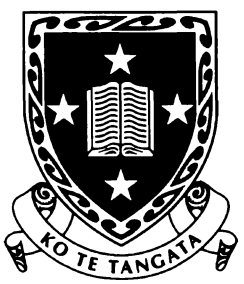
Speech Analysis and Synthesis using an Auditory Model

A thesis submitted
for the degree of
Doctor of Philosophy

By

Dale Anthony Carnegie



The
University
of Waikato
Te Whare Wānanga
o Waikato

April 2000

Abstract

Many traditional speech analysis/synthesis techniques are designed to produce speech with a spectrum that is as close as possible to the original. This may not be necessary because the auditory nerve is the only link from the auditory periphery to the brain, and all information that is processed by the higher auditory system must exist in the auditory nerve firing patterns. Rather than matching the synthesised speech spectra to the original representation, it should be sufficient that the representations of the synthetic and original speech be similar at the auditory nerve level.

This thesis develops a speech analysis system that incorporates a computationally efficient model of the auditory periphery. Timing-synchrony information is employed to exploit the in-synchrony phenomena observed in neuron firing patterns to form a nonlinear relative spectrum intensity measure. This measure is used to select specific dominant frequencies to reproduce the speech based on a synthesis-by-sinusoid approach.

The resulting speech is found to be intelligible even when only a fraction of the original frequencies are selected for synthesis. Additionally, the synthesised speech is highly noise immune, and exhibits noise reduction due to the coherence property of the frequency transform algorithm, and the dominance effect of the spectrum intensity measure.

This noise reduction and low bit rate potential of the speech analysis system is exploited to produce a highly noise immune synthesis that outperforms similar representations formed both by a more physiologically accurate model and a classical non-biological speech processing algorithm. Such a representation has potential application in low-bit rate systems, particularly as a front end to an automatic speech recogniser.

Acknowledgements

I wish to thank my chief supervisor Dr. Geoff Holmes for taking over supervision of this project when two thirds of my original supervision panel left. Geoff's encouragement and proof reading of this thesis, particularly in its end stages is very much appreciated. Thanks also to Dr. Lloyd Smith who inherited me way back in the beginning, and whose suggestions and guidance provided this thesis topic.

Special thanks are due to Debbie, it has been a very rough year, and your support has been active, and unwavering. Debbie has taught me so much, and helped in so many ways, that this simple acknowledgement seems a poor reward for all her efforts. I wish to also thank my parents, they have always been there for me when I needed them, and their constant support has been invaluable.

Thanks to the Department of Physics and Electronic Engineering, particularly Dr. Bill Munro who helped with early proof reading, and Dr Howell Round for his encouragement and support.

Dedication

I wish to dedicate this thesis to my children, Kieran and Teigan.

The last 12 months have been particularly difficult for them. The future holds a promise of so much, and I shall do everything I can to help them fulfil their own potentials.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Humans with normal hearing, perceive speech much better than any hardware or software processor. An example of this can be seen in the performance of non-biologically based speech recognition systems that exhibit a significant deterioration in performance as the Signal to Noise Ratio (SNR) falls below 25 dB. A human would have no trouble recognising the speech at this noise level. The error rates of machines are often more than an order of magnitude greater than those for humans, even in quite, wideband, read speech (Lippmann, 1997).

Traditionally, the desired information is extracted from the speech waveform (in either the time or frequency domain) by using statistical inference methods. The vast majority of fronts ends to automatic speech recognisers (ASR) are based on standard signal-processing techniques such as filter banks, homomorphic (cepstral) analysis or linear predictive coding (Jankowski et al., 1995). These approaches are sensitive to anything that changes the relative magnitude of in-band energies, and hence reduced bandwidths (i.e. over the telephone) or noise can severely degrade their performance (Holton, 1995).

Auditory models can be substituted for these classical techniques and it is reasonable to assume that if they adequately model the human system, they should provide superior performance to these classical techniques. Ideally the model would incorporate characteristics of the human auditory system, particularly the perceptual tolerance to acoustic deviations such as noise and phonemic variability. These features can be utilised to (for example) reduce the bit rate in speech coders or to produce a robust (i.e. noise immune, speaker independent) front end to an automatic speech recogniser.

From a functional approach, the auditory pathway can be broadly classified into two components, the peripheral and the central. The input into the periphery is the acoustic signal, and the output is some processed representation, which in turn, forms the input into the central part. The auditory periphery then, can be considered as a form of preprocessor that removes irrelevant information from the acoustic signal, and converts it to some appropriate internal representation. The processing principles of the central part are associated with cognition. Within these divisions, the auditory periphery contains the outer, middle and inner ears as well as the neural centres composing the auditory brain stem and auditory midbrain, and the central element is essentially the auditory cortex (Ghitza, 1993).

Many traditional speech analysis/synthesis techniques are designed to produce speech with a spectrum that is as close as possible to the original. Ghitza (1987) suggests that this is not necessary, and proposes that the representations of the synthetic and original speech be matched at the auditory nerve level, that is, at the output of the auditory periphery. The justification for this arises from the observation that all information that is processed by the higher auditory system must exist in the auditory nerve firing patterns, since the auditory nerve is the only link from the auditory periphery to the brain. Consequently, for speech analysis purposes, Ghitza proposes that it is only necessary to retain the properties of the speech signal that determine the auditory nerve firing patterns.

Ambikairajah et al. (1989) classify cochlear models into one of two broad categories, analytical and computational. Analytical models capture the behaviour of the cochlea by reproducing its response to any given stimulus, whereas accuracy is substituted in a computational model in order to gain computational efficiency. This argument can be extended to include the entire auditory periphery, not just the cochlea. Ambikairajah et al., envisage an application for computational models as a front end to a speech recognition system if the computation times can be sufficiently reduced.

Ideally then, an auditory model, that is computationally efficient, could be useful for low bit rate coding, and could form a (potentially real-time) front end to an automatic speech recogniser that increases recognition (as compared to a classical front end), is noise immune and is also speaker independent.

Ghitza (1987) proposes such a computational model of the auditory periphery, followed by a nonlinear relative spectrum intensity measure (which he terms the "in-Synchrony-Bands-Spectra" or SBS) that utilises timing-synchrony information to exploit the in-synchrony phenomena observed in neuron firing patterns. This approach can be considered as an efficient method of extracting a minimal set of Discrete Fourier Transform (DFT) maxima necessary to retain the main features of the speech. Speech synthesis is performed using McAulay and Quatieri's synthesis-by-sinusoid approach (1986) where the frequency of the sinusoids are obtained from the relative spectrum intensity measure. Ghitza's experiments produced highly intelligible and natural speech (though some tonal artifact is exhibited), and indicated that the number of sinusoids required for synthesis can be reduced by approximately 70 percent, offering a potential for reduced data rate. Additionally, the model appeared to be highly noise immune.

Although this model showed considerable promise for low bit rate coding, and as a front end to ASR, it has not been thoroughly examined to determine its effectiveness in these situations.

This thesis constructs a computational model of the auditory periphery and represents the speech waveform using the in-Synchrony-Bands-Spectrum. The two essential questions that are pursued in this thesis are, "How applicable is this model to low bit rate coding?", and "How does the resulting low bit rate model perform as a front end to ASR?" .

For low bit rate coding, the synthesised speech must remain intelligible, though quality may be degraded. To achieve this low bit rate, the number of frequency bins used in the synthesis will be reduced, and the frequency domain parameters such as amplitude, phase and frequency bin representation will be quantised. Questions arising from this approach are:

- What is the minimal frequency set that still retains the essential features of the speech?
- Is Ghitza's computational approach the best method to select these frequencies?
- What is the lowest bit rate that can be achieved from the reduction of the frequency set and quantisation of other parameters?

- What is an appropriate measure of the success (or otherwise) of this system? In other words, given a speech representation, to what extent can it be measured to have preserved the phonetic information that is perceptually relevant?
- Although a loss of naturalness is expected as the number of frequency components is reduced, will the system be more noise immune than some classical, non-biological algorithm?

As a front end to an ASR, questions to be answered include:

- Will it be more suitable as a front end than some classical equivalent?
- Will it be more suitable as a front end than a more physiologically accurate system?
- How does white noise and cocktail-party noise affect the performance of the system? Are more or fewer frequency components required in such an environment?
- To what extent does the frequency elimination and parameter quantisation affect the ASR performance? Can this be statistically classified? From this, can an optimum front end be constructed?

The answers to these questions will determine to what extent the computational model can successfully perform in real speech processing applications. Its failure would indicate that the simplifications made in order to obtain computational efficiency are not valid.

1.1.1 Implementation

This thesis begins answering these questions by implementing Ghitza's model in its described form (Ghitza, 1987). The filter shapes, frame sizes, windowing routines, and high-frequency emphasis routines are varied to determine a best working model. From this, the number of frequency components used in the synthesis process are varied, dropping in some experiments, to as low as three. Deficiencies in the system are noted, and corrected.

As the processing is performed in the frequency domain, it is easy to adjust the amplitude, phase and frequency bin quantisation levels, since the exact levels are not necessary for either intelligibility or speech recognition purposes. Objective methods of scoring intelligibility and recognition are investigated and implemented, and intelligibility and

recognition experiments are performed to find an optimum level of number of frequency components and parameter quantisation levels. This system is then compared with an auditory model. One of the most respected and referenced auditory models is that developed by Seneff (1988). This model is implemented, and compared to the modified Ghitza system. Another comparison is performed with the classical (non-biological) Linear Predictive Coding 10 (LPC10) system, obtained from the US Department of Defence (1997). All models are subjected to significant white and cocktail-party noise levels (in some cases the SNR is less than 0 dB), and their performance noted.

Analyses of the intelligibility and recognition results are undertaken to objectively evaluate the performance of the three systems. This analysis is performed using a squared error approach, graphical trend analyses, and a statistical neural network multiple regression. These results are used to obtain a best working implementation of Ghitza's model, (incorporating several significant modifications) to produce a system that outperforms the Seneff and LPC for recognition results, possesses a very low bit rate, and has superior intelligibility results (for the same bit rate) to the original Ghitza system.

1.1.2 Performance Measurement

A long-standing question that arises when studying a particular auditory model is how to evaluate its performance. Of interest is how closely the model representation can describe the actual human internal representation. Historically, auditory models were quantitatively evaluated only in the context of speech recognition, where the model formed the front end to a particular recogniser. Such a measure however, provides little information on how well the auditory model predicts the internal human representation. Algorithms and tests currently exist to quantify the:

- Quality of the reproduced speech
- Intelligibility of the reproduced speech
- Automatic (machine) recognisability of the reproduced speech

Subjective measurements are of some use in the initial characterisation of the speech processing system. Such an evaluation can quickly eliminate some processing options if a particular result is markedly inferior to some alternative. However, an objective

measurement of the system performance is an essential requirement. Specifically, this project requires the quantitative analysis of the effects of changing the process parameters and the effects of adding noise to the original speech. Without such a numerical performance measure, it would be difficult to justify the superiority of one simulation over another.

Ghitza makes no claims concerning the quality of the speech processed using this algorithm, and in fact, reports that it suffers from some tonal artefact. However, Ghitza does claim that the intelligibility of such reproduced speech is high. Intelligibility tests exist that rate the speech along six phonetically distinctive features, and an error analysis of the speech for these six features can provide diagnostic information on the poorly modelled parts of the auditory model. This thesis attempts to determine if the reduced information set (i.e. speech reproduced with substantially fewer frequencies) aids the automatic machine recognition of the processed speech. Consequently, the last two factors will be used to measure the success (or otherwise) of the various speech processing algorithms, and quality evaluations such as the Mean Opinion Score (a five level subjective rating of speech quality) or the Bark Spectral Distortion Rating (an objective estimate of the Mean Opinion Score, Watanabe and Hayashi, 1995) are not employed.

Two forms of objective measurement are considered. One is an intelligibility measure, which is implemented initially by subjective listening trials, but subsequently by the more objective Diagnostic Rhyme Test (DRT). This test will provide a measure of the extent to which the auditory system under investigation can preserve phonetic information that is perceptually relevant for the six phonetically distinctive speech features of voicing, nasality, sustention, sibilation, graveness and compactness. The second measurement is automatic recognition, where the processed speech is fed into an automatic speech recogniser, the Hidden Markov Model Toolkit (HTK) of Cambridge University Engineering Department, from which an absolute recognition score can be obtained.

Analyses of the results are initially in the form of an auditory inspection of the resulting reconstructed speech, and a visual inspection of the numerical data and graphical spectra. A more quantitative analysis using a sum of squared errors provides a useful technique for comparing recognition results. The final analysis employs a neural network approach capable of handling the nonlinear relationship of the input variables to the recognition score.

From these analyses, a final "optimal" model is obtained, using a modified version of Ghitza's algorithm that forms the best compromise of high recognition, high intelligibility, and low bit rate.

1.2 Thesis Structure

In this thesis, speech is processed and synthesised using many different Digital Signal Processing (DSP) routines. Initially filtering, and/or interpolation and decimation may be required to appropriately condition the speech signal (particularly in terms of sampling rate). The speech may need to be preemphasised using some form of numerical differentiation (or simply numerical differencing). For the processing required in this project, the speech must be transformed to the frequency domain. To achieve this, the data is divided to a frame size that captures the pseudo-periodic nature of the speech, and contains data points totalling some power of two (to permit the use of the Fast Fourier Transform (FFT)), and is appropriately windowed. The Hamming window is normally used throughout this thesis, being the traditional compromise window, but various other windowing options are investigated. Frame overlapping is necessary to mask the discontinuities that occur at the frame boundaries. The speech is processed in the frequency domain and is then synthesised using McAulay and Quatieri's (1986) synthesis-using-sinusoids approach. Finally the speech is transformed back to the time domain using an Inverse FFT, and the frame overlap is removed by a weighted linear interpolation. The details and derivations of these algorithms are not essential for an understanding of this thesis, and are therefore contained in the first four Appendices.

Auditory models endeavour to create a computational model that mimics the performance of the human ear. The ear is loosely divided into three parts, the outer ear, middle ear and inner ear. Depending upon their complexity, auditory speech models attempt to process the speech data in a similar manner, and so an understanding of physiology of the human ear is required in order to appreciate the auditory model. A summary of these physiological considerations is presented in Chapter 2.

Non-biological models of speech processing have consistently failed to attain the level of speech recognition that the human auditory system is capable of achieving. These poor results have encouraged investigations into auditory models as front ends to automatic speech recognition systems. The work of Young and Sachs (1979), Sachs and Young (1980), and Delgutte and Kiang (1984) yielded considerable insight into the response of auditory nerve fibres to particular stimuli. Their work has formed the basis of many auditory models. Chapter 3 reviews the background of auditory modelling, with particular emphasis on the models of Ghitza and Seneff, and their place in contemporary research. An objective measurement of the output of the speech processing system is a vital component of this research, and quantitative methods of speech evaluation are also discussed in this chapter. From this, a technique for speech intelligibility measurement, and another for automatic speech recognition is obtained.

With the conclusion of this background material, a thorough investigation of the Ghitza model is conducted in the next two chapters. An appropriate sampling rate is determined, followed by an analysis of the effects of different windowing routines. Phase shift is considered, and a routine is developed to join the discretely processed (and therefore discontinuous) frames. Following consideration of the SBS spectra, the first reconstructed speech results are produced, and investigation is begun on how the setting of the SBS Threshold alters the intelligibility of the reconstructed speech. The effects of high frequency emphasis, and the changing of the bandpass filter shapes sheds further insights into the SBS algorithm, which is completed in Chapter 4 by a subjective investigation into how the model rejects impulse and white noise.

The analysis to this point has fixed some DSP and SBS routines and parameters, and allows compression experiments to proceed with these features determined. For low bit rate coding, not only can the number of frequencies used in the reconstruction be lowered, but amplitude and frequency bin representation can be quantised. It is predicted that phase is unlikely to be an important parameter for intelligibility purposes. These claims are investigated, both subjectively, and by the Diagnostic Rhyme Test in Chapter 5. The results yield considerable information concerning Ghitza's algorithm, and highlights some deficiencies, specifically in unvoiced, and low amplitude speech. Ghitza's original model is modified to counteract these deficiencies, primarily by the introduction of a Voiced-Unvoiced-Silence classifier, and selective frequency bin energy amplification. The

improved model is then tested against an Otago University digit corpus and the TIMIT sentence corpus. These experiments, as well as resulting in the modification of Ghitza's model, also provide boundaries of amplitude, frequency bin and phase quantisation for later testing.

At this point, the SBS process parameters, and quantisation possibilities have been reduced to the extent that it is possible to investigate the remaining combinations using the computationally intensive HTK recognition program (Chapter 6). The effects of varying SBS Threshold, frequency bin, amplitude and phase quantisation levels, frame overlap, variations of the Voiced-Unvoiced-Silence classifier, and high frequency emphasis by differencing are investigated in the presence of 9 Signal-to-White-Noise ratios, and 19 Signal-to-Cocktail Party-Noise ratios. The results are analysed graphically, and explained in terms of the behaviour of the system. Two sets of best results are found, depending upon whether the noise is white or cocktail party in nature.

So far, only the SBS process has been used to select the contributing frequencies. The Seneff model is a more physiologically accurate interpretation of human hearing, and is considered as a front end to the compression process in Chapter 7. Comparison is made between the performance of the Seneff selected frequencies to the SBS selected frequencies, and an LPC10 algorithm is included as a representative of a non-auditory based approach to speech processing.

Analyses of the results are performed by a neural network regression system, from which a greater understanding of the relationship between the input variables and the output recognition score is obtained in Chapter 8. In this chapter, a final speech processing algorithm is formed, that is the best compromise between high intelligibility, high recognisability and low bit rate coding. The resulting optimum model is compared to Ghitza's original model, and its superiority confirmed in terms of both recognition and intelligibility scores.

The final chapter, Chapter 9, summarises the results of the thesis, discusses the contributions made to research and suggests avenues for future work.

1.3 Experimental Methodology

This thesis makes use of a number of different speech sources and databases. When the speech spectra needs to be kept reasonably simple for analysis purposes, the vowels AH and IY are investigated, as they exhibit clear formant structures, most with sufficiently high amplitude to dominate the effects of added noise. During the initial stages of this thesis, a restricted test set was used to help characterise the system. In addition to the two vowels, sentence analysis was performed using the sentences "Sally sells seashells by the seashore" and "We were away a year ago". These early tests provided subjective indications of the system's performance to varying SBS Thresholds, bandpass filter shapes, preemphasis using numerical differentiation, frame overlap, as well as the effects of adding white and impulse noise.

To further characterise the system, 10 isolated words were selected (Section 5.2.3) and used to investigate the effects of amplitude and phase quantisation. A detailed examination of the spectra of these words provided an upper and lower limit to these quantisation levels, and set the parameters for a more formal analysis. A Diagnostic Rhyme Test (DRT) comprising 96 test words (Section 3.6) was then conducted using one male speaker, and 40 untrained listeners. This test provided valuable information concerning the performance of the SBS process with respect to the word types voicing, nasality, sustention, sibilation, graveness and compactness (Section 5.4), and highlighted significant limitations of the SBS algorithm. A subset of the DRT words were chosen, and their spectra investigated in detail to determine the cause of these limitations.

Two main deficiencies were detected, in the loss of energy from the reduced frequency synthesis, and in the failure to reproduce speech frames indicative of unvoiced speech. An energy adjustment algorithm, and a Voiced-Unvoiced-Silence classifier were subsequently constructed and implemented in the simulation to counteract these problems. The simulation was then tested using a selection of sentences from the TIMIT (Section 5.8), and the Otago Digit (Section 5.9) multi-speaker databases. Using these sentences and isolated words, the levels of SBS Threshold, amplitude, frequency bin and phase quantisation levels were refined, to allow testing using the HTK speech recognition toolkit (Section 3.7).

Speech recognition experiments are performed using the HTK toolkit, and the TIMIT recommended training database, and test core base. The TIMIT sentences are tested without added noise, and also in the presence of added white and cocktail-party noise (as discussed in the previous section). Initially, 26 recognition tests are performed varying the parameters of the SBS model and quantisation parameters. The best results from these tests are compared to 18 recognition tests that are the equivalent bit-rate and noise level variations of the Seneff model, and also to the LPC recognition results. The SBS recognition results are seen to consistently outperform the other two. With the aid of neural network multiple regression analyses, a best level of the SBS model is obtained, and tested with 100 DRT tests, obtained from four speakers, two male, two female.

A large number of speech databases (of both isolated words and complete sentences) are employed to produce a speech analysis/synthesis system that is not speaker dependent (in terms of being optimised for a particular speaker). Very high levels of white and cocktail party noise are added to more fully appreciate the noise immunity that this form of processing takes. Quality of speech is sacrificed, though intelligibility remains high, and the resulting system provides excellent recognition scores, even when the Signal to Noise Ratio drops below 0 dB.

Chapter 2

2 Physiology of an Auditory System

2.1 Overview

Auditory models have shown much promise as pre-processors for both low bit rate coding and speech recognition. Such models fall into one of two broad categories, analytical and computational. Analytical models attempt to uncover the detailed workings of the inner ear, whereas computational models attempt to capture relevant behaviour of the auditory system, but with an emphasis on computational efficiency rather than physiological accuracy. Models of both types are presented in Chapter 3, and Ghitza's model (1987) which is extensively explored in this thesis, is an example of the latter.

Both types of model attempt to mimic some part of the auditory process, and so an understanding of the physiology involved is necessary. A human auditory system consists of an outer, middle and inner ear section, each with totally different functions. An overview of the system is presented in Figure 2.1, and indicates the anatomical features involved together with their mode of operation and function in the hearing process.

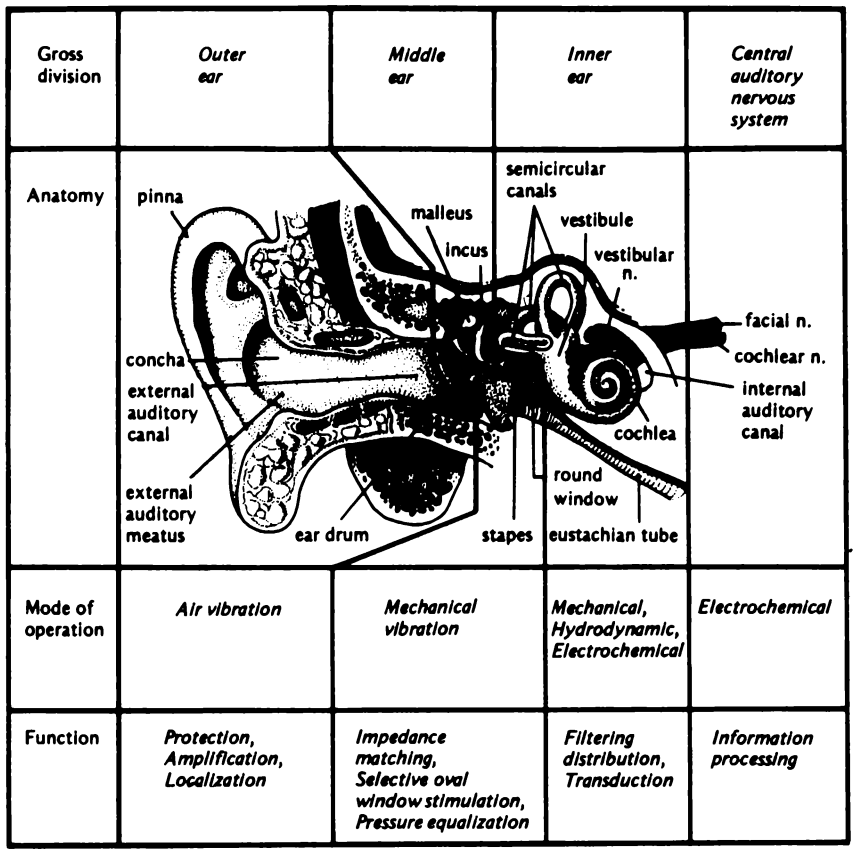


Figure 2.1 : Overview of the Auditory System (From Yost & Nielsen, 1977)

2.2 The Outer Ear

The sound collection process in animals begins in the outer ear, which consists of the pinna and the external auditory canal. The deep centre portion of the pinna is referred to as the concha, which leads to the meatus and hence to the external auditory canal. Experiments have shown that the outer ear results in an amplification in intensity of between 10 and 15 dB over a frequency range from approximately 1.5 kHz to 7 kHz. This increase in amplitude is a result of the resonances of the external auditory canal (2.5 kHz) and the concha (5 kHz) (Yost & Nielsen, 1977). The outer ear also offers the rest of the auditory system protection against foreign bodies and changes in temperature and humidity.

2.3 The Middle Ear

The outer ear terminates at the tympanic membrane, commonly called the ear drum. The tympanic membrane completely seals off the external auditory canal, and provides the middle ear with protection against foreign bodies. The compression and rarefaction of the air molecules as sound is channelled down the external auditory canal forces the tympanic membrane to vibrate with a similar frequency, and this vibration is then passed along to the ossicular chain (the malleus, incus and stapes). In order for this chain to operate efficiently, the middle ear must not be a closed cavity, otherwise external environmental factors affecting pressure could force a movement of the membrane. To overcome this, the Eustachian tube provides for pressure equalisation by connecting the middle ear to the nasal passages.

2.3.1 The Ossicles

One end of the malleus is attached to the centre of the tympanic membrane, and the other is tightly bound by ligaments to the incus. The incus is then secured to the stapes whose footplate is anchored against the oval window (Figure 2.2).

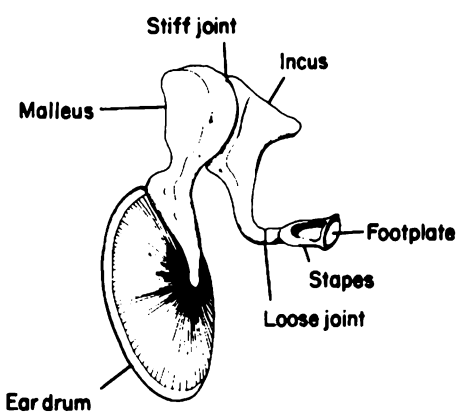


Figure 2.2 : The Auditory Ossicles
(From Rosenberg, 1982)

The inner ear is filled with a fluid that can be moved when pressure is exerted on the oval window membrane that designates the transition from the middle to the inner ear region. To overcome the attenuation resulting from the impedance mismatch of the vibratory wave crossing from a gas into a liquid media, two forms of amplification are introduced. First, the ossicles work as a lever system, as the length of the manubrium (the "handle") and the neck of the malleus is longer than the long process of the incus. Hence the force at the tympanic membrane is increased by a factor of 1.3 at the stapes. This increase and the fact that the stapes has only 1/17th of the area of the tympanic membrane results in a theoretical maximum pressure increase between the tympanic membrane and the stapes of $17 \times 1.3 = 22$ (27 dB) at the stapes footplate (Yost & Nielsen, 1977).

In practice, this increase is strongly frequency dependent and varies between 20 and 25 dB over the range 100 Hz to 2.5 kHz. This, combined with the amplification provided by the outer ear provides the mechanism whereby air pressure variations can be made to move the dense fluids of the inner ear.

2.3.2 Attenuation of High Intensity Sounds

Sounds in excess of 80 dB above the quietest sound someone can hear may cause contraction of the middle ear muscles (stapedius muscle), resulting in a reduction in the transmission of pressure through the ossicular chain. This attenuation is frequency dependent, and for loud sounds may offer a maximum reduction in amplitude of 10 to 30 dB. It is more effective at low frequencies, and probably has no effect above 2 kHz. Onset time for this muscular contraction varies between 10 ms for very loud sounds, to 150 ms for lower intensities, though no protection exists for sudden, loud transients.

The ossicles themselves can also protect the inner ear from damage caused by very loud sounds. For sounds of low intensity, the motion of the ossicles causes a pumping action of the stapes at the oval window, but for high intensity sounds, the motion of the stapes is more of a rotation so that the amplitude of oscillation of the tympanic membrane is not proportionally increased (Ainsworth, 1976).

2.3.3 Alternative Mechanisms for Inner Ear Stimulation

Two methods, other than via the ossicular chain exist for an acoustic stimulus at the tympanic membrane to be transmitted to the inner ear. The stimulus may be transmitted by bone conduction, i.e. the sound travels via the bones of the skull, or by air conduction through the middle ear cavity.

If the middle ear ossicles were missing, the inner ear could be directly stimulated by air pressure variations in the middle ear cavity. However, such a condition would result in both the oval and round window ends of the basilar membrane (see later) moving in opposite directions, resulting in very large system losses. Indeed, the threshold of hearing by this route alone is some 60 dB above that by the normal ear (Rosenberg, 1982). Stimulation of the inner ear as a result of bone conduction does not ordinarily occur, though direct application of a vibrator to the skull does result in a response from the inner ear.

2.4 The Inner Ear

The inner ear, located in the temporal bone, consists of the semicircular canals, the vestibule, and the cochlea (refer Figure 2.1). The main purpose of the vestibule concerns balance rather than auditory perception, so it will not be discussed here. The cochlea is a 35 mm long tapering tube that is coiled to $2\frac{5}{8}$ times in humans and contains the primary auditory organ of the inner ear. It is partially divided throughout its length by a thin spiral shelf of bone, known as the osseous spiral lamina, across which the basilar membrane connects to the outer wall of the cochlea, completing its division into two passages the full length of the canal except for a small opening at the apex called the helicotrema. The lower passage of the canal (scala tympani) has an opening known as the round window which is totally covered with a thin membrane. A partially uncoiled cochlea is presented in Figure 2.3. At the apical end, the basilar membrane is 0.52 mm wide, is flaccid and under no tension. The basal end is narrower (0.16 mm) and stiffer than the apical end and may be

under a small amount of tension. These factors contribute to the vibratory pattern of the membrane in response to acoustic stimulation.

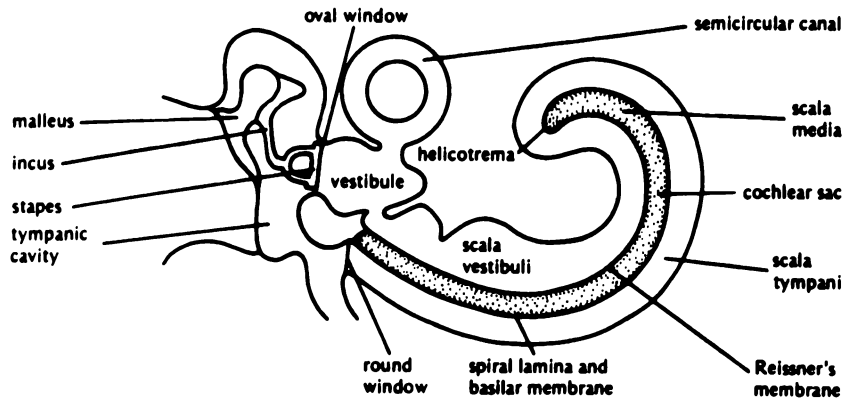


Figure 2.3 : Schematic Diagram of the Middle Ear and Partially Uncoiled Cochlea (From Yost & Nielsen, 1977)

The organ of Corti is situated on top of the basilar membrane and runs the full length of the cochlear duct. It is the receptor organ that generates nerve impulses in response to vibrations passing through the fluid environment of the inner ear. The organ of Corti includes a layer of supporting cells that rest on the basilar membrane and contains rows of hair cells. These hair cells extend up into the fluid within the cochlear duct and the tops of some touch an awning-like membrane, the tectorial membrane, that arches over them. This membrane is attached at only one end and can therefore move independently of the basilar membrane.

2.4.1 Hair Cells

There are about 30 thousand hair cells, lined up along the organ of Corti in two distinct groups. Approximately 5000 of these cells are situated on the basilar membrane close to where the tectorial membrane is attached to the wall of the cochlear duct. These are referred to as the inner hair cells and line up in a single row that runs the length of the basilar membrane. It is on these cells that 95% of the nerve fibres terminate (O'Shaughnessy, 1987). The remaining cells, known as the outer hair cells, line up in three rows at the basal

end and five rows at the apical end. The outer hair cells are sensitive to a bending motion across the basilar membrane, and are capable of detecting very faint sounds, but are unable to register the precise frequency. The inner hair cells react to a motion along the basilar membrane and are less able to register the presence of weak signals, but can specify with precision the region of the basilar membrane that is maximally displaced (Sekuler & Blake, 1985).

Both types of cells terminate in cilia, those belonging to the outer hair cells contact the tectorial membrane. It is the bending of the cilia from both the inner and outer hair cells that triggers the electrical signals that travel from the ear to the brain. The tectorial membrane tends to move in a different direction to the basilar membrane in response to an applied acoustic stimulus. These opposing motions cause the cilia of the hair cells to bend, triggering complicated electrical changes within the hair cells. The hair cell releases a chemical transmitter substance which is picked up by nerve endings surrounding the base of the hair cells (terminations of nerve cells whose axons form the auditory nerve). The exact nature of these electrical signals is not of interest in this application, but the relationship between acoustic stimulus and mechanical movement of the basilar membrane is.

2.4.2 Mechanical Response of the Inner Ear

Acoustic air pressure is transmitted in the form of vibratory patterns to the inner ear via the stapes. The stapes moves the oval window and sets up vibratory undulations in the fluid of the cochlear sac at the organ of Corti. The mechanical response of this organ, and that of the basilar membrane is translated into neural responses in the auditory branch of the eighth cranial nerve. The vibration of fluids in the cochlea causes the basilar membrane to move, bending the cilia of the hair cells. This triggers the nerve at the base of the hair cell to initiate a neural potential which is then transmitted along the auditory nerve.

2.4.3 The Basilar Membrane

As mentioned previously, the basilar membrane becomes wider and more flaccid as the distance from the stapes to the helicotrema increases, therefore the natural period of vibration of the membrane heading towards the helicotrema will correspondingly decrease. Because of the variation in stiffness, different frequencies will obtain their highest amplitude at different points along the membrane (refer Figure 2.4). Multiple simultaneous frequencies at the cochlea will each produce different maximum displacements somewhere along the basilar membrane's length. Effectively, as any complex signal is being resolved into different points of maximal displacement, the basilar membrane can be viewed as performing a role similar to a series of bandpass filters. This theory, which evolved from the original place theory of the 19th century, was rather elegantly substantiated by the Nobel laureate (1961), Georg von Békèsy, to whom we owe much of our understanding of the functioning of the basilar membrane.

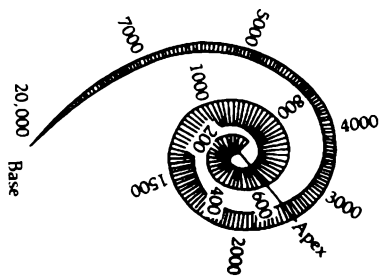


Figure 2.4 : Regions of Resonant Frequencies in the Basilar Membrane (From Sekuler & Blake, 1985)

Higher frequencies obtain their maximum displacement very quickly, and hence tend only to stimulate the basilar membrane at the basal end, before being rapidly attenuated. Lower frequencies however, travel the length of the membrane before obtaining their maximum somewhere close to the apical end. The amplitude of the membrane displacement increases as the intensity of the sound is increased. The waves themselves may be modelled as a series of travelling waves that always start in the base and travel towards the apex, with its distance of travel being highly frequency dependent (lower frequencies travel further) (Figure 2.5). There is obviously some time delay between displacement at the apex and

that at the base (or stapes), and so there is a phase shift along the cochlear partition for various frequencies.

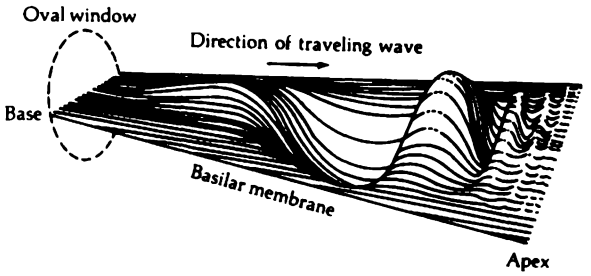


Figure 2.5 : Instantaneous Waveform of a Travelling Wave Along the Basilar Membrane
(From Sekuler & Blake, 1985)

Although certain frequencies result in a maximal displacement of the basilar membrane, frequencies higher and lower than this "resonant" frequency will also displace the membrane but to a lesser extent. As high frequencies are rapidly attenuated as the travelling wave progresses from the basal to the apical end of the membrane, the membrane can be viewed as a filter with a sharp high-frequency roll-off. Also, lower frequencies progressively displace the membrane less and less at that point, so that location also has a gradual low-frequency roll-off. Hence a particular location along the basilar membrane acts as a bandpass filter of the vibrating motion, with a gradual incline slope and a sharp decline slope.

