 8 and 110 samples. To eliminate this problem, the test cases would have to be controlled much more tightly (for example by only using waveforms in which the onsets could be observed directly, or by constructing the input files from individual notes placed at precise times), or equipment with precise specifications would have to be employed in their generation. However, this is at odds with the desire to move towards test cases involving recordings of real performances.

7.5 Conclusions

This chapter has given a detailed assessment of the results described in the previous chapter, when they are considered in various contexts. It has hopefully become apparent that it cannot simply be stated whether such an analysis technique works or does not. Different contexts present a variety of criteria, and there is often an element of subjectivity that is difficult to account for. In spite of these difficulties, a number of precise measures of success have been arrived at, and the performance of the onset detection method illustrated.

The experimental design was also vindicated in many respects -- it has been shown that different transition types give different results, as do different timbres. It is for these reasons that a wide range of examples must be considered, and we have shown that an even greater range than used here would be required to illustrate any timbral grouping by analysis parameters. It was largely due to the lack of any such existing experiment that detailed comparisons with other work proved impossible.

A persistent problem with such analysis techniques is the profusion of parameters which are often involved. Whilst an experienced user may be prepared to familiarise

themselves with a complicated system, minimising the number of parameters will lead to more readily usable applications. We therefore considered reducing the number of parameters from an initial six (which gave the results in the previous chapter) to three and then to one. This was equivalent to fixing the analysis method (leaving only the onset detection parameters variable), then reducing the number of parameters associated with onset detection. Whilst this inevitably increased some errors, table 7.3 shows that for a number of realistic applications this increase may be tolerable.

In summary, this chapter has demonstrated that the onset detection method which has been developed could be usefully employed in the automation of a variety of realistic tasks. The investigation of a wide range of timbres and unusual cases gives us some confidence in the method's utility and flexibility. Finally, it has been shown that reasonable performance can be achieved without resorting to a large number of parameters.

Chapter 8
In Conclusion...

In this final chapter, the thesis is summarised, before its impact on the current state of affairs described in chapter 2 is assessed. Ways in which the techniques described are limited, and the directions in which they may be developed in the future are then discussed.

8.1 What Has Been Achieved

Initially, the abstract task of analysing audio for some kind of internal structure was decomposed into the sub-problems of: segmentation, analysis of the identified segments, and high level interpretation. This view is widely applicable and has been utilised successfully in speech processing, but rarely in the analysis of musical audio. In a musical context, the segmentation stage is most usefully related to the task of detecting note onsets. It was these observations, along with the relative rarity of work devoted solely to onset detection, which motivated further study of the onset detection problem.

8.1.1 Harmonic Wavelet Analysis

The harmonic wavelet analysis of David Newland was selected, as it provided a particularly suitable combination of variable frequency resolution and multiple time resolutions. Also, this transform has not formed the basis of any other work in the field and this was another important reason to investigate its properties further.

The transform is easily implemented via the FFT, and we then considered enhancement of the modulus plane based on observations of human auditory perception. The result should not be thought of as an auditory model, but rather as a time-frequency representation with salient features enhanced via the implementation of a number of the observed properties of the human auditory system. Specifically, these were: weighting to account for varying frequency sensitivity and mappings to both the sone and decibel loudness scales. The application of an adaptive normalisation technique to overcome the problems associated with detecting slowly rising onsets was also described.

8.1.2 Highlighting Change and Detecting Onsets

The problem of detecting points in time at which some kind of change was taking place in the transformed modulus plane was investigated. The technique of forming two vectors representing adjacent regions of the modulus plane, and calculating a vector distance between them was found to be suitable for this purpose. Although distance measures had previously been used in assessing similarity, they had not been employed in the context of onset detection. Peaks in the distance measure provide pointers to times at which change is occurring in the modulus plane. Various types of change (including onsets and offsets) are highlighted by this method, and previously published studies concerning the nature of onsets were utilised in establishing a method of determining whether a particular change was indeed an onset.

8.1.3 Experimental Design for Analyses of Musical Audio

It was found that there had been relatively little discussion of this aspect of previous work, and this motivated the detailed description of the experimental design process in chapter 5. We focused on the problem of properly exercising an onset detection technique that would be applied to various types of musical examples. However, the guidelines which were derived regarding the identification of variables, and the subsequent construction of a suitable set of test cases, would be directly applicable in other areas of musical analysis.

Consideration of the common structure inherent in monophonic musical examples allowed the identification of a number of the sources of variability in those examples. A list of those variables and their likely ranges was given, as was a discussion of how a suitable set of timbres could be selected.

Of course, the ideal solution is not always achievable in practice and the experiment which was conducted was inevitably somewhat restricted. It was decided that timbre and transition type were of crucial importance in onset detection, therefore these variables were investigated most thoroughly, with single test cases used to show the effect of others. It was also vital that actual onset times could be ascertained, and the use of a MIDI controlled synthesiser enabled this and other aspects of the test cases to be controlled.

The final set of test cases involved a piece with a repeated note and a range of intervals, played using 12 diverse timbres, in legato and staccato styles (where applicable). Separate test cases demonstrated results in the presence of reverberation, dynamic variation, low notes, short notes, vibrato, tremolo and drum sounds (with

overlapping cymbals). Although it might be thought that such artificially controlled test cases would present a lower level of difficulty than live performances, the test cases were, in fact, artificially difficult in many ways. This was especially true of the legato pieces (which included no gaps whatsoever, when this would not be possible in a real performance), the repeated note (where a repeated attack was often not evident when no gap was present) and the additional test cases (which included extreme possibilities that would be rare in actuality).

8.1.4 Assessment of Results

In order that results can be evaluated in different contexts and compared with other work in the area, a precise statement of performance is required. In that respect, it is intended that the various methods of presentation employed in chapters 6 and 7, and the detailed results in appendix E, provide a detailed enough picture of the results obtained in the current work.

Again, previous work in the area provided few precedents for such a presentation. There was, therefore, little scope for comparison with published techniques and chapter 7 was primarily concerned with assessing the results in the context of a number of typical applications. As expected, a small number of timbres emerged as particularly problematic, and likely sources of difficulty were discussed.

After presenting the best results achieved, over a wide range of the possible parameter settings and modulus plane transformations, the possibility of restricting the method to a minimal number of parameters was considered. This is important if the method is to be incorporated in an easily usable application, and it is therefore encouraging that results were not vastly degraded and acceptable performance was maintained in a number of the application contexts.

It is important to note that, although the error tolerances set for each application were intended to be as representative as possible, there are no standards in this area. As a result, the figures quoted in the previous chapter provide a guide to the method's performance, but are not immutable. In addition, some of the difficulties in assessing performance have already been noted.

8.2 Main Contributions

It is the author's belief that the importance of the onset detection problem in a wide range of audio analyses has not been adequately acknowledged in much previous work in the field. A solution could also be usefully employed in a number of applications, some of which have been discussed in detail. Inter onset intervals

provide cues to higher level musical structure, onsets are crucial in timbre classification, and onset locations can guide further analyses such as pitch identification. Therefore, a reliable onset detection technique, applicable to a wide range of timbres in various situations would impinge on many of the areas of research already identified.

Although the subtleties in properly assessing results have been noted, the previous chapter has demonstrated that the technique which has been developed could be applied in a range of applications, with a high degree of reliability and accuracy. In addition, this has been achieved without incorporating a large number of parameters. It has also been shown that the method is flexible enough to be applied to non-monophonic examples, and this idea will be expanded in the following sections.

We first turn our attention to the detailed components of the work which has been presented, describing the novel aspects at each stage.

8.2.1 Harmonic Wavelet Analysis

It is only relatively recently that wavelet transforms have been employed in the context of audio analysis. In particular, harmonic wavelet analysis has hardly been investigated in that context at all.

The current work has shown that the attractive theoretical properties of harmonic wavelet analysis can be exploited in practical applications, and that its weaknesses, in terms of time resolution, are not insurmountable.

The combination of wavelet analysis and perceptually motivated transforms has also been demonstrated, and has been found to improve the results of further analyses.

8.2.2 Highlighting Change and Detecting Onsets

The use of a distance measure in detecting onsets was central to the work. Minkowski's measure (with a range of exponents) was used, and various kinds of change in the audio have been shown to give rise to peaks in distance.

This represents a novel technique for detecting change, but to restrict the results to note onsets, classification of the distance peaks was required. Methods of seeking partial onsets have been investigated previously in the context of onset detection, but the prior use of a change detecting measure restricts further analysis to relevant points in time. There are a number of other advantages in adopting this technique. For example, the broadband sensitivity of the change detection method means that a collection of partials commencing within a short space of time, even if they are not rapidly rising, will produce a peak in vector distance. Also, as was illustrated in the

discussion concerning the reduction of the number of parameters, the partial onset detection phase becomes less critical, since it has already been established that some event has taken place.

Finally, the range of parameters and analysis techniques investigated clearly illustrated their effect in this context, which should provide pointers for future work.

8.2.3 Experimental Design for Analyses of Musical Audio

The design of a corpus of test cases became an integral part of the work at an early stage. This was primarily because there seemed to be no such set of benchmark pieces already in existence, and this lead to some difficulty in evaluating and comparing previous work on audio analysis.

The framework presented in chapter 5 can therefore be seen as a first step in the establishment of a widely available body of pieces which could be used in evaluating new analysis techniques. Of course, the requirements of different techniques will vary, but the promotion of such practices is crucial if the field is to evolve.

8.2.4 Assessment of Results

Having established a set of test cases, assessment of the results obtained was of crucial importance. Again, there appeared to be too little emphasis placed on this aspect of previous work, and the previous chapter was intended to give as clear a picture of the results achieved in the current study as possible.

A number of traditional techniques were employed, but a number of domain specific metrics were also defined, based on the techniques employed and various potential applications. These showed that the restricted time resolution of the harmonic wavelet analysis was not as serious a restriction as might have been anticipated, and also that the errors which were encountered were not prohibitive when considered in the context of a number of realistic applications.

Further, it is hoped that the way in which the results were considered suggests ways in which other audio analysis techniques might also be evaluated.

8.3 Limitations

Before describing ways in which the current work might be advanced, it is important to consider the limitations of the techniques which have been described. Any assumptions made during the course of the work were stated at the appropriate places, and a few limitations in scope have already been mentioned.

8.3.1 Type of Onsets Detectable

The method relies on a note onset introducing new elements in the modulus plane within a short space of time. Although results were previously achieved with tones consisting of only a single partial, and very slowly rising envelopes (see [Tait & Findlay 95]), these are the types of timbre to which the method is least well suited. The results in chapter 6 have verified this to some extent, and there are bound to be some timbres for which the deterioration in performance would be unacceptable. It is also likely that if partials started over an extended period of time, the peak position would be misleading and the onset difficult to verify. The questions of how accurately a listener would be able to pinpoint the onsets in such cases, and whether the decrease in accuracy has less impact for such timbres, have no clear answers.