The basilar and tectorial membranes are displaced in different directions, but are connected by the hair cells. This displacement results in some shear force being applied to the cilia, which is thought to be the stimulation of the nerve fibres at the base of the outer hair cells. As we are using a physiological model to form the front end of our auditory system, the exact form of the slopes of a bandpass filter that would model the behaviour of the basilar membrane is of considerable interest, as is the maximum firing rate of the receptor neurons. These are discussed in Sections 2.6 and 2.4.4 respectively.

2.4.4 Pitch Perception

2.4.4.1 The Volley Theory

The maximum firing rate of the receptor neurons becomes important because the neurons require at least 1 ms recovery time between discharges, and so are not able to exceed a firing rate of 1000 impulses per second (Greenberg, 1995). The auditory system is obviously capable of hearing sounds of a higher frequency than 1 kHz, so the neural signal cannot be realised by individual fibres. This limitation could be overcome if separate nerve fibres fired not in unison but in a staggered fashion. For instance, if two out of phase fibres each fire at 1000 impulse per second, the combined effect would be a 2000 impulse per second transmission. This is known as the volley theory and was first proposed by Wever and Bray (1937).

2.4.4.2 The Place Theory

An alternative to the volley theory is the place theory, which maintains that different frequencies of vibration maximally displace different areas of the basilar membrane, in turn activating different hair cells and hence different auditory nerve fibres (e.g. Clark, 1996). Therefore, frequency information is encoded according to the location along the basilar membrane most disturbed by fluid vibration. Evidence to support the place theory can be obtained by studies linking damage to a limited portion of the basilar membrane to a loss in ability to hear certain frequencies. The frequencies affected depend upon the region of the membrane damage, as predicted. Simmons et al. (1965), placed electrodes in the ear of a man who was deaf because of basilar membrane damage, and upon applying a current to the electrodes, a tone was heard, differently placed electrodes producing the perception of different pitch. Such findings considerably strengthened the validity of this spatial approach to frequency perception.

This model does not account for the behaviour of lower frequency waves that produce a very broad pattern of displacement, such that the point of maximal displacement can not be easily identified. However, humans can easily perceive the pitch at these low frequencies, and therefore the conclusion must be made that pitch depends upon something besides place coding.

Using the volley theory that utilises the firing rate of auditory nerve fibres to register such low-frequency information overcomes this limitation, but it has been shown (Rose et al., 1967) that auditory fibres cannot fire in synchrony to high frequency tones. So pitch perception may be mediated by two neural mechanisms, a frequency code at low frequencies and a place code at higher frequencies.

2.5 The Auditory Nerve

As mentioned above, the cochlea converts sound energy into neural impulses, which is transported out of the inner ear by the auditory nerve. The interpretation of this information by the neural analyzers to locate and identify sound sources are beyond the scope of this thesis, however this nerve forms an important part of any auditory model for speech analysis.

2.5.1 Frequency Characteristics of Auditory Nerve Fibres

The auditory nerve consists of about 50,000 individual fibres, 90 percent of which carry information picked up from the inner hair cells, the remainder from the outer hair cells. These cells are active (firing 10-50 times per second (O'Shaughnessy, 1987)) even when no stimulus is present, and therefore an incident sound must alter this spontaneously occurring random neural chatter in order to be detected. Each fibre has a different response to incident sound frequencies, and a definite threshold that must be overcome in order for that sound to be able to evoke a response. Furthermore, each fibre has a certain Characteristic

Frequency (CF) for which the required intensity threshold to detect a sound is significantly lower than that required for other frequencies.

At a very low sound intensity, only the Characteristic Frequency will produce an activity in the fibre, and as the volume is increased, previously ineffective frequencies begin to produce measurable increases in the fibre's activity. A plot of the intensity threshold for different frequencies is referred to as the frequency tuning curve for that fibre (Figure 2.6). This frequency tuning arises from the way the fibres are connected to the basilar membrane, with those fibres originating from the apex of the cochlea having a low CF, whilst those from the base "prefer" higher frequencies.

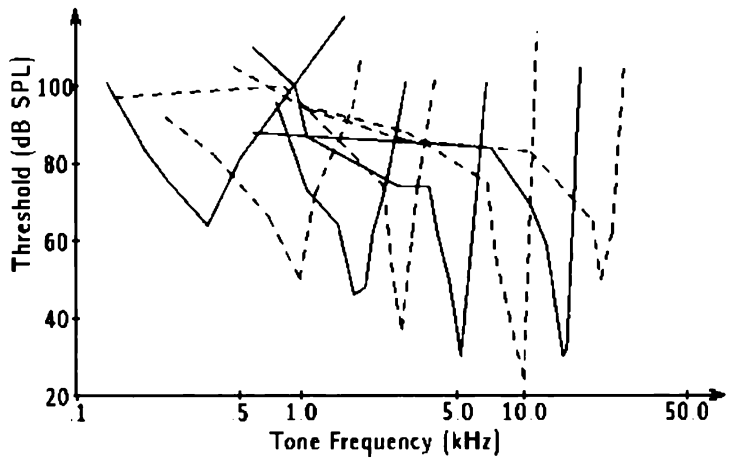


Figure 2.6 : Representative Tuning Curves of Cat Auditory Fibres for Eight Frequency Regions (From O'Shaughnessy, 1987)

2.5.2 Perception of Loudness

An increase in sound intensity increases a particular fibre's activity, up to a certain limit where the firing rate is at its maximum possible level, i.e. the fibre has saturated. However, there is an obvious audible difference between a sound intensity of say 80 dB_{spl} and 120 dB_{spl}, though there is no distinguishable neural response. This indicates that some

additional neural information must enter into the coding of sound intensity. It is proposed (Kiang, 1968, also Ainsworth and Meyer, 1993) that different neurons may operate over different levels of sound intensity. So while the response of any particular neuron may only cover a 40 dB range, groupings of such neurons can cover a far larger level of sound intensities.

Another possibility is that as the sound level increases, a larger number of fibres will respond as the required intensity threshold is overcome. For very soft sounds, only that fibre whose Characteristic Frequency is the same as the frequency of the sound will respond, however as the intensity increases, neighbouring fibres whose CFs are slightly different will also become active. Hence the higher the sound level, the more fibres respond.

2.6 Bandpass Filter Modelling of the Basilar Membrane

Adequate modelling of the motion of the basilar membrane is an essential component of any successful auditory simulation. As mentioned previously, this motion can be simulated as the output signal of a bandpass filter whose frequency response reflects the mechanical tuning characteristics at that place. The point of maximal displacement of the basilar membrane has an approximate logarithmic variation with frequency, as indicated in Figure 3.4, and so the simulation bandpass filters can be placed with their CFs evenly spaced on the logarithmic frequency axis.

The corresponding cochlear filter functions vary somewhat between models, but in general endeavour to replicate neural data (for example, Kiang et al., 1965). Filters based on such data are highly overlapping and those with high CF exhibit a far steeper decline than their low frequency counterparts (for example, Allen 1985, Ghitza 1987, Seneff 1988, Wang et al., 1992).

The number of filters employed (and therefore their Centre Frequency spacing) also varies, Ghitza has varied the number between 85 (1986, 1988) and 100 (1987) (all using a 3% logarithmic frequency step), Seneff uses 40 spaced 0.5 Bark apart, Watanabe and Hayashi (1995) (following the model of Wang, Sekey and Gersho (1992) employ 15, with a 1 Bark spacing. The number of filters effectively models the sampling of the basilar membrane motion by the inner hair cells, which are assumed to be relatively uniformly distributed along the length of the membrane. The effect of altering the number of filters varies with the particular model employed and the details are discussed in the next chapter where various models are examined in some depth.

2.7 Summary

This chapter has explained the physiological functioning of the outer, middle and inner ears and has explored the coding of sound into neural signals. Combining the volley and place theory for the triggering of the hair cells from the basilar membrane provides a reasonable explanation of the neural perception of pitch, whilst the neural representation of loudness seems to involve both the discharge rate within individual fibres and the spread of activity among different fibres. Auditory models attempt to imitate this behaviour and will be dealt with in the next chapter.

Chapter 3

3 Auditory Models and Evaluations

3.1 Overview

Many approaches to computer speech recognition are based on a spectrographic approach to feature extraction, where the energy of the speech is measured as a function of frequency, and parameters derived from the resulting spectrum are compared to a template or rule. Such techniques involve the computation of FFTs, extraction of LPC and cepstral coefficients, and processing by filter banks. These approaches are sensitive to anything that changes the relative magnitude of in-band energies, and so their performance is often severely degraded by situations of practical interest; for example the reduced spectral bandwidth of phone lines, or the presence of background noise (Holton, 1995).

Invariably, these non-biological models of speech processing and recognition, fall well short of the level of speech recognition that the human auditory system is capable of achieving. Such poor performances have led to an increased level of investigation into auditory models as front ends to recognition systems. There now exists a wealth of physiological data, particularly at the level of the auditory nerve, though the efficient computational coding of such systems remains a difficult task. Another complicating factor is that beyond the periphery, the physiological properties of the system are not at all well-defined, and hence researchers face the task not so much of trying to duplicate the human auditory processing system, but rather of finding a system that works.

There have been many auditory models proposed for speech processing. These models are based on behaviour seen in physiological preparations and/or transformations inferred from psychophysical data. As noted by Klatt (1982), many of these auditory models, can be simulated to first order by a functional model that includes:

(1) A linear preemphasis filter to model the signal boost provided by the resonances in the external ear canal and the middle ear.

(2) A set of approximately linear bandpass "critical-band" filters spaced equally along a Mel or Bark frequency scale to model the mechanics of the basilar membrane.

(3) Half-wave rectifiers to account for the transformation that takes place at the hair cells.

(4) Lowpass filters with short time constants (at least for the high frequency channels).

(5) Lateral suppression circuitry to sharpen peaks in the output spectra.

(6) Partial adaptation of filter outputs that emphasize onsets and possibly offsets.

(7) A log transformation of filter outputs to approximate the phone scale of loudness.

More refined models may also include features such as the attenuation of low-frequency components by middle ear muscle activation, basilar membrane nonlinearities that may account for combination tones and the nonlinear upward spread of masking with increased stimulus intensity.

Researchers examining how the response of the cochlea may be processed to provide a relevant representation of a speech signal generally utilise a computational model to simulate either the direct firing activity or another related representation of the cochlear output. There appear to be two main hypotheses concerning the perceived structural properties of the central processor (Ghitza, 1994): the *place/nonplace* approach, that indicates if the central processor utilises explicit knowledge about the fibre's tonotopic place of origin in the cochlear partition, and the *rate/temporal* approach that indicates whether the central processor uses instantaneous firing rate measurements alone or higher-order firing statistics (e.g., interspike interval statistics). These models are consistent with the physiological theories of the functioning of the inner ear presented in Section 2.4.4.

Early models only considered the average firing rate statistics gathered from primary auditory neurons. However, studies of the responses of auditory neurons to vowel-like sounds suggest that the spectral pattern (as represented by average firing rates versus frequency) changes with stimulus level. The majority of fibres in the region of the first and

second formant frequencies saturate at their maximum firing rate as the level increases to approximately 60 db SPL and output invariance with changes in level is more nearly preserved in the pattern of interspike intervals (Young and Sachs, 1979: Sachs and Young, 1980). This feature has given rise to many models that attempt to estimate the dominant frequency components in the output from the periphery.

Within these *place* and *rate* components, the following categories are traditionally used: First, a *place/rate* category, where the central processor possesses explicit knowledge of place and uses only instantaneous rate information. Second a *place/temporal* category where place information is used together with detailed temporal information of local neural responses, and finally a *nonplace/temporal* category, where place information is omitted altogether and the only sources of information are the temporal properties of the global neural responses.

Particular auditory models are heavily referenced in the speech processing literature. With respect to the three categories listed above, often mentioned is the *place/rate* cochlear model of Lyon (for example, Cohen 1989, Cooke 1992, Lazzaro 1997), the EIH and SBS *nonplace/temporal* models of Ghitza (for example, Cohen, 1989, Jenison et al. 1991, Cooke 1992, Kajita and Itakura, 1994, Alwan, 1995, Jankowski 1995) and the Synchrony/Mean-Rate *place/temporal* model of Seneff (for example, Beet 1990, Jenison et al. 1991, Cosi 1993, Dermody et al, 1993, Ghitza 1994, Alwan 1995, Jankowski 1995, Holton 1995, Lazzaro 1997).

One thesis objective is to compare a computational model of the auditory periphery with a more biologically correct model (as well as a classical non-biological algorithm). The above list of auditory models is by no means exhaustive, but the models of Lyon, Ghitza and Seneff can be considered as indicative of the *place/rate*, *nonplace/temporal* and *place/temporal* models that exist in the literature. The SBS model of Ghitza and its promise of a reduced frequency representation was chosen as the computational auditory model. As a comparison, the more physiologically accurate model of Seneff was selected as it is extremely well respected by contemporary researchers in speech processing, and is one of the most

referenced auditory models. Details of the Ghitza and Seneff models are presented in Sections 3.3 and 3.4, respectively.

3.2 Speech Synthesis Using Sinusoids

In many speech production models, the speech waveform, $s(t)$, is assumed to be the output of passing a vocal cord (glottal) excitation waveform through a linear system that represents the characteristics of the vocal tract. For many applications the simplification can be made that the glottal excitation will be in one of two possible states corresponding to voiced or unvoiced speech. The excitation function is often represented as a periodic pulse train during voiced speech, where the spacing between consecutive pulses corresponds to the pitch of the speaker and is represented as a noise-like signal during unvoiced speech. However, McAulay and Quatieri (1986) have successfully demonstrated that this binary voiced/unvoiced excitation model can be replaced by a sum of sine-waves. The motivation for this is that voiced excitation, when perfectly periodic, can be represented as a Fourier series decomposition in which each harmonic component corresponds to a single sine wave. In the general case, when the periodicity is not exact or the speech is unvoiced, the sine waves will be aharmonic. Passing this sine-wave representation through the time-varying vocal tract results in a representation for the speech waveform, which for a given frame is described by

$$s(n) = \sum_{l=1}^{L} A_l \cos(\omega_l n + \varphi_l) \qquad \textbf{Equation 3.1}$$

where A_l and ϕ_l represent the amplitude and phase of each sine-wave component associated with the frequency ω_l, and L is the number of sine waves.

When speech is perfectly periodic, the sine-wave parameters correspond to the harmonic samples of the Short-Time Fourier Transform (STFT), in which case Equation 3.1 reduces to

$$s(n) = \sum_{l=1}^{L} A_l \cos(nl\omega_0 + \phi_l) \qquad \textbf{Equation 3.2}$$

If the STFT of *s(n)* is given by:

$$S(\omega) = \sum_{n=-N/2}^{N/2} s(n)\exp(-jn\omega) \qquad \textbf{Equation 3.3}$$

then Fourier analysis gives an estimate for the amplitude of $A_l = |S(l\omega_0)|$ and for the phase of $\phi_l = \arg S(l\omega_0)$. The magnitude of the STFT will have peaks at multiples of ω_0. When the speech is not perfectly voiced, the peaks of the periodogram will not necessarily be harmonic, but can still be used to identify an underlying sine-wave structure. In this case the sine-wave amplitudes and frequencies correspond to the peaks of the periodogram, and the phases are computed from the real and imaginary parts of the STFT.

The assumption of a rectangular window in the above, will result in a poor sidelobe structure that will compromise the performance of the estimator, and other windowing routines (designated by *w(n)*) must be employed, though these will be at the expense of broadening the mainlobes of the periodogram estimator. A Hamming window is a common compromise (Appendices A and D), and McAulay and Quatieri (1995) state that the window width must be at least two and one-half times the average pitch period in order to maintain the resolution properties that are needed to justify the peaks of the periodogram. They also note that the placement of the analysis window is vital for computing the phases, and indeed an error in the symmetrical placement of the window about the centre of the analysis frame could lead to a phase error of the order of π.

For unvoiced speech, this sinusoidal representation is valid provided the frequencies are "close enough" that the ensemble power spectral density changes slowly over consecutive frequencies. This is generally satisfied with windows of the order of 16-20ms in length (though McAulay and Quatieri have gone as low as 10ms).

Given a set of frequencies, amplitudes and phases for a particular analysis frame, synthetic speech can be computed from

$$\hat{s}^k = \sum_{l=1}^{L^k} A_l^k \cos\left(n\omega_l^k + \phi_l^k\right) \qquad \textbf{Equation 3.4}$$

but since the sine-wave parameters will be time-varying, discontinuities at the frame boundaries will result unless some smoothing process is introduced. Numerous methods have been proposed to perform this task, most involving some technique of sine-wave tracking, linear interpolation of the amplitude and cubic interpolation of the phase characteristics (e.g. $\theta(t) = \zeta + \gamma t + \alpha t^2 + \beta t^3$) (McAulay and Quatieri, 1986). Such procedures are computationally intensive, even if simplifications such as harmonic matching are applied. McAulay and Quatieri (1995) use an overlap-add interpolator in which Equation 3.4 is applied to the sine-wave data on frames k-1 and k to generate the waveforms $\hat{s}^{k-1}(n)$ and $\hat{s}^{k}(n)$ respectively. These are then appropriately weighted, overlapped and added according to

$$\hat{s}(n) = \omega_s(n)\hat{s}^{k-1}(n) + \omega_s(n - T)\hat{s}^{k}(n)\,(n - T) \qquad \textbf{Equation 3.5}$$

where $\omega_s(n)$ is the overlap-and-add synthesis window that is designed such that

$$\omega_s(n) + \omega_s(n - T) = 1 \qquad \textbf{Equation 3.6}$$

They have shown that there is no discernible difference using the Triangular and Hanning windows (note that the Hamming is only a modified form of the Hanning – refer Appendix D), and so the window which is most computationally simple appears to be the obvious choice. Their experiments indicate that not only is the synthetic speech indistinguishable from the original, but there is no improvement from using the matching and interpolation algorithms in place of the simple overlap-add method.

In summary then, a speech synthesis system is possible based on a sum of sinusoids of appropriate frequency, amplitude and phase, providing consideration is given to windowing and frame recombination issues. The Ghitza and Seneff models will be used as a front end to select these frequency components, and the speech will be synthesised by employing the appropriate sinusoids.

3.3 Ghitza's SBS Auditory Model

3.3.1 Overview

Ghitza (1987) proposes a system that endeavours to synthesize speech "so as to match the representations of the synthesized and the original speech at the auditory nerve level" by creating a simple approximation of the auditory periphery (up to the auditory nerve level) followed by a nonlinear relative spectrum intensity measure that uses timing-synchrony information to exploit the in-synchrony phenomena observed in the neuron firing patterns. This representation he terms the "in-Synchrony-Bands Spectrum" (SBS), and when combined with the sinusoidal representation system suggested by McAulay and Quatieri (1986), provides for a one-step rule for synthesis by analysis.

As the auditory nerve is the only link from the auditory periphery to the brain, all the information that is processed by the higher auditory system must exist in the auditory nerve firing patterns. Measurements of the firing patterns of cats' auditory nerve fibres in response to stimuli (Sachs and Young (1979), Delgutte and Kiang (1984)) indicate that use should be made of both the temporal characteristics as well as the firing rate. These experiments also demonstrated that as the stimulus intensity increased, more fibres fired in synchrony with the stimulus periodicity. Ghitza employs the temporal nonplace approach (Carlson et al., 1975), assumes that the relative intensity of different spectral portions of the signal is in the number of fibres that fire synchronously (regardless of the fibre's characteristic frequency (CF)), and also assumes that the phase response of the fibre's firing is irrelevant (Goldstein and Srulovicz, 1977).

The speech analyser consists of two stages, one to model the peripheral auditory processing structure up to the level of the auditory nerve, and the other to provide a heuristic nonlinear relative spectrum intensity measure. The first stage consists of a simple model of the cochlear filters, and comprises 100 highly overlapping filters equally spaced on a logarithmic scale with a three percent frequency step. The frequency response of these filters is similar to the tuning curves of the auditory nerve fibres, with the filters up to 1 kHz

possessing a +18 dB per octave incline on the low frequency side and a -18 dB per octave roll-off on the high frequency side. Those filters whose CF lies above the 1000 Hz mark possess the same incline slope, but have a very high roll-off.

The second stage of the model involves forming the *in-synchrony bands,* defined as "a region of L_n successive filters having the same dominant frequency f_n, where the dominant frequency is the frequency of the strongest component in the filter's output signal". L_n must be greater than some threshold in order for a region to be declared as an in-synchrony band. The SBS then, is the collection of L_n's at frequencies f_n. Details of spectral energy distribution are not considered, only the frequency of the dominant component is measured.

Ghitza implemented this model using a 20 ms speech-frame, analysed 100 times per second. Due to the nonlinear positioning of the filter's CFs (modelling the non-uniform distribution of fibres along the Basilar membrane), most of the frequency components in the first formant are represented in the SBS display, whilst successive formants become progressively poorly illustrated. The synthesis-by-analysis system employed by Ghitza, uses the SBS spectrum to select which frequency components are to be used in the speech reconstruction, and then performs this reconstruction by employing the original amplitudes and phases of the speech at those frequencies.

Ghitza's results (stated without a value of the SBS threshold, though the maximum number of sine wave components was set at ten) give a saving in the number of frequency components employed in the reconstruction of between 60 and 80 %, with male utterances resulting in a 10% greater saving than female. He reports natural and highly intelligible speech, but that it includes some tonal artefact, due to the failure of the SBS to adequately reproduce voiceless frames. The claim is also made that noise does not seriously affect the performance of the system, and indeed, some noise reduction is exhibited due to the coherence property of the Fourier Transform and the dominance effect of the SBS.

The most interesting feature to draw from Ghitza's model, is his claim that only ten frequency components at the most, are sufficient to obtain natural and highly intelligible synthesised speech, though problems remain with the unvoiced frames. However, the potential exists (as he states) for an efficient speech coding system based upon this

technique, given that the average number of sinusoidal components required for synthesis is reduced by about 70%.

3.3.2 Implementation

A block diagram of the software implementation of Ghitza's model is shown in Figure 3.1. The speech is lowpass filtered to telephone bandwidth, divided into an appropriate frame size, windowed, and for some experiments pre-emphasised, before being converted to the frequency domain by a Fast Fourier Transform. The frequency domain data is fed into the 100 overlapping bandpass filters (whose CFs are given in Appendix G), and for each filter, the dominant frequency is extracted. Following this, the SBS spectra is formed by summing the number of contiguous filters that have the same dominant frequency. The SBS spectra is then used to select frequencies for speech synthesis, using either the original or quantised forms of that frequency's amplitude and phase.

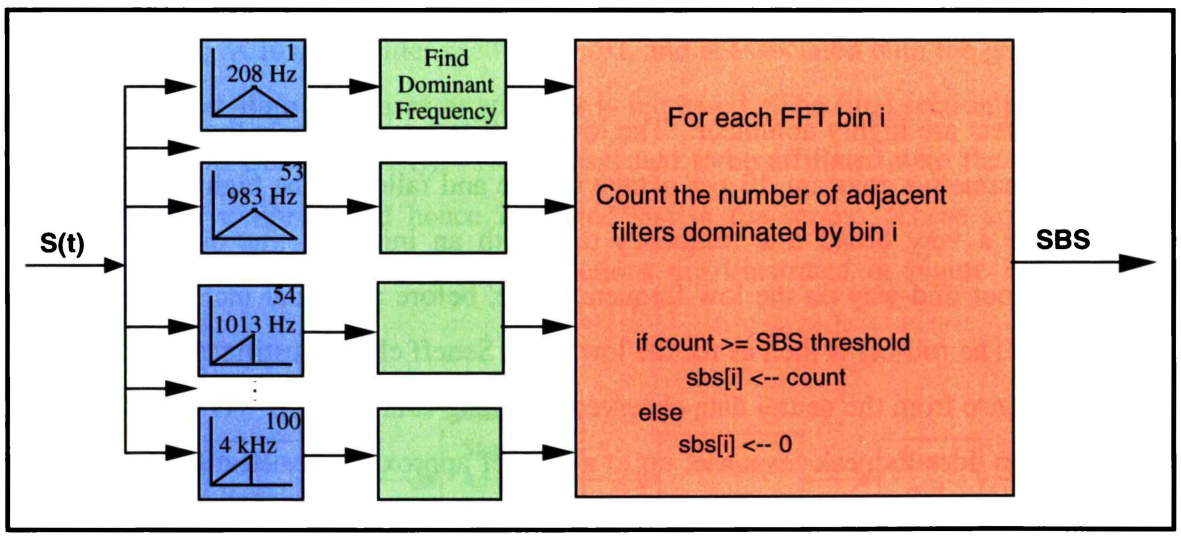


Figure 3.1: Implementation of Ghitza's SBS Model

Subsequent chapters detail experiments performed using Ghitza's model with varying frame sizes, windowing routines, filter shapes, pre-emphasis levels, and SBS threshold values.

3.4 Seneff's Auditory Model

3.4.1 Overview

Seneff's model consists of a set of 40 independent channels over the frequency range 130 Hz to 6400Hz, with the bandwidth of each channel set to approximately 0.5 Bark. The speech is sampled at 16 kHz. Forty channels were chosen as a compromise between spatial resolution of the cochlear output and the amount of computation involved. The first step in the model is the formation of a linear critical-band filter. Following this is a nonlinear stage to capture the prominent features of the transformation from basilar membrane vibration to the probabilistic response properties of auditory-nerve fibres. The output from this stage represents the probability of firing as a function of time for a collection of similar fibres acting as a group, and are delivered to two parallel modules, one to determine the envelope amplitude (the mean discharge rate), the other to measure the extent to which information near the CF of the linear filter dominates the output (the synchronous response, Figure 3.2).

The 40 filters are highly nonlinear. The low CF filters approximate the triangular shape of Ghitza's, namely a constant dB per octave incline and falloff. The high CF filters (above 2 kHz) have a very broad low frequency tail, with an initial incline as per the low CFs, flattening out mid-way on the low frequency side, before a gradual incline attains the peak response. The roll-off is then as for the low CFs. Seneff claims justification for this form of filter response from the neural data observed by Kiang et al., (1965). As the CF of the filters increase, so does its peak response, up to a gain of approximately 10 dB for the 6kHz final filter. This pre-emphasis of the speech data is common in traditional spectral analysis techniques, and has a physiological basis as discussed in Section 2.2.

After the filtering stage, Stage II of Seneff's model is the transformation from basilar membrane vibration to the auditory-nerve fibre responses. This stage incorporates such nonlinearities as dynamic range compression, half-wave rectification, short-term and rapid adaptation and forward masking. At the conclusion of this stage, the output represents a probability of each modelled auditory nerve firing.

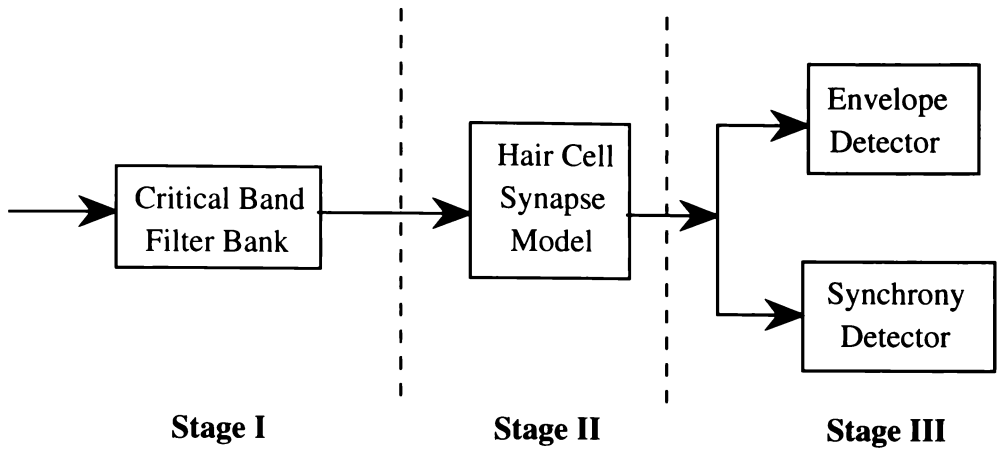


Figure 3.2: Overview of the Seneff Auditory Model

Most of the components in Stage II are nonlinear, and therefore the ordering is important. Seneff draws a physiological comparison with each of these components. The hair-cell current shows a distinct directional sensitivity, modelled by Seneff as a half-wave rectification process. It is assumed that short-term adaptation occurs in the synaptic region between the hair cell and the nerve fibre (Eggermont, 1973), so this section follows the rectification process. A lowpass filter section is next, and is associated with the gradual loss of synchrony in nerve-fibre responses as stimulus is increased. The final section provides a rapid Automatic Gain Control (AGC). This is assumed to be affiliated with the refractory phenomenon of nerve fibres, and hence is placed last in the series. These components together with Seneff's suggested auditory affiliations are illustrated in Figure 3.3, (from Seneff (1988)).

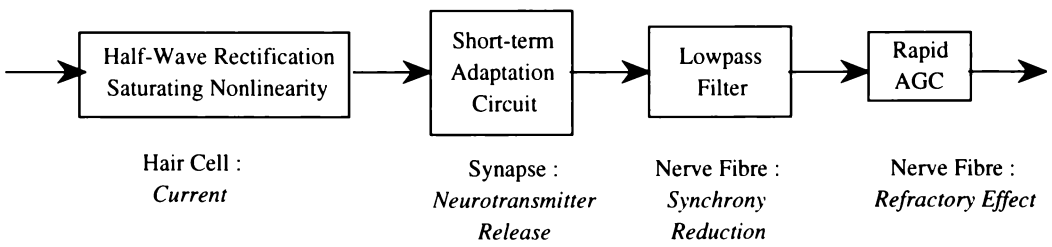


Figure 3.3: Stage II of the Seneff Model with Auditory System Affiliations

The output from Stage II is fed into two parallel and independent processes, to obtain both a synchrony spectrum and one which models the average firing rate of each channel's auditory

nerve fibres. It is the synchrony spectrum that is of most interest for the frequency selection processes employed in this thesis. The synchrony detector takes the Stage II outputs and via a Generalized Synchrony Detector (GSD) attempts to produce a spectral representation that preserves the prominent peaks at the formant resonances. The GSD is based on the ratio of the estimated magnitude of a sum waveform to the estimated magnitude of a difference waveform. The input waveforms are the output and delayed output from each channel of Stage II, where the delay period corresponds to the centre frequency of the corresponding auditory filter. Only the channel whose CF is closest to a particular input frequency will have a large response, adjacent channels will be significantly smaller.

The mean-rate detector was coded, but not actually used in this project. Of more interest was the synchrony detector. Each channel output from Stage II is processed through a GSD tuned to the centre frequency of the corresponding peripheral filter in Stage I. This tuning is implemented by splitting the Stage II output signal into two, feeding one directly to the GSD, and delaying the other by a specific period as illustrated in Figure 3.4.

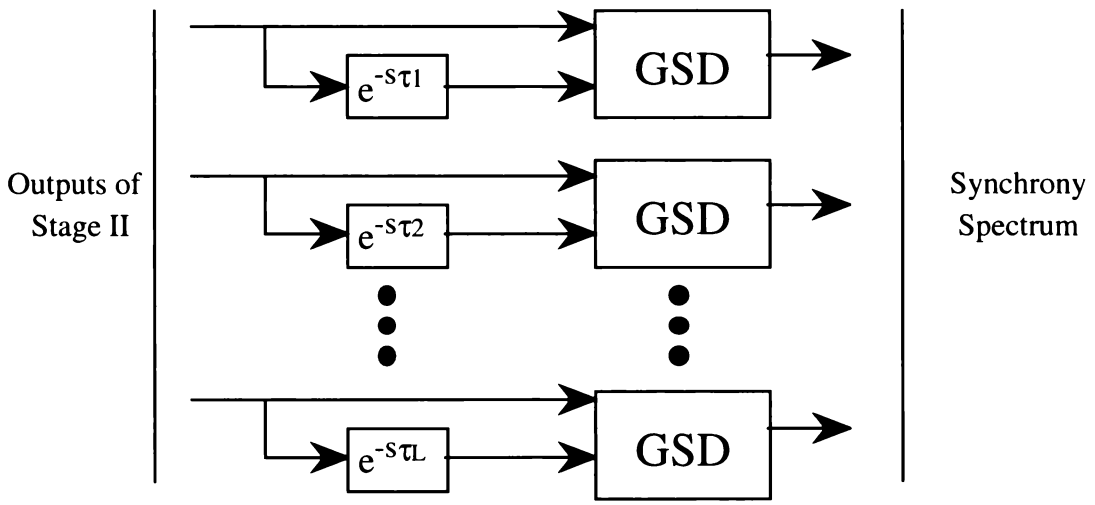


Figure 3.4: Synchrony Branch of Seneff Stage III

In the GSD, the two signals are summed (x) and differenced (y). A threshold δ, (set at a level slightly greater than the spontaneous discharge rate) is subtracted from the summed signal (x) to prevent a response from very weak signals. If the input to the GSD is perfectly periodic with the delay period, the difference waveform (y) will be zero, and therefore the

ratio of x to y will be infinitely large, and hence must be constrained by a final saturating component. This process is illustrated below in Figure 3.5.

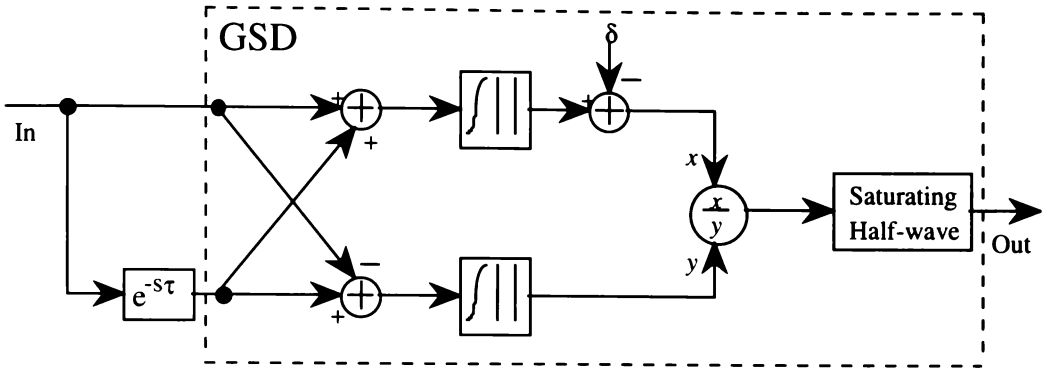


Figure 3.5: Seneff's Generalized Synchrony Detector (GSD)

3.5 LPC

The source filter model of speech production assumes that a source (voicing or fricative excitation) is passed through a filter (the vocal tract response) to produce speech. One of the simplest implementations of this model is known as a Linear Predictive Coder (LPC) synthesiser, of which the most common variant is termed the LPC10e (so called because ten coefficients are typically employed). At every frame, the speech is analysed and the filter coefficients, energy of excitation, and voicing decision (plus a pitch value if voiced) are computed. The decoder passes a regular set of pulses or white noise (depending on whether the speech is voiced or unvoiced respectively) through the linear filter, and multiplies it by the gain to resynthesise the speech.

Tremain's (1982) implementation of the LPC10, samples the speech at 8 kHz, and partitions it into frames containing 180 samples (i.e. a frame rate of 44.44 frames/second). The covariance method is used to compute the linear predictive solution, and a set of generalised reflection coefficients are encoded. Ten coefficients are computed for voiced speech, but when an unvoiced decision is made, only four coefficients are used. In the stored (or transmitted) bit stream, 41 bits are used for the reflection coefficients, seven for pitch and the voiced/unvoiced bit, five bits for the gain, and one additional bit is employed for synchronisation. The 54 bits at the 44.44 frames/second rate result in a total bit rate of 2400

bps. The reconstructed speech does suffer from a loss of naturalness, and occasionally a loss of intelligibility.

The LPC system employed for comparison purposes in this thesis is the US Department of Defence's LPC-10 2400 bps Voice Coder, Release 1.5, October 1997. This software was obtained from the web, without modification to the source code. As such, this discussion has only outlined the basis behind the LPC, without providing extensive details. The reader is referred to Tremain (1982) for a more formal treatment of the LPC10 code.

3.6 Isolated Word Intelligibility Analysis

The evaluation of speech intelligibility is complicated by inferences from message context, i.e. from sources other than the phonemic information features of the speech signal. Such characteristics include knowledge of (from Voiers, 1977):

(1) the structure of language,
(2) explicit or implicit situational constraints on response options,
(3) the circumstances occasioning the communication,
(4) the dialectal and idiolectal characteristics of the speaker, and
(5) the immediate prior history of the speech signal itself.

To determine to what extent a speech representation based on the auditory system can preserve phonetic information that is perceptually relevant, a range of isolated word psychophysical experiments have been developed. The Fairbanks Rhyme Test (FRT) limits the stimulus uncertainty to a single phoneme, and hence improves control over the effects of interphonemic constraints (Fairbanks, 1958). Fairbanks' test is concerned only with consonant recognizability as consonants carry most of the useful information in speech and are, at the same time, much more sensitive than vowels to most forms of signal degradation (Voiers, 1983). In order to effectively control other contextual factors, explicit restriction of the response options is required, for example the Modified Rhyme Test (MRT) (House et. al., 1965), which provides six response options. Each option must satisfy the requirements of monosyllabicity, Consonant-Vowel-Consonant (CVC) form (though occasionally this is violated), and constant orthographic representation of vocalic nucleus within a given

ensemble. The requirement of one-letter consonant representation used by Fairbanks was relaxed, and no attempts were made to achieve a phonetic balance.

In a test such as the MRT, each option varies from the others in either the initial or final consonant, though if such restrictions are applied in an arbitrary or unsystematic manner, one may simply substitute one set of unknown contextual constraints for another. Depending on the nature of the response options, listeners may be forced to give responses which are misleading to their perceptions of other features, or they may be deprived of the opportunity to reveal specific deficiencies of the speech being evaluated.

The problem of any scheme that provides multiple options (not only those of the CVC form) is that all options must be equally attractive, and must differ from the correct response in terms of a single elementary attribute or distinctive feature. For example, in the response set:

bee pea vee dee me

each erroneous option differs from the correct response, "bee" in only one aspect. These tests are adequate provided the listener is not subjected to repeated exposure to the test material. For repeated use with the same listener, it is preferable to have different randomizations of the material.

Aside from the lexical limitations in devising a set of five or six real words incorporating symmetric and complete minimal pair confusions of features, there is also the problem with the scanning and long response selection time in some listeners. Inevitably there are also wasted distracters that rarely occur as error responses. A compromise to these early rhyme tests (FRT and MRT) is the Four Alternative Auditory Feature Test (FAAF) (Foster & Haggard, 1987), but the limitations with repeated use, and choice of options – though reduced, still remain.

These limitations can be overcome to a large extent if only two response options are provided to the listener. With such a testing scheme, erroneous responses can, (aside from the effects of guessing), be unequivocally attributed to the characteristics of the test system

employed, and "ambiguity as to the specific cause of an erroneous response can be eliminated by the use of minimally contrasting response options" (Voiers, 1977).

3.6.1 The Diagnostic Rhyme Test

One of the most commonly employed two response test schemes, is termed the Diagnostic Rhyme Test (DRT) developed by W. Voiers in stages until its final version in 1983 (the DRT is an ANSI standard, ANSI S3.2 – 1989). In general, the DRT test attempts to evaluate how well phonetic information is perceived by a human listener.

Voiers' database consists of 96 pairs of confusable words spoken in isolation by several male and female speakers. Each word is of the CVC type, and words in a pair rhyme with their initial consonant varying by a single distinctive feature. The listener must judge which of the two words has been spoken.

The words are equally distributed among six phonetic features, voicing, nasality, sustention, sibilation, graveness and compactness. The voicing and nasality features are representative of virtually every classification system. Sustention corresponds to the continuant-interrupted feature of Jakobson et al. (1952) and to the affrication feature of Miller and Nicely (1955). Sibilation corresponds to the strident-mellow opposition of the former and to the duration feature of the latter. Graveness and compactness are incorporated by the place feature of Miller and Nicely and are taken directly from the system of Jakobson et al. Vowel-likeness is used to distinguish the glides from the true consonants (Voiers, 1983). This is summarised in Table 3.1

voicing	the nature of the source (periodic or non-periodic),
nasality	the presence or absence of a supplementary resonator,
sustention	the continuant-interrupted contrast of Jakobson et. al.
sibilation	the strident-mellow contrast
graveness	from Miller and Nicely's place of articulation – grave or acute
compactness	also from Miller and Nicely's place of articulation, compact or diffuse

Table 3.1: The Six Phonetic Distinctive Features of the DRT Test Words

The phonemic taxonomy on which the DRT is based is shown in Table 3.2. Voiers lists the following constraints in assembling the corpus of test words:

1. For half of the items designed to test for the apprehensibility of *voicing*, both critical phonemes involve friction (i.e., are affricates or affricatives); for half, the critical phonemes are stops.
2. Half of the *nasality* items in each vowel context involve a grave phoneme-pair (i.e., /m-b/); half involve an acute pair (i.e., /n-d/).
3. Half of the items designed to test for the apprehensibility of *sustention* involve a voiced phoneme-pair; half, an unvoiced pair.
4. Half of the items concerned with *sibilation* involve voiced phonemes; half, unvoiced phonemes.
5. In the case of *graveness*, items were constructed such that, for each vowel context, one item lies in the voiced "plane", one in the unvoiced; one lies in the sustained "plane", one in the interrupted.
6. *Compactness* items were constructed such that both states of vowel-likeness, sibilation, voicing, and sustention were given equal representation in the test, though not in all vowel contexts.

There is no definitive standard for testing speech intelligibility, with the MRT, FAAF and DRT all employed by speech researchers. A brief view of the literature shows Gatehouse (1989) using the FAAF, Ozawa and Logan (1989) the MRT, Smith (1969), Duggirala et al. (1988), Calistri and Kallman (1986), van Santen (1993), Ghitza (1994), Laflamme et al., (1996), Deisher and Spanias (1997), Greenspan et al. (1998), Sheffield et al. (2000) the DRT. This is by no means an exhaustive list, but does show there is variability in the method used to determine speech intelligibility, each method with its own merits.

The DRT was selected as the tool for intelligibility testing in this thesis, as it is straightforward to implement, provides intelligibility information over six forms of speech (Table 3.1), and is an acceptable evaluation tool used by many researchers.

	Voicing	Nasality	Sustention	Sibilation	Graveness	Compactness	Vowel-like
m	+	+	-	-	+	-	-
n	+	+	-	-	-	-	-
v	+	-	+	-	+	-	-
ð	+	-	+	-	-	-	-
z	+	-	+	+	-	-	-
ʒ	+	-	+	+	0	+	-
ʤ	+	-	-	+	0	+	-
b	+	-	-	-	+	-	-
d	+	-	-	-	-	-	-
g	+	-	-	-	0	+	-
w	+	-	+	-	+	-	+
r	+	-	+	-	-	-	+
l	+	-	+	-	0	0	+
j	+	-	+	-	0	+	+
f	-	-	+	-	+	-	-
θ	-	-	+	-	-	-	-
s	-	-	+	+	-	-	-
ʃ	-	-	+	+	0	+	-
ʧ	-	-	-	+	0	+	-
p	-	-	-	-	+	-	-
t	-	-	-	-	-	-	-
k	-	-	-	-	0	+	-
h	-	-	+	-	0	+	-

Table 3.2: Consonant Taxonomy Used in Construction of the DRT

3.6.1.1 Administering the DRT

The DRT is quite flexible in the manner in which it can be administered. Crews of eight to ten listeners are recommended (Voiers, 1977) and it is generally desirable to use recordings of the test material by more than one speaker. The DRT is usually constructed so as to test each feature every seven items, where the seventh item is a meaningless filler. Other arrangements are suitable, and, within reason, multiple randomizations of the test materials and of the arrangement of response options on the listener's response form can be tolerated. Word presentation is recommended to be at the rate of one word every 1.33 seconds (Voiers, 1977). Experiments by Hick (1952) found that the time it takes humans to make a discriminative response increases as the logarithm of the number of response options, and thus the optimal rate for the six option MRT would be approximately one item every four

seconds, resulting in a considerably longer testing time than the DRT for the same number of test words. The DRT is one of the primary evaluation tools used in this project, and the implementation and scoring of the DRT is detailed further in Chapter 5.

3.7 Sentence Recognition Analysis

As Ghitza (1992) notes, a long-standing question that arises when studying a particular auditory model is how to evaluate its performance. One approach has been to apply the model as a front end to automatic speech recognition systems, but this has a significant drawback in that the measured performance is of the overall system, front end (auditory model) and back-end (recogniser) combined. In general, there is no way of distinguishing which end generated the error, and hence there is no clear picture of how well the auditory model performs. Additionally, developers of auditory models wish to measure to what extent the model representation can describe the actual human internal representation, a concern that is not addressed if it is implemented in a speech recognition system.

This thesis treats the back-end (the recogniser) as a constant, and measures the output change as the front end (auditory model) parameters are varied. It can reasonably be assumed that in this situation, any change in the output is solely due to the changes implemented in the front end. Such a system will provide valuable information concerning the *relative* recognisability of the processed speech.

A common approach to speech recognition is to use Hidden Markov Models (HMMs). This approach creates a statistical model (rather than a direct template) of each word in the vocabulary. For an unknown input word, a probability is calculated for all words in the vocabulary to see if it matches this input, and the word with the highest probability is chosen as the answer. The HMM represents each word in the vocabulary by a set of states, together with the probabilities of transitions from state to state. When a word is spoken, the process begins in the initial state, makes state transitions at time intervals equal to the separations between speech frames, and concludes at the final state. At each time frame, a vector of acoustic parameters is emitted. The states themselves are not observable (hence the name *hidden*), and the name *Markov* is applied because of the state transitions from a first-order

Markov chain: that is, the probabilities of transitions from a given state do not depend on the traversal of prior states (Grant, 1991).

The Hidden Markov Model Toolkit (V1.2) (HTK) from the Speech Group of the Cambridge University Engineering Department[1] was selected to obtain an objective speech recognition measure of the SBS, Seneff, and LPC algorithms. HMM techniques are commonly employed by speech recognition researchers, and the HTK is a very popular implementation of HMM. This thesis is making no claim concerning the "highest" absolute recognition result obtainable from SBS processing, but rather is endeavouring to compare SBS variations against other algorithms. The HTK can be successfully employed to provide a quantitative comparison measure between these different algorithms. In essence, we are using the HTK as an "objective listener", to complement the results of informal listening and DRTs.

3.7.1 The HTK System

There are eight stages to the HTK system. The first stage creates a time-ordered label file (*.lab*) and in this application, operates on the time-aligned phonetic transcription files (*.phn*) of the TIMIT database (discussed in Section 3.7.2). The *.lab* files (in their default form) replace the TIMIT transcriptions as below:

V replaces *iy ih eh ae ix ax ah ax-h uw uh ao aa ey ay oy aw ow ux*
L replaces *l el r y w er axr*
N replaces *m n en ng em nx eng*
C replaces *ch j jh dh b d dx g p t k z zh v f th s sh hh hv pcl tcl kcl qcl bcl dcl gcl epi*
S replaces *sil h# #h pau*
Occurrences of *q* are deleted.

A *.lab* file must be created for each of the 3696 training files and the 192 test files, but does not need to be repeated when noise is added to the test files.

---

[1] Entropic Research Laboratory gained exclusive marketing rights to HTK, and developed it to V2.2 until the company was sold to Microsoft in 1999, and the HTK software subsequently withdrawn from sale.

The second stage performs a linear prediction analysis of the speech waveform. It divides the input file into a sequence of overlapping frames. A Hamming window is applied to each of these frames, a preemphasis coefficient is set (0.97 in these tests), and linear prediction filter coefficients are computed, converted to LP Reflection coefficients, and written to file with a *.iref* suffix. All the training files and every variance of the test files must be processed by this stage. This stage makes use of the TIMIT header (1024 bytes) information which (amongst other details) states the number of sample points in the file. This value is compared by stage 2, to the actual (read-in) number of sample points, and, if the two numbers are different, an error status occurs. In the SBS and Seneff processing of the TIMIT files, the data is truncated to the nearest complete frame (128 points), and therefore, the output generally contains fewer data points than the original. Consequently, the header information must be modified, otherwise stage 2 of the HTK system will not successfully execute. To achieve this, a routine was written to search the header file for the variable containing the number of samples in the file, and then decrement this value by at least the number of data points in a single windowed frame.

Stage 3 provides initial estimates for the means and variances of a single HMM by repeatedly using the Viterbi alignment to segment the training speech files and then recomputes the means and variances by pooling the vectors in each segment. A prototype HMM definition must be provided that defines the required HMM topology, i.e. it has the form of the required HMM except that means, variances and mixture weights are ignored. The output from this stage is fed to stage 5. A maximum of ten estimation cycles are used, an arbitrary compromise between accuracy and computation time.

Stage 5 performs basic Baum-Welch re-estimation of the parameters of a single HMM. A maximum of 20 estimation cycles are allowed for this, and the output is used to seed stage 7. Stage 7 uses an embedded training version of the Baum-Welch algorithm. For each training utterance, a composite model is effectively synthesised by concatenating the phoneme models given by the transcription. Each phone model has the same set of accumulators allocated to it as used in stage 5, but now they are updated simultaneously by performing a standard Baum-Welch pass over each training utterance using the composite model (Young, 1990).

Stages 4, 6 and 8 are essentially the same, but operate on the outputs of stages 3, 5 and 7 respectively. Initially these stages call a Viterbi recogniser that matches a network of HMMs against one or more speech files and outputs a transcription in HTK format for each. The output from this (in the form of HTK Label format files) is compared using a Dynamic Programming algorithm, with the corresponding transcription file. Output statistics include % Corr and Acc, defined as:

%Corr: The percentage number of labels correctly recognised, given by:

$$\%\text{Corr} = \frac{H}{N} \times 100 \qquad \textbf{Equation 3.7}$$

where H is the number of correct labels, N is the total number of labels in the defining transcription files

Acc: The accuracy computed by

$$\text{Acc} = \frac{(H - I)}{N} \times 100 \qquad \textbf{Equation 3.8}$$

where I is the number of insertions. To simplify the graphical outputs, only the % Corr results from stage 4 are plotted in the results chapters – remembering that it is the comparative trends that are important, not the final absolute values.

3.7.2 Test and Training Considerations

The TIMIT corpus of read speech has been designed to provide speech data for the acquisition of acoustic-phonetic knowledge and for the development and evaluation of automatic speech recognition systems, and so was chosen to provide the test and training sentences for the HTK experiments. TIMIT is sponsored by the Defence Advanced Research Projects Agency (DARPA), and the text corpus design was a joint effort by the Massachusetts Institute of Technology (MIT), Stanford Research Institute (SRI) and Texas Instruments (TI).

TIMIT contains a total of 6300 sentences, ten sentences spoken by each of 630 speakers from eight major dialect regions of the United States. The dialect regions correspond to the graphical areas (defined by the Language Files, Ohio State University Linguistics Dept., in 1982) in the U.S.A. where the speaker lived during their childhood years with the exception of the "Army Brat" classification where the speakers moved continually during their childhood. The dialect regions are New England, Northern, North Midland, South Midland, Southern, New York City, Western and Army Brat, coded DR1 through to DR8 respectively in Table 3.3.

The TIMIT material comprises two dialect "shibboleth" sentences designed at SRI (designed to expose the dialectal variants of the speakers), 450 phonetically-compact sentences designed at MIT (designed to provide a good coverage of pairs of phones), and 1890 phonetically-diverse sentences selected at TI (designed to add diversity in sentence types and phonetic contests). TIMIT codes these sentences as SA, SX and SI respectively.

The TIMIT database provides the speech database for the HTK experiments. An initial consideration is the number of files to be used for both the training of the HTK system, and the testing of the speech processing algorithms. Tests were performed on a restricted range of 17 TIMIT files (both male and female speakers) to characterise the system, and to attempt to determine whether there was any improvement in recognition upon varying either the SBS threshold, or the quantisation parameters. Due to this small training set, the results from this series of experiments were erratic – no discernible trend in recognition scores were evident.

The TIMIT documentation includes suggestions for a training/testing subdivision. The suggested training set comprises 462 speakers (both male and female), each speaking three SI and five SX sentences, (refer Table 3.3), providing a training base of 2310 SX sentences and 1386 SI sentences. Despite the formidable size, it was decided to use the entire suggested 3696 sentence training base. Such a large training base can be justified (computationally) by the fact that the training files need only be processed once for each processing algorithm (i.e. varying SBS threshold, LPC).

	Number of Speakers	SX	SI	Total
DR1	38	190	114	304
DR2	76	380	228	608
DR3	76	380	228	608
DR4	68	340	204	544
DR5	70	350	210	560
DR6	35	175	105	280
DR7	77	385	231	616
DR8	22	110	66	176
Totals	**462**	**2310**	**1386**	**3696**

Table 3.3: Composition of the TIMIT Training Data

The TIMIT restrictions on the testing files are:

- That no speaker should appear in both the training and testing portions.
- All the dialect regions should be represented in both subsets, with a least one male and one female speaker from each.
- The amount of overlap of text material between the two subsets should be minimized – if possible no texts should be identical.
- All the phonemes should be covered in the test material, and preferably, each should appear multiple times in different contexts.

The test data must be reprocessed for each S/N level of added noise (both cocktail party and white), and repeated for every processing algorithm. In terms of computation time, it was necessary to use the TIMIT suggested "core test portion" of 24 speakers (two male and one female) from each of the eight dialect regions. As for the training data, each speaker in the test core reads five SX and three SI sentences, with no two speakers reading the same eight sentences. The result is a core test base of 192 sentences each processed over 28 different noise levels for every change in the speech processing algorithms (i.e. each variable change requires the retesting of 5376 TIMIT speech files).

3.8 Summary

This chapter has presented an overview of auditory modelling, with particular emphasis on Ghitza's SBS model and Seneff's Synchrony/Mean-Rate model. These models can be used to select specific frequencies for speech synthesis using the synthesis by sinusoid process of McAulay and Quatieri. The SBS process selects frequencies based on their ability to have the dominant response in contiguous bandpass filters, whereas the Synchrony/Mean-Rate model employs a synchrony detector to select which of 40 auditory nerve fibres has the highest probabilistic response. Ghitza has reported preliminary results on the SBS algorithm suggesting that highly intelligible speech can be reconstructed using only a fraction of the original frequencies.

The synthesised speech can be evaluated by subjective listening tests, or by objective measures. These objective measures can be grouped into three categories, intelligibility, quality, and speech recognition performance. For the speech reconstructed by the algorithms developed in this thesis, intelligibility and speech recognition performance were chosen as the appropriate objective measures, and were implemented by the DRT and HTK respectively. The results are to be compared with a non-auditory process, the LPC10.

Chapter 4

4 Speech Coding using SBS

4.1 Overview

Ghitza's auditory model, based on the In-Synchrony-Bands-Spectrum (SBS), as described in Section 3.3, is a natural candidate for speech compression. Justification for this comes from the nature of the SBS speech reconstruction - the speech is reproduced using only a fraction of the original frequency bins. This chapter presents a preliminary analysis of Ghitza's model, and discusses the effect of varying the windowing routines, frame size, filter shape, and frame overlap, both on uncorrupted speech, and speech in the presence of significant levels of white and cocktail party noise. Such are the number of possible variations of all these parameters, that it is not feasible to test each one formally and individually via (say) an HTK process, or formal Diagnostic Rhyme Test (DRT) analysis. To account for this, speech was compressed using SBS, and the resulting intelligibility measured using *informal* DRTs. At the conclusion of the chapter, the possible variations of SBS processing will be reduced down to 26, which will be formally tested by the HTK process (Chapter 6).

4.2 Speech Sampling

The experiment described in Ghitza (1987) lowpass filters the speech to 5 kHz, pre-emphasises by 6 dB/octave and then samples at 10 kHz. Software analysis is performed on 20 ms Hamming windowed frames, with 50 % frame overlap (50 % with the preceding frame, 50 % with the succeeding frame, i.e. 100% data redundancy). This produces an analysis rate of 100 frames per second.

The digitising boards used in this project sample at 8 kHz, so the lowpass filter has a maximum cutoff at 4 kHz. Fast Fourier Transforms (FFTs) are extensively used, and, optimally, require 2^N data points (where N is an integer). To retain the necessary frame periodicity required to perform Fourier analysis, the choice of frame sizes was limited to either 128 or 256 data points (corresponding to 16 and 32 ms respectively). A 32 ms frame was trialled, however SBS spectra and reconstructed speech produced very similar results to the 16 ms frame, and the 256 data point frame was therefore discarded. A 16 ms frame is a more common frame length, and is closer to that employed by Ghitza, and was therefore chosen for this implementation.

In traditional speech analysis, speech is typically pre-emphasized prior to Fourier analysis. Some form of pre-emphasis can be justified from an auditory standpoint from the resonances in the concha and external auditory canal (Chapter 2). The gain in acoustic pressure provided by the outer ear generally extends from 1.5 to 7 kHz. Cutoff frequencies for this model are well below 7 kHz, so pre-emphasis does not amplify higher frequencies than can be physiologically justified. However, this 6 dB per octave preemphasis does provide a gain to those frequencies below 1.5 kHz, which is not mirrored in the human auditory system. Alternatives to this preemphasis were examined, and will be discussed later in the chapter.

Our implementation of Ghitza's model is shown in the diagram below. Blue squares indicate an input into the simulation, whether the speech waveform, or some argument to dictate the form of processing. Orange blocks indicate some conversion process, not necessary in the final implementation, but essential for experimentation. The main processing is performed in the green squares, initially involving signal conditioning (segmentation, windowing, transform and SBS processing), and then the various quantisation routines. The purple blocks indicate *wav* format I/O that was used in later experiments.

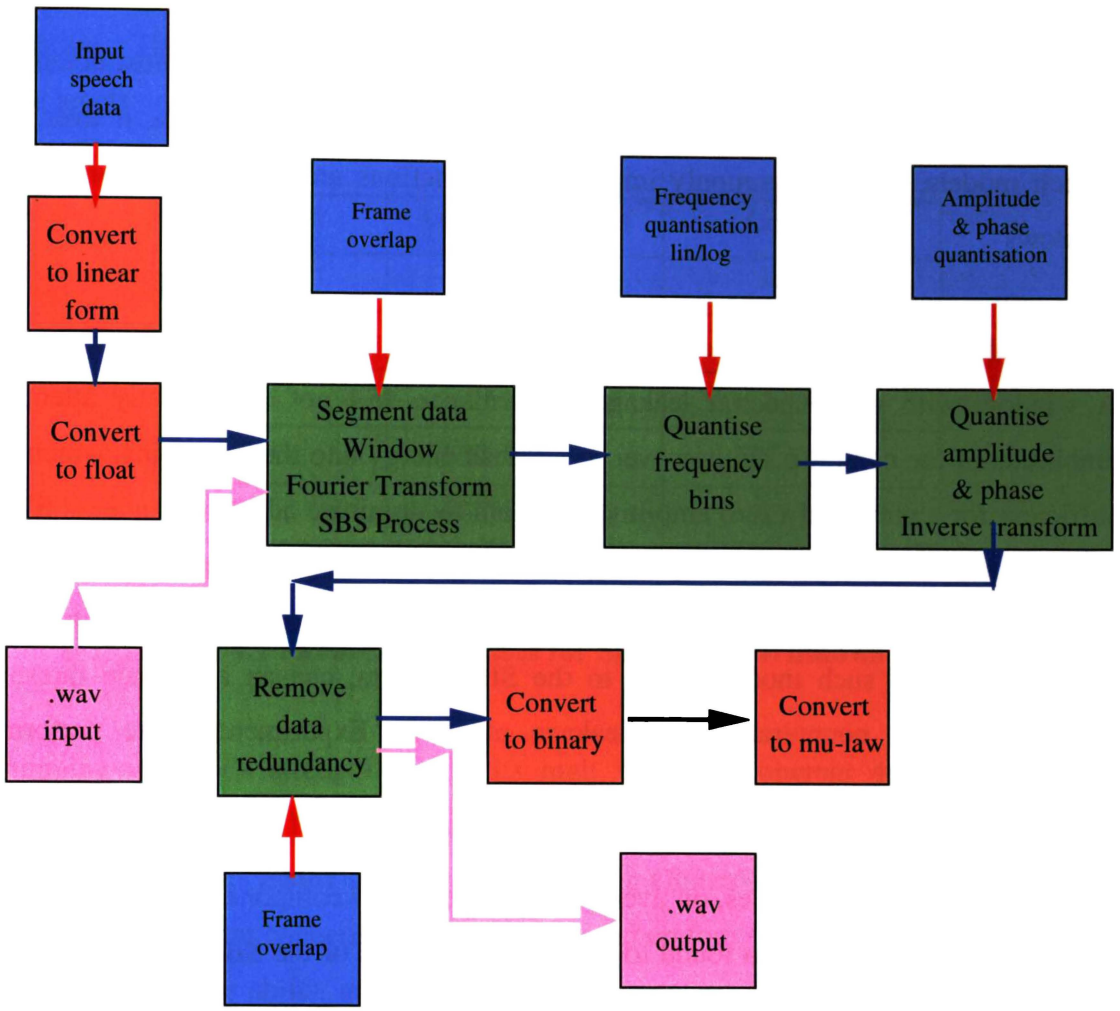


Figure 4.1 : Simulation Implementation

4.3 Windowing in the Simulation

The need for frame windowing is well understood, and the shape of the windowing function can have a major influence on the characteristics of the reconstructed speech. A selection of windows were trialled in this simulation, including the Rectangular, Triangular, Hanning, Hamming, Blackman, Blackman-Harris and the Kaiser. Important parameters for window selection include coherent gain, side-lobe falloff and spectral leakage. These issues are discussed in Appendices A and D.

Rectangular and Triangular windows possess such poor characteristics, most notably in the height of the side-lobes and the side-lobe fall-off, that they are seldom, if ever, used in speech models. More commonly implemented functions are the Hanning and Hamming windows

Of critical importance is whether leakage and frequency smear will alter the SBS spectrum. A window with poor spectral leakage performance will not significantly attenuate the amplitude of the central peak, however it will leak energy into the other bins, which may be sufficient for a previously zero amplitude bin to now dominate adjacent bandpass filters and hence acquire a non-zero SBS amplitude. Conversely, a well performed window in terms of suppressing leakage, will attenuate the central peak, and so reduce its SBS amplitude. Given the possibility of such modifications to the SBS spectra, can an amplitude threshold be chosen that will counteract these leakage effects? Experiments were performed to investigate this.

An SBS threshold of ten gives, on average, three frequency components per frame of speech data, and this value was also found to reject practically all of the SBS amplitudes generated as a result of spectral leakage. However, when significant energy was distributed between two frequency bins (smearing of the central peak), this value of the threshold generally resulted in both frequencies being selected. For further data reduction, an additional algorithm that rejects one of these frequencies could be possible with no cost in terms of intelligibility. This will be explored later.

Of concern is that the leakage of energy from a frequency bin, combined with the selection of a reduced set of frequencies, will result in a deterioration in the amplitude of the reconstructed frame. In extreme cases (speech input rather than pure sine waves) this effect may render the frame inaudible. The sub-unity coherent gain figures of most of the windowing routines further aggravates this processing loss. Experiments performed with various levels of SBS thresholding, including no SBS processing, and a control experiment that only windowed the data, i.e. no Fourier transformation, verified this. Even when windowing was the only process performed on the data, significant energy loss was evident for all but the rectangular window option. As expected, the higher the SBS threshold, the

larger the energy loss as successively more frequency bins were removed from the reconstruction. These results are presented below in Table 4.1, and are classified using 6 subjective levels: nil, very small, small, moderate, considerable and severe.

Window	Amplitude Loss	Leakage
Rectangular	nil	severe
Hamming	small	moderate
Hanning	moderate	small
Blackman	considerable	very small
Kaiser	considerable	very small

Table 4.1 : Processing Losses for Simulation Windows

The Hamming window's strength lies in its high value of coherent gain, and leakage properties superior to the Triangular and Rectangular windows, and is therefore used extensively in speech processing applications, including Ghitza's original SBS model. Loss of frame energy is of great concern in this project, and therefore a large value of the coherent gain is desirable. Phase shifts, and noise rejection are not so important and poor spectral leakage performance may be acceptable given an appropriate SBS threshold. Indeed, in some speech frames, this may be preferable as less frequency smearing will result. Given these constraints, the Hamming does appear to be a suitable window for further SBS experimentation.

4.4 Phase Shift

A side effect noted during the leakage experiments was the amount of phase shift that occurs during the process. This was found to be related to the FFT thresholding that is a consequence of the SBS process. By setting a value of SBS threshold such that only one frequency component (for the test sinusoid) is retained, it was found that the more energy was removed from the frame, the higher the degree of phase shift. Considering the matter in the frequency domain, we are eliminating a large amount of information upon reconstruction, namely all the phases and amplitudes of the other N/2-1 bins, where N is the

number of sample points per test frame. It can not reasonably be expected for either the amplitude or phase to be retained upon inverse transformation if these deleted frames possessed non-zero data.

Phase shift was investigated only qualitatively, as it is expected that during the compression experiments, phase would be discounted. Results indicate that the higher the amplitude of the selected frequency component, the smaller the degree of phase shift. For progressively smaller central frequency amplitudes, the phase shift grew more pronounced. This is expected as a small central frequency amplitude indicates a considerable energy loss from the frame and therefore a more corrupt reconstructed waveform.

4.5 The Bandpass Filters

Ghitza's speech analyzer is composed of two stages, the first modelling the peripheral auditory structure up to the level of the auditory nerve. This stage is based on a simple model of the cochlear filters, consisting of a bank of 100 highly overlapping filters equally spaced on a logarithmic scale with a 3 percent frequency step (Appendix G).

Ghitza divides his bandpass filters into two categories; those whose centre frequencies lie below 1000 Hz, and those whose CFs are above. The lower range filters are created to be symmetric about the CF, with an 18 dB per octave incline and an -18 dB per octave decline. The higher range filters retain the same incline slope, but have a very sharp drop-off. The exact slope of the drop-off is not stated by Ghitza, though he does indicate that the slope must be very large, and is satisfied in this simulation by a decline slope of -120 dB per octave. These frequency responses are similar to the tuning curves of the auditory nerve fibres (as discussed in Chapter 2). Filtering in the log-amplitude log-frequency domain is accomplished by adding the input log spectrum with the log-frequency response of the filter. This is equivalent to multiplying the time domain responses. An interesting variation is to multiply time domain response by the log gain of the filter. This produces speech that is less sensitive to SBS thresholds, but does require high frequency pre-emphasis to avoid masking of high formants by lower ones. Such variations of Ghitza's model are discussed later.

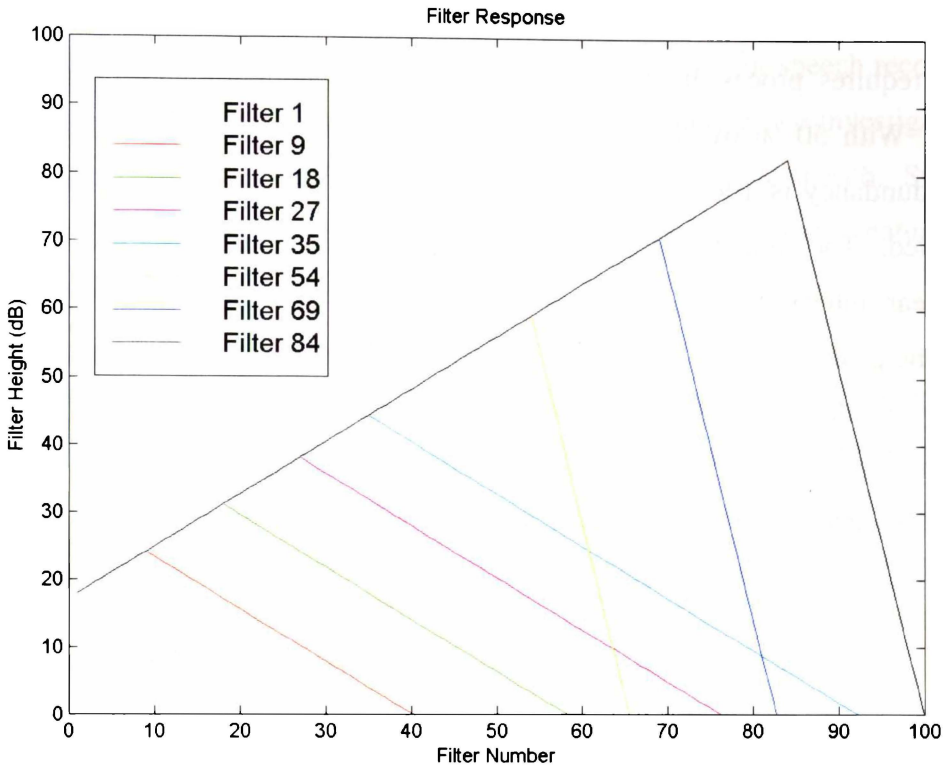


Figure 4.2 : Ghitza's Bandpass Filters for SBS Processing

The structure of the overlapping filters is vital in determining the resulting SBS spectrum. All filters span the lowest frequency bins, increasing the likelihood that a low frequency will form the dominant component of that and subsequent filters. Conversely, higher frequencies may fall into only a few filters that have non-zero response. In these cases, a large SBS amplitude is not possible. Overall, the result of this is to give less weighting to the higher frequency components of any speech frame; furthermore, if the SBS threshold is set too high, these frequencies will be discarded completely.

The maximum height of the first filter (at the CF) is set at 18 dB. This is an arbitrary value, as it is not the absolute value of any filter height that is important - rather the relative height with respect to the other filters. This relative height is set by the incline and decline slopes, and is independent of the initial filter amplitude. To verify this, simulations were run with initial filter heights of 1, 18 and 36, and no changes to the SBS spectra were noted.

4.6 Combining Overlapping Frames

Frame overlap requires processing to recombine the processed frames into a continuous speech output. With 50 % overlap (64 data points) for both preceding and succeeding frames, data redundancy is 100%, and there are effectively twice as many data points as originally sampled. The simplest way to recombine the overlapping frames was to perform a weighted linear interpolation between the two overlaps. Interpolation weighting is determined by the position of the data point. Those data points near the end are more likely to be altered by windowing and FFT discontinuities, and so are given a lower weighting (zero at the end points). Similarly, the data points towards the centre of the frame are assumed more reliable, and given a maximum weighting of 100. There is a linear progression in weightings from the centre to the end points. This is shown diagrammatically below.

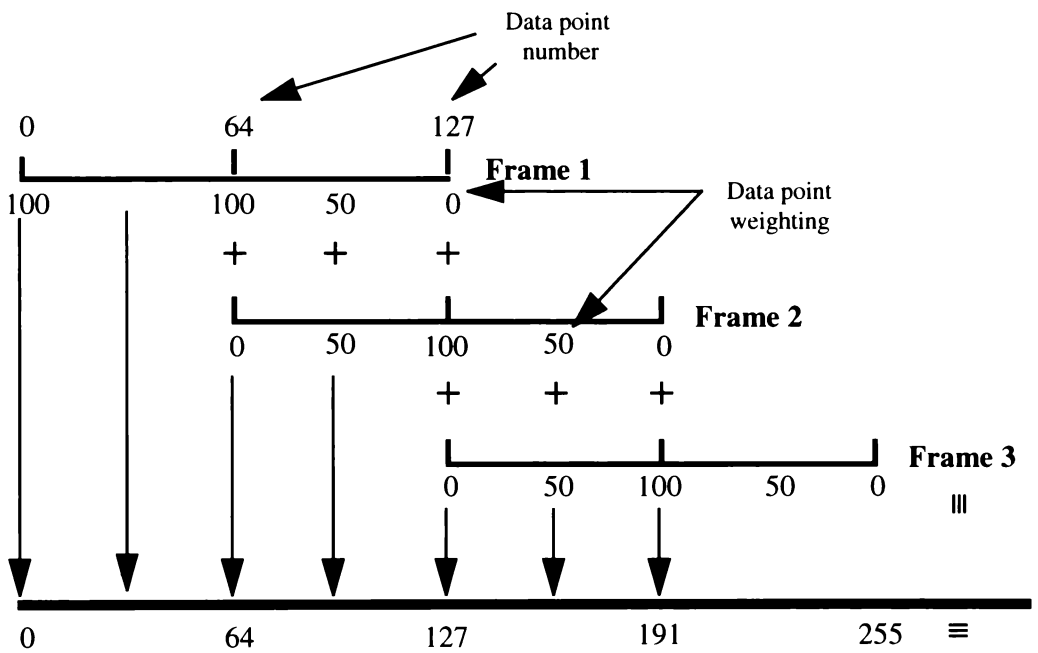


Figure 4.3 : Combining Overlapping Frames

4.7 Reduced Frequency Speech Synthesis

Ghitza employs an upper bound of ten frequency components for speech reconstruction. As SBS processing suggested an application in low bit rate coding, we investigated how many frequency components were actually required to create intelligible speech. Starting with ten SBS selected frequencies, the reproduced speech sounded reasonably natural, though did suffer from the tonal artefacts Ghitza reported.