8.3.2 Extension to Polyphonic Examples

Perhaps the most interesting extension of the work would be to polyphonic input. It has been shown that results can be achieved in the presence of overlapping notes, and the method should also be somewhat resistant to background noise. However, there is one restriction which becomes apparent when polyphonic examples are considered. It has been explained that, in the monophonic case, the window size used in calculating average vectors for the distance measure should be less than half the length of the shortest note in the input. What this really means is that, ideally, no two onsets in the input should be closer than the combined length of the left and right windows.

In the polyphonic case, if this condition holds then peak detection and classification could be attempted as normal. However, it is likely that there will be coincident onsets, and onsets occurring very close together. The problem would then be to set a suitable window size, such that every onset was highlighted, whilst implementing some method of identifying coincident (and almost coincident) onsets. In any case, peaks will appear that correspond to a number of onsets which have occurred within such a short space of time that only one peak is evident.

8.4 Future Directions

It is hoped that the previous sections have demonstrated that the work described herein constitutes a valuable addition to research in the field, as well as making clear its limitations. It is hoped that the thesis will provide inspiration for further work, and in this final section, ways in which the various threads could be pursued in the future are discussed.

8.4.1 Harmonic Wavelet Analysis

The theoretical underpinnings of harmonic wavelet analysis have only been established relatively recently. It is therefore possible that advances in this area could facilitate improvements in the results presented here, especially since the method does not rely on any peculiarities of the time-frequency decomposition.

For example, the transform is not time invariant, in that identical inputs translated by different time periods will give rise to slightly different coefficients. A way of overcoming this is described in [Newland 95], but it was felt that the repeated calculation of the transform introduced too great a computational overhead. There is therefore scope for investigating the effect of using the time invariant version of the transform.

In addition, [Newland 94A] indicates that subsequent work will address the potential time resolution problems currently associated with harmonic wavelets. In the current context, this might be achieved by simply lowering the frequency resolution. If the frequency bands were wider than a single semitone, time resolution would be improved and it is also possible that the problems encountered with vibrato would be alleviated. In this situation, the gap between higher harmonics would be relied on to distinguish consecutive notes whose fundamentals were encompassed by the same frequency band.

There is also considerable potential for extending the perceptual transforms applied to the modulus plane, if that area was of particular interest. It has been explained why this aspect of the work was not heavily emphasised; however it is possible that a full auditory model could be based on the harmonic wavelet transform. For example, the division of the frequency scale can be controlled, and could be brought even closer to that observed in the auditory system (as in [Brookes et al 96]). There are also various other observed phenomena such as masking and so on, which could be mimicked in the modulus plane.

In this respect, the adaptive normalisation procedure may merit further investigation. It does not relate directly to any psychoacoustic phenomenon, however it has been shown to improve results with slowly rising timbres, and its refinement may lead to the reduction of errors for these worst case timbres.

8.4.2 Highlighting Change and Detecting Onsets

When the method is broken down into its constituent stages, the ways in which it might be augmented become more apparent. For example, the use of smaller windows was shown to give more accurate results in some cases, although it was not

appropriate in others. This suggests that greater accuracy could be obtained if some method of automatically detecting the optimal window size for a particular example could be devised. If the input could be characterised in this way, we might also then consider automating the selection of the onset detection parameters, or even adjusting these during the course of a single analysis.

Although this may seem overly optimistic, there was one source of information which was not explored in depth. Chapter 4 showed that different timbres (specifically, a rim shot and a french horn) produced markedly different types of peak in their distance functions. The percussive sound had two prominent and narrow peaks (one onset and one offset); whereas the slower onset generally gave rise to wider, more complicated peaks. It is therefore possible that the distance function as a whole may provide interesting information about the input audio, aside from just peak locations. For example, the shape and extent of peaks may guide further analyses.

A range of exponents were tested when calculating the distance, but there are other distance measures aside from Minkowski's measure. Although [Feiten & Gunzel 93] found that most suitable, the context was somewhat different and the investigation of other distance measures for onset detection would be of interest.

Finally, we might consider what the distance measure was comparing. Calculation of the average vector in each window met most of the requirements of onset detection, but how might changes such as those in the example with glissandi be highlighted? Due to the continuous change in this case, the form of the peaks was again quite different to that in other cases, and further study here may provide clues. If we wish to disregard such smooth changes (or categorise them in a more precise fashion), the comparison between windows must be more elaborate. For example, if we could identify lines of significant modulus values in the modulus plane, a distance measure could be calculated based on whether these continue or are broken at a specific point in time. This idea can be related to the synchrony strands of [Cooke 93]. An implementation would both give rise to negligible distance in the presence of glissandi and continuous variations, whilst providing information on their shape.

Of course, this is considerably advanced from the current work and simpler extensions to the distance measure might also be considered. For example, the modulus values could be weighted based on the idea that those near the point in time under consideration are more important than those at the outer edge of each window. This would be a first step towards incorporating relevant aspects of the partials' timing in the analysis.

Lastly, the identification of other types of change has not been considered – offsets were often visible in the musical examples, and the application of the method to data other than sound would be possible.

8.4.3 Experimental Design for Analyses of Musical Audio

The experimental design of chapter 5 was intended to be sufficiently detailed for the task of exercising many analyses based on monophonic musical audio. Continuation of this process should be aimed at utilising the idea for other types of analyses, and constructing similar such designs for analyses of polyphonic examples.

The most obvious way in which the test cases themselves might be altered is via the use of recordings of real performances. Ideally, ways in which such performances might be captured along with precise timing information should be investigated. For example, sensors which can be attached to musical instruments are sometimes used to synchronise other aspects of a performance, and the expertise accumulated in this field may provide a solution. Another possibility may be the use of software based physical models of real instruments, to provide a combination of realism and controllability.

Also, although the domain was restricted to monophonic musical examples, some potential in the analysis of polyphonic examples has been demonstrated, and previously published examples have included the location of footsteps in background music [Tait & Findlay 96]. This suggests that further work might include detailed investigations of performance in such situations.

8.4.4 Assessment of Results

Assuming that the audio examples used are accompanied by precise timing information, the biggest problem in assessing results is deriving suitable metrics for the purpose. Only the synchronisation case could be precisely quantified with some confidence, and even then we saw that different timbres may imply different tolerances. This may seem unusual, however it seems reasonable that the precise timing of a timbre with a long rise time should not be as crucial as that of a percussive timbre.

What is required is a study of the type of applications which arise, and the levels of accuracy demanded by typical users of those applications. Such a study would provide greater insights than considerations of auditory acuity, or the theoretical scenarios of the previous chapter.

Another aspect of this part of the work which is open to further study is the comparison of various existing techniques based on some single benchmark set of test

cases. This task has been pursued with various high level analyses (see [Desain & Honing 94]), and would also be of value in the context of low level audio analyses.
Finally, we have only considered assessing the results for their accuracy. However, if a large body of test cases were analysed, useful information concerning the best parameter sets and types of analysis for different timbres may be amassed.

Appendix A
Custom Voices

Table A.1 – Voices I5.1 (Cello), C4.2 (Steel Drum) and I8.8 (Timpani).

Parameter	Cello	Steel Drum	Timpani
Common			
Configuration	A-B	A-B	A-B
Effect Dep	0	0	0
Wheel PM	Off	Off	Off
Envelope AR	+68	+8	
Vector			
Level		Y-18	
Detune		X+7, Y-5	
Element Tone A			
Wave	040	007	049
Freq Shift		+12	+12
Velocity Sense	+3	+1	+1
LFO AM	11	0	
LFO PM	10	0	
LFO Rate	44		
LFO Spd	6	21	
Element Tone B			
Wave		078	
Freq Shift		-12	
Volume	0		0
Velocity Sense		+1	
LFO PM		0	

Table A.1 gives parameter settings for the non-preset SY22 voices used in the experiment of Chapter 6. These were Cello (I5.1), Steel Drum(C4.2) and Timpani

(I8.8). If a voice is initialised (by selecting 'init. voice' in the setup menu), then adjusting the parameters specified above to the values shown gives the timbre which was used in the experiment. Parameters which remain unchanged, or become irrelevant as a result of other settings, are not specified.

Appendix B
Algorithms and Implementation

This appendix gives more detailed descriptions of the algorithms mentioned throughout the text, and discusses some implementation issues. Pseudo-code is given, however detailed error checking and so on are not shown. Of course, there will be many ways of accomplishing the same tasks, however it is hoped that the methods presented here are readily understood. All of the analyses were implemented in the application described in appendix D, which also provided screen shots and the framework in which to conduct the main experiment.

B.1 Harmonic Wavelet Analysis

The input audio is converted to the application's internal signal format in a representation-dependent fashion, such that the maximum possible sample value is mapped to 1 and the minimum to -1. The resulting signal is used for display and further analysis.

The first step of the wavelet analysis is to calculate a Fourier Transform of the signal. The FFT algorithm (from [Press 92]) expects $N = 2^m$ input samples, so the selected sample range is first zero-padded to the next power of 2, and we will denote this signal $x(t)$. The discrete Fourier transform is then as follows.

$$X(k) = \sum_{t=0}^{N-1} x(t) e^{-j2\pi kt/N}$$

The non-negative part of the spectrum contains $2^{(m-1)} + 1$ complex coefficients, three of which are copied directly (the purely real values representing zero and the Nyquist frequency, and the first complex coefficient). The remainder are partitioned into frequency bands – various schemes are possible, and octave bands (doubling in size) were used initially. However, these were then further partitioned into 12 semitone bands, each having $2^{\frac{1}{12}}$ as many coefficients as the previous. Of course, this raises a number of issues: firstly, not all octave bands contain enough coefficients to be subdivided into semitones and also, the size of many bands will not be an exact integer. These are resolved by only subdividing octaves in which there are enough coefficients to create semitone bands of at least 2 coefficients, and rounding the limits

of the semitone bands to the nearest integer (this introduces only a small error, as discussed in [Newland 94A]).

The FFT is calculated in place, and the first complex coefficient will be at position 3 in the original signal array (indexed from 1..N). This is considered to be the first octave scale level (Θ_1) in the wavelet transform, consisting of only one coefficient, and there are $m-1$ octaves in total. The first index and number of coefficients in a given octave Θ_i can be calculated as follows.

$$octave\_start\_index(\Theta_i) = 2^{i-1} + 2$$
$$octave\_num\_coeffs(\Theta_i) = 2^{i-1}$$

Given this information, we can test whether an octave has enough coefficients to be subdivided into semitones (others are disregarded in further processing). It happens that octave number six, with 32 coefficients commencing at index 34, is the first such octave.