Ghitza believes that tonal artefact arises as a consequence of attempting to reproduce speech exhibiting broad band spectra (for example unvoiced speech), with a greatly reduced discrete spectral set. This explanation appears intuitively correct, but experiments were designed to verify this. Ghitza produced a hybridized system that reconstructed speech using the original frame for unvoiced segments, and the SBS controlled synthesis for those that were voiced. He claimed this practically eliminated the tonal artefacts.

Implicit in this analysis is that some decision on voiced/unvoiced frames needs to be made. Ghitza uses the Gold-Rabiner pitch detector (1969), though the voicing decision was so important to intelligibility (especially in view of DRT results, discussed later), that a dedicated Voiced/Unvoiced/Silence (VUS) classifier was designed (Section 5.7). Using this VUS algorithm, the quality and intelligibility of the speech improved markedly when the original voiceless frames were employed in the speech reconstruction.

4.7.1 SBS Threshold Setting

The goal here was to emphasise intelligibility rather than speech quality, so experiments were performed varying the SBS threshold to determine what the limit would be in order for the speech to remain highly intelligible. Increasing the SBS threshold so the number of frequency components was steadily reduced, resulted in a lower limit of three frequencies (on average) for intelligible reconstructed speech. This limit is a quite definite one. Speech reconstructed using an average of two frequencies per frame scored very poorly on a limited set of DRT tests, and four frequencies scored only marginally higher than three. While a claim may be made for increased naturalness from the use of four rather than three frequencies, this was not tested.

An SBS threshold of ten yields this average number of three frequencies per frame for most forms of speech input and reduces the frequency domain information (compared to the input speech) by 95 %. While tonal artefact is obvious, especially in the unvoiced frames, and some word groups score significantly lower than average (discussed later), SBS processing still appears to be applicable to speech compression.

4.7.2 SBS Reconstruction of Vowels

Which particular frequencies are selected is of considerable interest. It is desirable that the SBS selected frequencies bear some relationship to the formants of the tested phoneme. This does appear to be the case, as evidenced by the vowels IY and AH (spoken by a North American adult male) shown in Figure 4.4. The power spectra of these vowels are shown both before and after SBS processing. Comparing the frequency bins corresponding to the reconstructed spectral peaks, shows close correlation with the expected formant frequencies for these vowels (Table 4.2). The only variation from this is the peak at about 1.2 kHz for the vowel AH, which cannot be attributed to formant energy. However, all other peaks for both vowels do show strong correlation with Table 4.2 (North American vowel sounds).

Vowel	AH	IY
Fundamental Frequency	129	136
Formant Frequency F1	570	270
Formant Frequency F2	840	2290
Formant Frequency F3	2410	3010

Table 4.2 : Typical Formant Frequencies for the Vowels AH and IY

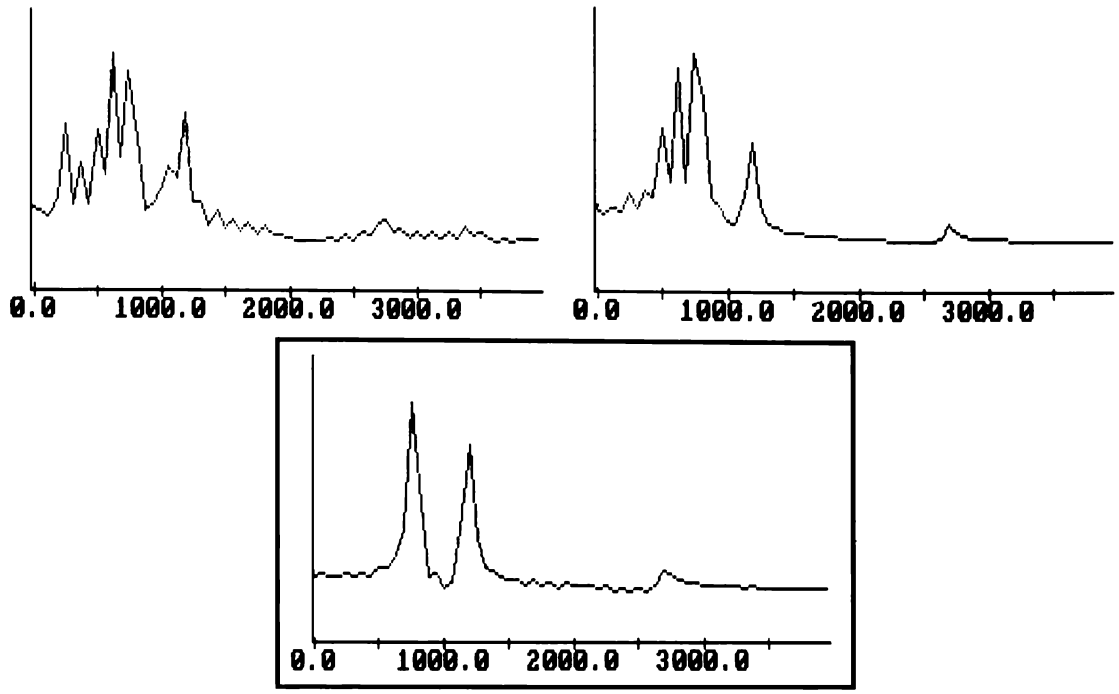


Figure 4.4 : Power Spectra for Vowel AH

(a) Before SBS processing, (b) After SBS Processing - Threshold 4, (c) After SBS Processing (Threshold 10)

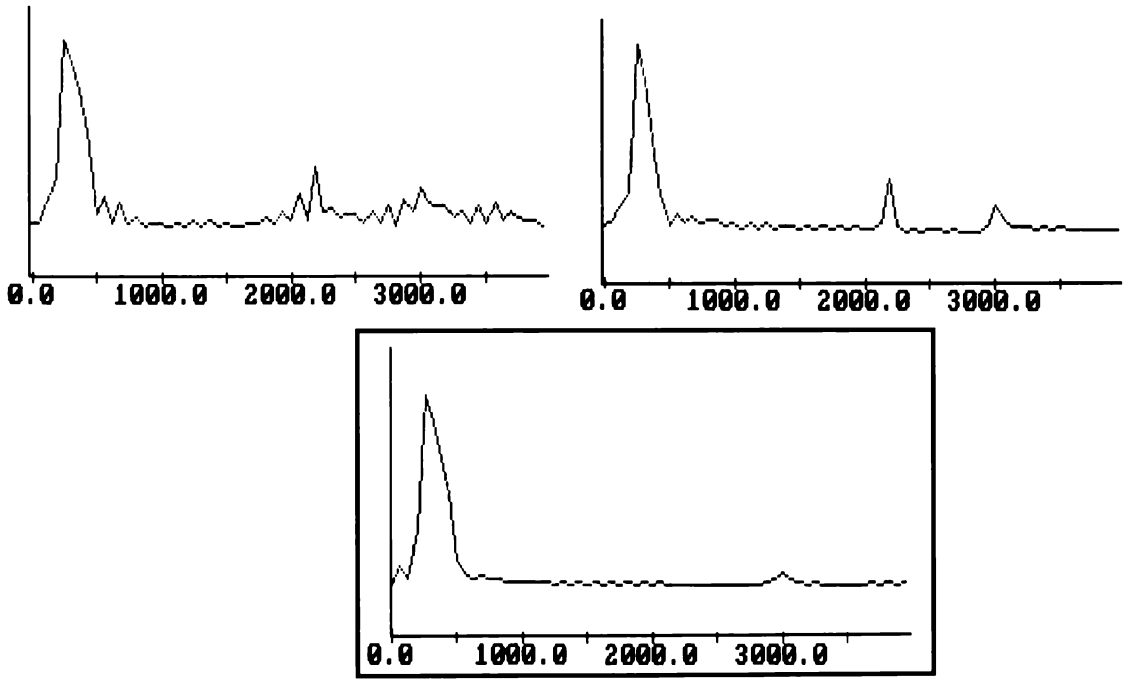


Figure 4.5 : Power Spectra for Vowel IY

(a) Before SBS Processing, (b) After SBS Processing - Threshold 4, (c) After SBS Processing (Threshold 10)

The figures above also give a graphical indication of the effects of setting a value of the SBS threshold. A threshold of four yields five distinct peaks for the vowel AH, a threshold of ten only produces three. For the vowel IY, there are four peaks with the threshold set at four (the broad peak contains two frequencies at 290 and 440 Hz), but only two when the threshold is raised to ten.

4.8 High Frequency Emphasis by Numerical Differentiation

High frequency emphasis is commonly achieved by differencing the data points, (a form of linear differentiation and therefore high pass filtering). Differencing is the subtraction of data point x_n from the succeeding data point, x_{n+1}.

As explained in Appendix C, numerical differentiation is an inherently unstable process, and whilst differencing techniques are commonly employed in speech processing, very little information exists as to whether higher order polynomial approximations succeed in improving the quality of reconstructed speech, that is, improving the quality of the high frequency emphasis. To examine the possibilities of employing a differentiation routine with known superior numerical accuracy, differentiation was performed using two central difference techniques, and compared to the results obtained from the linear differencing and undifferenced speech. SBS thresholds of four and ten were tested. By varying the SBS thresholds, the effect of introducing additional frequency components could be investigated. The effects should be similar, but experiments were performed to investigate this.

Analyses were performed on four samples of speech. The first two were the vowels AH and IY, chosen because of the clearly defined formant structures. The other two samples, were phrases, "We were away a year ago", and "Sally sells seashells by the seashore". Both phrases are rather demanding of SBS processing, especially the latter with the predominance of unvoiced speech. Using the equations developed in Appendix A, each speech sample was processed by:

(a) linear differencing

(b) first order central differentiation

(c) third order central differentiation

(d) no differentiation

and the quality and intelligibility of the reconstructed speech compared for the two SBS thresholds. Typical vowel frame spectra following linear differentiation are shown below:

		SBS Amplitude			
Bin Number	**Frequency**	**Linear Diff.**	**1st O. Central**	**3rd O. Central**	**No Diff.**
4+5	~300	18	18	18	19
7	438	19	19	21	41
11	688	9	10	-	-
16	1000	3	5	7	-
36	2250	9	8	8	8
49	3063	4	8	8	3

Table 4.3 : SBS Spectra for Vowel IY Using Ghitza's Bandpass Filters

		SBS Amplitude			
Bin Number	**Frequency**	**Linear Diff.**	**1st O. Central**	**3rd O. Central**	**No Diff.**
4	250	10	10	10	16
8	500	-	-	-	8
10	625	25	26	26	8
12+13	~800	10	10	9	10
19	1188	26	27	28	25
43+44	~2700	8	12	12	7
56	3500	4	-	-	5

Table 4.4 : SBS Spectra for Vowel AH Using Ghitza's Bandpass Filters

All other bins (of which there could be up to ten) have an SBS amplitude that never exceed a threshold value of four. Note that in some cases, when a frequency is represented in two adjacent bins, both will have some SBS amplitude, e.g. bin 12 for the vowel AH typically has SBS amplitude of five, as does bin 13. If these were combined, the resulting SBS amplitude would be of sufficient magnitude to include this frequency in the reconstruction. If the bins are treated separately, this would not generally be the case. Hence a modification to Ghitza's model is to combine such bins as in the above table. This will be investigated as a consequence of frequency bin quantisation in later chapters.

For both vowels, differentiation places greater emphasis (in terms of increased SBS amplitudes) on those frequencies around 600 Hz, while reducing the amplitude of lower frequency components. The first and third order central differentiation techniques yield very similar results, though tend to produce less frequencies with non zero SBS amplitudes than the linear technique. However, in the reconstruction, this effect could also be achieved by the setting of an appropriate SBS threshold.

Listening to speech reconstructed using these differencing techniques indicated only very slight differences between them, especially for speech reproduced with an SBS threshold of four. For the threshold of ten, the differences were more marked, with the third order central technique reproducing the phrase at a higher pitch than either the linear or the first order central. The SBS threshold of ten (three frequencies per frame) contained notably more tonal artefact for all forms of differencing than speech processed with a threshold of four (seven frequencies per frame). However, it was apparent for this implementation of the bandpass filters, frequency emphasis by differentiation did not increase the intelligibility of the reconstructed speech.

4.9 Filter Modifications

Before progressing with amplitude, phase and frequency quantisation, some variations on how the speech spectra were modified by the band pass filters were investigated. Ghitza's filter has some physiological justification (Section 3.3), where the log amplitude of the speech spectra is added with the log response of the filter. This places a very large emphasis

on frequencies close to the filter's CF, but in terms of maximising SBS amplitude, it could be that applying a smaller modification to the spectral amplitude would allow a large frequency amplitude to dominate more frames.

To test this, the amplitude of the speech was multiplied by the log amplitude of the filter response. Such a scheme effectively reduces the weighting of those frequencies closest to the filter's CF, compared to the other frequency components spanned by that filter. In this implementation, a large frequency component, especially one located at the lower frequency range, will have the ability to dominate more adjacent filters as the filters will not so highly amplify any of their spanned frequencies. In other words, the natural amplitude of the frequency bin becomes more important than its proximity to a filter CF.

The expected result is that the lower frequencies will have larger SBS amplitudes, and will contribute more to the speech reconstruction, as they are more likely to be above the SBS threshold than higher frequencies. For intelligibility purposes, this low frequency emphasis is not entirely unreasonable as the bandpass filtered input speech will normally only contain the first three formants, the first two of which contribute more to intelligibility than does the third.

With no pre-emphasis, the results were inferior to the original filter construction. The modification did result in larger SBS amplitudes as expected for the lower frequency components. The extent of this was such that intelligibility actually decreased, due to lower formants almost totally masking higher ones, and not in themselves being sufficient to consistently identify the word.

Two methods were explored in an attempt to rectify this - frequency emphasis and altering the filter slopes so that all possessed -18 dB per octave decline, i.e. the filters were symmetric about their CF. This broadens the frequency span of filters 54 to 100, resulting in a larger number of filters spanning the higher frequencies, and hence increasing the likelihood that they can achieve significant SBS amplitudes. This is not physiologically justifiable, but was hoped to help re-establish the higher frequency formants in the speech reconstruction.

An example of the spectra for the test vowels is provided in Table 4.5 and Table 4.6, for both -18 dB and -120 dB per octave declines for the higher frequency filters, each both with and without differencing.

		SBS Amplitude			
Bin Number	**Frequency**	**-120 dB No Diff.**	**-120 dB Diff.**	**-18 dB No Diff.**	**-18 dB Diff.**
4+5	~300	98	19	98	19
7	438	-	35	-	20
34	2125	-	9	-	-
36	2250	-	39	-	45
49	3063	-	10	-	11

Table 4.5 : SBS Spectrum for the Vowel IY with Modified Bandpass Filters

		SBS Amplitude			
Bin Number	**Frequency**	**-120 dB No Diff.**	**-120 dB Diff.**	**-18 dB No Diff.**	**-18 dB Diff.**
4	250	14	9	14	9
10	625	26	23	23	23
12+13	~800	57	15	60	13
19	1188	-	47	-	51

Table 4.6 : SBS Spectrum for the Vowel AH with Modified Bandpass Filters

All other frequency bins have a zero SBS amplitude.

These tables provide some indication of the high frequency information loss if no differencing is used. As expected, the SBS amplitude are higher, and unlike Table 4.3 and Table 4.4, these are the only frequencies with non-zero SBS amplitudes. The frequencies selected for speech reconstruction are therefore less likely to change between frames than the original filter implementation. Using the differentiation procedures described above, the

speech becomes far more intelligible, though some of the higher frequencies selected by Ghitza's method have been lost. Broadening the filter spans by having a constant -18dB per octave decline regardless of frequency, does not significantly alter the SBS amplitudes and generally does not alter the selection of frequencies for reconstruction. There was no significant audible difference.

An investigation into the effect of narrowing the filter spans was found to produce a "smeared", low-frequency dominated spectra, with inferior intelligibility performance to either the original filter shape or the variations discussed above. It is reasonable to conclude that the filter shape suggested by Ghitza is a good compromise solution – producing a spectra with a reasonably large amplitude as well as a broad frequency range (i.e. the spectra is not overly dominated by the lower frequencies).

4.10 The Effects of Noise on SBS Reproduced Speech

Ghitza claims that noise does not seriously affected his model, even when it is as high as 0 dB peak-vowel-to-average-noise ratio (Ghitza, 1987). In fact he claims that there may be some noise reduction due to the coherence property of the Fourier transform and the dominance effect of the SBS. The latter could be an excellent form of noise reduction, given that the SBS controlled speech synthesizer will only be selecting high-energy components. A broad-band form of noise could be expected to contribute some energy modification to each frequency bin, but should be unable to mask a frequency with a large SBS amplitude or sufficiently modify a low-energy bin so that it dominates a large number of successive filters. Restated, the expected result from adding white noise to a speech segment and SBS processing, is that, whilst the amplitudes of the SBS spectrum may change, it would not be expected that new frequencies with significant SBS amplitudes will be created. Frequencies with large SBS amplitudes will be expected to retain their dominance in any speech frame.

To support this, Ghitza provides results obtained from a female speaker, and compares the SBS spectrum of the original processed speech, to the SBS spectrum of the speech when the noise level is 3 dB peak-vowel-to-average-noise ratio. No new SBS components were generated, and whilst the amplitudes of existing SBS spectra were altered, each spectra

maintained its relative dominance. Ghitza presents an overly simplified view of his model's performance in the presence of noise.

An initial observation is that he has only presented results for what appears to be white noise added to an uttered vowel. The SBS process appears to work extremely well on vowels, so it is to be expected that noise immunity will be superior for this form of SBS processed speech. Also, the actual graphical results he presents show no additional SBS components, even of very low amplitude upon the addition of this noise. For the levels of white noise added in this project, additional SBS spectral points almost invariably result. The setting of an appropriate threshold level eliminates many of these additional SBS components, but only very high threshold levels successfully remove them all.

A second point of interest, is that SBS processing will reject white noise with relative ease given the broad-band nature of the noise, and that the SBS synthesizer chooses only high-energy components for the speech reconstruction. However, other forms of noise may not be so easily managed - for example, impulse noise. The following sections qualitatively examine the effects of adding white and impulse noise to both vowels and test phrases.

4.10.1 White Noise Experiments

Analyses were performed on the vowels IY and AH, as well as the two phrases "We were away a year ago" and "Sally sells seashells by the seashore". White noise was added at several signal to noise ratios. In all cases, an objectionable tone was audible in the reconstructed speech, although the utterances were still easily intelligible. The amplitude of this added tone was related to the signal to noise ratio. Resynthesis using higher SBS thresholds lowered the tonal artefact, but did not eliminate it (Carnegie et al. 1990).

Figure 4.6 shows the SBS spectrum, across time for the utterance "We were away a year ago". With the SBS threshold set to ten, an average of three frequencies are selected for use in resynthesis. The SBS representation is, for the most part, two or three formants of the voiced portion of the utterance. Silence, at beginning and end, shows more scattered activity with up to five selected frequencies.

Frication is tested with the phrase "Sally sells seashells by the seashore" in Figure 4.7. Again an average of three frequencies are used in the resynthesis. However, unlike the previous test sentence, this phrase is severely distorted by the presence of tonal artefact, predicted by Ghitza for speech that is embedded in noise.

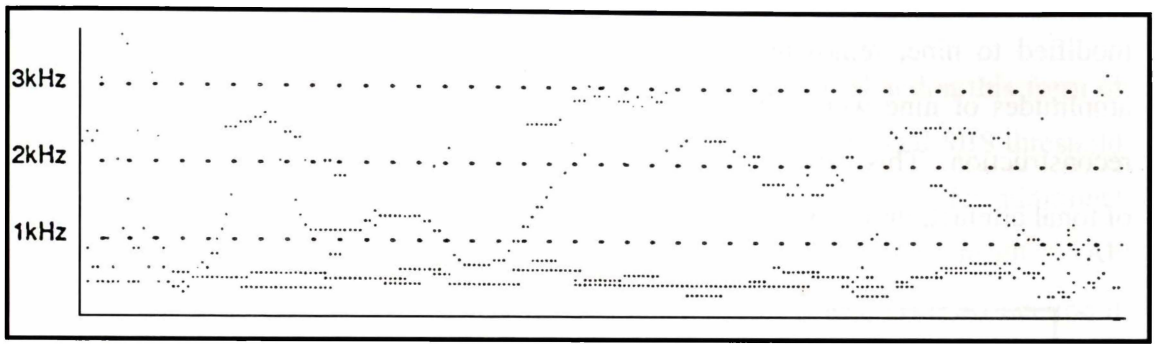


Figure 4.6 : SBS Spectra Over Time for "We were away a year ago"

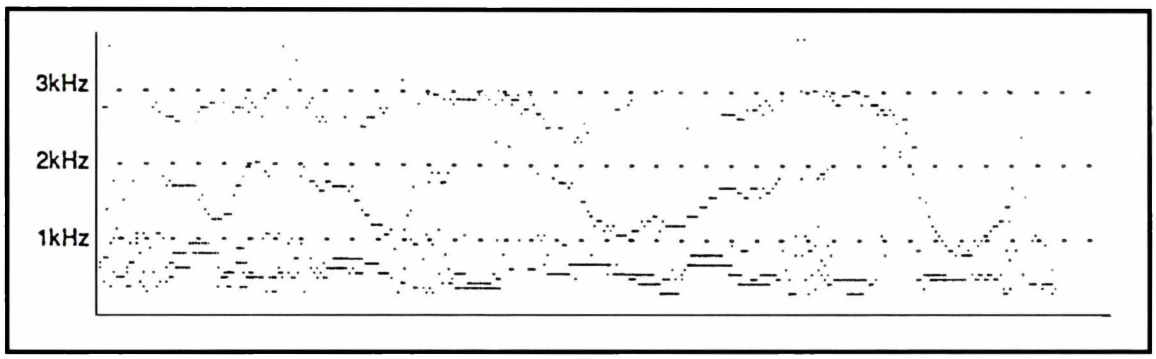


Figure 4.7 : SBS Spectra Over Time for "Sally sells seashells by the seashore"

This introduced artefact was further investigated by the deliberate introduction of white noise to both test phrases. The frication phrase is further degraded, and even the first test phrase produces an objectionable tone, the amplitude of which is related to the signal to noise ratio. The SBS spectra across time for the first test phrase in the presence of added white noise is shown in Figure 4.8

In most test frames, addition of noise created additional, small amplitude SBS components. These were more common for the phrases than for the vowels, especially during an unvoiced, or a low energy frame. The large SBS peaks associated with the vowel formants limited these additional components, and there were some frames for which the formants so

totally dominated that no additional components were formed. This was the exception rather than the rule. However, even for the unvoiced frames of the uttered phrases, the amplitudes introduced by noise into the SBS spectra never exceeded the threshold used for speech reconstruction. The change to reconstructed speech comes from modification of "above-threshold" SBS amplitudes. In some cases, an original amplitude of eleven was modified to nine, removing it from those frequencies sent to the synthesizer, and some amplitudes of nine were increased to ten, allowing that frequency to now be selected for reconstruction. This variation of existing SBS amplitudes is responsible for the introduction of tonal artefact, though intelligibility remains very high.

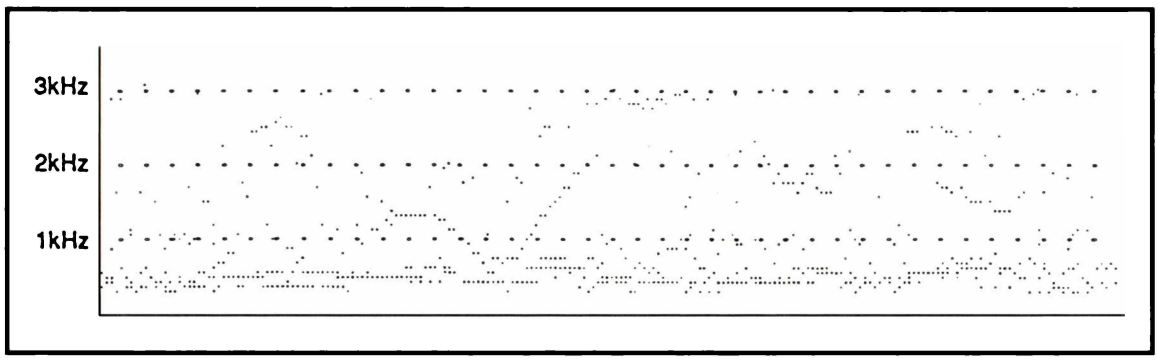


Figure 4.8 : SBS Spectra Over Time for "We were away a year ago" with added white noise

4.10.2 Experiments with Impulse Noise

Impulse noise was also added to the test utterances to establish the model's immunity to this form of interference. The addition of impulse noise was simulated by adding spikes of random amplitude at random intervals to the speech waveform. The probability of spike occurrence in a given frame was varied from 10% to 50%, and the amplitude of the spike given a random value between zero and the maximum amplitude the waveform.

On reconstruction, these spikes were smeared into short tonal bursts, which were clearly audible. Whilst this form of added noise did not affect the intelligibility of the utterances, the spikes did not seem to be greatly attenuated by the SBS model.

Running the above noise experiments through a model using 100 bandpass filters symmetric about the CF, again showed that greater intelligibility resulted from the introduction of

linear differencing (the higher polynomial differencing techniques were not explored). For the case of white noise, the tonal artefact increased upon differencing, and with impulse noise, the spikes were broadened out and audibly changed from a click, to sounding more like a whistle. However, even though the effect of noise became more audible, the increase in intelligibility made differencing for this form of the filter shape essential.

It was at this stage of the investigation that the decision was made to abandon this form of filter modification and return to Ghitza's original model. Apart from reduced SBS threshold sensitivity, this form of the filters created extra processing steps, did not offer additional noise immunity and provided no noticeable improvement in intelligibility results. All experiments from this point onward were conducted using filters with a -120 dB decline if their CF's were above 1 kHz, and no pre-emphasis.

4.10.3 SBS Noise Conclusion

Ghitza was justified in his claim of noise reduction when white noise is added to a spoken vowel. It does appear that he simplifies the matter considerably by not discussing the likelihood of additional, small amplitude SBS components that result after the noise has been added. With the tests performed here, it was only on a few vowel frames that additional SBS components did not appear. With spoken phrases, the low amplitude SBS spectrum changed considerably. However, speech reconstruction will only be affected if the added noise affects frequencies with an SBS amplitude close to the set threshold. For example, for a threshold of ten, and the addition of white noise resulted in a frequency with a noise free SBS amplitude of nine to increase to SBS amplitude ten (or higher), then the synthesised speech would be affected (as that frequency would now be selected). This is a reasonably rare occurrence, and speech intelligibility in the presence of white noise is high.

The model does not appear to provide significant immunity to impulse noise, most of the added spikes appear in the reconstructed speech with the addition of significant tonal artefact that further degrades speech quality. There is too much contextual information to infer any absolute intelligibility results from these tests, though it is evident that the model is considerably more resistant to white noise than impulse noise.

4.11 Summary of SBS Processing

This chapter has thoroughly investigated many of the components of SBS processing, and certain decisions concerning the form of the SBS implementation can now be made, and applied to subsequent chapters.

The compromise of FFT speed, and ensuring sufficient periodicity of the speech, results in the selection of a 128 point, 16 ms frame size being adopted (Section 4.2). The best windowing option is the Hamming, being a compromise solution between the considerations of coherent gain and spectral leakage (Section 4.3 and Appendix A). Both the frame size and windowing selection are common in speech processing literature.

A certain amount of data redundancy that results from overlapping frames can reduce the discontinuities between FFT processed windows, and so will be retained, though the amount of frame overlap will be user selectable, and will be investigated in the later recognition experiments (Section 6.4.3). A weighted interpolation scheme (Section 4.6) successfully combines these frames, and smoothes out much of the discontinuity. The setting of the SBS threshold is obviously crucial to the final quality of the reconstructed speech, and will be one of the major aims of the later recognition experiments.

After investigation into the effects of altering the shape of the overlapping bandpass filters, both with and without emphasis by numerical differentiation (Sections 4.8 and 4.9), it was decided to revert to Ghitza's original filter shapes, with an incline slope of 18 dB per octave, a decline of –120 dB per octave. One difference experiment will be performed in the HTK tests, but the informal results to date do not indicate that an improvement will result.

There is now some evidence to support Ghitza's claims of good SBS performance in the presence of white noise. The addition of impulse noise also did not significantly degrade the performance of the SBS algorithm. The concentration of the later HTK experiments will be to quantify the model in terms of its performance in the presence of noise, though in these instances, the introduced noise will be white noise and cocktail party noise (added speakers at varying amplitudes).

Chapter 5

5 The Compression Experiments

5.1 Overview

SBS processing, using 128 point FFTs, returns 64 frequency bins. With 125 frames analysed per second, assuming each bin requires nine bits to encode phase (ranging from $-\pi$ to π) and seven bits to encode amplitude, the data rate might seem to be increasing rather than decreasing (the assumptions of nine bits for phase and seven bits for amplitude are empirically derived - there is no improvement in intelligibility or quality obtained if these values are increased). A measure of the bit rate for SBS processing without quantisation is given by:

$$\begin{aligned} &\#\text{ frames per second} \times \#\text{ of frequency components} \\ &\times (\#\text{ bits for the frequency bin} + \#\text{ bits phase} + \#\text{ bits amplitude}) \\ &= 125 \times 64 \times (0 + 9 + 7) = 128{,}000 \text{ bits per second.} \end{aligned} \qquad \textbf{Equation 5.1}$$

The promise of SBS processing lies in eliminating some of these frequency bins. Ghitza reduced the data rate to 27.5 kbit/s by using ten frequency components per frame. Experiments discussed here indicate intelligible speech can be synthesised using an average of only three frequencies per frame, in which case the bit rate is now 6.0 k bits per second (plus the overheads to encode which frequencies are selected).

Subjective listening tests with this model indicate that phase has little effect on intelligibility. Taking phase out of the calculation results in a bit rate of 2.6 k bits per second (without any amplitude or frequency bin quantisation), and with 100% data redundancy resulting from the frame overlap. This was the motivation behind performing

an extensive set of compression/quantisation experiments, varying the number of frequency components, and the frequency/phase/amplitude quantisation levels. It is hoped that without further processing in the form of Vector Quantisation or LPC methods, that the bit rate can eventually be reduced to approximately 1 kbit/second, and still produce intelligible speech.

5.2 Subjective Quantisation Tests

When analysing the effects of a large number of parameter changes, it is infeasible to carry out formal intelligibility tests for every variation given that such testing is a very time consuming process. Consequently, an informal pilot study was conducted in order to eliminate some of the possible variations, and hence reduce the number of formal tests.

5.2.1 The Importance of Amplitude in Test Phrases

To investigate the role of amplitude representation, the phrase "We were away a year ago" was examined. This investigation involved the amplitude quantisation being lowered from seven bits down to one bit, while the phase quantisation was held constant at nine bits. At this stage there was only one listener providing subjective feedback on the intelligibility of the reconstructed speech. Very little degradation was found going from seven bits to six, and in fact only a slight loss of intelligibility resulted from progressively decreasing the amplitude representation down to three bits. Past this, however, at quantisation levels of two and one bits, the phrase was hardly recognisable. No attempt was made to limit the level of contextual information in this evaluation, and it cannot be expected that DRT scores should be high for isolated words at this three bit amplitude quantisation level. It is apparent, however, that this is the lowest level capable of producing intelligible output.

5.2.2 The Importance of Phase in Test Phrases

A similar experiment was performed to test the effect of phase quantisation. The amplitude quantisation level was held constant at seven bits, while the phase was tested at 9, 6, 3, 2 and 1 bits. The degradation of intelligibility was rather hard to detect, at least with this subjective analysis. To further investigate the effects of phase, a range of experiments were conducted with the amplitude quantisation set at three bits, and the phase again set at 9, 6, 3, 2 and 1 bits. The results were as before, no noticeable degradation in speech intelligibility

was exhibited under these test conditions when phase representation was successively reduced to the one bit quantisation level.

The next step was to investigate whether phase information was actually needed at all for intelligibility. Phase was eliminated and experiments performed with the amplitude quantisation set to seven bits, then to three, and the results compared to the runs where the phase was fully represented given these values of the amplitude levels. Again there was no observed effect on speech intelligibility, though speech quality was degraded. Recognising the limited validity of these experiments, given only one listener presenting subjective results on one test phrase (with strong contextual inferences), it does appear that phase has limited impact on intelligibility.

5.2.3 The Importance of Amplitude and Phase in Isolated Words

The next set of experiments used a selection of words arbitrarily selected from the MRT list (House et al., 1965), presented in Table 5.1

cut	king	peace
tease	shook	oil
hit	pay	sad
sang	peace_sentence	

Table 5.1 : Test Words From the MRT List

The word "peace" was employed in a sentence, as well as in isolation, hence the inclusion of peace_sentence in the above table. These words were chosen from the MRT list rather than the DRT to provide a greater coverage of test words. Several different levels of amplitude and phase quantisation were trialled, and are shown in Table 5.2. Note that for the last run, the SBS threshold was lowered to seven to test whether an increased number of frequency components significantly altered the results of the quantisation experiments.

	Amplitude Representation	**Phase Representation**	**SBS Threshold**
Run 1	7	9	10
Run 2	5	3	10
Run 3	5	0	10
Run 4	3	0	10
Run 5	5	3	7

Table 5.2 : Amplitude and Phase Quantisation Tests for MRT Words

The first run is a control, any distortions introduced would be due to the recording and playback equipment and the SBS processing, not due to amplitude or phase quantisation. Two runs were then conducted with the amplitude initially limited to 2^5 (= 32) distinct levels, and the phase limited to 2^3 (= 8) levels, and then repeated with no phase information at all. In almost every case, no subjective difference in intelligibility was noted between runs 2 and 3. This supported the earlier proposition that phase does not contribute significantly to speech intelligibility. In general (the exception is discussed later), this five bit amplitude representation produces highly intelligible speech.

With no phase information, the amplitude was then lowered to a three bit representation. The results were markedly inferior, severe distortion occurring for a large number of the test words. For example, the "*ng*" disappeared from the word "*king*", the "*k*" could not be heard in the word "*shook*", and the "*t*" seemed to be truncated in the word "*cut*".

The final variation was to lower the SBS threshold to seven, and repeat run 2. Lowering the threshold to this value, introduced on average an additional two frequency components per frame. For some words, such as "*king*" and "*peace*", this improved the clarity of the speech, but for others it was hard to detect any difference. As mentioned above, there was an exception to word intelligibility using five bit amplitude quantisation. This was the word "*peace*". The reproduction of this word was very poor, even when the SBS threshold was lowered to seven. An extensive investigation was undertaken to find out why this word was so different, and what would be required to improve its intelligibility.

5.2.4 The Effect of Phase on the Intelligibility of the Word "Peace"

The following trials (Table 5.3) were performed on the word "*peace*" with amplitude and phase quantisation set both above and below those values used in the previous trials.

	Amplitude Representation	Phase Representation	SBS Threshold
Run 1	9	0	10
Run 2	7	0	10
Run 3	7	3	10
Run 4	7	7	10
Run 5	4	3	10
Run 6	4	2	10
Run 7	3	3	10
Run 8	2	3	10
Run 9	2	2	10

Table 5.3 : Amplitude and Phase Quantisation Tests for the Word "Peace"

There was no audible difference between run 1 and run 2, and even comparing these results to the five bit amplitude/no phase of the past set of experiments, no apparent difference in intelligibility resulted. All results to date have supported this, so we could reasonably expect to set the upper limit for amplitude quantisation to five, i.e. there is an insignificant improvement in intelligibility (with this experimental technique) in increasing the amplitude representation above five bits. There was also no difference (audibly) between runs 3 and 4, nor when these were compared to the five bit amplitude/three bit phase of the previous trials. From this, the value of three was selected as our upper limit for phase quantisation. Lowering the phase below two produced some audible distortion, though this depends upon the amplitude quantisation (discussed later). Whether lowering the phase representation would actually affect intelligibility or not needs to be evaluated by quantitative experiments.