The first index in an octave to be subdivided is also the first index in the first semitone band in that octave. If a semitone band in octave Θ_i is denoted S_{ij} (where j ranges from 1 to 12), then the first index of each semitone band can be calculated as follows.

$$semitone\_start\_index(S_{ij}) = round(octave\_start\_index(\Theta_i) \times 2^{\frac{j-1}{12}})$$

An inverse FFT is then performed on each semitone band thus delimited, giving a number of complex wavelet coefficients representing equal-length periods of time in the original signal. In semitone band S_{ij}, the wavelet coefficient S_{ijc} could be calculated from the Fourier components in the band as follows.

$$S_{ijc} = \frac{1}{N_{ij}} \sum_{k=start}^{start+N_{ij}-1} X(k) e^{-j2\pi kc/N_{ij}} \quad \text{where} \quad start = semitone\_start\_index(S_{ij}),$$

N_{ij} is the number of coefficients in the semitone level as calculated from successive start indices, and coefficients are numbered from zero.

Again, the FFT algorithm used expects the length of its input to be a power of two, so zero padding may be required. Assuming that FFT and IFFT subroutines are available, that will allocate memory and perform zero padding to the next power of two as required, the wavelet analysis stage can be accomplished as in the following pseudo-code.

```
frequency, num_coeffs : array(1..max_scale_levels) of Integer;
-- store corresponding frequency and number of coefficients for each scale level
type Scale_level is array (Integer range <>) of Float;
```

```
scale : array (1..max_scale_levels) of Scale_level;
...
subdivisions:=12;
FFT(X[1..N]);
level:=1;
num_octaves:=log2(N)-1;
for octave in 1..num_octaves loop
   octave_start:=(2**(i-1))+2;
   octave_num_coeffs:=2**(i-1);
   if (octave_num_coeffs>(subdivisions*2)) then
      for semitone in 1..subdivisions loop
         actual_start:=octave_start*(2**((semitone-1)/12));
         next_actual_start:=octave_start_index*(2**((semitone)/12));
         semitone_start:=round(actual_start);
         next_semitone_start:=round(next_actual_start);
         num_coeffs[level]:=next_semitone_start-semitone_start;
         scale[level]:=IFFT(X[semitone_start..next_semitone_start-1]);
         frequency[level]:=actual_start;
         level:=level+1;
      end loop;
   else
      --disregard these octaves
   end if;
end loop;
max_level:=levels-1;
```

The reader will note the derivation of associated frequency from the calculated start index of each semitone band in the above (for later use). The relationship of the harmonic wavelet decomposition to the equal-tempered musical scale is stated in [Newland 94A] as follows. Given the customary tuning of A to 440 Hz, middle C would have a frequency of 261.6 Hz. Conveniently, the closest band in the harmonic wavelet analysis covers the range 256 to 271.2 Hz (the nearest actual starts), thus straddling the expected value (other notes and semitone bands are related similarly).
At this point, the wavelet coefficients have been calculated and are available for further processing. However, much of this processing utilises time slices from the time-frequency plane. The derivation of these requires calculation to traverse the variable resolution scale levels. As this is done so often, a regular two-dimensional structure is created in which each scale level contains the same number of coefficients as the highest resolution level. This is done by simply replicating coefficients in lower scale levels, and results in an array of coefficients HWA_{TS} where the possible values

of T (time) and S (semitone) range from 1 up to the maximum values calculated, according to the properties of the input audio. Continuing the above code in a similar fashion, this could be expressed as follows.

```
type Vector is array(1..max_scale_levels) of Float;
HWA : array(1..max_time) of Vector;
...
time_divs:=num_coeffs[max_level];
for level in 1..max_level loop
   factor:=time_divs/num_coeffs[level];
   for coeff in 1..num_coeffs[level] loop
      coeff_start:=(coeff-1)*factor;
      for offset in 1..factor loop
         time:=coeff_start+offset;
         HWA[time][level]:=scale[level][coeff];
      end loop;
   end loop;
end loop;
```

An associated array of modulus values can now be calculated from each complex coefficient in the usual fashion.

$$Mod_{TS} = \sqrt{\mathrm{Re}(HWA_{TS})^2 + \mathrm{Im}(HWA_{TS})^2}$$

The modulus values are then normalised for further processing and display (discussed further in appendix C), by finding the maximum value in the whole plane, and dividing every modulus value by it. The following code fragments illustrate the calculation of modulus values combined with finding the maximum, followed by normalisation.

```
mod : array(1..max_time) of Vector;
...
max_mod:=0;
for level in 1..max_level loop
   for time in 1..time_divs loop
      mod[time][level]:=
            sqrt(Re(HWA[time][level])**2+Im(HWA[time][level])**2);
      if mod[time][level]>max_mod then
         max_mod:=mod[time][level];
      end if;
   end loop;
end loop;
...
```

```
for level in 1..max_level loop
   for time in 1..time_divs loop
      mod[time][level]:=mod[time][level]/max_mod;
   end loop;
end loop;
```

As is stated in [Newland 95], the wavelet coefficients are derived using two applications of the FFT on the input data, so that (since the modulus calculation and normalisation are $O(n)$) the whole algorithm is $O(n\log(n))$.

B.2 Perceptual Transforms

This section describes the details of the loudness scales and weighting applied to the modulus plane.

B.2.1 Loudness Scales

Two units of loudness were investigated: the decibel and the sone. The number of decibels corresponding to some amplitude A is calculated as follows.

$$decibels = 20\log_{10}\left(\frac{A}{A_0}\right)$$

Usually, A_0 is a reference amplitude intended to represent the threshold of hearing (at some reference frequency), however the normalisation applied to the modulus plane made such a reference inappropriate. Therefore, A_0 was set to the smallest amplitude which would be mapped to a grey value for display (as it happens, 50 grey levels were used so that $A_0 = 0.02$).

The following formula is used to calculate a number of sones from some amplitude A [Stevens 55].

$$sones = kA^{0.6}$$

In general, k is dependent on units used – again, there was no obvious candidate so it was set to 1.

To implement these scales, each modulus value Mod_{TS} is treated as an amplitude and mapped to a loudness value according to one of the above formulae. This involves a straightforward traversal of the modulus plane as follows.

```
A_zero:=0.02;
for level in 1..max_level loop
   for time in 1..time_divs loop
      mod[time][level]:=20*log10(mod[time][level]/A_zero);
      -- e.g. decibel scale
```

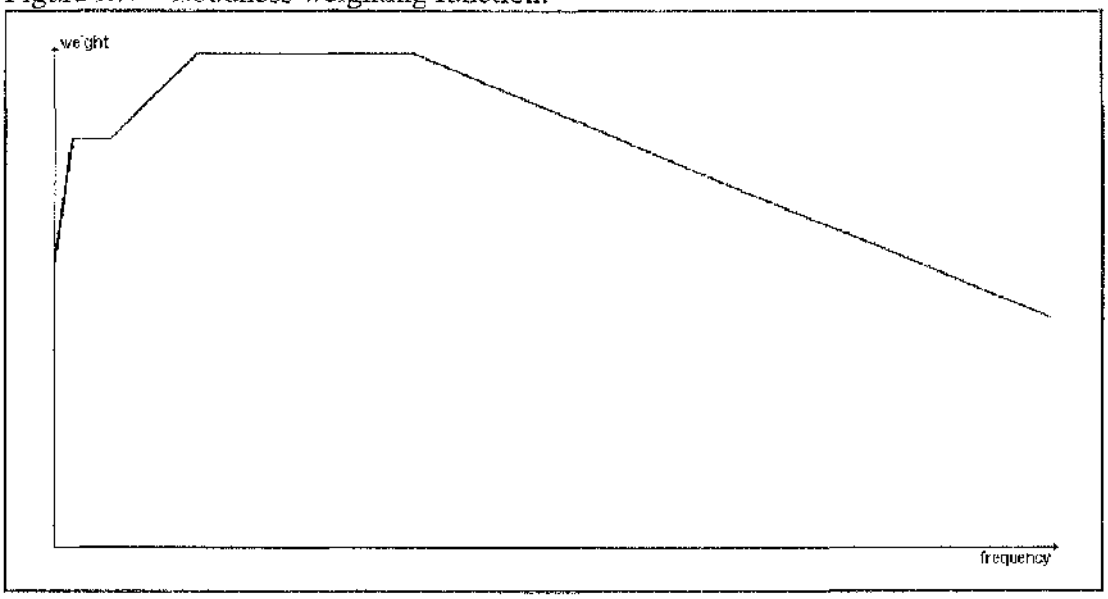
```
    end loop;
end loop;
-- renormalise, as above
```

Finally, the modulus plane is re-normalised (as above) after mapping to either loudness scale. The whole process involves first performing one of the above calculations for each modulus value (during which, the maximum is again found), and then dividing each value by the maximum to re-normalise the plane (so that it is $O(n)$).

B.2.2 Equal Loudness Weighting

Figure B.1 graphs the loudness weightings which were applied to the output of the wavelet analysis. The graph shows weights for frequencies ranging from 0 Hz to 22050 Hz.

Figure B.1 – Loudness weighting function.



This piece wise linear approximation was derived in [Stevens 72] from the results of many studies of loudness perception, and the original graph in that paper was used to calculate the breakpoints in the following table.

Table B.1 – Frequency-weighting breakpoints.

frequency	80 Hz	400 Hz	1250 Hz	3150 Hz	8000 Hz	12500 Hz
weighting	0.62	0.83	0.83	1.0	1.0	0.83

A function which would cover the required range of frequencies was created by linear extrapolation of the first and last line segments to the high and low values not covered in the survey. The scale levels in the wavelet analysis are then traversed, the centre frequency of each being used to derive a weighting, which is then applied to the modulus values in that semitone band. Obviously, this process is $O(n)$, so that it does not adversely affect the efficiency of the overall analysis (pseudo-code is given below).

```
function weight(freq:Float) return Float is
   Float w1:=0.62,w2:=0.83,w3:=1.0;
   Float f1:=80,f2:=400,f3:=1250,f4:=3150,f5:=8000,f6:=12500;
begin
   if (freq>f6) then -- extrapolated
      return w2-(((w3-w2)/(f6-f5))*(freq-f6));
   elsif (freq>f5) then
      return w3-(((freq-f5)/(f6-f5))*(w3-l2));
   elsif (freq>f4) then
      return w3;
   elsif (freq>f3) then
      return w2+(((freq-f3)/(f4-f3))*(w3-l2));
   elsif (freq>f2) then
      return w2;
   elsif (freq>f1) then
      return w1+(((freq-f1)/(f2-f1))*(w2-l1));
   else -- extrapolate
      return w1-(((w2-w1)/(f2-f1))*(f1-freq));
end weight;
...
for level in 1..max_level loop
   for time in 1..time_divs loop
      mod[time][level]:=weight(frequency[level])*mod[time][level];
   end loop;
end loop;
-- renormalise, as above
```

B.3 Adaptive Normalisation

This technique was outlined in §6.2.1, and a detailed description of the algorithm is now given.

The modulus plane is scanned along the time axis, and the maximum modulus value at each time position is found. Initially, the axis was divided into windows, but the technique was found to be most useful when this stage was as responsive as possible, so time slices (at the resolution of the highest scale level) were eventually used. In other words, for each time slice Mod_τ (including all possible values of S), a maximum value Max_τ is found.

These maximum values are then used as a guide to scaling the modulus values. To ensure that discontinuities are not introduced, smoothing is again used (a small number of adjacent maxima are averaged at each point). The maxima are then traversed, and if a value is above some threshold (representing silence), all the modulus values at that point in time are divided by it (so that the maximum itself is scaled to 1). So that artificial discontinuities are not introduced, maximum values less than the threshold are divided by $1+\left(\frac{threshold-1}{threshold}\times maximum\right)$ which goes from 1 to *threshold* as the maximum goes from 0 to *threshold*. Pseudo-code for this process is given below.

```
maxima : array(1..max_time) of Float;
...
for time in 1..time_divs loop
   maxima[time]:=0;
   for level in 1..max_level loop
      if mod[time][level]>maxima[time] then
         maxima[time]:=mod[time][level];
      end if;
   end loop;
end loop;
...
smooth(maxima[1..time_divs]);
...
for time in 1..time_divs loop
   if maxima[time]>:=threshold then
      divisor:=maxima[time];
   else
      divisor:=1+(((threshold-1)/threshold)*maxima[time]);
   end if;
   for level in 1..max_level loop
```

```
            mod[time][level]:=mod[time][level]/divisor;
        end loop;
    end loop;
```

This involves 2 passes over the modulus plane and, like the other transformations, is $O(n)$.

B.4 Vector Distance

When implementing analyses of the wavelet transform, its multiresolution nature presents some interesting options. It has been explained that the coefficients are treated as a rectangular array, with redundancy at lower resolution scale levels. Based on the description of the method already given in chapter 4, it is evident that if a given scale level has coefficients whose extent in time is greater than the windows used in the analysis (for example), then that level will affect the vector distance much less than higher resolution levels (since it changes less often). Thus, there is also redundancy in the calculation which could be exploited to give more efficient algorithms. That said, pseudo-code for the most straightforward algorithm will be given.

Calculation of vector distance involves first deriving average vectors from the two windows under consideration (note that these are stored for the onset detection phase). Repeatedly calculating the average would give an algorithm approaching $O(n^2)$ (for long windows), so a running average is used which keeps the calculation $O(n)$ (as shown below).

```
function divide_vector(v,d) return Vector is
    -- divide each element of V by d

-- the following operations are defined for pairs of vectors V1 and V2 with equal
numbers of elements

function sum_vectors(v1,v2) return Vector is
    -- result_vector[i]:=v1[i]+v2[i]

function subtract_vectors(v1,v2) return Vector is
    -- result_vector[i]:=v2[i]-v1[i]
...
function minkowski(v1,v2:Vector;exp:integer) return Float is
    float  sum:=0;
begin
    for element in 1..num_elements loop
```

```
        sum:=sum+(abs(v2[element]-v1[element])**exp);
    end loop;
    -- apply root (not in adopted method)
    return sum**(1/exp);
end minkowski;
...
vector_distance : array(1..max_time) of Float := (1..max_time=>0);
left_avg_vec,right_avg_vec,left_sum_vec,right_sum_vec : Vector;
...
left_sum_vec:=sum_vectors(mod[1..window_size]);
right_sum_vec:=sum_vectors(mod[(window_size+1)..(2*window_size)]);
for time in (window_size+1)..(time_divs-window_size) loop
    left_avg_vec[time]:=divide_vector(left_sum_vec,window_size);
    right_avg_vec[time]:=divide_vector(right_sum_vec,window_size);
    vector_distance[time]:=minkowski(left_avg_vec[time],
                                                    right_avg_vec[time],p);
    left_sum_vec:=subtract_vectors(left_sum_vec,mod[time-window_size]);
    left_sum_vec:=add_vectors(left_sum_vec,mod[time]);
    right_sum_vec:=subtract_vectors(right_sum_vec,mod[time]);
    right_sum_vec:=add_vectors(right_sum_vec,mod[time+window_size]);
end loop;
```

B.5 Peak Detection

A detailed description of the peak detection method was also given in chapter 4. However, a pseudo-code version of the algorithm is included for clarity.

The verification of a potential peak at time t involves simply checking the peak height against a percentage threshold, and counting the number of successive decreases in amplitude on each side of the peak (as shown below, using parameters as in the text).

```
if ((vector_distance[t]*100)>peak_thresh) then
    i:=1;
    while ((vector_distance[t+i]>vector_distance[t+i+1]) and
                (vector_distance[t-i]>vector_distance[t-i-1])) loop
        i:=i+1;
    end loop;
    if (i>peak_reqd) then
        -- peak at time t
    end if;
end if;
```

B.6 Peak Classification

Again, the peak classification technique was explained in chapter 4. However, a precise version of the algorithm is presented here. Given a potential onset time t, and onset detection parameters as defined in the text, the following pseudo-code segment implements the onset detection method (note that the average vectors stored in the vector distance calculation are utilised).

```
partial_onsets:=0;
if (t-last_onset time>min_len) then
   for level in 1..max_level loop
      if (right_avg_vec[time][level]>band_thresh) then
         diff:=right_avg_vec[time][level]-left_avg_vec[time][level];
         pc_change:=(diff/left_avg_vec[time][level])*100;
         if (pc_change>change_thresh) then
            partial_onsets:=partial_onsets+1;
         end if;
      end if;
   end loop;
   if (partial_onsets>num_partials) then
      -- onset at time t
   end if;
end if;
```

A potential problem with the above algorithm arises when clusters of nearby peaks (separated by less than *min_len*) are considered. If a number of nearby peaks would all qualify as onsets, the above technique selects the first. However, it could be argued that the highest peak in such a group should be selected, and this was the method which was actually used.

Appendix C
Formats and Conventions

This appendix describes the data formats and graphing conventions which are used throughout this document.

C.1 General Graphing Conventions

All of the analyses were conducted using the application described in Appendix D, and all of the resulting graphs (and audio waveforms) were screen shots from that application. In addition, it should be noted that all of the graphs are normalised, both horizontally and vertically. This eases comparisons, but implies that different scales are relevant to each. Partly for this reason, detailed scales are not shown, although indications of the extremes on the axes are given.

C.2 Audio

All of the audio examples consist of 16-bit samples at a rate of 44.1 kHz, which were captured using the inbuilt hardware on a Sun SPARCstation-10 and stored as Sun audio files. The graphs (of amplitude against time) were produced by plotting the maximum and minimum samples in a sliding window as in [Foster et al 82]. Normalisation is applied, so that the graphs are not all to the same scale — they are shown the same size for clarity and ease of comparison with diagrams of other analyses, although the sounds were not all of the same duration.

C.3 Wavelet Analyses

The wavelet analyses show semitone level against time. For each complex coefficient, the modulus value is calculated and mapped to a grey scale level. A filled rectangle is then drawn, having relatively the same extent in time and frequency as the corresponding coefficient. Having said this, the semitone levels often do not cover exactly the same number of pixels on the Y-axis. This is because normalisation is again employed, and rounding implies that the vertical extent of semitone levels in the graphs may vary slightly. Finally, the grey levels are converted to dot densities when printed in black and white.

C.4 Vector Distance

The graphs of vector distance against time are also normalised, so that the height of peaks in different graphs cannot be compared. Another point to note in these graphs is that they are plotted so as to occupy the same horizontal space as the waveforms and modulus planes. This implies some degree of zero padding at the beginning and end (since vector distances are not calculated when either the left or right window would extend beyond the end of the data).

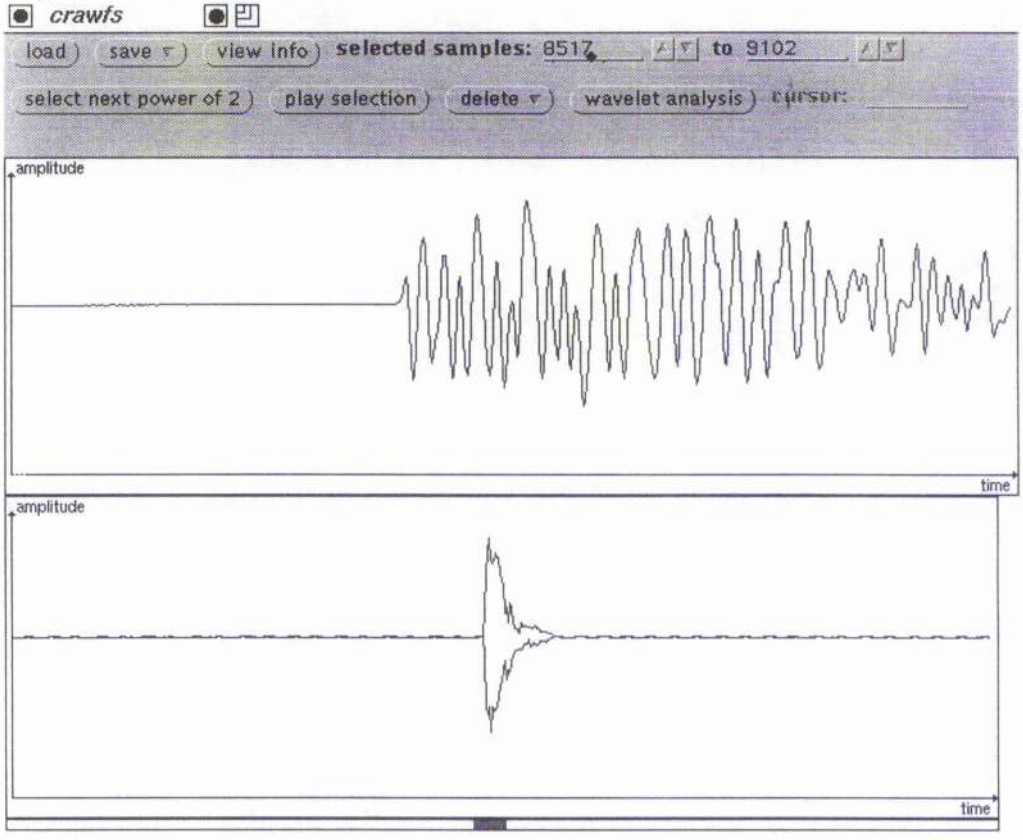
Appendix D
Application Guide

All of the analyses and almost all of the diagrams in this work were generated using an application written by the author. This appendix describes its operation (note that the conventions used in displaying the various types of data are described in appendix C).