However, unlike the previous test cases, it does appear that the word "*peace*" is better reproduced with three (or two) bits of phase information, than it is when phase is eliminated. "*Peace*" reproduced in run 5 appeared to be superior to run 2 (seven bit amp, no phase), and

about equal to runs 2 and 3 of the previous trial. This importance of phase is unusual, and was not exhibited in any of the other test words.

"*Peace*" contains about the same average number of frequency components per frame as the other words (≈ 2.7), however, it does seem to have been recorded with a markedly lower amplitude (about 23%) than the other words. It seems likely that the elimination of the phase information reduced the energy of the frame to such an extent that significant portions of it became inaudible. (Note that if phase is eliminated, the imaginary component of the Fourier transform becomes zero [Im = Amplitude $\times$ sin (Phase)] and energy is therefore lost from the system). Applying an arbitrary gain factor to the word seemed to reduce the importance of the phase information, but the results were not conclusive. The amplification of low energy frames to improve reproduction is discussed in more detail later.

5.2.5 Subjective Upper and Lower Quantisation Levels

To summarise this section, results indicate that five bit amplitude quantisation with no phase generally produces intelligible results, even if consonants are recorded at low amplitude or have a broad frequency spectra. Sharp, strongly vocalised words can be recognised (under these test conditions) with only three bits amplitude information. If the word is recorded at low amplitude and no gain is applied, then some phase information (generally not more than three bits), does increase intelligibility. A strong indication though, is that reducing amplitude quantisation below three bits (assuming no phase), renders practically all words unrecognisable.

5.3 Quantitative Intelligibility on Isolated Words

The next step in this analysis was to quantify the effects of reducing phase and amplitude resolution. In order to do this, some standard form of intelligibility test was needed. Considering the various forms of word intelligibility tests discussed in Chapter 3, specifically the FRT, MRT, FAAF, and the DRT, it was decided to employ the latter technique. The DRT was chosen as it is the easiest of the options to implement, and more effectively removes the problems of contextual interference and listener response time than do the other methods.

5.3.1 Implementation of the DRT

The six test features were alternately presented over a six word cycle - no filler was employed. There were four test groups, each comprising between nine and twelve untrained listeners, selected from third year and graduate electronics research students. The response forms required each listener to indicate their choice of word by circling the correct response from the two options listed. Each group were used for two different trials, providing eight distinct tests (Table 5.5). This could introduce some listener preference, but with the 96 word options presented only once previously this was unlikely. The likelihood of introducing any such bias was further reduced by presenting the least intelligible test first. The words were presented at approximately 1.4 second intervals.

5.3.2 Scoring the DRT

The results are corrected for guessing by subtracting the number of errors from the number of correct answers, and the results provide percentage scores for each of the features as well as an overall score obtained by averaging the individual feature scores. Such a correction to remove the effect of guessing effectively maximizes the subject's *a priori* uncertainty, and thus minimizes the contribution of the contextual features (Voiers, 1983). Numerically, this adjustment takes the form:

$$S = \frac{R - \left(\frac{W}{n-1}\right)}{T} \times 100 \qquad \textbf{Equation 5.1}$$

where S is the "True" percentage-correct responses, R is the observed number of correct responses, W is the observed number of incorrect responses, n is the number of choices that the listener can select between, and T is the total number of items involved. For the DRT, as there are only two response options, this simplifies to

$$S = \frac{R - W}{T} \times 100 \qquad \textbf{Equation 5.2}$$

As mentioned previously, this formula is only valid if the incorrect members of a set of permissible responses are equally attractive, and also if the listener's response depends on a single discriminative decision. Note that this second condition is not satisfied with Harvard PB word lists, in which a correct response depends on several possibly interdependent decisions - one for each of several phonemes. However, where stimulus uncertainty is confined to a single phoneme or phonemic feature (i.e. as in the DRT), this formula works very well. Equation 5.1 returns a figure of merit that is free of the effects of context, and therefore permits meaningful comparisons between intelligibility tests that use different sized response sets.

5.4 Testing Amplitude and Phase Quantisation Effects Using the DRT

The previous subjective tests on isolated words provided some indication of what form the group DRT experiments should take. These tests should more accurately indicate the phase and amplitude quantisation effects on words in isolation using untrained listeners, and ultimately yield a value for amplitude and phase quantisation to be used in the final low-bit coding scheme (where this quantisation would be combined with frequency quantisation, and elimination of most of the frame overlap). Appendix E lists the pairs of words presented to the listener, and were chosen to test the response of the simulation to words indicative of voicing, nasality, sustention, sibilation, graveness or compactness (explained earlier). Four variations of this DRT experiment were performed as presented in Table 5.4.

Run #	Amplitude Representation	Phase Representation
1	full	full
2	full	none
3	5 bit	none
4	3 bit	none

Table 5.4 : Quantitative DRT Intelligibility Experiments

Run 1 was the control experiment. The recording and playback facilities were not ideal, the playback especially was in a noisy open lab environment, played through a portable cassette player which itself introduces tape hiss and other distortions to the recorded speech segments. Additionally the speech was bandpass filtered (telephone bandwidth), which will also introduce speech distortion. It was not too surprising then, that run 1 did not achieve 100% intelligibility scores over all six word groupings. This noise corruption is discussed later.

Run 2 was designed to qualitatively measure the effect phase information has on intelligible speech synthesis. With the exception of the word "*peace*", all previous testing indicated that phase information was not necessary. The question posed was, are words such as "*peace*" reproduced poorly because of low amplitude spectra as predicted in Section 5.2.4, or does phase information actually contribute measurably to word intelligibility? Runs 3 and 4 were designed to quantify amplitude quantisation effects. The claim was tentatively made from the earlier experiments that amplitude coding at five bits does not severely distort intelligibility. These runs test this claim.

Each run was conducted with two independent sets of between nine and twelve untrained listeners, selected from third year and graduate electronics students. Each group was subjected to two tests (Table 5.5); the test expected to be least intelligible was presented first. The words were digitally recorded by a male speaker (New Zealand accent) using a Scott Instruments data acquisition board, and played to the subjects on a Sanyo portable cassette player. Subjects were seated in a laboratory at varying distances with some ambient background noise. This is far from ideal for testing purposes, but is more realistic in terms of reproducing an "average" listening environment and should still provide good comparative results for the different runs.

Group	Number	DRT Tests
1	12	3 bit amp - no phase / full amp - no phase
2	9	5 bit amp - no phase / full amp - no phase
3	9	3 bit amp - no phase / full amp - full phase
4	10	full amp - no phase / full amp - full phase

Table 5.5 : DRT Experiments per Group

The results (corrected for guessing) are presented in Table 5.6, and are plotted in Figure 5.1. DRT scores evaluating a channel vocoder and a linear predictor coding system, operating at 2.4 kbit/s/s have been scored above the 60% level (Voiers, 1983). The SBS in its present form performs well below this, and obviously needs some improvement before it can be implemented as a low bit speech coding device.

word type	**Intelligibility Result**			
	run 1	**run2**	**run3**	**run4**
voicing	77	34	30	17
nasality	99	68	54	35
sustention	75	56	45	33
sibilation	78	65	48	47
graveness	57	31	29	8
compactness	93	79	69	34

Table 5.6 : DRT Intelligibility Results Corrected For Guessing

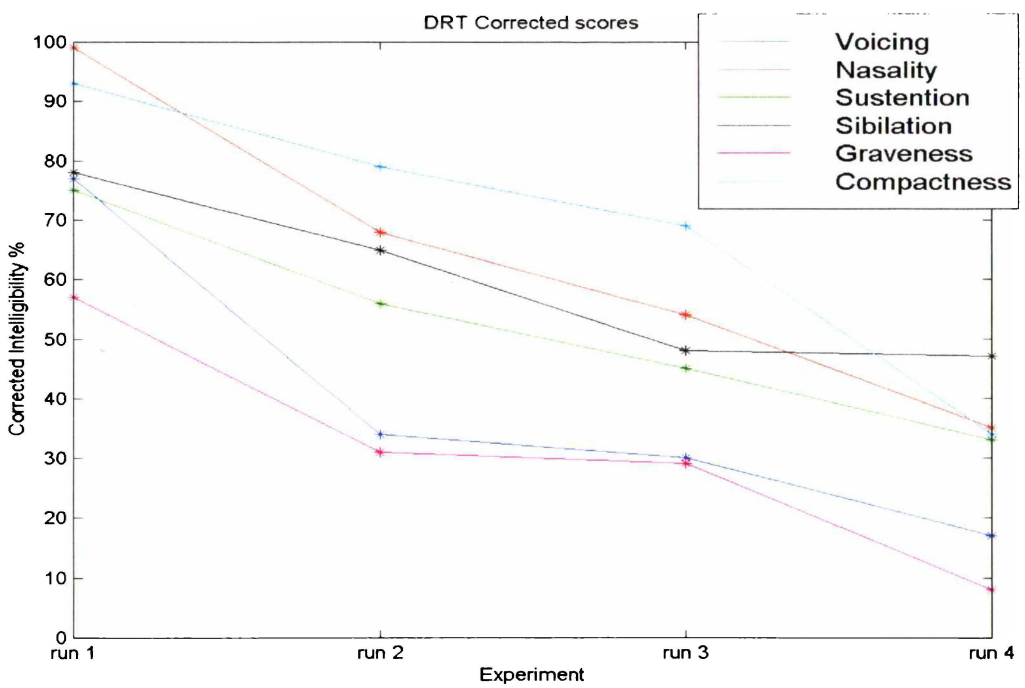


Figure 5.1 : DRT Intelligibility Results Corrected for Guessing

5.4.1 Errors in the DRT

Three issues must be decided before attempting to draw any quantitative conclusions from these results. The first is what is the margin of error for each tabular point? Voiers (1983) evaluated the test-retest reliability of the DRT with the same and different testing crews, and found the value to be very high, both for the total score and for the major diagnostic scores. With listening crews of eight to ten members, he found the standard error of the DRT total score to be typically one percentage point, and only slightly higher for the individual diagnostic scores. With such a small expected margin of error in the test, error bars have not been included in the result plots.

5.4.2 The Effects of Noise on the DRT

The second consideration is the sub-100% results from run 1, and the effect this has on subsequent experiments. Voiers has endeavoured to estimate the effect of both Gaussian and "babble" (i.e. cocktail party noise - created by combining many voices at once) noise to the six diagnostic scores. Gaussian noise tends to act as a low pass filter, and Voiers has also evaluated the degradation of test words for a low pass filter of varying cutoff frequencies. Whilst the loss of intelligibility following low pass filtering is similar to that produced by Gaussian noise, there are some differences, so these need to be taken into account when evaluating the overall signal degradation.

Gaussian noise severely effects graveness and sustention, and also heavily degrades the results for sibilation. Of the six groupings, voicing and nasality should be most resistant to this form of noise. A similar trend is evident for the cocktail-party noise - graveness and sustention are most vulnerable - voicing, nasality and sibilation are quite resistant. Surprisingly, compactness which depends on, among other things, the higher second-formant frequencies is most immune to cocktail-party noise masking, though it was appreciably effected by the Gaussian noise. The lowpass filter trends are similar to those for Gaussian noise, except that sustention is quite resistant.

In the testing environment, noise originates from many sources, including background machine hum, voices from outside the window, tape hiss, etc. This noise is not purely Gaussian, nor strictly multiple-speaker, but a combination of the two, with particular

frequencies most at risk, i.e. those corresponding to the pitch of the computer and monitor hums. It seems reasonable to model the noise feature as a combination of the two. The degraded run1 DRT test then, can provide some indication of the effects that this added noise has had on the speech that is not part of the SBS process itself, rather indicative of the testing conditions.

Qualitatively, combining Voiers' Gaussian and babble (cocktail party) noise effects, the expectation is for voicing and nasality to be the least degraded, sustention and graveness the most. Compactness and sibilation will be degraded only if the noise is predominantly Gaussian. Experimental results show severe degradation of graveness and sustention as predicted. The difference between the two degradation levels is slightly larger than expected, but does follow the lowpass filter noise trend. Sibilation is less degraded than graveness or sustention, and loss of compactness intelligibility is lower still. Voiers predicts compactness to be less degraded than sibilation over a wide range of speech-to-Gaussian noise ratios (and lowpass filter cutoffs), and to be equally effected by any babble noise. This trend is observed in the experimental results, and again, as expected, nasality is very well reproduced. Voicing however, defies all such predictions by being substantially degraded. Speaker bias may have affected the absolute value of any of these diagnostic scores, and as Voiers suggests, future DRT experiments should include more than one speaker to correct for this. External noise must also be controlled, or at least accurately measured before any absolute intelligibility score becomes meaningful.

5.4.3 Failures of the SBS Model

The third and final issue is the degradation of the speech due to the SBS processing, rather than the effect of the noise in the testing environment. Figure 5.2 shows the degradation of each of the six diagnostic scores, from the no-processing test through the final three bit amplitude trial. This plot shows how SBS processing degrades each feature, and gives an indication of model deficiencies that need to be addressed. Once again, only a general trend may be inferred, the numerical intelligibility value will be affected by noise.

The degradation in the six diagnostic scores progressing from run 1 to run 2 can be attributed to distortions introduced by SBS processing, rather than from extraneous noise

sources (accounted for in the initial degradation results). Voicing undergoes a major loss in intelligibility, as do (to a lesser extent) nasality and graveness. The claim from previous experiments was that there should be very little degradation (if any) in discarding phase information. This is clearly not indicated by Figure 5.2, though one attribute that will affect this run score is the loss of frame energy resulting from the elimination of phase. Another earlier claim is that there would not be a large intelligibility loss in progressing from full amplitude representation to five bit quantisation, indicated by run 2-3 in Figure 5.2. In fact, all scores except sibilation show a smaller loss in intelligibility here than in any of the other SBS processed runs, so to some extent this claim has been supported. A large degradation over all diagnostic scores progressing from run 3 to run 4 is predicted by the earlier three bit amplitude quantisation experiments.

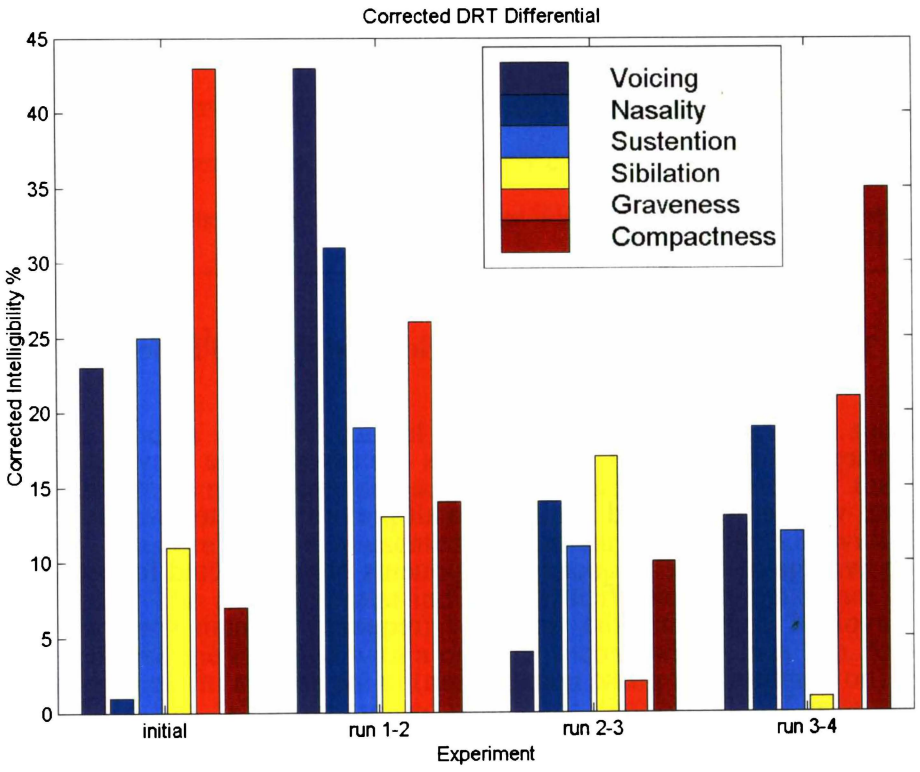


Figure 5.2 : Degradation of DRT Intelligibility Scores

5.5 The Intelligibility of Stops and Fricatives Reproduced by SBS Processing

The DRT voicing diagnostic scores are unaccountably low, and so to further investigate the effects of SBS processing on such words, a selection of voiced and unvoiced fricatives, stops and affricatives presented in O'Shaugnessy (1987) were tested (Table 5.7). These words were subjected to detailed signal analysis, and the results compared with the audio effects, in an effort to identify this particular failure of SBS reconstruction.

fluff	valve	thin
then	sass	zoos
shoe	measure	how
pop	bib	tot
did	kick	gig
church	judge	

Table 5.7 : Fricatives and Stops Selected for SBS Processing

The analysis software was DaDISP V1. For reasons discussed in the previous chapter, the SBS threshold level was set at ten, and similarly to all previously tested words and phrases, this resulted in an average of three frequency components being selected for reconstruction. The initial test involved comparing the time and frequency domain spectra of the SBS processed speech (no amplitude or phase quantisation) to the spectra of the original recorded speech. It was hoped that such a process could be used to predict which forms of speech would be subject to severe degradation upon SBS processing. The expectation was that any unvoiced speech (exhibiting a broad band frequency spectra) would be most prone to this degradation, and close attention was paid to any phoneme possessing such characteristics.

5.5.1 Analysis of Time and Frequency Domain Spectra

The first step involved very careful analysis of the time domain spectra of the original recording of the word, and identifying the phoneme transition points. For example, one recording of the word "*fluff*" involved 4952 data points. The two areas of particular interest are the initial and final "*f*" sounds. Analysis of the spectra suggested that the initial "*f*" spanned the data points from approximately 1 through 2048. The "*lu*" spanned points 2048 to 3328, and the final "*ff*" from 3328 to about data point 4440. Thereafter the remaining data points were silence (the initial silent points were deleted).

Once these points of interest in the time domain were identified, frequency power spectra were obtained at various intervals. Again taking the test word "*fluff*" as the example, spectra were obtained starting from points 256, 640, 1024, 1536, 3672, 3928, 4181 and 4440. The frequency analysis was performed using a 128 point FFT algorithm intrinsic to DaDISP. This analysis was repeated for both samples of every test word, resulting in a total of 136 frequency spectra to analyse and compare to the SBS processed versions (see step 3).

5.5.2 Audio Investigation

Step 2 in this process involved careful listening to the SBS processed speech, subjectively noting which phonemes were degraded, and qualitatively noting the extent of this degradation. These results are presented in summary in Table 5.8, with the phoneme of interest in bold. The coding format indicates that if two phonemes are of interest in the word, then the first one is coded with a one, the second with a two. So, for example, in the word "***valve***", the first "*v*" is represented with a one, the second "*ve*" is represented by a two, and in these tests, the first "*v*" is not reproduced at all, whilst the second "*ve*" is only faintly reproduced. When there is only one phoneme of interest, for example in the word "***th****in*", then the reproduction of that phoneme is coded with an X, and for "*thin*", the phoneme was only faintly reproduced. The results were very similar for both speakers, and the results following are an "average" of the two.

5.5.3 The SBS Spectra

Step 3 of this analysis was to examine the SBS spectra of each test word and compare it to the generated frequency power spectra obtained in step 1. By doing so it was hoped that some estimation of a phoneme's SBS intelligibility could be obtained from the magnitude and the number of the selected SBS components. The justification for this arises from the premise that unvoiced speech will have a broadband frequency spectrum, and therefore it is unlikely that any single frequency will dominate a large number of adjacent filters, i.e. the SBS amplitudes for such a frame will be comparatively small - generally lower than the threshold. The expectation is for unvoiced speech to have a small number of SBS selected frequencies, and those frequencies which are selected should have SBS amplitudes only just in excess of the threshold.

word	**Reproduction**			
	not at all	**faintly**	**good**	**very good**
fluff	1 2			
valve	2	1		
thin		X		
then			X	
sass		2		1
zoos	2		1	
shoe				X
measure				X
how				X
pop				1 2
bib			2	1
tot		1 2		
did		2	1	
kick				1 2
gig				1 2
church				1 2
judge				X

Table 5.8 : Qualitative Intelligibility of SBS Reconstructed Fricatives and Stops

A vowel-like phoneme however, will have its energy centred around particular energies representative of the vowel formants. Frequencies close to this formant area would be

expected to contain a large amount of the frame's total energy, and could therefore be expected to have a significant SBS amplitude, at least appreciably larger than SBS processed unvoiced frames. Depending upon the location and energy distribution, previous experimentation indicates that between three and six frequencies could be selected.

Results were obtained for all 34 test words. An example, again for the word "*fluff*" is presented in the table below. The first column gives the frame number, the second the maximum SBS amplitude of that frame, and the final column gives the number of frequency components that the SBS processing would select for reconstruction.

frame	max	#freq
0	17	2
1	13	3
2	12	1
3	30	2
4	17	4
5	12	5
6	12	3
7	15	2
8	12	3
9	14	2
10	9	0
11	14	2
12	15	4
13	11	2
14	16	3
15	11	3
16	14	2
17	14	1
18	12	1

frame	max	#freq
19	14	3
20	11	2
21	11	2
22	9	0
23	14	4
24	16	2
25	22	3
26	10	1
27	19	2
28	12	4
29	19	3
30	14	3
31	18	2
32	15	1
33	30	5
34	34	5
35	34	4
36	27	5
37	32	4

frame	max	#freq
38	27	4
39	38	4
40	25	4
41	65	1
42	26	4
43	37	3
44	22	4
45	20	5
46	38	3
47	20	6
48	29	4
49	22	5
50	15	5
51	12	2
52	24	2
53	13	4
54	18	3
55	11	3
56	16	2

frame	max	#freq
57	22	3
58	22	4
59	16	4
60	14	3
61	15	3
62	22	2
63	14	3
64	10	1
65	14	6
66	16	3
67	23	2
68	12	3
69	15	2
70	15	3
71	12	3
72	22	2
73	24	4
74	17	1
75	21	

Table 5.9 : SBS Maxima and Selected Frequencies For the Word "*Fluff*"

5.5.4 Results

In general, no definite information concerning reproducibility or intelligibility could be gained from examination of the number of components used in the speech reconstruction. Both poorly and clearly reproduced speech could exhibit periods of both high and low numbers of selected frequency components.

The SBS selection of several frequencies per frame for reconstruction given a broadband (unvoiced) input spectra was unexpected, but can be explained by noting that low frequency amplitudes could actually be an order of magnitude above any neighbouring values, i.e. 0.003 compared to 0.0003. Whilst neither of the frequency bins are likely to be audible due to the low energies involved, this order of magnitude difference was sufficient for that energy bin to dominate many adjacent filters, and hence obtain a significant SBS amplitude. These effects were widely spread, reasonably randomly positioned amongst the frequency bins, and therefore could result in a wide number of SBS selected bins, from zero to six.

Despite the effects of these variations in the low amplitude frequency bins, a general trend is that unvoiced speech has a significantly lower value of SBS maximum than voiced speech. There is an exception to this trend in that some unvoiced segments do not possess the white-noise type of spectra. For example, the unvoiced "*sh*" in word "*shoe*" contains a significant proportion of the frame's energy, and therefore it exhibits a large amplitude SBS maxima.

SBS spectra then, can not be used by itself as a voicing detector, though low values of SBS maxima do correspond to silent periods and some (but not all) unvoiced segments of speech. This information will be used later in an attempt to develop a reliable voicing detector, though this is an aside from the SBS development. In terms of improving the SBS model, these results imply that the problem with speech reproduction using this model is not so much whether the speech is voiced or unvoiced, but rather the magnitude of the power spectra for the selected frequencies.

5.6 Energy Problems

Attenuation of energy as a consequence both of windowing and, more significantly, elimination of most of the frequency bins, has a marked effect on the intelligibility of SBS reconstructed speech. The results of the DRT discussed previously, indicate a definite problem with some forms of words. Initially it was assumed that this was the result of unvoiced frames exhibiting a broad band spectrum. Ghitza discusses this briefly in his paper (1987), and notes that tonal artifact is very evident for unvoiced frames. The experiments we performed support this, but some completely voiced words also recorded very low intelligibility scores.

Upon closer investigation, it was found that the amplitudes of some frequency components were too low to be audible. In such cases, although the selected frequencies should be sufficient to reconstruct the frame, the energies in these frequency bins were so low (due to the two attenuation factors mentioned), that the word was not intelligibly reconstructed. Experiments were performed to test this theory.

5.6.1 Energy Experiments on Isolated Words

A selection of twelve trial words were analysed, selected from the fricatives and stops of Table 5.7. In the first instance, an examination was made of the energy of each frame, and how well it was audibly reproduced. To determine whether low energy was responsible for the poor reproduction, low energy frames were replaced initially by the original (unprocessed) frames (a test similar to that involved in Ghitza's determination of the source of tonal artifact), and later were amplified by some constant factor.

The replacement of a poorly reproduced frame with its original non-SBS processed frame improved the word's intelligibility in every case, but this could be due to reasons other than low energy, and so the gain factor was applied to the selected frames.

With only one exception (discussed later) specifically increasing the amplitude of low energy frames also increased the intelligibility. For example, in the word "*fluff*", the "*f*'s were originally inaudible in the reconstructed speech, and an examination of the frequency spectra indicated that frame energy at the beginning and the end of the word was significantly lower than in the middle. More quantitatively, the word "*fluff*" contains 55 processed frames, and the "*f*'s appear to take up frames 0-13 and 29-55. The energies in these frames were amplified by a factor of ten, with the result that the "*f*'s could now be heard upon reconstruction, although the quality was poor.

Continuing this method of analysis in accordance with the following table yielded similar results. In some cases the gain factor was far too large, and severe clipping of the waveform and distortion of the reconstructed speech resulted. In such cases, the gain was reduced to a factor of five. There were two male speakers for each word, though only one utterance of the word was selected for gain analysis. The word chosen was that of the pair that exhibited the clearest spectra, e.g. the ability to identify the phoneme segments in the speech spectra.

word	**frames analysed**	**poorly reproduced sound**	**frames poorly reproduced**	**applied gain**
fluff	75	initial and final f	20-33, 49-75	10
valve	91	initial and final v	0-21, 63-99	10
thin	52	th	0-15	10
then	63	th	0-17	5
sass	89	initial and final s	0-26, 43-89	10
zoos	71	z and s	0-11, 34-71	10
pop	45	initial and final p	0-10, 20-45	10
bib	32	final b	14-32	10
tot	48	final t	27-48	10
did	46	final d	17-46	10
kick	58	final k	22-58	10
gig	44	final g	19-44	5

Table 5.10 : Isolation of Poorly SBS Reproduced Phonemes

Whilst the segmentally amplified words did, in most cases, become more intelligible, it was found that amplifying unvoiced frames produced a marked increase in tonal artifact. The problem with this form of energy modification is that the amount of gain has to be adjusted for each frame, and some decision needs to be made as to which frames need the energy amplification. If speech compression was not the priority, then a solution to this would be to transmit the original frame energy along with the amplitudes and phases of the SBS selected frequencies in that frame. However, since low bit rate is important, an alternative techniques must be found.

5.6.2 Conserving RMS Energy

A useful measure of the energy content of a frame is the rms values of all the amplitudes of the frequency bins. Upon SBS reconstruction, most of these frames will be eliminated, so the energy must be transferred from them to the retained bins. This conversion is performed via the following algorithm:

$$\frac{N}{2} \times E_{rms}^2 = \left(n_1^2 + n_2^2 + \cdots + n_{k-1}^2 + 1\right) x_k^2 \qquad \textbf{Equation 5.3}$$

where

N = number of points in analysed frame (= number of energy bins*2)

E_{rms}^2 = the rms value of all the frequency amplitudes in that frame

x_1 = $n_1\ x_k$, $x_2 = n_2\ x_k$, ≡ $x_{k-1} = n_{k-1}\ x_k$

x_k = the value of the lowest frequency amplitude

k = the number of frequencies used in the speech reconstruction.

Such a scheme removes the problem of deciding which frames to amplify; it applies some variable gain factor to all frames. This energy value is coded in the amplitude bits of the transmitted speech and so will not increase the overall bit rate.

The subjective results of applying this rms energy requirement to the twelve trial words produced mixed results. Overall, intelligibility improved greatly from the non modified frames, and this form of selective frequency bin amplification will become a standard component of the speech analysis/synthesis system.

5.7 Voiced-Unvoiced-Silence Classification

The zero crossing parameter is an indicator of the frequency at which the energy is concentrated in the signal spectrum. Voiced speech is produced as a result of excitation of the vocal tract by the periodic flow of air at the glottis. The spectrum of glottal air flow decreases with frequency at approximately 12 dB/octave thereby producing a concentration of energy at low frequencies in the speech signal. Voiced speech usually shows a low zero-crossing count - typically in the range 0 to 30. Unvoiced speech is produced due to excitation of the vocal tract by a noise-like source at a point of constriction in the interior of the vocal tract. While the spectrum of the noise source is flat, the vocal-tract response usually increases with frequency. Thus, the unvoiced speech has a concentration of energy at high frequencies, and shows a high zero-crossing count - typically in the range 10 to 100.

The zero-crossing count for silence can vary considerably depending upon the characteristics of the recorded noise. Often, the spectrum of room noise is concentrated at low and middle frequencies, and in such cases the zero-crossing count would be lower than that of unvoiced speech, and comparable to that of voiced.

The energy parameter depends upon the recording environment. Influencing factors include such features as the microphone sensitivity, amplifier settings, and A/D quantisation. In general, silence exhibits the lowest energy, followed by unvoiced frames, with voiced frames possessing the largest energy value. Atal and Rabiner (1976) found that the signal energy is a very good discriminator between speech and silent frames, and that the zero-crossing parameter performed well for the voiced-unvoiced decision.

A problem with the energy parameter is that different speakers use widely varying talking levels. To make the voicing decision as accurate as possible, the signal level is scaled on a long-term basis (to ±1 - full range of *.wav* file format). However, this means that the background silence is also scaled so that, for a weak talker, the measured energy of the silence is often much larger than the mean energy for the silence distribution. In such instances, methods such as training the algorithm, or smoothing the data before VUS classification have been used to improve the accuracy. For this project, the values of energy and zero-crossing thresholds, used in the VUS classification, are set manually.

The justification for employing a VUS discriminator in the simulation is two-fold. Firstly, Ghitza's model of using the SBS threshold to reduce the number of frequency components used in the speech reconstruction, performs poorly for unvoiced speech (Section 5.4). This is to be expected given the broad spectrum nature of such speech. By identifying such speech segments, and increasing the number of components used in its reconstruction, the expectation is that the resulting speech will be of higher quality. This however, increases the bit rate - an outcome at odds with our objective of finding a low bit rate system, and hence the second justification for the discriminator. If a silent frame can be identified, all frequencies can be rejected, decreasing the overall bit rate of the system.

Using the values of zero crossings suggested by Atal and Rabiner as a starting point, and determining the average energy of each frame by manual inspection, a VUS classifier was developed for the Otago database (discussed later in Section 5.9), and later for the TIMIT sentences. The energy value offered an immediate classification. Below a certain energy value (low_energy_threshold [*let*]), the frames were almost invariably silent, and above another energy value (high_energy_threshold [*het*]), the frames were voiced. For energies between these two extremes, account needed to be taken of the zero crossing value.

A mid_energy_threshold (*met*) was determined, as were three zero crossing values, low_crossing_threshold (*lct*), mid_crossing_threshold (*mct*) and high_crossing_threshold (*hct*). If the frame energy was greater than *let*, but less than *met*, then the frame was classified as silent if the number of zero crossings was less than *mct*, else it was assigned to be unvoiced.

For energies greater than *met* but less than *het*, a zero crossing < *mct* resulted in a voiced classification. If the zero crossing was > *hct* the frame was unvoiced. Finally , a comparison was made of the percentage difference in energy from *met* to the percentage change in zero crossings × 40 to *lct*. If the energy showed the greater variation, the frame was designated voiced, else it was assigned unvoiced. This is illustrated in Figure 5.3.

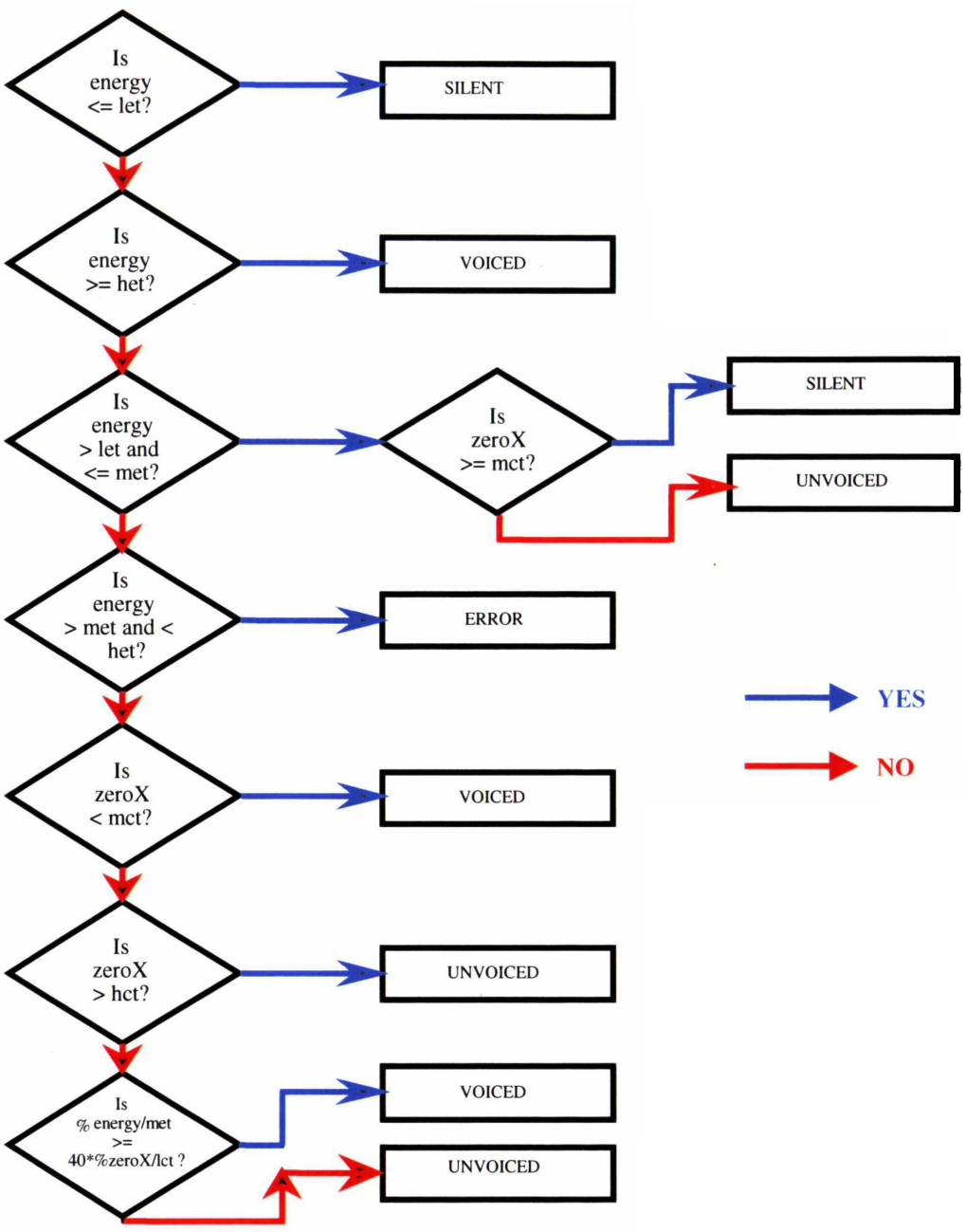


Figure 5.3 : VUS Determination

The results of this discriminator were extremely good. None of the frames marked as silent contained any speech (that could be detected), all of the voiced frames appeared to belong to voiced phonemes. There were some instances of high energy silent frames (containing recording hiss) and some low energy voiced frames being designated as unvoiced, but as this merely results in more frequencies being assigned to that frame, there was no resulting degradation in reconstructed quality (in fact the opposite is true).

5.8 TIMIT Database

Experiments performed so far have consisted of only one or two speakers, generally recording isolated test words. Improvements to the recorded quality of the test speech and speaker variability were obtained by employing the TIMIT speech data base (discussed in Section 3.7.2). In an attempt to keep the speakers as close to an Australasian accent as possible, the speakers selected from this database (for this limited analysis) originated from either the eastern states, or were likely to have been widely travelled. The groups most closely meeting this criteria were New England, New York City and "Army Brat" classifications, and both male and female speakers were selected. In general, the accents were not too pronounced, and were more similar to New Zealand speakers than the other TIMIT options.