D.1 The Main Window

Figure D.1 shows the main window, which allows the user to view and edit an audio waveform, as well as selecting portions for further analysis.

Figure D.1 – Main window.



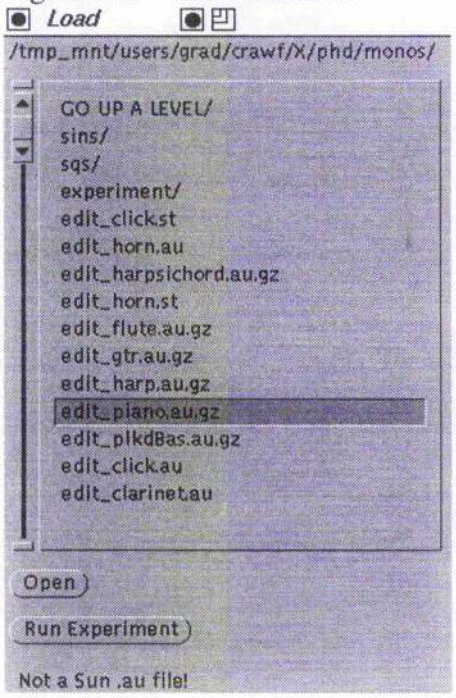
The grey bar beneath the lower waveform is the *selection indicator* , showing which part of the audio is currently selected. Above that, the *global display* always shows the whole envelope of the currently loaded sound (or a test waveform initially), whilst a close up of the current selection is shown above in the *zoom display*. Lastly, the uppermost panel gives access to various functions, described in the following subsections.

D.1.1 Loading A Sound

Clicking the *load* button displays a window like that in figure D.2. This shows the current directory and its contents. Navigation is achieved by clicking on a directory name (or 'GO UP A LEVEL/') followed by clicking *open*.

Sound or state files (see D.1.6) are opened similarly. An error message will be displayed on attempting to open a sound file which is not of a supported type (currently supported are mono .au files of either 8KHz ulaw encoding, or 44.1KHz 16 bit encoding). State files are recognised by suffix only.

Figure D.2 – Load window.



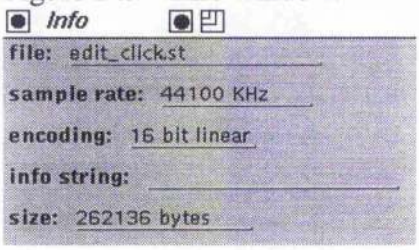
The button labelled *run experiment* is used to run the experiment described in Chapter 5 in the current directory. This involves opening every file with the suffix '.ons' — these are expected to contain a list of actual onset times. For all such files, the

corresponding state file is opened, and a range of analyses conducted (as described in chapter 6). The results are summarised in a number of files left in the current directory. If the state file is not present, the corresponding audio file is opened and a state file generated.

D.1.2 Viewing Information About the Sound

Clicking the *view header info* button displays a window like that in figure D.3, showing the information contained in the currently loaded file's header (this cannot be edited).

Figure D.3 – Info window.



D.1.3 Adjusting the Selection

The current selection can be altered in a number of ways. The simplest is by clicking the left and middle mouse buttons in either the *global display* or the *selection indicator* to set the beginning and end of the selection (respectively). The pair of sample values representing the current selection are shown in the uppermost panel, and these can be edited directly by clicking on them and typing.

Lastly, the wavelet analysis (described in appendix B) is designed to take input containing a number of samples equal to some power of two. This is not crucial (as zero padding takes place when necessary), however most efficient use of the wavelet analysis is achieved by first clicking the button labelled *select next power of 2*. This simply extends the current selection (if possible), so that it contains a number of samples equal to some power of two.

D.1.4 Editing the Sound

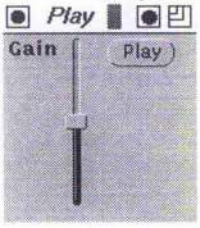
Three types of editing are possible via a pop-up menu accessed by clicking on the button labelled *delete*▽. The sound can be trimmed by choosing *delete left of selection* or *delete right of selection*. These remove the required portion of the sound and update the display and other information accordingly. The *delete selection* function is similar,

except that the parts of the sound before and after the selection are joined to create the new sound.

D.1.5 Playing the Sound

The current selection can be played via the internal audio device of the machine on which the application is running. Clicking the *play selection* button shows the window in figure D.4.

Figure D.4 – Play window.



The slider can be moved to adjust the playback volume, and clicking the *play* button sends the current selection to the audio device (note that play cannot be interrupted).

D.1.6 Saving

The (potentially edited) .au file can be saved by clicking the button labelled *save*▽ and choosing *.au*. This causes the dialogue box in figure D.5 to be displayed, which allows the name of the saved file to be specified and the file to be written (by clicking *save*). The name must have the suffix '.au', and this will be appended if it is not typed. Care should be taken as any existing file of the same name will be overwritten without warning.

Figure D.5 – Naming a file for saving.

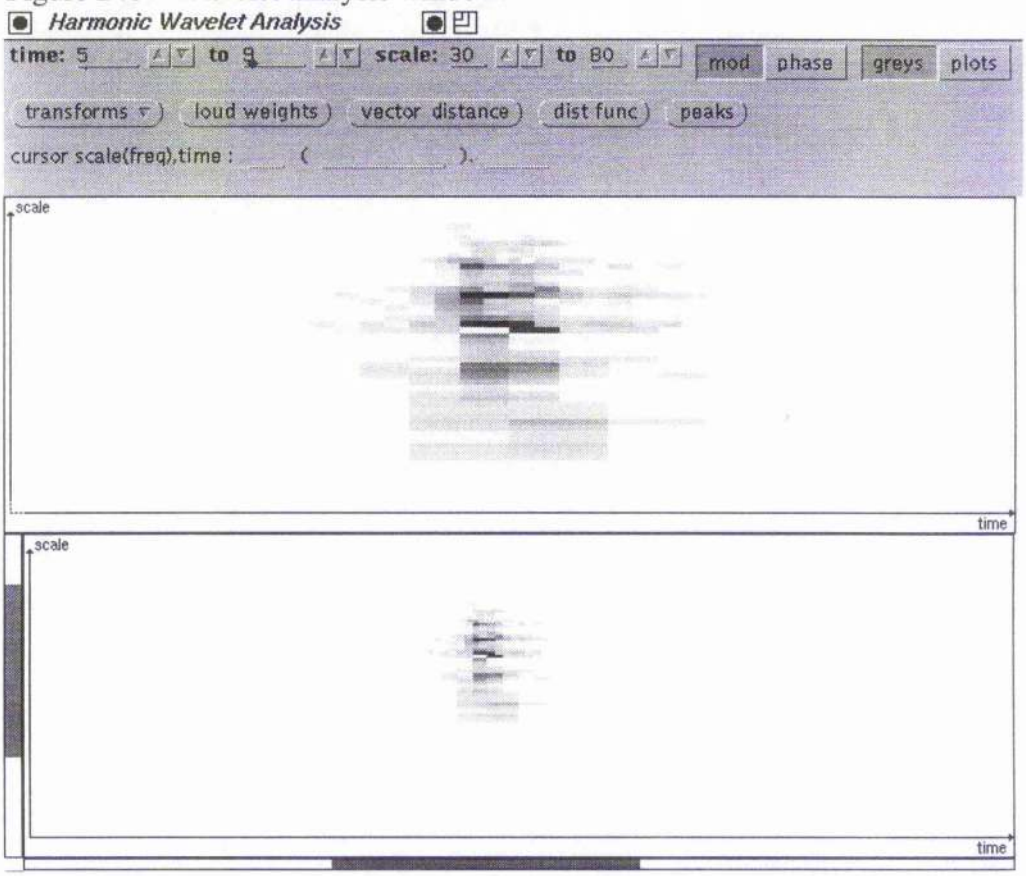
In addition, if any further analyses have been carried out, they can be saved in what is known as a state file by choosing the *state* option via the *save*▽ button. This simply replaces the '.au' suffix in the name with '.st' and saves the audio, wavelet analysis (plus transformations) and vector distance function in a single file (again, existing state files of the same name are overwritten without warning).

D.2 The Harmonic Wavelet Analysis Window

Clicking the button labelled *wavelet analysis* in figure D.1 causes a harmonic wavelet analysis of the current selection to be performed (as described in appendix B), and the results displayed in a window like that in figure D.6.

This has a similar form to the main window, with a control panel, zoom display and global display arranged similarly. The main difference is that the two-dimensional nature of the data calls for two selection indicators (to the left of and below the global display).

Figure D.6 – Wavelet analysis window.



D.2.1 Display Options

Either the modulus or phase plane can be displayed, and the user specifies which by clicking on one of the choices labelled *mod* and *phase*.

The other display option dictates whether the data is shown as grey scale boxes, or plotted as a series of line segment graphs. This is set similarly using the choices labelled *greys* and *plots*.

D.2.2 Adjusting the Selection

The area of the plane currently selected is represented by a pair of scale (or semitone) levels, and a pair of times (or coefficients). Because of the multiresolution nature of the data, the lower scale value dictates the resolution of the time selection.

As in the main window, the selection can be changed by typing in values directly. However, clicking in the global display has no effect on the selection -- clicks of the left and right mouse buttons in each of the selection indicators change their upper and lower values (respectively).

D.2.3 Perceptual Transforms

The perceptual transformations described in appendix B can be carried out by clicking on the button labelled *transforms\/*. This raises a pop-up menu, listing the available transforms. A given transform is applied to the whole modulus plane (regardless of current selection), and is applied to the current plane (so that several transforms may be applied in sequence).

In addition, the graph of equal loudness weightings (figure B.1) can be displayed by clicking the button labelled *loud weights*.

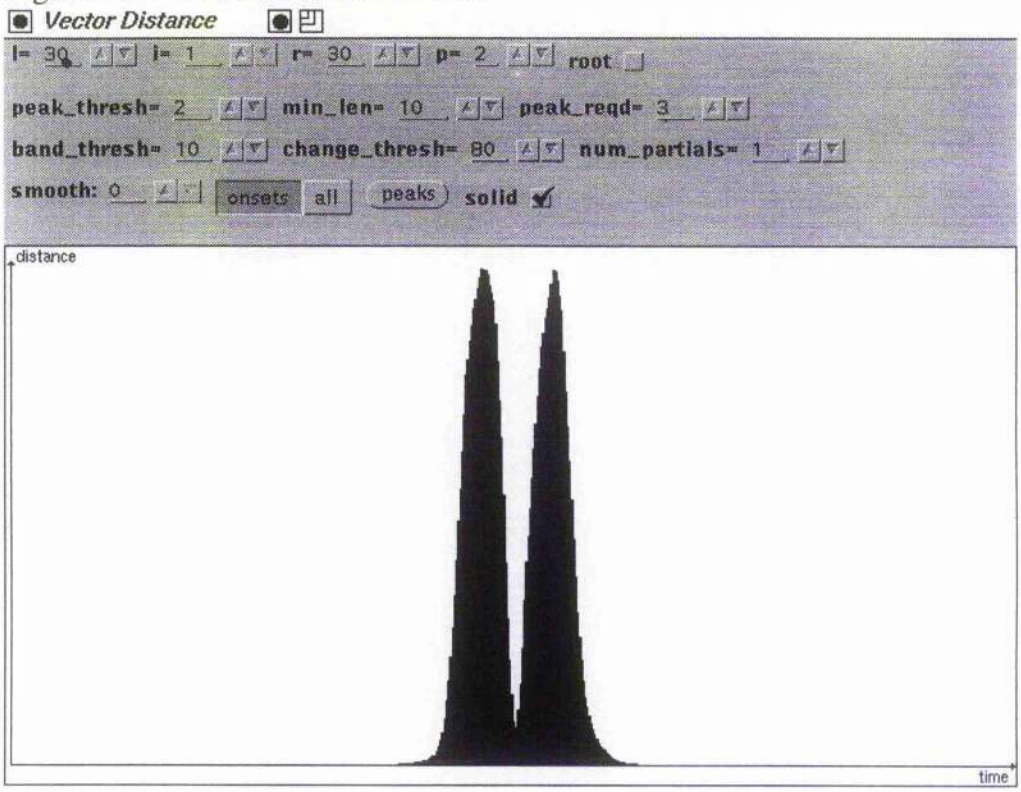
D.2.4 Further Analysis

The vector distance function (described in chapter 4) can be calculated by clicking *vector distance*, which raises the window described in the next section. The results of this calculation can be overlaid on the modulus plane by clicking the button labelled *dist func*. Lastly, the button labelled *peaks* is included for convenience, and behaves identically to that described later in section D.3.3.

D.3 The Vector Distance Window

Clicking on the button labelled *vector distance* in the wavelet analysis window causes the vector distance function to be calculated for the current selection, and displayed as in figure D.7. The settings of the various parameters involved in calculating the function, as well as the peak and onset detection parameters, are shown uppermost in the window. These are named exactly as in chapter 4, where detailed descriptions are given.

Figure D.7 – Vector distance window.



D.3.1 Changing Analysis Parameters

All parameters are initially set to default values, but can be adjusted by simply clicking and typing (with the exception of whether or not the root is taken in the calculation of Minkowski's measure, which is switched on and off by clicking in the checkbox labelled *root*). Changing any of the parameters involved in calculation of the vector distance causes the function to be recomputed and the display updated. Alteration of the peak or onset detection parameters results in the audio, wavelet analysis and vector distance displays being redrawn (so that any previously generated peak marks are removed).

D.3.2 Smoothing the Vector Distance Plot

The vector distance function can be smoothed, by averaging of adjacent values. This is accomplished by clicking in the field labelled *smooth:* and typing the number of values to average. The new smoothed function is then computed, and the display updated.

D.3.3 Locating Peaks and Onsets

In interpreting the vector distance function, the user can choose to find all peaks, or only those identified as onsets, by highlighting the required option in the choices labelled *onsets* and *all*. Clicking on the button labelled *peaks* then causes the function to be analysed, and the resulting points highlighted by vertical marks on the audio, wavelet analysis and vector distance displays.

Appendix E
Detailed Results

This appendix supplements the results given in chapters 6 and 7, and is similarly organised (separating the main body of tests from the others, and presenting the results of the tests on the main body both by timbre and by method).

E.1 Main Body

Again, the results in this section were obtained from the legato and staccato versions of the timbres listed in 6.1.

E.1.1 Results by Timbre

Table E.1 lists the results by timbre, corresponding to those plotted in the graph of figure 6.7. The average error (over all seven notes) is given, along with the minimum and maximum errors. The next 2 columns correspond to the number of early detections and the number of late detections. These are followed by the analysis parameters (exponent, loudness scale and an indication of whether or not adaptive normalisation was applied). Lastly, the averages of all the onset detection parameter sets which resulted in the lowest average error are given (often many parameter sets produce the same average error).

Table E.2 is similar, but shows the effect of applying the root in Minkowski's measure.

The last of the experiments conducted on the main body of test cases was to investigate the effect of applying peak adjustment. The results of this (with no root applied) are given in Table E.3. In this case, the peak adjustment technique dominates the results, so that many sets of analysis parameters produce the same average difference. The table is thus partitioned into sets of parameters for each timbre.

Table E.1 – No root applied, no peak adjustment (overall average error=411).

timbre	avg	min	max	<	>	p	loud scale	adapt. norm?	*change _thresh*	*num_ partials*	*band_ thresh*
cello	1415	356	2396	1	6	5	sone	yes	69	0	8
cello_stacc	306	4	588	5	2	1	sone	yes	83	6	17
dist_gtr	278	26	1270	3	4	4	dB	no	75	5	19
dist_gtr_stacc	71	14	134	4	3	1	dB	no	76	4	28
flute	1062	378	1710	1	6	3	dB	yes	73	2	5
flute_stacc	191	38	310	3	4	3	dB	yes	72	3	17
horn	844	42	1750	2	5	1	dB	yes	64	3	13
horn_stacc	731	344	1160	0	7	1	sone	no	72	2	17
marimba	238	56	576	4	3	3	dB	no	71	2	15
mute_gtr	151	6	306	4	3	4	dB	no	73	3	23
mute_gtr_stacc	294	110	478	7	0	5	dB	no	73	3	24
oboe	860	680	1128	1	6	3	sone	no	74	2	15
oboe_stacc	572	440	800	0	7	5	dB	no	72	3	23
organ	548	83	811	2	5	3	dB	yes	77	4	20
organ_stacc	167	31	375	6	1	3	sone	yes	65	3	50
pno	323	72	1056	2	5	1	sone	no	72	3	14
pno_stacc	66	4	220	5	2	1	sone	no	73	2	14
sax	297	16	600	0	7	1	sone	no	72	3	33
sax_stacc	151	62	270	2	5	1	dB	no	73	4	29
steel	178	45	355	5	2	5	dB	no	86	5	23
timp	152	44	228	6	1	1	dB	no	75	3	16
timp_stacc	138	31	311	7	0	2	sone	no	75	4	15

Table E.2 – Root applied, no peak adjustment (overall average error=407).

timbre	avg	min	max	<	>	p	loud scale	adapt. norm?	*change _thresh*	*num_ partials*	*band_ thresh*
cello	1469	164	2332	0	7	1	sone	no	75	6	3
cello_stacc	306	4	588	5	2	1	sone	yes	83	6	17
dist_gtr	278	26	1270	3	4	4	dB	no	76	5	20
dist_gtr_stacc	71	14	134	4	3	1	dB	no	76	4	28
flute	939	506	1454	1	6	5	dB	yes	73	2	5
flute_stacc	191	38	310	3	4	3	dB	yes	72	3	17
horn	844	42	1750	2	5	1	dB	yes	64	3	13
horn_stacc	731	344	1160	0	7	1	sone	no	72	2	17
marimba	238	56	576	4	3	3	dB	no	70	2	17
mute_gtr	151	6	306	4	3	4	dB	no	73	3	26
mute_gtr_stacc	294	110	478	7	0	5	dB	no	73	3	26
oboe	860	680	1128	1	6	3	sone	no	74	2	15
oboe_stacc	572	440	800	0	7	5	dB	no	72	3	23
organ	548	83	811	2	5	3	dB	yes	77	4	20
organ_stacc	167	31	375	6	1	3	sone	yes	65	3	50
pno	323	72	1056	2	5	1	sone	no	72	3	14
pno_stacc	66	4	220	5	2	1	sone	no	73	2	14
sax	297	16	600	0	7	1	sone	no	72	3	33
sax_stacc	151	62	270	2	5	1	dB	no	73	4	29
steel	178	45	355	5	2	5	dB	no	86	5	23
timp	152	44	228	6	1	1	dB	no	75	3	16
timp_stacc	138	31	311	7	0	2	sone	no	75	4	15

Table E.3 – No root applied, with peak adjustment (overall average difference=503).

timbre	avg	min	max	<	>	p	loud scale	adapt. norm?	*change thresh*	*num_ partials*	*band_ thresh*
cello	926	228	1580	1	6	1	dB	yes	93	5	5
cello_stacc	458	52	1340	4	3	1	dB	yes	73	3	20
cello_stacc	458	52	1340	4	3	2	dB	yes	73	3	22
cello_stacc	458	52	1340	4	3	3	dB	yes	73	3	16
cello_stacc	458	52	1340	4	3	4	dB	yes	73	3	16
dist_gtr	367	6	846	3	4	1	sone	no	78	4	25
dist_gtr_stacc	290	70	706	2	5	1	dB	yes	77	6	50
flute	815	470	1454	1	6	1	dB	yes	73	2	6
flute	815	470	1454	1	6	2	dB	yes	73	2	6
flute	815	470	1454	1	6	3	dB	yes	73	2	6
flute_stacc	172	14	458	5	2	1	dB	yes	88	6	10
flute_stacc	172	14	458	5	2	2	dB	yes	76	4	18
flute_stacc	172	14	458	5	2	3	dB	yes	73	3	24
flute_stacc	172	14	458	5	2	4	dB	yes	73	3	24
flute_stacc	172	14	458	5	2	5	dB	yes	73	3	26
horn	656	6	1450	2	5	2	dB	no	73	3	6
horn	656	6	1450	2	5	3	dB	no	73	2	6
horn_stacc	349	48	968	1	6	4	dB	no	73	2	17
horn_stacc	349	48	968	1	6	5	dB	no	73	2	17
marimba	428	16	1088	2	5	1	dB	no	83	2	19
marimba	428	16	1088	2	5	2	dB	no	83	2	20
marimba	428	16	1088	2	5	3	dB	no	83	2	18
marimba	428	16	1088	2	5	4	dB	no	83	2	18
marimba	428	16	1088	2	5	5	dB	no	83	2	18
mute_gtr	528	26	1366	1	6	2	dB	no	95	3	26
mute_gtr_stacc	384	38	754	4	3	1	sone	no	84	4	26
oboe	435	80	1352	2	5	4	dB	yes	68	2	23

Table E.3 continues over...