The TIMIT database is sampled at 16 kHz, and so only the filtering (10th order Chebyshev) and decimation techniques were required to convert the data into a form useable by the simulation. Output files were generated to display the number of contributing frequencies for each frame, the original and modified amplitudes of all selected frequency components, the rms energies of each frame and also an audio version of the reconstructed speech.

For testing with this simulation, the TIMIT test sentences below (Table 5.11) were chosen from the phonetically-compact selection (the "SX sentences"), and were selected on the basis that they contained similar phonemes to words tested in isolation in previous experiments, and were spoken by at least one speaker from one of the three preferred dialect regions.

The code NE refers to a New England speaker, NY to New York City, and AB to "Army Brat". As mentioned earlier, these particular phrases were chosen as they contain phonemes either tested in the previous 34 word experiment, or in the DRT trials. Particular emphasis was placed on including the "*f*", "*v*", "*th*", "*s*" and "*z*" fricatives and "*p*", "*b*", "*t*", "*d*", and "*g*" stops.

Phrase	Speaker
Young people participate in athletic activities.	NY, AB
Biblical scholars argue history.	NE, AB
Tugboats are capable of hauling huge loads.	NE
The haunted house was a hit due to outstanding audio-visual effects.	NE, NY
Non-profit organisations have frequent fundraisers.	NE, NY
The most recent geological survey found seismic activity.	NY
Should giraffes be kept in small zoos?	NE, NY
Butterscotch fudge goes well with vanilla ice-cream.	NY, AB
Movies never have enough villains.	NE, AB
A moth zigzagged along the path through Otto's garden.	NY

Table 5.11 : TIMIT Test Phrases

5.8.1 Frame Overlap

The data redundancy (the amount of data shared between a frame of interest and its preceding and succeeding frames) employed by Ghitza was 100%, (50% succeeding, 50% preceding). The choice of frame overlap becomes a trade-off between speech quality and low-bit rate. To determine this, twice the number of frequencies were utilised for tagged unvoiced frames (from the VUS detector of 5.7), frequency quantisation was set at four bits logarithmic, amplitude quantisation was eight bit linear, and the phase quantisation set to six bit linear. Of interest was the effect of the number of frequencies used in the reproduction, so two runs of the experiment were performed, one with the SBS Threshold set to five, the other with the threshold set to ten.

At 100% frame overlap, essentially every point in the input waveform is processed twice - i.e. the bit rate is twice that of the situation where there is no overlap. At 0% overlap, the frame discontinuities are audibly evident as crackles or clicks. Only minor differences were audibly detected upon reducing the redundancy from 100% to 50%, not in the form of actual clicks, just subtle changes. The difference going from 50% redundancy to 25% is far more marked. Although some digit utterances were relatively unaffected, the majority displayed deteriorated speech quality and frame crackle. In general, the runs with the higher SBS

threshold (i.e. fewer frequencies used in the reconstruction) were more degraded than the runs with the lower value, but the deteriorating trend with decreasing frame overlap was very similar.

From this analysis, we conclude that a good compromise between quality and low-bit rate is achieved by the utilisation of a 50% data redundancy. A 25% value is still highly intelligible, but obviously inferior in terms of quality. These experiments are undertaken in the HTK trials, with redundancy values of 100%, 50%, and 25%, and are detailed in Chapter 6.

5.9 Parameter Selection

To further characterise the speech synthesis/simulation, a multi speaker database was required. For diagnostic simplicity, a database containing isolated words was used, specifically, the University of Otago's speech corpus comprising 21 speakers, each of whom speak each digit 4 times (http://kel.otago.ac.nz/hyspeech/corpusinfo.html). This corpus was employed in a new series of speech analysis experiments on isolated words, where the phoneme of interest could be more easily identified than in the TIMIT sentences. The quality of this corpus is not high, but does provide one of the few selections of New Zealand speakers, and is entirely suitable for measuring the effects of simulation quantisation. A subset of four speakers, two male and two female, were chosen from these 21 samples, and the best recording of the four utterances of each digit selected, resulting in a base of 40 recordings.

5.9.1 Frequency Quantisation

With all parameters set as previously, and data redundancy set to 100%, the representation of the frequency bins was explored using this Otago Digit Database. The maximum number of frequencies that can be used in the reconstruction of a frame is *framesize*/2, where *framesize* is the number of data points in any particularly Fourier frame. If these frequency bins are assumed to range from DC through to *sample_frequency*/2 (in accordance with the Nyquist/Shannon Sampling Theorem), then for a framesize of 128 points, and a sampling frequency of 8 kHz, each frequency bin would span 62.5 Hz. The introduction of a

bandpass filter to reduce the data spectrum to telephone bandwidth, 300 Hz - 3300 Hz means that the first four and the last twelve of these bins are discarded.

To determine the valid frequency bins, the bandpass region is divided (either linearly or logarithmically - depending upon the user's preference) by 2^{freq_bits}, where *freq_bits* has a maximum value of six corresponding to 64 levels. The nearest 62.5 Hz bin to each of these resulting frequency steps is designated as a valid bin, into which all nearest frequencies are to be stored. Frequency components are combined in these bins by adding the real and imaginary components.

Twelve experiments were performed as per Table 5.12:

SBS Threshold = 5 **(62.5 Hz bin width)**	**SBS Threshold = 10** **(62.5 Hz bin width)**
6 bits - linear spacing	6 bits - linear spacing
4 bits - linear spacing	4 bits - linear spacing
2 bits - linear spacing	2 bits - linear spacing
4 bits - logarithmic spacing	4 bits - logarithmic spacing
3 bits - logarithmic spacing	3 bits - logarithmic spacing
2 bits - logarithmic spacing	2 bits - logarithmic spacing

Table 5.12 : Frequency Quantisation Experiments Using the Otago Corpus

In general, the four bit logarithmic spacing could not be distinguished from the six bit linear spacing. Linear spacing of four bits sounded artificially high pitched, whilst two bit linear spacing was unacceptably horrible. Often, three bit logarithmic representation could not be distinguished from the four bit log case, the few exceptions being for the higher value of the SBS Threshold. The two bit log coding was high pitched, and noticeably artificial.

From this we conclude that four bit logarithmic frequency representation provides good quality sound reproduction (compared to the lower bit-rate representations), three bit is satisfactory if low-bit rate is a priority.

5.9.2 Voiced Frames

With all simulation parameters set as above, but with SBS Threshold set to five, and frequency representation set to four bit logarithmic, the corpus was tested to determine the effect of increasing the number of frequency components used to represent unvoiced frames. The voiced/unvoiced/silence distinction is determined by the algorithm discussed in Section 5.7.

Two runs were performed:

i) No difference in the number of frequencies used to reconstruct unvoiced frames

ii) Twice the number of frequencies used to reconstruct unvoiced frames (c.f. voiced frames)

Of particular interest in this test, are the digits beginning with the unvoiced "*f*" and "*s*". Indeed, upon reconstruction, the run using an increased number of frequencies produced clearer "*four*", "*five*", "*six*" and "*seven*", with little effect on the other digits.

These tests indicate that increasing the number of frequencies used to represent unvoiced frames does increase the quality of the reconstructed speech, though this will be at the expense of an increased bit rate.

5.9.3 Preemphasis

A detailed investigation of preemphasising speech via the use of numerical differentiation tools is discussed in Appendices A and C. Using the standard differencing technique (Equation A.10), the corpus was preemphasised, and applied to the corpus. The resulting sound was invariably high-pitched and artificial. Hence, for this filter shape, the introduction of preemphasis lowers the quality of the reconstructed speech.

5.9.4 Amplitude and Phase Quantisation

The amplitude of a speech signal can vary significantly. It is unlikely that much of a sound wave will possess high amplitude values, and that the majority of the sample points will possess a low or moderate magnitude. For quantisation then, a logarithmic coding scheme appears to be intuitively applicable to amplitude representation. Phase variation on the other hand, progresses in a far more linear fashion.

To investigate the audible effects of the amplitude variations, two series of experiments were performed, firstly with the unvoiced frames having twice the number of frequencies as the voiced, and then secondly with 2.5 times the number of frequencies. For each run (20 in total), the phase was kept constant at six bits (linear) and the amplitude was varied as Table 5.13

Amplitude coding form	Amplitude bits
linear	8
linear	6
linear	5
linear	4
linear	3
logarithmic	6
logarithmic	5
logarithmic	4
logarithmic	3
logarithmic	2

Table 5.13 : Amplitude Quantisation Experiments

Lowering linear representation from eight bits to five bits produced only a slight deterioration in intelligibility. At four bits, the difference was significant, and a three bit linear representation was very poor. Logarithmically, there was only a slight audible difference in reducing the bits from six to three. The two bit logarithmic form was still highly intelligible, but markedly different from the others. Comparing the logarithmic and linear representations, four bit logarithmic was only marginally different to the three bit logarithmic. The increase in the unvoiced frequencies did not affect this trend.

To determine the significance of the phase, for each of the amplitudes in Table 5.13, the phase was varied as per Table 5.14. The results indicated that the amplitude setting did not greatly affect the results of phase variation. In general, no differences were detected between a phase representation of six bits, and a phase representation of two bits. This may at first seem surprising, but it must be remembered that these experiments were performed using a reduced frequency reconstruction, and that whilst highly intelligible, the speech is already of a degraded quality, particularly for the higher SBS thresholds settings (SBS >3).

6 bits
2 bits
1 bit
0 bit (i.e. set to a constant)
random

Table 5.14 : Phase Quantisation Experiments

Audible differences were detected in further dropping the phase representation to one bit (two states), and at zero bits, the quality deteriorated considerably. Random phase produced (predictably) mixed results.

In summary, for most of the test cases, the intelligibility and quality of the reconstructed digits were the same for a three bit logarithmic amplitude/one bit phase coding as they were for an eight bit linear amplitude/six bit phase representation. A caution to note here, is that as the SBS threshold is increased and fewer frequencies are selected, changes in the quantisation values result in a more easily perceived difference.

5.9.5 SBS Thresholds

The final parameter to vary in this series of experiments is the SBS threshold setting. It is the SBS processing that distinguishes Ghitza's model from all others, and its value has a large influence over the quality and intelligibility of the reconstructed speech. Ghitza (private communication, 1997) does not recall the actual values he used in his prototype

system, and his papers give no clues to the setting. Experiments indicate that with an SBS threshold setting of ten, three frequencies per frame (on average) are utilised in the speech reconstruction. With a setting of one, about ten frequencies are employed.

Experiments were run with SBS settings of 3, 4, 5, 7 and 10, with no preemphasis, four bit logarithmic frequency representation, 100% data overlap, twice the voiced number of frequencies for the unvoiced frames. The experiments were initially run with an amplitude/phase setting of eight bits linear amplitude/six bits phase and repeated with a three bit logarithmic amplitude, one bit phase (Table 5.15) to further substantiate the claim made in Section 5.9.4.

As expected there was absolutely no detectable difference for the lower SBS values, but for threshold settings of seven and ten, minor quality differences were noted. The general trend indicated no perceived differences in increasing the threshold from three to four. Some minor changes were heard upon further increasing the threshold to five, but the quality and intelligibility are still high. The setting of seven resulted in an obviously degraded output, and the final setting of ten produced intelligible but heavily distorted speech.

SBS Threshold	**Amplitude/Phase Quantisation**
3	8 bit linear / 6 bit linear
4	8 bit linear / 6 bit linear
5	8 bit linear / 6 bit linear
7	8 bit linear / 6 bit linear
10	8 bit linear / 6 bit linear
3	3 bit logarithmic / 1 bit linear
4	3 bit logarithmic / 1 bit linear
5	3 bit logarithmic / 1 bit linear
7	3 bit logarithmic / 1 bit linear
10	3 bit logarithmic / 1 bit linear

Table 5.15 : Variation of SBS Thresholds on Intelligibility

To summarise these results, very intelligible, reasonable quality speech can be obtained with an SBS threshold setting of five. The improvement in the reconstruction that results from lowering this threshold to three is not large. Settings greater than five are poor, are more affected by lower amplitude quantisation (in terms of number of bits), and should only to be considered when low quality is tolerable.

5.10 Final Parameter Selection

Considering the amount of CPU time required for the HTK experiments, the fixed model features are:

(i) 128 point, 16 ms frame
(ii) Hamming window
(iii) initial filter height of 18
(iv) linear weighted interpolation between frames
(v) speech input to be in the TIMIT format

The variations to be considered in the model are

(i) a choice between 6 bit and 3 bit logarithmic amplitude quantisation
(ii) a choice between 8 bit and 1 bit phase representation
(iii) a choice between 6 bit and 3 bit logarithmic frequency bin quantisation
(iv) a choice of 100%, 50% or 25% data overlap
(v) a option to use differencing (linear) for high frequency emphasis
(vi) an option to increase the number of frequencies used in non-voiced frames
(vii) a selection of 1, 2, 3, 4, 5, 6, 7, or 10 as the SBS threshold

The forms of speech to be considered for processing are

(i) uncorrupted speech from the TIMIT database
(ii) speech from the TIMIT database corrupted with white noise
(iii) speech from the TIMIT database corrupted by one additional speaker
(iv) speech from the TIMIT database corrupted by two additional speakers

(v) speech from the TIMIT database corrupted by 4 additional speakers

The SBS processing is to be compared with

(i) no processing
(ii) Seneff 256 point processing
(iii) Seneff 128 point processing
(iv) Seneff sliding 128 point frame processing
(v) LPC processing

Chapter 6

6 HTK Results

6.1 Overview

HTK experiments were performed to investigate the remaining 26 SBS cases. In this analysis, it is not so much the absolute value of the recognition scores that are important, but the comparative trends, and their use in determining the superior processing system.

6.2 The HTK Experiments

The HTK experiments were performed using five different forms of the TIMIT sentences. The initial test determined the HTK results of the training and test files when no noise was added. To determine the performance of the speech processing algorithms in the presence of noise, white noise at nine S/N values was added to each test file. Cocktail Party noise was examined by first adding one speaker at nine S/N values, then two speakers at five S/N values, and finally four speakers, again at five S/N values. A total of 26 experiments (on each of these file types) were performed using the SBS speech processing algorithms, a further 18 were performed based on the Seneff algorithms, and finally an LPC 10 algorithm was tested and compared to the above results.

The SBS experiments involved varying the SBS threshold, quantisation levels of frequency, amplitude and phase, variation of the unvoiced modifier (additional frequencies selected to reconstruct unvoiced frames), frame overlap and the inclusion of differencing. These are summarised below in Table 6.1. The order in which these experiments were performed may seem strange, however, each experiment takes approximately one week, so the essential experiments were performed first, the remainder included for completeness as time permitted, and as further knowledge of the characteristics of the SBS algorithm were discovered.

Label	SBS Threshold	Freq/amp quantisation	Phase quantisation	Data redundancy	Unvoiced modifier	Differencing
A	1	6	8	100	1	No
B	3	6	8	100	1	No
C	5	6	8	100	1	No
D	5	3	0	100	1	No
E	2	6	8	100	1	No
F	4	6	8	100	1	No
G	10	6	8	100	1	No
H	10	3	0	100	1	No
I	5	6	8	100	0.6	No
J	4	6	8	100	0.5	No
K	5	6	8	100	0.4	No
L	5	3	0	50	1	No
M	5	3	0	25	1	No
N	5	3	0	100	1	Yes
O	7	6	8	100	1	No
P	7	3	0	100	1	No
Q	6	6	8	100	1	No
R	6	3	0	100	1	No
S	5	3	1	100	1	No
T	3	3	0	100	1	No
U	3	3	1	100	1	No
V	6	3	1	100	1	No
W	5	3	2	100	1	No
X	0	6	8	100	1	No
Y	0	3	0	100	1	No
Z	0	3	1	100	1	No

Table 6.1 : Variation of SBS Quantisation Parameters

An SBS Threshold of five was chosen as the best compromise to obtain good cocktail party and white noise results, so most of the variation of frame overlap, frequency/amplitude/phase quantisation, differencing and unvoiced modifier is done at this level. The SBS Threshold was varied from zero through to ten, missing only eight and nine (as these points yield little additional algorithm information). Additional experiments were performed at thresholds of three, and six as these provide the best results for white and cocktail party noise respectively, a threshold of zero was used as a baseline indicator, and thresholds of four, seven and ten provide useful comparisons with the other thresholds.

From Chapter 5, the levels for amplitude and frequency quantisation were selected as six bit logarithmic for an upper limit, and three bit logarithmic as the most stringent limit. For many of the experiments, phase information was deleted to more easily see the effect of varying the other parameters, it has a maximum at eight bit coding, though from Chapter 5, a value of one bit was trialled, and a two bit scheme provided additional information. Only one differencing variation was employed, as the previous chapter suggests no increase in intelligibility for differencing with the form of bandpass filters selected.

6.2.1 SBS Recognition Plots

With such an abundance of results, for clarity, not all variations are plotted together. Instead, a reduced representation to illustrate the effect of a particular variation is displayed.

To examine the effects of SBS variation, view runs *a*, *b*, *c*, *e*, *f*, *g*, *q*, *o*, and *x*, (with maximum number of quantisation levels), also (separately) runs *d*, *h*, *p*, *r*, *t*, and *y*, (for minimum quantisation levels), and runs *s*, *v*, *u*, and *z*, (with an extra phase representation) as per Figure 6.1, Figure 6.2, Figure 6.3, and Figure 6.4, (though only a reduced representation of *a*, *b*, *c*, *e*, *f*, *g*, *q*, *o*, and *x* is actually plotted).

To compare the effects of frequency, amplitude and phase quantisation, it is appropriate to view (*c*, *d*, *s*, *w*), (*g*, *h*), (*q*, *r*, *v*), (*o*, *p*), (*b*, *t*, *u*) and (*x*, *y*, *z*), as per Figure 6.6, Figure 6.7, Figure 6.8, and Figure 6.9.

The influence of frame overlap, (or data redundancy) can be seen from observation of runs *d*, *l* and *m* (Figure 6.10 and Figure 6.11).

The effect of differencing the speech signal can be viewed in runs *d* vs *n*, Figure 6.12.

The effect of the unvoiced modifier compared to no modifications, is illustrated by runs *c*, *i*, and *k*, and also *f* vs *j*, as in Figure 6.13, Figure 6.14, Figure 6.15, and Figure 6.16.

6.3 Calculation of Signal to Noise Ratios

For these experiments, the signal to noise ratio (S/N) is displayed on the coordinate (x) axis. This ratio, expressed in decibels, can be calculated by

$$S/N = 10\log_{10}\left(\frac{P_{signal}}{P_{noise}}\right) \qquad \textbf{Equation 6.1}$$

where P_{signal} is the power estimation of the speech signal, and P_{noise} is the power estimation of the added noise (either white or cocktail party). This expression is equivalent to multiplying the log ratio of the signal voltage to the noise voltage by 20. For the S/N values used in these experiments, the power of the signal is estimated by calculating the power spectrum ("*psd*") of the signal using the Welch method of spectral estimation, summing over the contributing frequency bins, and comparing the result to a similar calculation performed on the noise data. The resulting S/N values are rather approximate given the random nature of the added noise, and the variability of the speech signal over the 3888 total speech files (3696 training, 192 test), but do give a good indication of the change in added noise.

For the two and four speaker cocktail party case, it is not feasible (in terms of time) to compare all the random added speech signals' psd values to the test speech signal, given that there are 192 test files, with the possibility of any two or four of these signals being added with one of five amplitudes. Instead, a good indication of the signal to noise ratio can be found by noting that

$$\frac{P_x}{P_y} = \left(\frac{A_x}{A_y}\right)^2 \qquad \textbf{Equation 6.2}$$

where A_x is the amplitude of test speech signal (assigned value one (maximum amplitude of a *.wav* file)), A_y is the amplitude of the speech signal to be added to the test signal (i.e. the noise), and P_x and P_y, are the corresponding power estimates). Using this, the Signal to Noise ratio becomes

$$SN = 20 \log_{10} \frac{A_x}{A_y} \qquad \textbf{Equation 6.3}$$

or, more conveniently

$$SN = -20 \log_{10} A_y \qquad \textbf{Equation 6.4}$$

6.3.1 White Noise

To test the ability of the SBS processing to resolve speech from white noise, white noise of nine different amplitudes (0.5, 0.4, 0.3, 0.2, 0.1, 0.075, 0.05, 0.025, and 0.01) were added to each of the speech files in the TIMIT test database. Using Equation 6.4, the Signal to Noise Ratio for each of these attenuation factors can be evaluated, and are displayed in Table 6.2. The resulting 6 to 40 dB Signal to Noise Ratio of the files in the presence of white noise should provide a very demanding test on the ability of SBS processing to reject such forms of speech corruption.

White noise attenuation	**0.5**	**0.4**	**0.3**	**0.2**	**0.1**	**0.075**	**0.05**	**0.025**	**0.01**
Signal-to-noise ratio	6	8	10.5	14	20	22.5	26	32	40

Table 6.2 : Signal-to-Noise Ratios for Added White Noise.

6.3.2 Cocktail Party Noise

The frequency tracking ability of the SBS processing implies good white noise suppression as discussed above, but not necessarily good cocktail party noise suppression. The reason for this is that the addition of another speaker introduces new dominant frequencies, which the SBS may consistently select. The critical parameter for the introduction of such noise is the amplitude at which the additional speakers are added, rather than the total signal-to-noise ratio. If the additional speaker is of a markedly lower amplitude, it would not be expected that those added (noise) frequencies would (in general) dominate (and therefore be selected for reconstruction) over the frequencies of the original speech file.

Three series of recognition experiments were performed to test this. The first involved adding one speaker (random selection from the TIMIT Test database), at nine Signal to Noise ratios, the second added two such random speakers at five Signal to Noise ratios, and the third experiment added four speakers, again at five SN values. The amplitude multiplier of the one and two added speakers were kept constant, but for the four added speakers, the amplitudes of each speaker were allowed to randomly vary around a capped maximum level.

The original amplitude of the test speech, and added speech varies greatly, and psd calculations of the original TIMIT speech files (no added noise), range from 0.41 through to 1.1. An average value appears to be about 0.7, but given the variability, it is important to realise that the S/N values listed below illustrate only the average trend, not a definitive measure for every test case.

For the case of one added speaker, the same nine amplitude ratios employed to generate the white noise, are used to attenuate the speech file to be added. For the two and four added speakers, only the first five amplitudes are used. The Signal to Noise Ratio for one added speaker cocktail party noise appears as the first (non-bold) row of data in Table 6.5.

For the situation of cocktail party noise consisting of two added speakers, the first speaker is added with the first five attenuation multipliers as per the one added speaker case. The second speaker is added at half the amplitude of the first, as per Table 6.3.

Attenuation of speaker 1	**Attenuation of speaker 2**	**Maximum S/N of single speaker**	**Total S/N of both speakers**
0.5	0.25	6	2.5
0.4	0.2	8	4.4
0.3	0.15	10.5	6.9
0.2	0.1	14	10.5
0.1	0.05	20	16.4

Table 6.3 : Attenuation and Signal-to-Noise Ratio of Cocktail Party Noise with Two Added Speakers.

For the case of four added speakers, the amplitude of each speaker is calculated randomly within a specified maximum (first column of Table 6.4). A point of interest is that due to the randomness of the speaker attenuations, the experiment that caps the individual amplitudes at 0.3 produces a smaller (total) Signal to Noise ratio than the cap at 0.4. It appears, however, that it is the maximum amplitude of any one speaker that primarily determines the cocktail party noise immunity of SBS processing, rather than the total S/N. As such, the values in the recognition plots are presented in order of the maximum speaker amplitudes (x axis), rather than total S/N.

Maximum attenuation	Attenuation of speaker 1	Attenuation of speaker 2	Attenuation of speaker 3	Attenuation of speaker 4	min. single Speaker SN	total SN (dB)
0.5	0.212	0.253	0.347	0.384	8.3	-1.5
0.4	0.108	0.317	0.146	0.189	10	2.4
0.3	0.251	0.264	0.299	0.274	10.5	-0.7
0.2	0.113	0.177	0.191	0.194	14.3	3.4
0.1	0.021	0.091	0.074	0.085	20.8	11.4

Table 6.4 : Attenuation and Signal-to-Noise Ratio of Cocktail Party Noise with Four Added Speakers.

Table 6.5 summarises the Signal to Noise Ratios for these three cases of cocktail party noise, rounding the results of Table 6.3 and Table 6.4 to eliminate the decimal places.

	Attenuation of Added Speaker								
# added speakers	0.5	0.4	0.3	0.2	0.1	0.075	0.05	0.025	0.01
1	6	8	11	14	20	23	26	32	40
2	2	4	7	11	16				
4	-2	2	-1	3	11				

Table 6.5 : Total Signal-to-Noise Ratios of the Three Cocktail Party Noise Experiments.

6.4 SBS Results

6.4.1 Variation of Recognition with SBS Threshold

For clarity in the following figures, not all the SBS threshold values are plotted (SBS values of two and four are not included), but those that are, are indicative of the system trend. The more heavily quantised runs (*d*, *h*, *p*, *r*, *t*, *y*), and (*s*, *v*, *u*, *z*) are not displayed, but show the same trend as the plotted runs (*x*, *a*, *b*, *c*, *q*, *o*, and *g*) for all noise cases.

The variation of recognition with increasing Signal to Noise ratio, where the noise is of the form of white noise, is illustrated in Figure 6.1. The *base* score in this plot (and subsequent plots) are the results of the TIMIT files resampled at 8 kHz, but with no windowing, transforms, nor SBS processing. These are the results to which all the speech processing algorithms must be compared, for results below the base line indicate inferior performance to no processing at all.

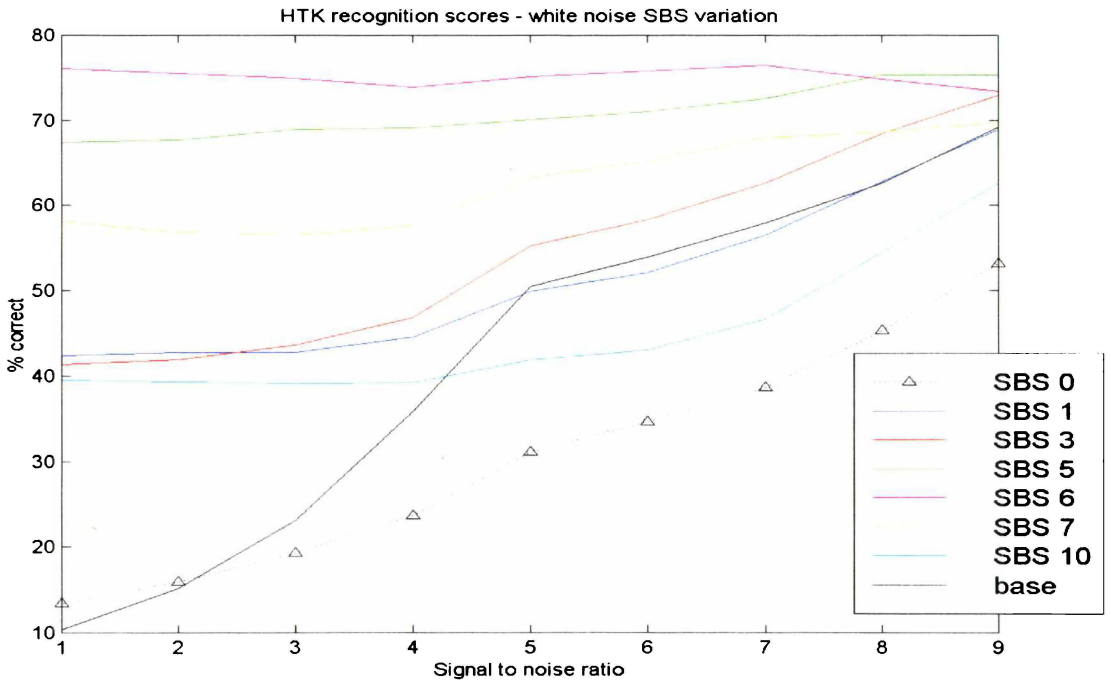


Figure 6.1 : Variation of Recognition with SBS Threshold for Nine S/N of White Noise

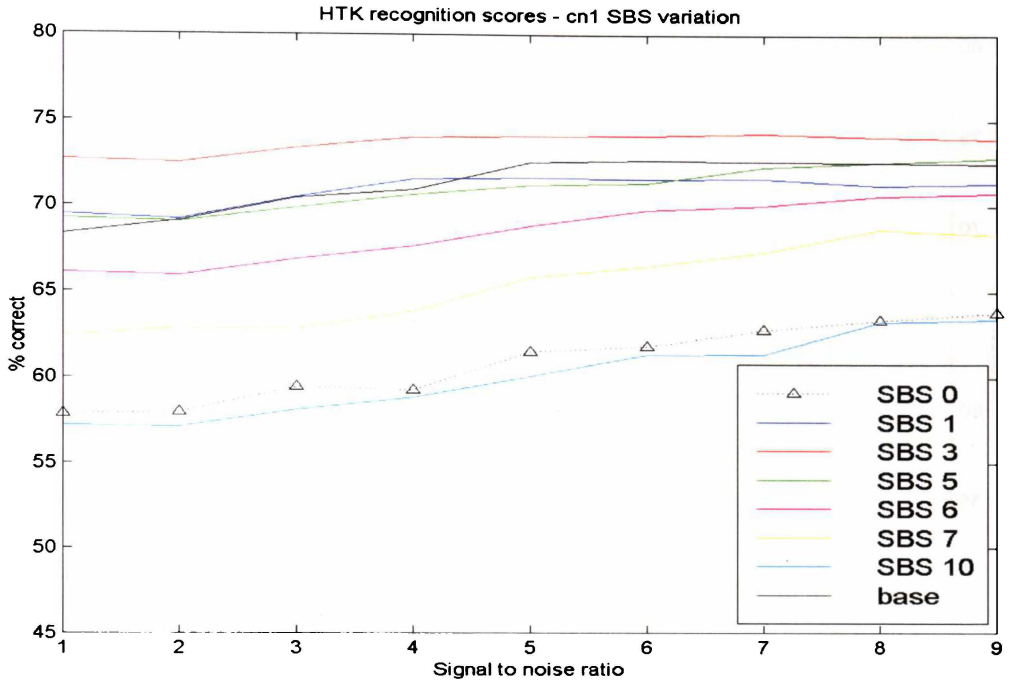


Figure 6.2 : Variation of Recognition with SBS Threshold for Nine S/N of One Added Speaker

The clear trend evident in Figure 6.1, is that recognition increases as the SBS threshold is raised from one to six, where the recognition is at its maximum. For further increases in the SBS threshold (i.e. a reduction in the number of frequency components), recognition decreases, until at SBS Threshold ten, it falls below the base score for the higher values of the Signal to Noise ratios (and closer to the line of no SBS processing at all).

For the case of one added speaker (Figure 6.2), there is the same increase in recognition with increasing SBS threshold as per the case of added white noise. Under this noise regime, however, the peak recognition occurs for an SBS threshold of three, after which point, recognition decreases. The base line level is significantly higher in this instance, and SBS thresholds above five have inferior recognition scores to this base level. The SBS ten trend (as per the white noise case) approaches the SBS zero line, indicating worse performance than if no SBS processing at all was used.

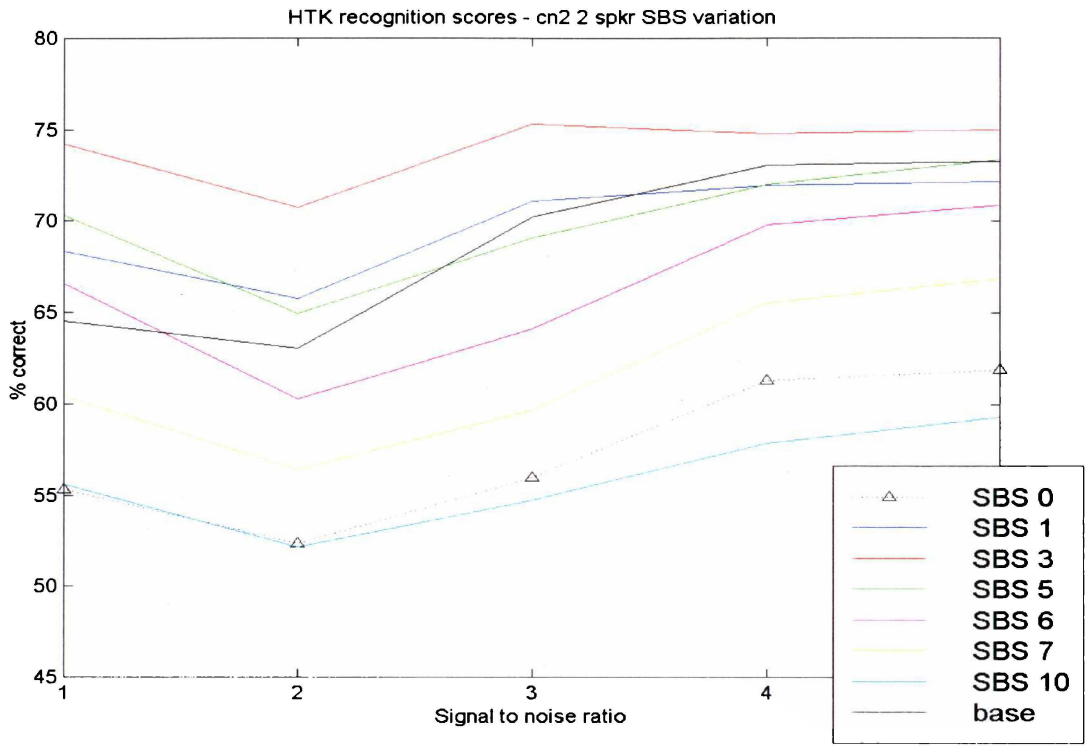


Figure 6.3 : Variation of Recognition with SBS Threshold for Nine S/N of Two Added Speakers

Figure 6.3 and Figure 6.4 show that a similar trend is evident for the instances of two and four added speakers respectively, again a peak at SBS three, a significant base-line level, recognition deteriorating markedly for SBS thresholds above five, and SBS ten approaching SBS zero. For these figures, the ordinate axis has been kept the same as the one speaker noise case to aid comparison.

All plots indicate that appropriate SBS processing can provide a significant increase in recognition over the base-line levels, for a wide range of signal to noise ratios. The difference, however, is that for the instances of added white noise, the peak performance is obtained for an SBS threshold of six (approximately five frequencies in the reconstruction), where for cocktail party noise, the performance peak is at SBS threshold of three, or an average of seven frequencies in the reconstruction. The setting of too high a threshold (ten) can be worse than the setting of no threshold at all.