Table E.3 (continued)

timbre	avg	min	max	<	>	p	loud scale	adapt. norm?	*change _thresh*	*num_ partials*	*band_ thresh*
oboe_stacc	467	32	976	0	7	1	dB	no	73	3	27
oboe_stacc	467	32	976	0	7	2	dB	no	73	3	27
oboe_stacc	467	32	976	0	7	3	dB	no	73	3	27
oboe_stacc	467	32	976	0	7	4	dB	no	73	3	27
oboe_stacc	467	32	976	0	7	5	dB	no	73	3	26
organ	489	197	1037	0	7	1	dB	yes	89	4	38
organ	489	197	1037	0	7	2	dB	yes	71	4	38
organ	489	197	1037	0	7	3	dB	yes	69	4	38
organ	489	197	1037	0	7	4	dB	yes	69	4	38
organ	489	197	1037	0	7	5	dB	yes	88	4	38
organ_stacc	442	33	887	4	3	1	sone	yes	95	3	45
pno	586	104	1104	1	6	1	dB	no	72	3	17
pno	586	104	1104	1	6	1	sone	no	73	3	17
pno	586	104	1104	1	6	2	dB	no	72	3	17
pno	586	104	1104	1	6	2	sone	no	72	3	17
pno	586	104	1104	1	6	3	dB	no	72	2	15
pno	586	104	1104	1	6	3	sone	no	73	3	15
pno	586	104	1104	1	6	4	dB	no	72	2	15
pno	586	104	1104	1	6	4	sone	no	72	2	15
pno	586	104	1104	1	6	5	sone	no	72	2	15
pno_stacc	637	132	1676	2	5	1	dB	no	73	3	17
sax	411	8	888	0	7	1	dB	no	73	3	27
sax	411	8	888	0	7	2	dB	no	73	3	29
sax	411	8	888	0	7	3	dB	no	73	3	27
sax	411	8	888	0	7	4	dB	no	73	3	27
sax	411	8	888	0	7	5	dB	no	73	3	27

Table E.3 continues over...

Table E.3 (continued)

timbre	avg	min	max	<	>	p	loud scale	adapt. norm?	*change _thresh*	*num partials*	*band_ thresh*
sax_stacc	538	230	814	0	7	1	sone	yes	61	4	50
sax_stacc	538	230	814	0	7	2	sone	yes	63	4	50
sax_stacc	538	230	814	0	7	3	sone	yes	61	4	50
sax_stacc	538	230	814	0	7	4	sone	yes	61	4	50
sax_stacc	538	230	814	0	7	5	sone	yes	60	4	50
steel	387	13	1203	3	4	1	sone	no	81	5	23
timp	687	44	2124	2	5	1	dB	no	90	0	50
timp_stacc	617	145	1513	2	5	1	dB	no	73	3	16

E.1.2 Results by Method

This section gives an alternative view of the experiment which gave rise to table E.1. The following two tables show the results obtained without and with adaptive normalisation applied, and are grouped to illustrate the effect of both loudness measures. For each analysis method, the average difference (over all onsets) is given, as well as the standard deviation of the errors, and the minimum and maximum error. The percentage of onset peaks correctly identified is also given, as is the number of spurious peaks wrongly identified as onsets.

Table E.4 – Adaptive normalisation not applied.

	decibels					sones				
p =	1	2	3	4	5	1	2	3	4	5
avg	748	708	698	714	694	721	715	735	796	804
stdev	768	763	747	787	743	781	744	736	785	753
min	12	4	6	6	4	4	4	12	7	2
max	6500	6724	6916	7076	7140	7724	6596	6660	6724	6788
found	100%	100%	100%	99%	98%	100%	100%	100%	99%	98%
spurious	0	0	0	0	0	0	0	0	0	0

Table E.5 – Adaptive normalisation applied.

	decibels					sones				
p =	1	2	3	4	5	1	2	3	4	5
avg	1378	1325	1355	1321	1273	1127	1159	1165	1226	1211
stdev	1641	1572	1762	1769	1746	1304	1358	1442	1650	1659
min	26	8	1	4	6	4	24	6	6	2
max	10607	9245	10671	10703	10735	9213	9213	9245	9245	9277
found	100%	100%	100%	99%	99%	100%	100%	100%	99%	97%
spurious	9	8	8	7	9	11	11	9	2	2

E.2 Additional Test Cases

This section contains the numerical results obtained from the additional test cases (discussed in §6.3.2).

Table E.6 – Additional test case results.

timbre	avg	min	max	<	>	p	loud scale	adapt. norm?	*change _thresh*	*num_ partials*	*band_ thresh*
pno+reverb	1183	453	1579	6	1	4	dB	no	72	3	10
pno+dynamics	528	79	1135	1	6	2	sone	no	73	1	3
pno-2 octaves	406	172	1052	2	5	1	sone	no	73	4	16
pno-3 octaves	781	89	2039	1	6	1	sone	yes	55	2	50
pno short notes	276	0	653	2	11	1	sone	no	76	5	6
vibrato	*	*	*	*	*	1	n/a	no	200	4	0
tremolo	368	115	829	4	3	3	dB	yes	83	2	7
drum pattern	138	4	680	26	6	2	dB	no	65	6	10

* Not all onsets were detected, and spurious detections were present.

Note that the test case with shorter notes had 14 onsets to be detected, the drum pattern had 32, whereas all the others had only seven. It should also be noted that some of the above results were derived in the same way as the main body of tests, whilst some used ad hoc parameter settings as described in chapter 6.

E.3 Restriction to Single Method

The results obtained with the single best method (see §7.1.2) were as in table E.7.

Table E.7 – Results for p=3, dB scale, no adaptive normalisation (overall average error=698).

timbre	avg	min	max	<	>	change_thresh	num_partials	band_thresh
cello	2305	884	6916	0	7	73	1	24
cello_stacc	1814	932	2804	0	7	72	2	20
dist_gtr	296	18	1238	3	4	77	5	20
dist_gtr_stacc	380	250	554	0	7	75	4	32
flute	1286	314	1774	1	6	74	1	2
flute_stacc	1396	862	1742	0	7	72	1	19
horn	1108	178	1610	0	7	73	1	6
horn_stacc	909	440	1680	0	7	72	2	17
marimba	238	56	576	4	3	71	2	15
mute_gtr	202	10	338	6	1	72	3	23
mute_gtr_stacc	408	238	542	7	0	73	3	25
oboe	883	656	1288	1	6	76	1	26
oboe_stacc	609	472	832	0	7	72	3	23
organ	611	427	877	1	6	73	5	13
organ_stacc	655	409	801	0	7	73	4	30
pno	542	184	896	0	7	72	2	15
pno_stacc	399	100	684	0	7	72	3	17
sax	365	144	568	1	6	72	3	25
sax_stacc	203	6	430	1	6	72	4	29
steel	214	45	445	5	2	72	3	26
timp	296	84	476	4	3	72	2	24
timp_stacc	217	65	377	2	5	74	3	19

Appendix F
CD Guide

This appendix acts as a guide to the CD accompanying this thesis. The first two tracks are monophonic audio examples. Subsequent stereo tracks illustrate analyses, and contain two separate channels of audio: the single channel audio input to the analysis and a click track generated from it (as described in §7.3).

Table F.1 – Track listing for accompanying CD.

track	figure	page	description
1	1.1	1	Repeated rim-shots
2	1.2	2	French horn solo
3	6.10	59	Cello
4	6.11	59	Staccato cello
5	6.12	60	Distorted guitar
6	6.13	60	Staccato distorted guitar
7	6.14	61	Flute
8	6.15	61	Staccato flute
9	6.16	62	French horn
10	6.17	62	Staccato french horn
11	6.18	63	Marimba
12	6.19	63	Muted guitar
13	6.20	64	Staccato muted guitar
14	6.21	64	Oboe
15	6.22	65	Staccato oboe
16	6.23	65	Organ
17	6.24	66	Staccato organ
18	6.25	66	Piano
19	6.26	67	Staccato piano
20	6.27	67	Saxophone

Table F.1 continues over...

Table F.1 (continued)

track	figure	page	description
21	6.28	68	Staccato saxophone
22	6.29	68	Steel drum
23	6.30	69	Timpani
24	6.31	69	Staccato timpani
25	6.33	73	Piano with reverberation
26	6.34	74	Piano with dynamic variation
27	6.35	75	Piano two octaves lower than original
28	6.36	75	Piano three octaves lower than original
29	6.37	77	Piano piece with short notes
30	6.38	78	Legato flute with amplitude modulation (tremolo)
31	6.39	79	Legato french horn with pitch modulation (vibrato)
32	6.41	80	Drum pattern
33	6.42	80	Square wave with glissandi
34	7.3	97	Real guitar recording
35	7.4	97	Real piano recording
36	7.5	98	Real french horn recording
37	7.6	99	Footsteps in background music

References

Andre–Obrecht, R. 1988. "A New Statistical Approach for the Automatic Segmentation of Continuous Speech Signals." *IEEE Transactions on Acoustics Speech and Signal Processing* 36(1):29–40.

Arcelo, V. B. 1995. "A Real–Time Onset Detector Using an Automatic Gain Control." Technical Report No. 11. Department of Electrical Engineering, San Jose State University.

Basseville, M. 1988. "Detecting Changes in Signals & Systems – A Survey." *Automatica* 24:309–326.

Bregman, A. S. 1994. *Auditory Scene Analysis: The Perceptual Organization of Sound.* London: Bradford Books.

Brookes, T., A. Tyrell, and D. Howard. 1996. "Musical Analysis Using a Real–Time Model of Peripheral Hearing." *Proceedings of the 1996 International Computer Music Conference.* San Francisco: Computer Music Association, pp. 79–82.

Brown, G. J., and M. Cooke. 1994. "Perceptual Grouping of Musical Sounds: A Computational Model." *Journal of New Music Research* 23:107–132.

Chafe, C., D. Jaffe, K. Kashima, B. Mont–Reynaud, and J. Smith. 1985. "Techniques for Note Identification in Polyphonic Music." *Proceedings of the International Computer Music Conference 1985* . San Francisco: Computer Music Association, pp. 399–405.

Cole, R. A., and L. Hirschman et al. 1992. "Workshop on Spoken Language Understanding." Technical Report No. CS/E 92–014. Oregon Graduate Institute.

Combes, J. M., A. Grossman, and Ph. Tchamitchian. 1990. *Wavelets, Time Frequency Methods and Phase Space.* Springer–Verlag.

Cooke, M. 1993. *Modelling Auditory Processing and Organisation.* New York: Cambridge University Press.