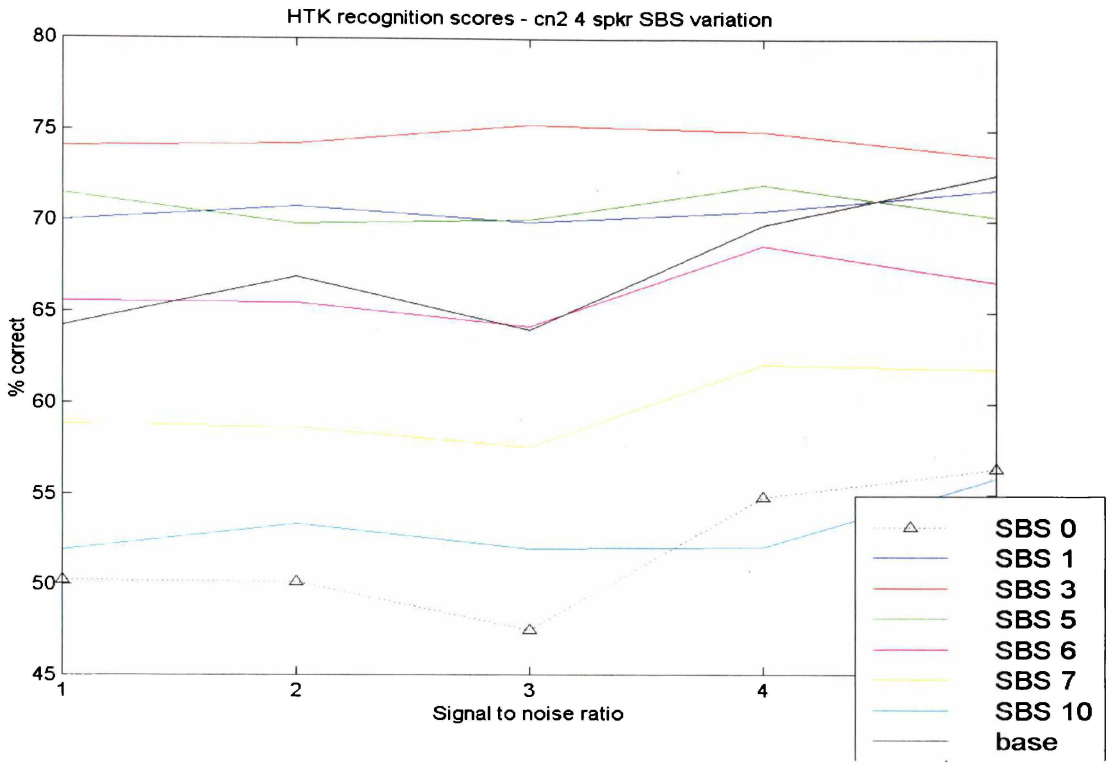


Figure 6.4 : Variation of Recognition with SBS Threshold for Nine S/N of Four Added Speakers

The broad band nature of the white noise is such that it dominates the lower SBS amplitude frequencies. A low SBS amplitude means that the frequency does not dominate many of the contiguous bandpass filters. White noise does exhibit the property that for a certain windowed frame, the energy of the noise in a particular frequency bin can be an order of magnitude larger than the energy in neighbouring bins. This energy difference is sufficient for it to dominate several contiguous bandpass filters, if those filters do not contain a strongly voiced frequency. The result is a low amplitude SBS value to that particular white noise frequency. In other words, white noise dominates the lower SBS amplitudes, and the setting of an SBS Threshold of six eliminates this form of noise, but not at the expense of the valid speech frequencies. This is further illustrated in the discussion on isolated vowels in Section 6.5.1.

Cocktail Party noise, however, has a very different frequency structure. The added noise can have moderate amplitude, strongly voiced speech, that results in moderate-to-high SBS amplitudes. Setting of an SBS Threshold sufficient to reject these added noise components, eliminates desirable frequencies as well. As a result, the SBS Threshold must be set far

more modestly, and a larger component of noise remains in the reconstructed speech. This is further illustrated for the case of an isolated vowel in Section 6.5.2.

In the absence of added noise, recognition peaks at an SBS threshold of three or four, as illustrated in Figure 6.5, corresponding to approximately seven frequencies in the reconstructed speech. White noise HTK results peak at SBS six, cocktail party noise at SBS three. SBS processing then, requires a higher threshold to ensure white noise frequencies are eliminated (and successfully does so, resulting in high recognition score), whilst cocktail party noise (which has difficulty rejecting all the noise components), tries to recreate the speech with extraneous frequencies, and subsequently scores a lower recognition result.

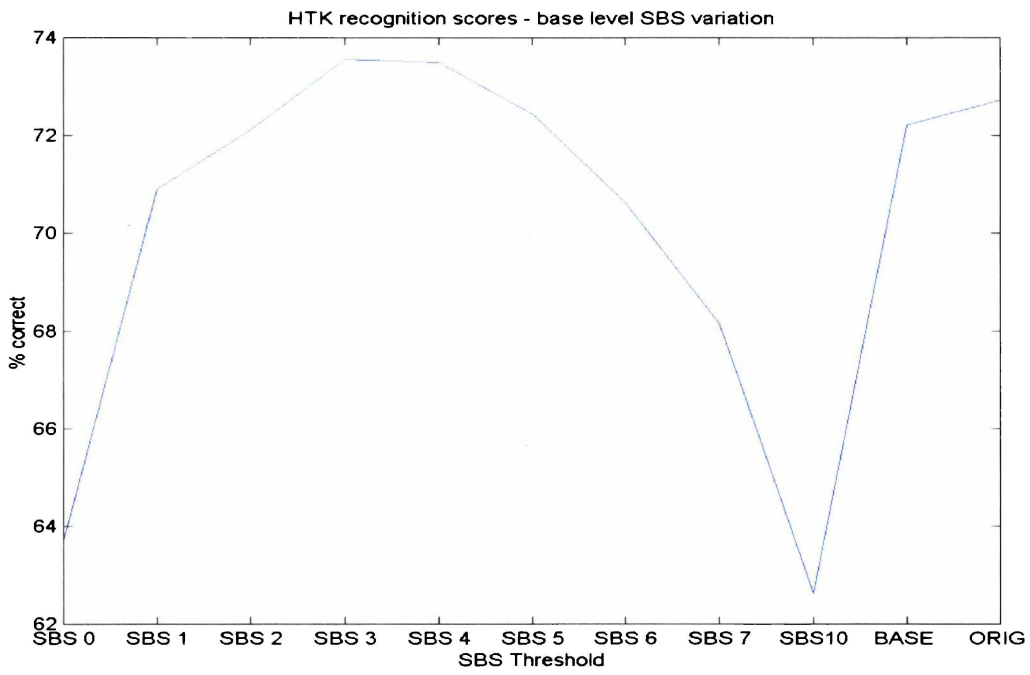


Figure 6.5 : Variation of Recognition with SBS Threshold in the Absence of Added Noise

From these results we see that white noise is best eliminated (for recognition purposes) with an SBS Threshold of six, Cocktail party noise, with an SBS Threshold of three. The difference is due to the differing SBS amplitudes exhibited by the two different forms of noise and how successful the SBS Threshold is at distinguishing between desirable and noise SBS amplitudes.

6.4.2 Variation of Recognition with Frequency, Amplitude and Phase Quantisation

The variation of recognition with altering quantisation levels for the nine white noise levels is illustrated below in Figure 6.6. In this figure, the solid lines indicate an algorithm that employs six bit log frequency and amplitude coding, and eight bit phase coding. The dotted lines, also indicated by the suffix "q" in the legend, indicate an algorithm that employs three bit log frequency and amplitude coding, and deletes the phase information entirely from the reconstructed speech. Dotted and triangle lines ("q1") employ three bit log frequency/amplitude and one bit phase, and finally, dotted lines and asterisx ("q2") indicate three bit log frequency/amplitude and two bit phase coding. The trend for SBS Threshold seven is very similar to that for SBS six, and so has been omitted in the above figure for clarity.

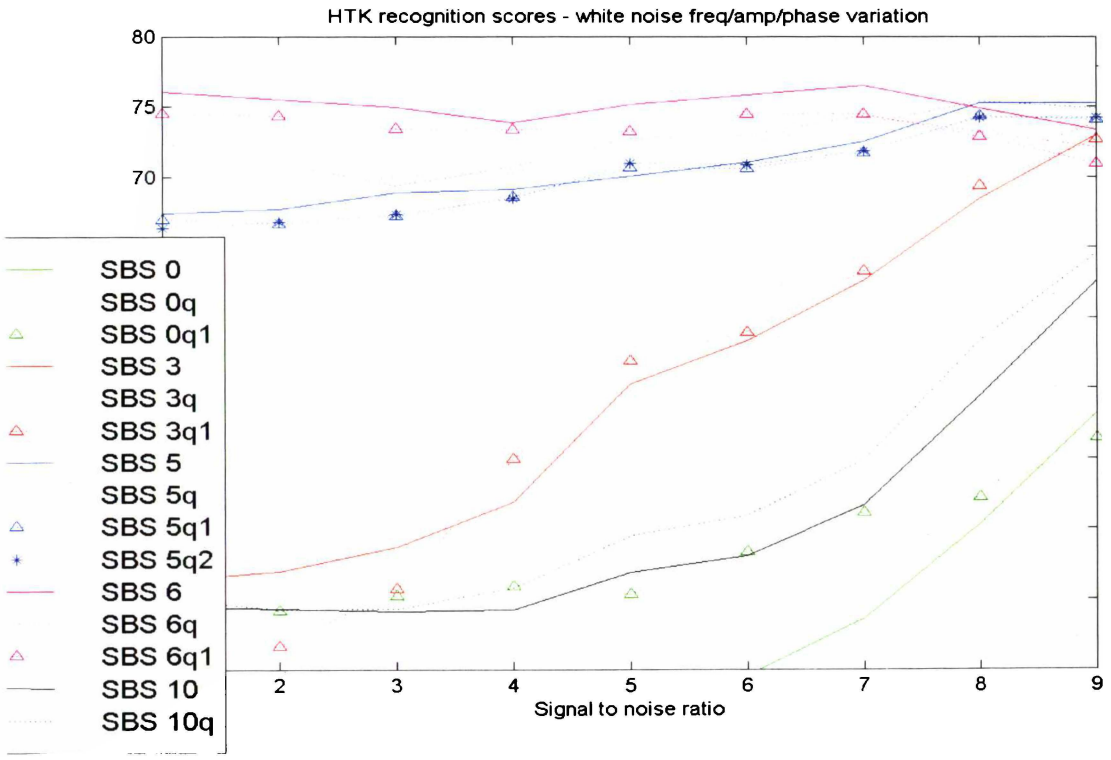


Figure 6.6 : Variation in Recognition with Frequency, Amplitude and Phase Quantisation Levels in the Presence of White Noise

For the introduction of white noise, the different quantisation levels appear to have a much smaller effect than a variation of SBS Threshold. The introduction of even one bit of phase

information produces a recognition score extremely close to that of the eight bit phase curve (except for the SBS zero case). Using two bits of phase information for an SBS Threshold of five, the resultant scores are practically identical to the one bit case. In general, the introduction of the single phase bit increased recognition by perhaps one percentage point (just outside the error bounds of this process), but could not increase the maximum intelligibility for SBS six.

When one additional speaker is introduced as a noise, the resulting variation in recognition with quantisation levels can be viewed in Figure 6.7. For all forms of cocktail party noise, the trends for threshold values of 3, 5, 6 and 7 were very similar, and so only the first two are illustrated in the following plots to increase clarity. For comparison purposes, the recognition axis (the ordinate axis) has the same range as that for the white noise.

Again, there is still not a large variation in recognition with the change in quantisation levels. The introduction of more stringent quantisation levels generally decreases recognition (except SBS ten), but this change is significantly less than the change resulting from alteration of SBS Thresholds. The one and two bit phase cases are extremely close together.

For two and four added speakers, as illustrated in Figure 6.8 and Figure 6.9 respectively, the trend is very similar, more stringent quantisation decreases recognition except for the extreme threshold values of zero and ten. Again the axes have been adjusted to be consistent with the white noise case, and the trends for thresholds six and seven omitted.

The comparison between the different quantisation schemes indicates that except for the extreme SBS thresholds (zero and ten), the three bit frequency/amplitude – zero bit phase, produces a slightly lower HTK score than the six bit frequency/amplitude – eight bit phase results. As mentioned, the variation of recognition with quantisation scheme is small compared to the variation that results as the SBS threshold is altered.

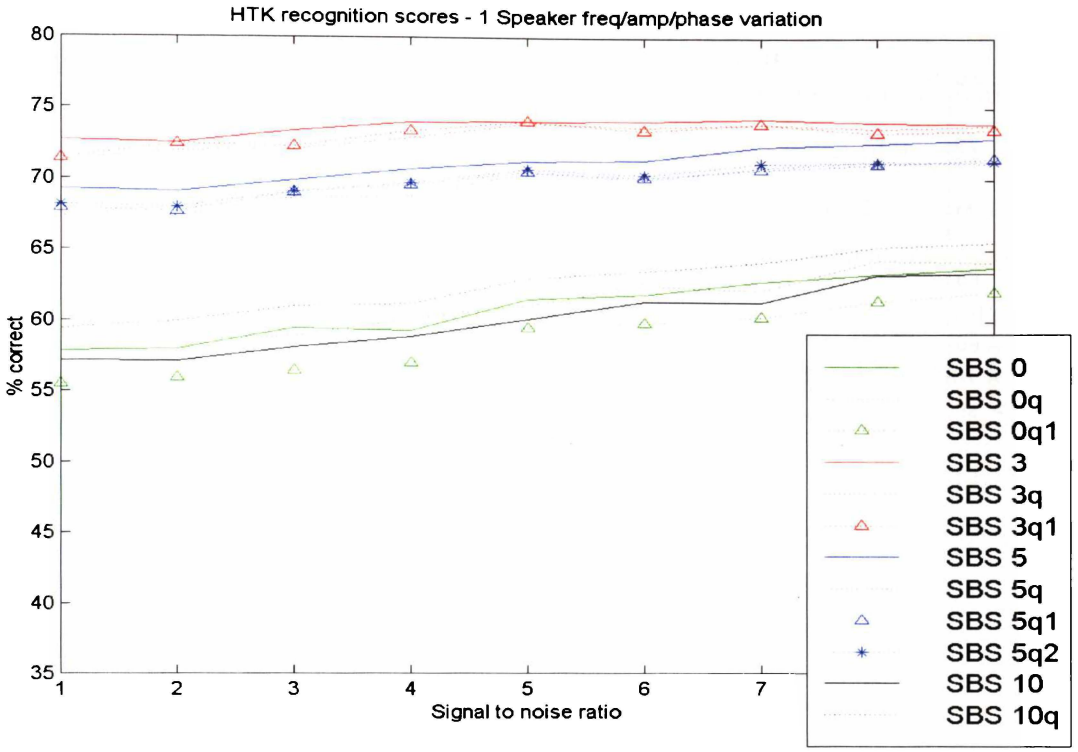


Figure 6.7 : Variation in Recognition with Frequency, Amplitude and Phase Quantisation Levels for One Added Speaker

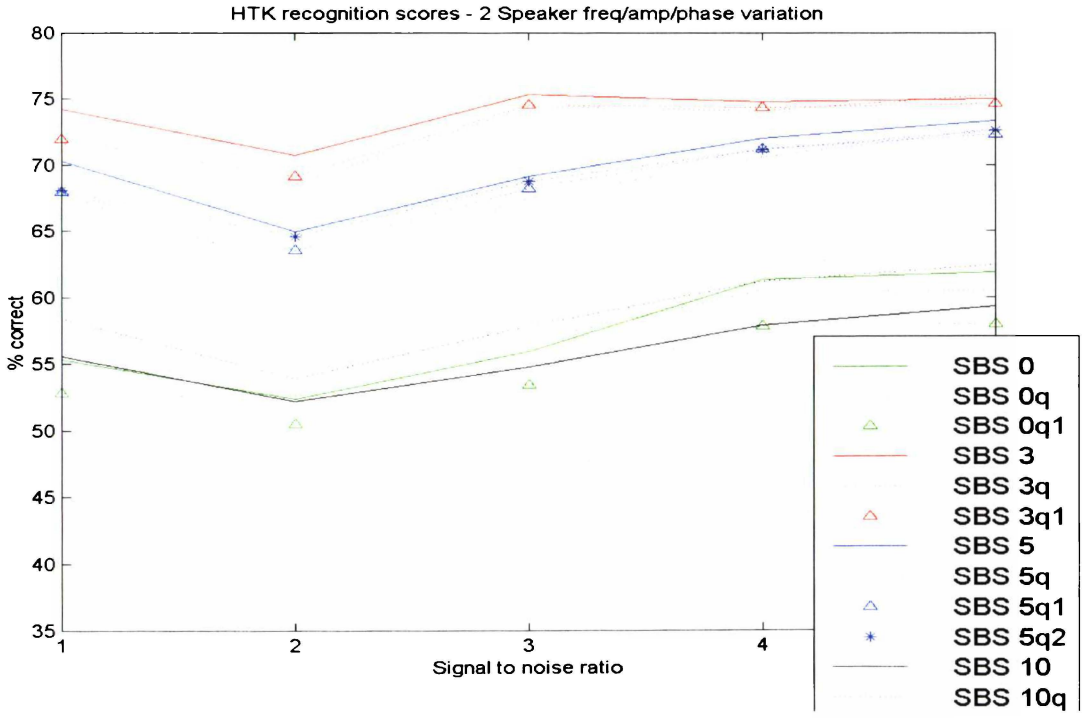


Figure 6.8 : Variation in Recognition with Frequency, Amplitude and Phase Quantisation Levels for Two Added Speakers

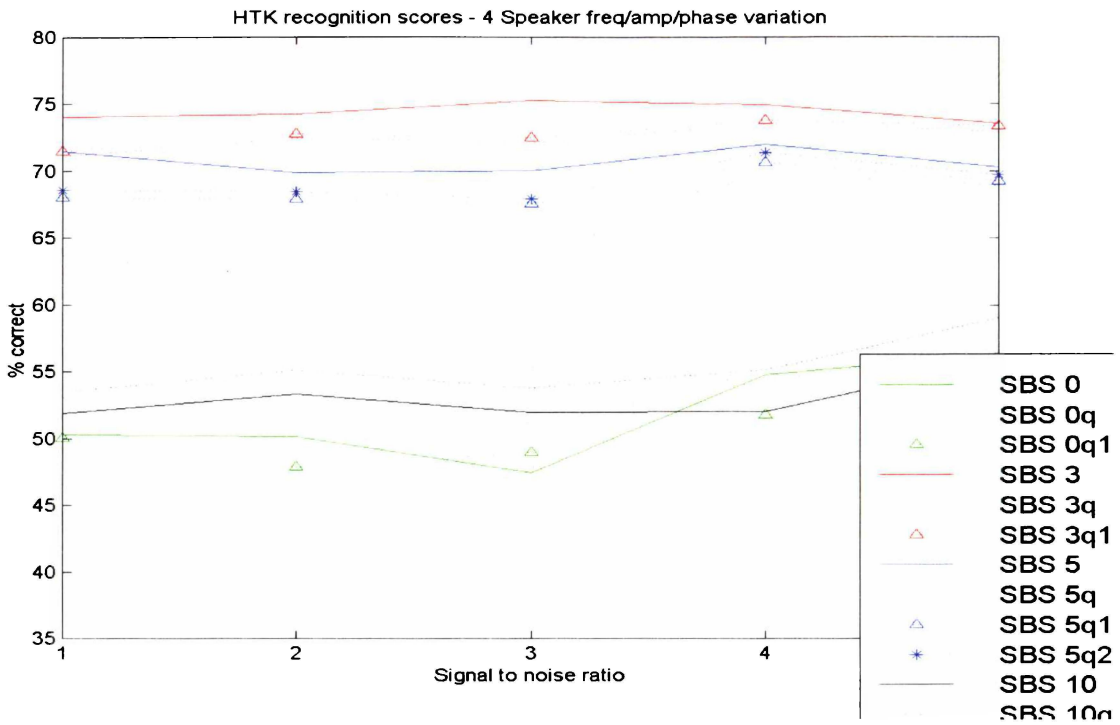


Figure 6.9 : Variation in Recognition with Frequency, Amplitude and Phase Quantisation Levels for Four Added Speakers

There is more of a change in the recognition scores as the quantisation parameters are varied than for the one speaker case, and hence it is possible to see the one bit phase plot situated between the zero and eight bit curves, and similarly for SBS five, the two bit plot lies between the one bit and eight bit curves as expected.

These results are consistent with the DRT scores that indicated that there was very little audible difference between the two schemes.

In summary, although in many instances it is difficult to separate the curves for zero bit phase from those that use a one bit phase representation, in general the one bit phase produces superior recognition results. The incorporation of a second phase bit does not significantly increase recognition scores. This information, combined with that from the listening tests (from Chapter 5), suggests retaining this one bit phase coding for the final model.

6.4.3 Variation of Recognition with Frame Overlap

For the cases of the four forms of added noise, the alteration in recognition with frame overlap is illustrated in Figure 6.10 and Figure 6.11 (note, that in these plots, the ordinate axis spans a smaller range than the previous plots to better highlight the differences in each run). The figures given as a percentage in the legend are the amount of data redundancy, in other words how often the data is repeated. A figure of 100% indicates that each data point is used twice, i.e. a 128 point frame, advancing 64 points for each new 128 point window. The figure of 50% indicates a slide of 32 points, and 25% indicates a slide of 16 points.

The 25% figure is the lower limit, since below this, the frame discontinuity produced by the finite windowing necessary for the FFT has proven to be auditorily significant (Section 5.8.1). At 25%, the weighted interpolation between frames can still smooth much of this discontinuity. All these figures are taken with an SBS Threshold of five.

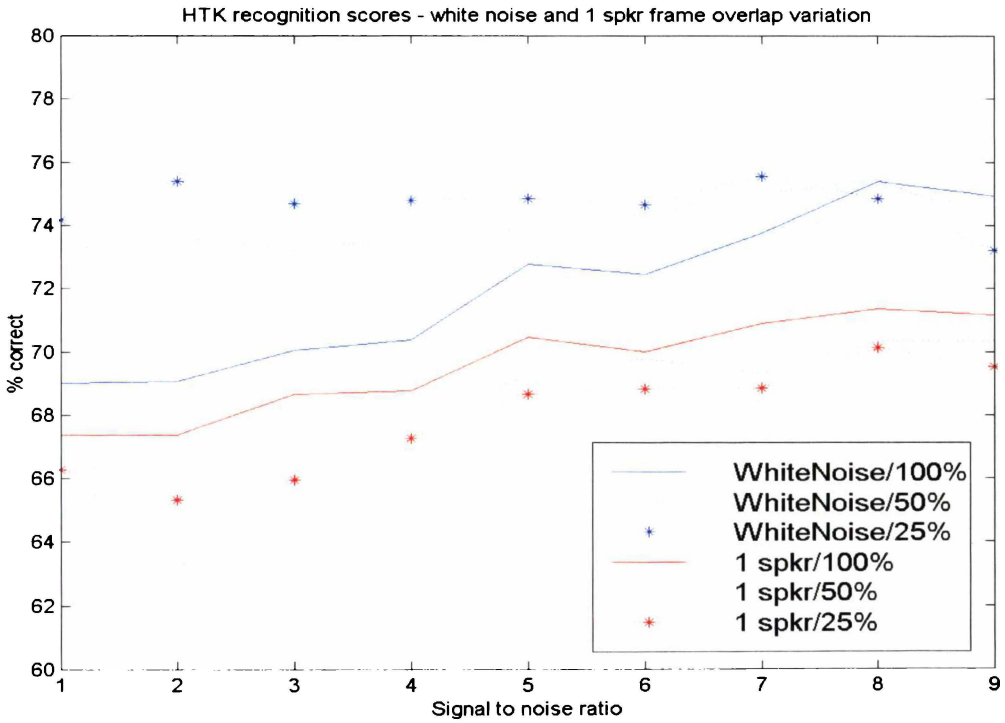


Figure 6.10 : Variation in Recognition with Frame Overlap for White Noise and One Added Speaker

For all the cases of Cocktail Party Noise, there is only a 1 or 2 % difference between the different forms of overlapping frames. The 100% case scores slightly higher in all three of

these cases, which is to be expected, as the frame overlap helps eliminate the discontinuities, and therefore provide less corrupted speech. The smaller the amount of overlap, the less "smoothing" of the signal occurs, the more the FFT frame discontinuities will affect the result.

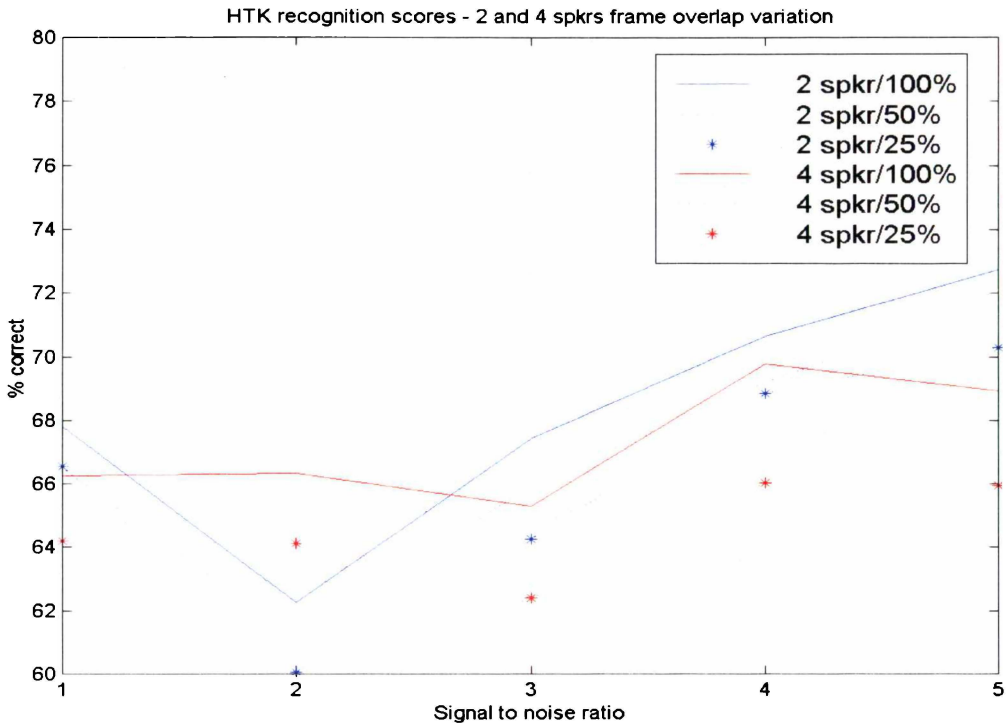


Figure 6.11 : Variation in Recognition with Frame Overlap for Two and Four Added Speakers

However, for the white noise, as illustrated in Figure 6.10, although there is very little difference between the 25% and 50% overlaps, there is a considerable difference in the 100%, which for the lower Signal to Noise ratios is significantly lower. It appears that for large levels of added white noise, the large frame overlap effectively adds more white noise into the resultant speech, lowering the recognition scores. As the level of added noise decreases, the 100% overlap curve converges with the 50% and 25% results.

The somewhat anomalous result of an increase in recognition with reducing frame overlap for speech corrupted by white noise, is to our advantage, in the development of a low bit rate coding scheme. It appears that a 25% overlap increases recognition for white noise events, and degrades cocktail party noise results by about 2%. The 50% overlap degrades the

cocktail party results by about half that of the 25% case, but does not provide as large an increase in the white noise recognition.

6.4.4 Variation of Recognition with the Inclusion of Differencing

Due to time constraints, and the auditory results from Section 4.8 that did not indicate positive results from differencing, only one HTK experiment was performed that included the Differencing Algorithm. The results, (illustrated in Figure 6.12) follow a very similar trend to those for the frame overlap, discussed in Section 6.4.3, in that for all the cocktail party noise results, the non differenced routines scores marginally higher (a few percentage points) than the differenced results.

For the white noise, however, the differenced routine is considerably more recognisable at the lower signal to noise ratios. White noise, as previously mentioned, tends to produce SBS amplitudes lower than those that result from voiced speech. However, the broadband nature of this form of noise is such that it can be expected to exhibit these low SBS amplitude across all the frequency bands. For the lower to mid frequencies, the high SBS amplitudes from the voiced speech mask most of these white noise contributions. For the higher frequencies, this is less likely, and in these instances, the frequencies SBS processing selects for speech reconstruction may originate from the white noise rather than the original speech. The Differencing Algorithm shifts the speech to a higher frequency range, and in doing so, the voiced speech begins to mask the higher frequency-white noise SBS amplitudes. The net result is an increase in the proportion of selected speech-produced frequencies to white noise produced frequencies, upon the introduction of the Differencing Algorithm.

Cocktail party noise has a totally different frequency and SBS spectra, and does not exhibit this dominance of the upper frequency range that the white noise did. Figure 6.12 indicates that recognition decreases with the introduction of differencing. This is consistent with the results of the informal listening tests of the previous chapter, as those results indicated that, for the filter shapes employed in Ghitza's SBS algorithm, high frequency emphasis decreased the intelligibility of the speech. We would hope, and generally do find, that there is a strong correlation between auditory intelligibility and HTK generated recognition

scores. Note, that the informal listening tests were not performed in the presence of *high amplitude* white noise, and so the white noise HTK results could not have been predicted.

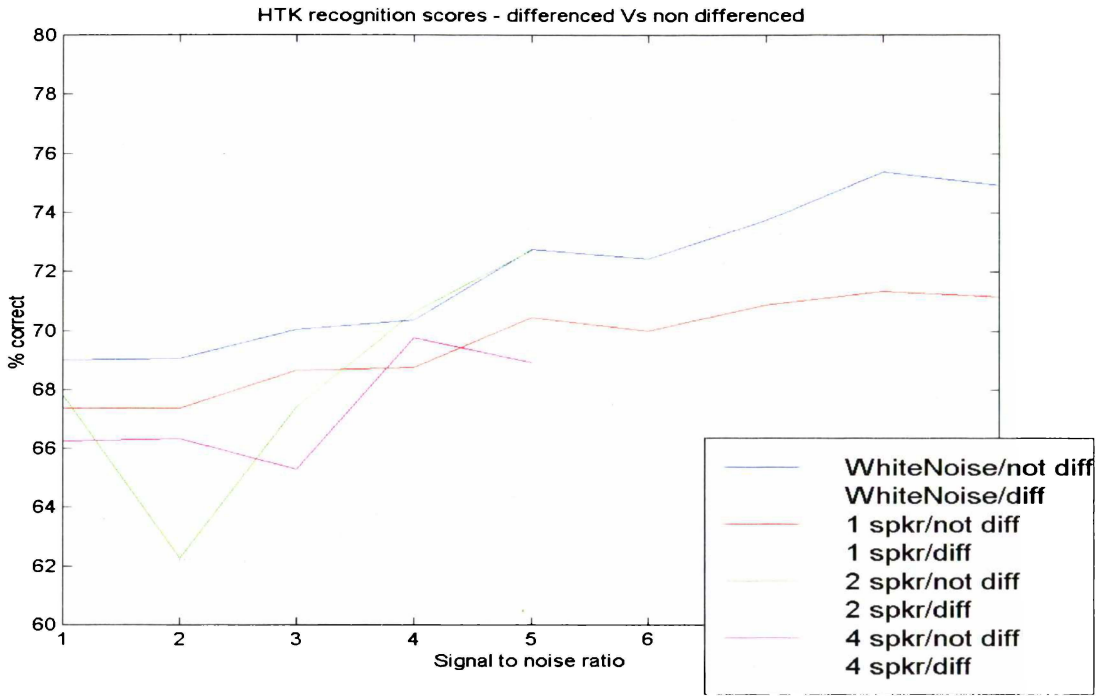


Figure 6.12 : Variation in Recognition with Differencing Included

6.4.5 Variation of Recognition with the Unvoiced Modifier

The DRT results (Section 5.7) indicated that considerable improvement in the recognition of the unvoiced results could be obtained by the introduction of an unvoiced modifier variable that effectively increases the number of frequencies used in the reconstruction of a frame if that frame can be identified as being unvoiced. The results are illustrated in Figure 6.13, Figure 6.14, Figure 6.15, and Figure 6.16 where the ordinate axis has been set to be the same for all four plots.

The expectation is that the inclusion of this variable would shift the curve towards the next lower SBS threshold level as what is effectively being achieved is an increase in the number of frequency components (employed in the speech reconstruction) as the unvoiced modifier fraction reduces.

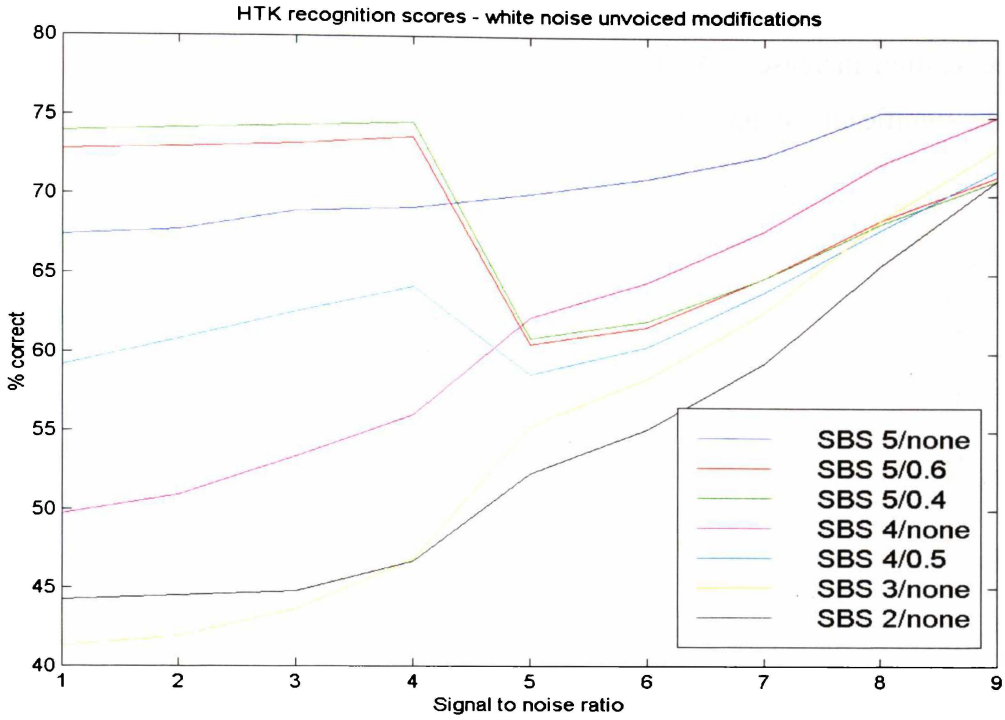


Figure 6.13 : Variation in Recognition with Unvoiced Modifier for White Noise

For the SBS Threshold five plots, an unvoiced modifier of (say) 0.6, produces some frames where the number of frequencies used in the reconstruction is more indicative of an SBS Threshold of four or even three. White noise scores peak at an SBS value of six (Section 6.4.1), and so the expectation is for the inclusion of the unvoiced modifier to move the recognition curve to follow the trend of SBS Threshold three, resulting in lower recognition scores.

The white noise plot (Figure 6.13) follows this trend for the high signal to noise ratio values. For the last five noise levels, the recognition score for the SBS five plot, decreases with the inclusion of the unvoiced modifier, towards the SBS three curve, a significant reduction. For the SBS four case, the curve moves down towards the SBS two curve, but in this instance, the difference between the curves is not large, and therefore there is a smaller drop in recognition.

The increase in recognition for the low S/N values is not well understood, and in fact was sufficiently unusual to warrant the repeat of these three HTK – white noise runs. This repeat produced identical results. Perhaps for these levels of white noise, the additional frequency

components selected originate solely from the original speech rather than the noise, and hence recognition increases. While testing of the algorithm did indicate this, it does not explain the significant recognition drop between S/N tabulated values of four and five.

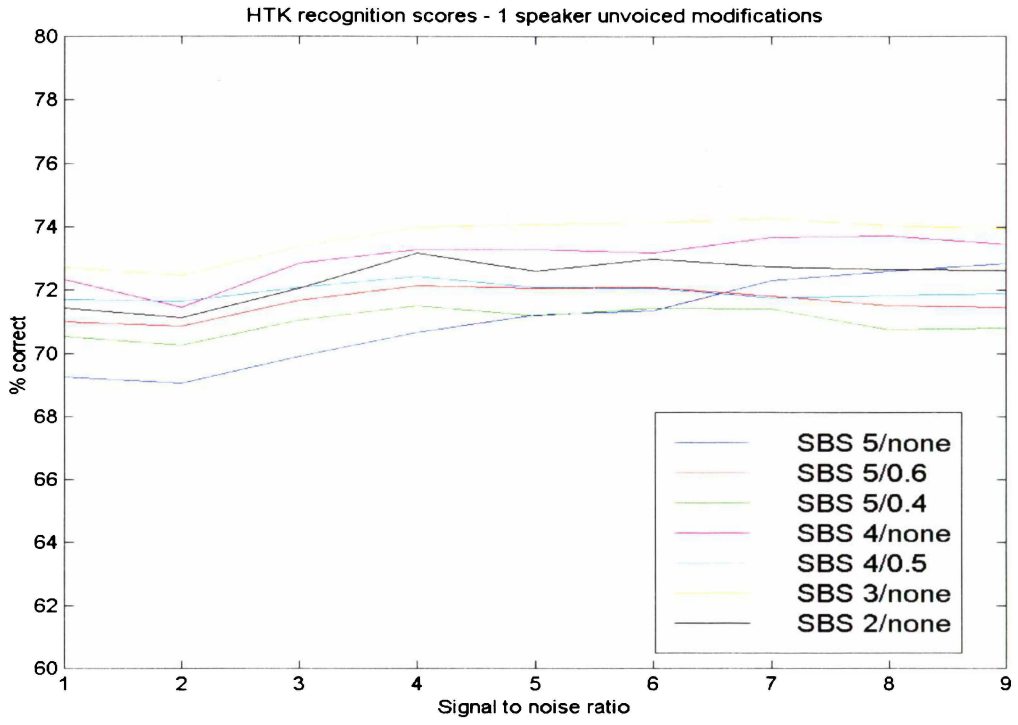


Figure 6.14 : Variation in Recognition with Unvoiced Modifier for One Added Speaker,

For noise of the form of one added speaker (Figure 6.14), the recognition results for SBS Thresholds of both three and four are higher than the results for a Threshold of five, so the expectation is for the inclusion of the unvoiced modifier for SBS five to increase the recognition score. However, the variation in recognition with varying SBS Threshold from five to three, is only a couple of percentage points, and so any variation introduced by the unvoiced modifier will be very slight, and difficult to detect.