Daubechies, I. 1988. "Orthonormal Bases of Compactly Supported Wavelets." *Communications on Pure and Applied Mathematics* 41:909–996.

De Poli, G., A. Piccialli, and C. Roads, eds. 1991. *Representations of Musical Signals.* Cambridge Mass: MIT Press.

Depalle, Ph., and L. Tromp. 1996. "An Improved Additive Analysis Method Using Parametric Modelling of the Short–Time Fourier Transform." *Proceedings of the 1996 International Computer Music Conference.* San Francisco: Computer Music Association, pp. 297–300.

Desain, P., and H. Honing. 1989. "The Quantisation of Musical Time: A Connectionist Approach." *Computer Music Journal* 13(3):56–66.

Desain, P., and H. Honing. 1994a. "Foot–tapping: A Brief Introduction to Beat Induction." *Proceedings of the 1994 International Computer Music Conference.* San Francisco: Computer Music Association, pp. 78–79.

Desain, P., and H. Honing. 1994b. "Advanced Issues in Beat Induction Modelling: Syncopation Tempo and Timing." *Proceedings of the 1994 International Computer Music Conference.* San Francisco: Computer Music Association, pp. 92–94.

Diagram Group. 1976. *Musical Instruments of the World: An Illustrated Encyclopedia.* New York: Facts on File.

Feiten, B., and S. Gunzel. 1993. "Distance Measure for the Organisation of Sounds" *Acustica* 78:181–184.

Foster, S., W. A. Schloss, and A. J. Rockmore. 1982. "Toward an Intelligent Editor of Digital Audio: Signal Processing Methods." *Computer Music Journal* 6(1):42–51.

Gordon, J. W. 1984. *Perception of Attack Transients in Musical Tones.* Ph.D. thesis. Department of Music, Stanford University.

Goto, M., and Y. Muraoka. 1995. "Music Understanding at the Beat Level – Real–time Beat Tracking for Audio Signals." in: *Working Notes of the Internlational Joint Conference on Artificial Intelligence 1995 Workshop on Computational Auditory Scene Analysis*, pp. 68–75.

Goto, M., and Y. Muraoka. 1996. "Beat Tracking Based on Multiple–agent Architecture – A Real-time Beat Tracking System for Audio Signals." *Proceedings of the 1996 International Conference on Multi-Agent Systems,* pp. 103-110.

Graps, A. 1995. "An Introduction to Wavelets." *IEEE Computational Science and Engineering* 2(2):1–18.

Gray, A. H., and J. D. Markel. 1976. "Distance Measures for Speech Processing." *IEEE Transactions on Acoustics Speech and Signal Processing* 24(5):380–391.

Grey, J. M. 1975. *An Exploration of Musical Timbre.* Report No. STAN–M–2. Stanford University Department of Music.

Grossman, A., M. Holschneider, R. Kronland–Martinet, and J. Morlet. 1987. "Detection of Abrupt Changes in Sound Signals With the Help of Wavelet Transforms." in: *Inverse Problems: Advances in Electronics and Electron Physics* Suppl. 19. Orlando, Florida: Academic Press, pp. 289–306.

Gustafson, D. E., A. S. Willsky, J. Wang, M. C. Lancaster, and J. H. Triebwasser. 1978. "ECG/VCG Rhythm Diagnosis Using Statistical Signal Analysis – I. Identification of Persistent Rhythms." *Biomedical Engineering* 25(4):344–352.

Howard, D. M., and J. Angus. 1996. *Acoustics and Psychoacoustics.* Reed.

International MIDI Association. 1983. *MIDI Musical Instrument Digital Interface Specification 1.0.* North Hollywood: International MIDI Association.

Jaffe, D. A. 1987a. "Spectrum Analysis Tutorial Part 1: The Discrete Fourier Transform." *Computer Music Journal* 11(2):9–24.

Jaffe, D. A. 1987b. "Spectrum Analysis Tutorial Part 2: Properties and Applications of the Discrete Fourier Transform." *Computer Music Journal* 11(3):17–35.

James Taylor Quartet. 1988. "Kook's Korner." from: *Wait A Minute.* CD 837 340–2, Polydor.

Jones, I. 1991. "What It Is." from: *Acid Jazz Vol. 3.* CDBGP 1025, Ace Records.

Kaiser, G. 1994. *A Friendly Guide To Wavelets.* Boston: Birkhauser.

Kristo, M. J., and C. G. Enke. 1989. "Reliable Peak–Finding for MS/MS." *International Journal of Mass Spectrometry and Ion Processes* 87:141–155.

Kronland–Martinet, R., J. Morlet, and A. Grossman. 1987. "Analysis of Sound Patterns Through Wavelet Transforms." *International Journal of Pattern Recognition and Artificial Intelligence* 1(2):97–126.

Kronland–Martinet, R. 1988. "The Wavelet Transform for Analysis, Synthesis and Processing of Speech and Music Sounds." *Computer Music Journal* 12(4):11–20.

Longuet–Higgins, H. C. 1976. "The Perception of Melodies." *Nature* 263:646–653.

Mallat, S. G. 1989. "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11(7):674–693.

Mani, R., and S. H. Nawab. 1995. "A Multiband Exponential Rate Operator for Musical Transient Analysis." *Proceedings of the 1995 International Conference on Acoustics, Speech and Signal Processing*, pp. 1233–1236.

Meddis, R., and M. J. Hewitt. 1991. "Virtual Pitch and Phase Sensitivity of a Computer Model of the Auditory Periphery. I: Pitch Identification." *Journal of the Acoustical Society of America* 89:2866–2882.

Mellinger, D. K. 1991. *Event Formation and Separation in Musical Sound.* Ph.D. thesis. Department of Computer Science, Stanford University.

Moore, B. C. J. 1982. *An Introduction to the Psychology of Hearing.* Academic Press.

Moorer, J. A. 1975. *On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer.* Report No. STAN–M–3. Stanford University Department of Music .

Newland, D. 1993. "Harmonic Wavelet Analysis." *Proceedings of the Royal Society of London Series A – Mathematical and Physical Sciences* 443:203–225.

Newland, D. 1994a. "Harmonic and Musical Wavelets." *Proceedings of the Royal Society of London Series A – Mathematical and Physical Sciences* 444(1922):605–620.

Newland, D. 1994b. "Wavelet Theory and Applications." *Proceedings of the 3rd International Congress on Air– and Structure–Borne Sound and Vibration*, pp. 695–713.

Newland, D. 1995. *Signal Analysis by the Wavelet Method.* Technical Report CUED/C–MECH/TR.65. Department of Engineering, University of Cambridge

Pearson, E. R. S. 1991. *The Multiresolution Fourier Transform and its Application to Polyphonic Audio Analysis.* Research Report 282. Department of Computing Science, University of Warwick.

Pearson, E. R. S., and R. G. Wilson. 1990. "Musical Event Detection From Audio Signals Within a Multiresolution Framework." *Proceedings of the 1990 International Computer Music Conference.* San Francisco: Computer Music Association, pp. 156–158.

Piszczalski, M., and B. A. Galler 1978. "The Analysis and Transcription of Musical Sound." *Proceedings of the 1978 International Computer Music Conference (volume 2).* San Francisco: Computer Music Association, pp. 585–618.

Plomp, R. 1976. *Aspects of Tone Sensation.* Academic Press.

Pollard, H. F., and E. V. Jansson. 1982. "Analysis and Assessment of Musical Starting Transients." *Acustica* 51(5):249–262.

Press, W. H. 1992. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press.

Rabiner, L. R., and B. H. Juang. 1986. "An Introduction to Hidden Markov Models." *IEEE Acoustics Speech and Signal Processing Magazine* January:4–16.

Satie, E. 1992. *Menus Propos Enfantins.* from: CD CLG 7037. Classical Gallery.

Schloss, W. A. 1985. *On the Automatic Transcription of Percussive Music – From Acoustic Signal to High–Level Analysis.* Report No. STAN–M–27. Stanford University Department of Music.

Schubert, F. 1994. *Symphony No.9 in C (Great) Andante – Allegro ma non troppo.* from: CD MM117, BBC.

Scott, H., and R. Wilson. 1992. *A Comparison of Filters for Audio Signal Segmentation in Audio Restoration.* Research Report 231. University of Warwick Dpeartment of Computing Science.

Shahwan, T. W. 1994. *An Adaptive Procedure for the Optimization of an Acoustic Onset Detector.* Technical Report No. 8. San Jose State University Department of Electrical Engineering.

Shuttleworth, T., and R. G. Wilson. 1993. *Note Recognition in Polyphonic Music Using Neural Networks.* Research Report 252. University of Warwick Dpeartment of Computing Science.

Smith, L. S. 1996. "Using an Onset–based Representation for Sound Segmentation." *Neural Networks and their Applications 1996,* pp. 274–281.

Solbach, L., R. Wohrmann, and J. Kliewer. 1995. "The Complex–valued Continuous Wavelet Transform as a Preprocessor for Auditory Scene Analysis." in: *Working Notes of the Intlernational Joint Conference on Artificial Intelligence 1995 Workshop on Computational Auditory Scene Analysis.*

Stevens, S. S. 1955. "Measurement of Loudness." *Journal of the Acoustical Society of America* 27(5):815–829.

Stevens, S. S. 1972. "Perceived Level of Noise by Mark VII and Decibels (E)." *Journal of the Acoustical Society of America* 51(2):575 – 601.

Strang, G. 1993. "Wavelet Transforms Versus Fourier Transforms." *Bulletin of the American Mathematical Society* 28(2):288–305.

Tait, C. 1995. "Audio Analysis for Rhythmic Structure." *Proceedings of the 1995 International Computer Music Conference.* San Francisco: Computer Music Association, pp. 590–591.

Tait, C., and W. Findlay. 1995. *Audio Analysis for Rhythmic Structure.* Technical Report no. TR–1995–11. University of Glasgow Department of Computing Science.

Tait, C., and W. Findlay. 1996. "Wavelet Analysis for Onset Detection." *Proceedings of the 1996 International Computer Music Conference.* San Francisco: Computer Music Association, pp. 500–503.

Vos, J., and R. Rasch. 1981. "The Perceptual Onset of Musical Tones." *Perception & Psychophysics* 29(4):323–335.

Wang, K., and S. A. Shamma. 1995. "Auditory Analysis of Spectro–temporal Information in Acoustic Signals." *IEEE Engineering in Medicine & Biology* March/April:186–193.

Wilson, R., A. D. Calway, E. R. S. Pearson, and A. R. Davies. 1992. *An Introduction to the Multiresolution Fourier Transform and its Applications.* Research Report 204. University of Warwick Dpeartment of Computing Science.