A claim can be made that for most of the S/N values, the inclusion of the unvoiced modifier moves the SBS five curve towards the SBS three curve. For the SBS four result, the SBS two curve is very close, and so there is no obvious trend in the movement of the curve with the unvoiced modifier included.

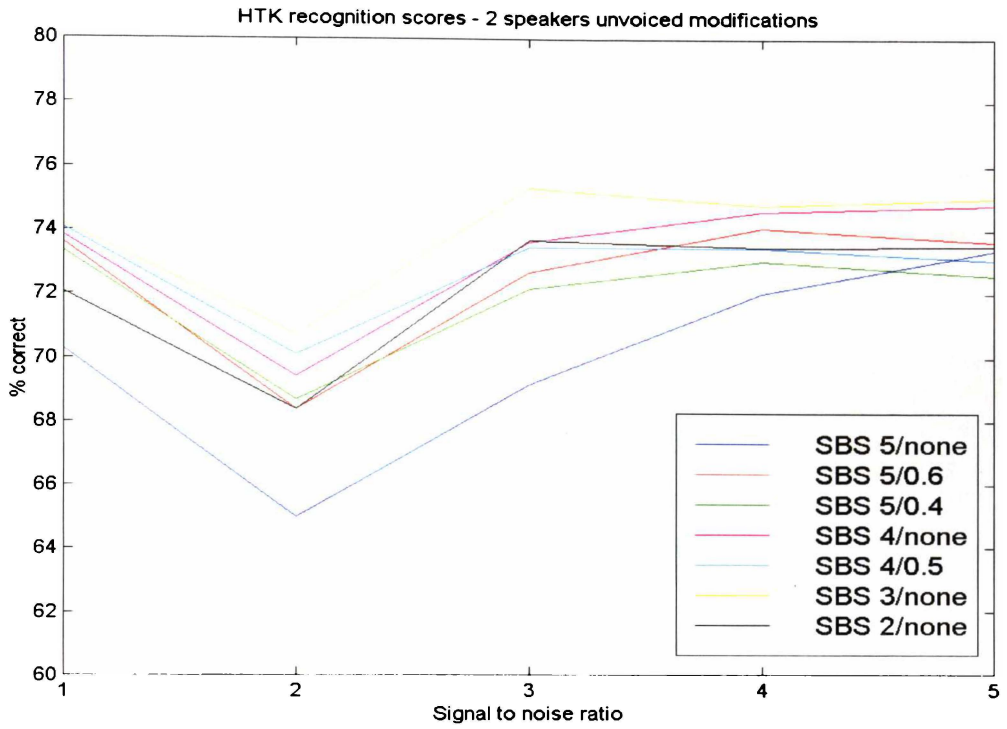


Figure 6.15 : Variation in Recognition with Unvoiced Modifier for Two Added Speakers

For the case of two added speakers (Figure 6.15), there is a reasonably large increase in recognition scores as the SBS Threshold is varied from five down to three. The expectation is for the SBS five with the unvoiced modifier recognition to be higher than the unmodified SBS five results, and this is indeed the case. The situation is more complex for the case of the unvoiced modifier with SBS four. Examination of the number of frequencies used to reproduce the frames, show that with the unvoiced modifier set to 0.5, the number of frequencies more closely reflects an SBS Threshold of two rather than SBS three. The SBS two curve is very close to the SBS four, and so any movement of the curve as a result of the unvoiced modifier is very difficult to detect

Finally, Figure 6.16 for four added speakers, shows a very similar trend to Figure 6.15. Again, recognition increases from SBS five to SBS four and SBS three, but decreases for SBS two. So, the expectation (and result) is that the unvoiced modifier increases the recognition results for SBS five (there being a significant difference between SBS five and SBS three), but decreases them slightly for SBS four given that the SBS two curve is so close.

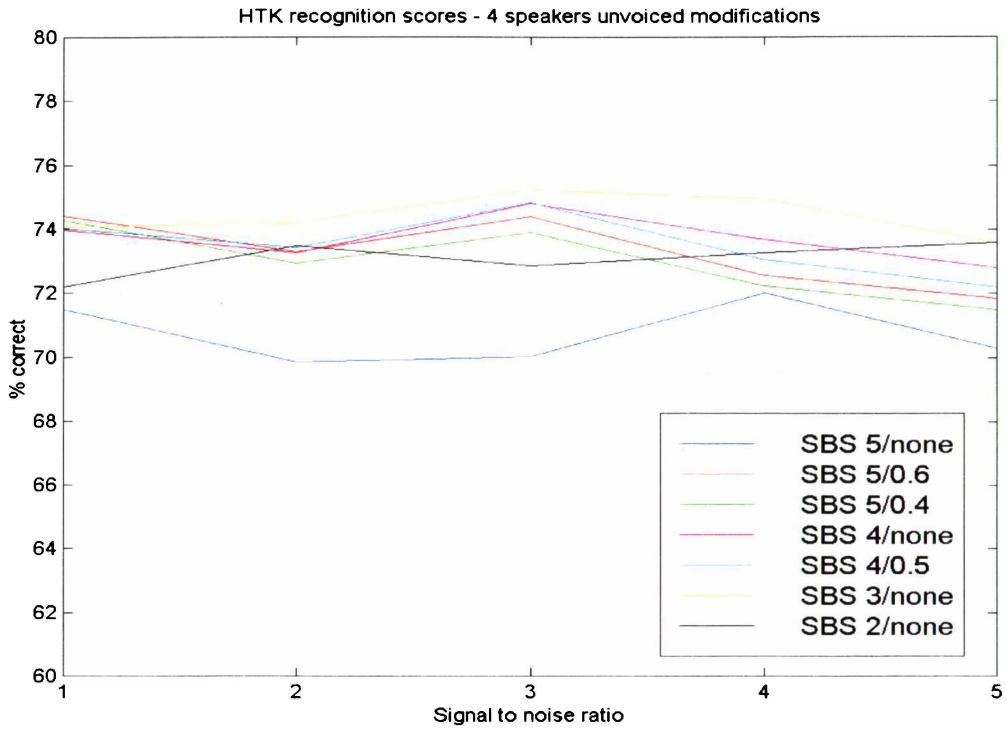


Figure 6.16 : Variation in Intelligibility with Unvoiced Modifier for Four Added Speakers

In summary, the introduction of the unvoiced modifier adds frequency components to the reconstructed speech, effectively decreasing the SBS Threshold. In terms of HTK recognition (without regard to low bit rate coding), it is more effective to completely change the SBS threshold, and maximise recognition in that manner, as the unvoiced modifier tends to produce results mid way between the integral SBS plots.

6.5 Vowel Analysis

The HTK speech recognition results (Section 6.4) clearly show that the SBS processed files exhibit a far greater white noise suppression than the non SBS-processed. To examine the reasons for this, it is illustrative to view a single vowel (for example "IY") in isolation. The spectra of this vowel (Figure 6.17), shows peaks at approximate bin numbers 8, 37 and 50, corresponding to frequencies 500 Hz, 2.8 kHz and 3.1 kHz respectively. Many other (smaller) amplitudes do of course exist.

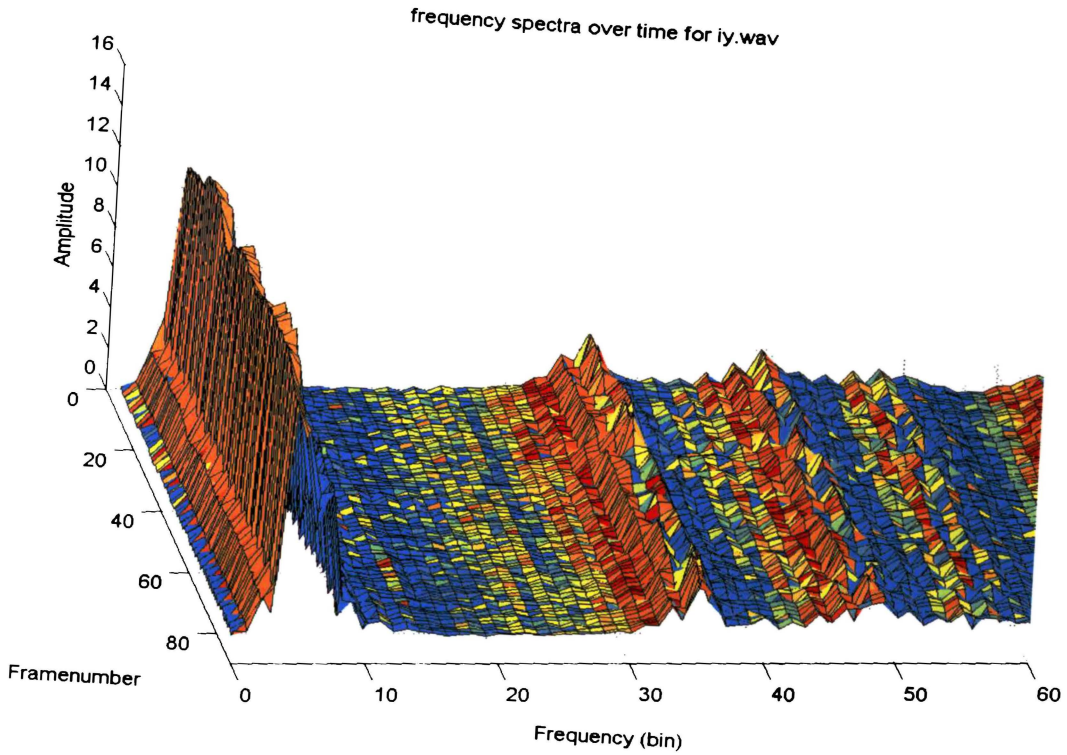


Figure 6.17 : Frequency Spectra Over Time for the Vowel "IY"

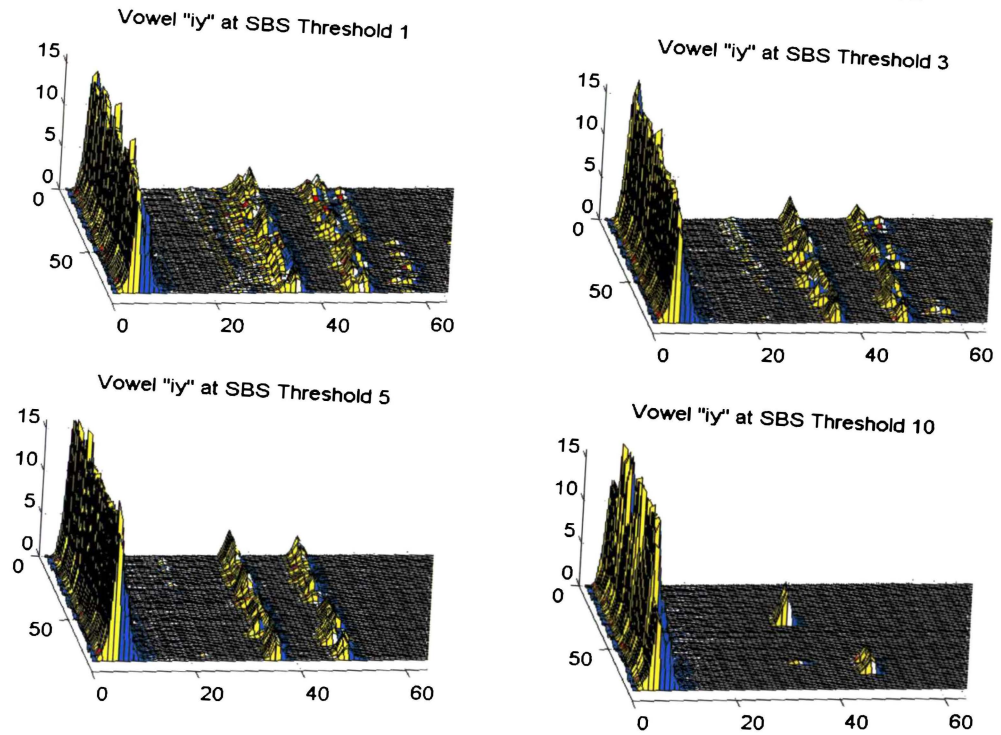


Figure 6.18 : Frequency Spectra Over Time for Vowel "IY" with Different SBS Thresholds

As the SBS threshold is increased from one through to ten (Figure 6.18), these lower amplitudes become increasingly suppressed, until, with SBS threshold at five, it is fair to say that they no longer contribute to the speech reconstruction (in terms of being audibly detectable). The energy from these suppressed frequencies is added to the remaining bins, so that these frequency components increase in amplitude. These results are entirely consistent with the earlier discussion of the SBS in Section 6.4.1.

6.5.1 White Noise

When white noise is added to the vowel (Figure 6.19), it becomes difficult to visually detect the dominating frequency bins, except for the first set of high amplitudes for bin number 8. This is particularly true of the low Signal to Noise ratio wave forms, as per the upper plot of Figure 6.19 (which has a S/N value of 6 dB, compared to the lower plot which has a S/N value of 40 dB).

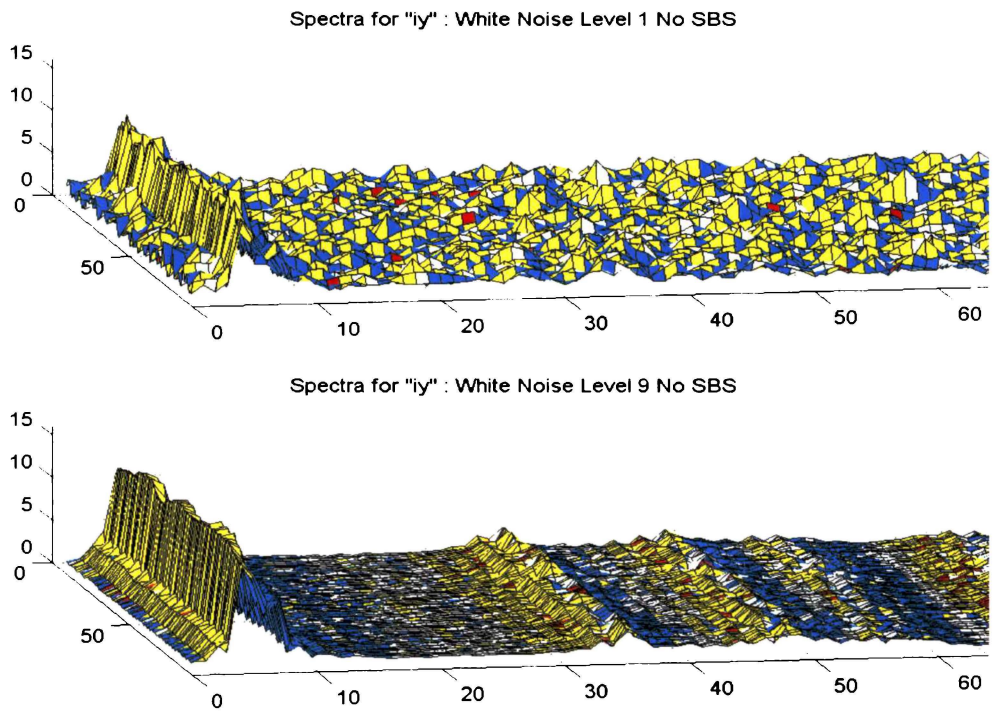


Figure 6.19 : Frequency Spectra Over Time for Vowel "IY" with Two Levels of Added White Noise

SBS processing endeavours to resolve the speech frequencies whilst rejecting the white noise. This is illustrated in Figure 6.20, for a high level of added white noise (S/N of 6 dB), Figure 6.21, for a moderate addition of white noise (S/N of 20 dB), and also Figure 6.22, which of the three, has the highest signal-to-white noise ratio (40 dB). These series of plots demonstrate how increasing the SBS threshold resolves the speech frequencies for each S/N white noise ratio.

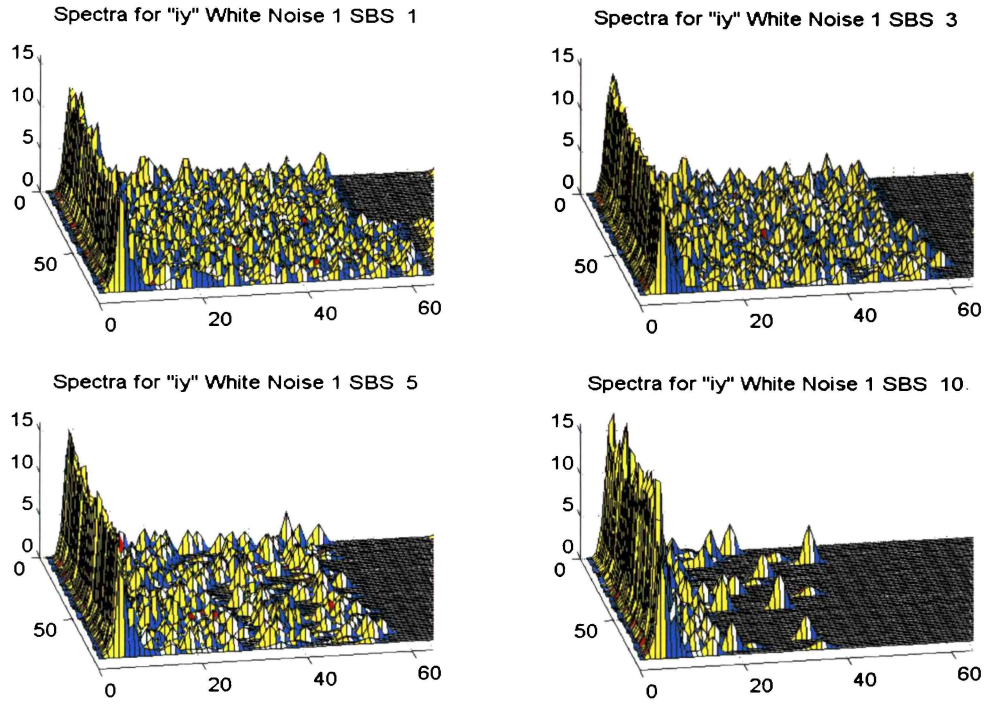


Figure 6.20 : Frequency Spectra Over Time for Vowel "IY" with High Added White Noise, and Varying SBS Thresholds

For all four SBS Thresholds, in the lower frequency bins, the original speech dominates the white noise, and is easily observed. For the low S/N of added white noise, only for an SBS Threshold of ten, does the second formant become visually apparent. Recognition has peaked well before this threshold, because such a high threshold value completely eliminates all the high frequencies as well as a significant portion of the *valid* mid-frequency range.

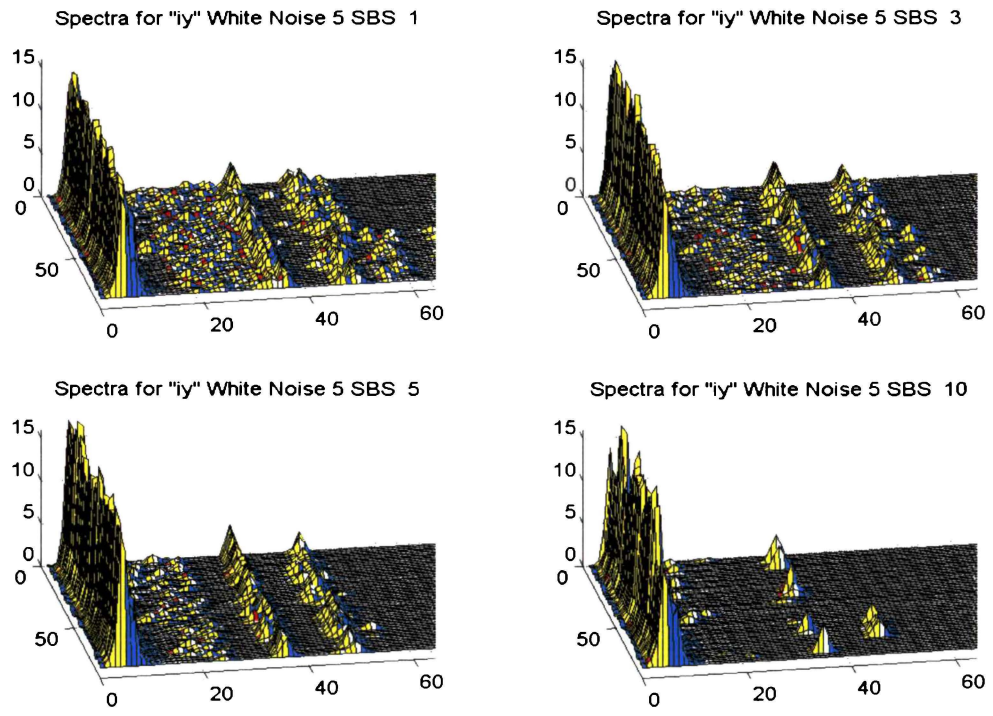


Figure 6.21 : Frequency Spectra Over Time for Vowel "IY" with Medium Added White Noise, and Varying SBS Thresholds

Even at the moderate S/N value of Figure 6.21, a visual inspection can begin to pick out the *frequency coherence* of the waveform once the SBS Threshold has reached the value of five. The term "frequency coherence" in this context refers to the waveform consistently exhibiting amplitudes of roughly equal value at the same bin numbers as time progresses. This is in contrast to amplitudes in one frequency bin at one time instant, but not there for the successive time increment (or frame number), i.e. amplitudes attributable to white noise. As the threshold is pushed even higher, to values of say seven and ten, this trend becomes more and more obvious, with the *non-coherent* frequencies being almost entirely suppressed.

For the highest signal-to-white noise ratio of Figure 6.22, by SBS five it is very difficult to visually distinguish the presence of the added white noise.

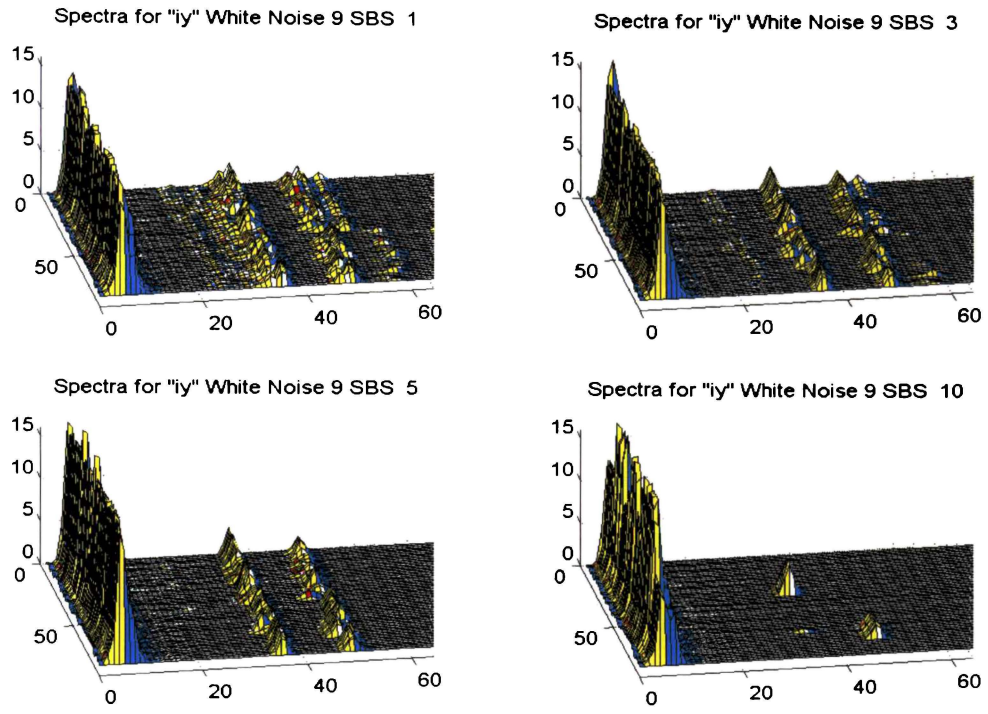


Figure 6.22 : Frequency Spectra Over Time for Vowel "IY" with Low Added White Noise, and Varying SBS Thresholds

Obviously, as the S/N is raised, the frequency coherence becomes evident at lower and lower SBS threshold values. To interpret this in view of the HTK results, refer to Figure 6.1. Even with an SBS Threshold setting of one, the recognition results are generally significantly greater than the "base" results, especially for the low signal to noise ratios. As the SBS threshold is further increased, the recognition results also increase, peaking at SBS six.

The conclusion from this is that because of its random nature, and the coherence inherent in the SBS algorithm, white noise is easily rejected by this form of processing. Increasing the SBS Threshold from zero to six increases the recognition score because of the ability of SBS to distinguish between the randomness of the noise and the structure of original speech. It is to be expected (and has been verified), that this form of processing would most suit strongly voiced speech, especially vowels. Because the threshold can be taken as high as six, and therefore eliminate a substantial amount of the added noise, it is expected that the white noise recognition scores would be superior to those of cocktail-party noise, where HTK results indicate a peak at SBS three. This is indeed the case.

6.5.2 Cocktail Party Noise

For cocktail party noise – the addition of another speaker, poses some problems when we are investigating a vowel in isolation. It makes no sense to add a sentence to this vowel, as that would greatly confuse the visual analysis. To allow an investigation of cocktail party noise for this isolated vowel, it was decided to add varying amplitudes of another isolated vowel, in this case "AH", whose spectra is illustrated in Figure 6.23.

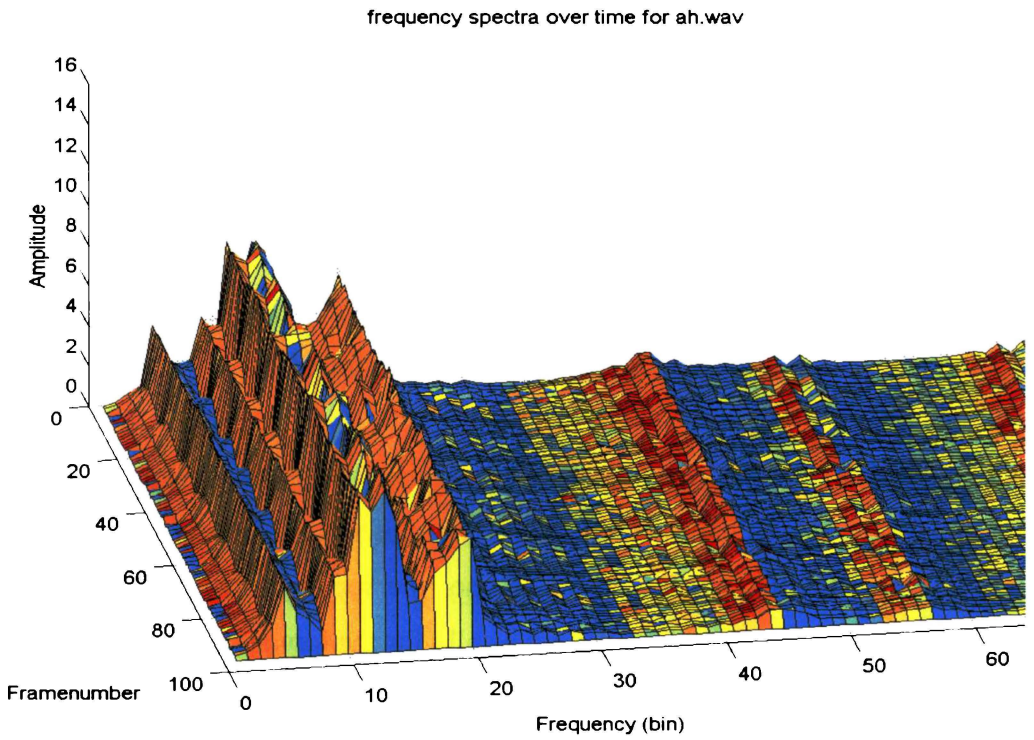


Figure 6.23 : Frequency Spectra Over Time for Vowel "AH"

This vowel possesses totally different formants to "IY", and the interaction between these formants yields a considerable insight into the functioning of the SBS processing. "AH" was added to "IY" at the nine signal to noise ratios explored in the HTK experiments, two of which are plotted in Figure 6.24 for Signal to Noise ratios of 6 dB and 40 dB for the upper and lower plots respectively. Upon inspection of these plots, one can observe the presence of frequencies corresponding to both vowel's formants.

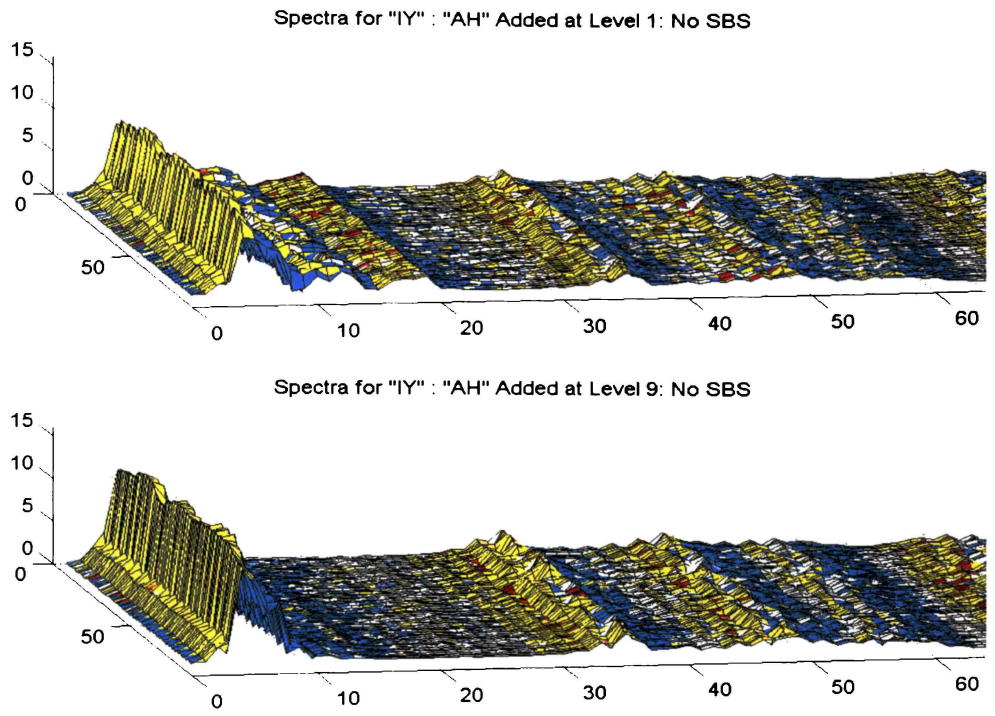


Figure 6.24 : Frequency Spectra Over Time for Vowel "IY" with Added Vowel "AH"

Similarly to the white noise case, SBS processing on three levels of this added noise is presented in Figure 6.25 for low S/N (6 dB), Figure 6.26 for moderate S/N (20 dB), and the lowest level of added cocktail party noise is presented in Figure 6.27 (40 dB S/N).

The interesting difference between this cocktail party noise, and the white noise considered earlier, is that even for the very high SBS Thresholds, the dominate frequencies of the added vowel are not attenuated at the S/N of Figure 6.25 – in fact they remain longer than the valid higher formants of the original vowel. Increasing the SBS Threshold past three does not eliminate these added noise components, but does eliminate some of the original "IY" frequencies.

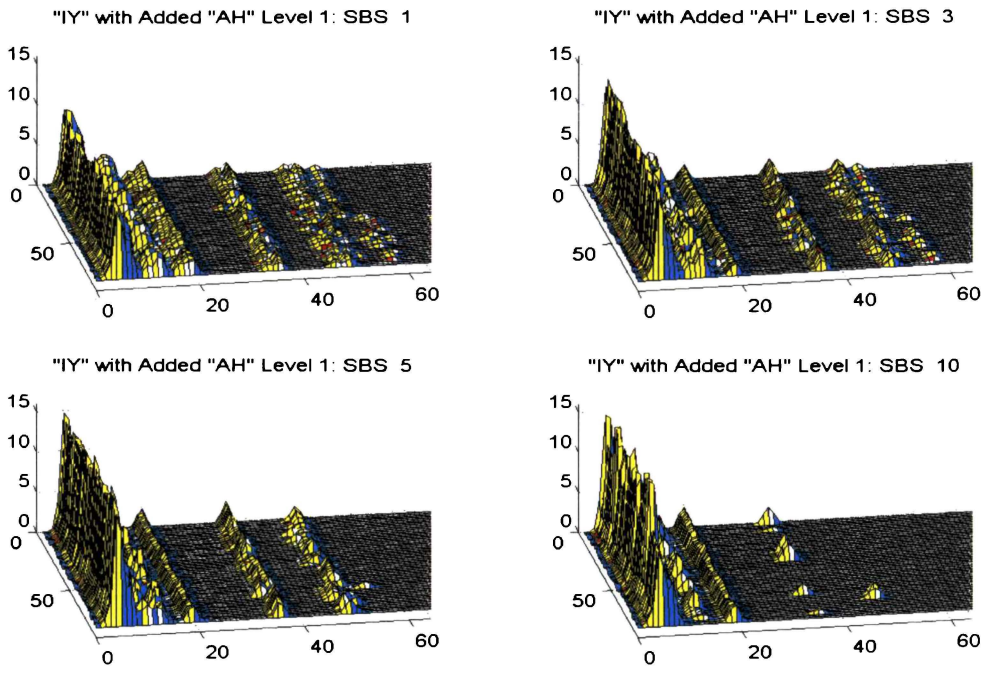


Figure 6.25 : Frequency Spectra Over Time for Vowel "IY" with High Added Vowel Noise, and Varying SBS Thresholds

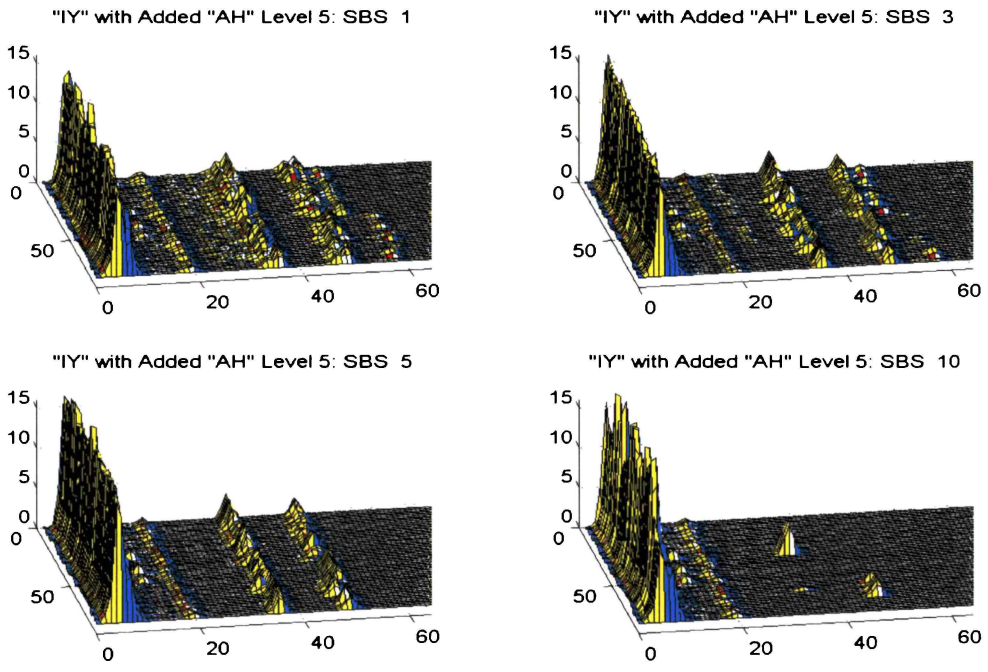


Figure 6.26 : Frequency Spectra Over Time for Vowel "IY" with Medium Added Vowel Noise, and Varying SBS Thresholds

Again, for a more moderate signal-to-noise ratio of Figure 6.26, the unwanted added vowel lower formant frequency remains for all SBS Thresholds, and past the threshold value of three, there is no real elimination of noise components.

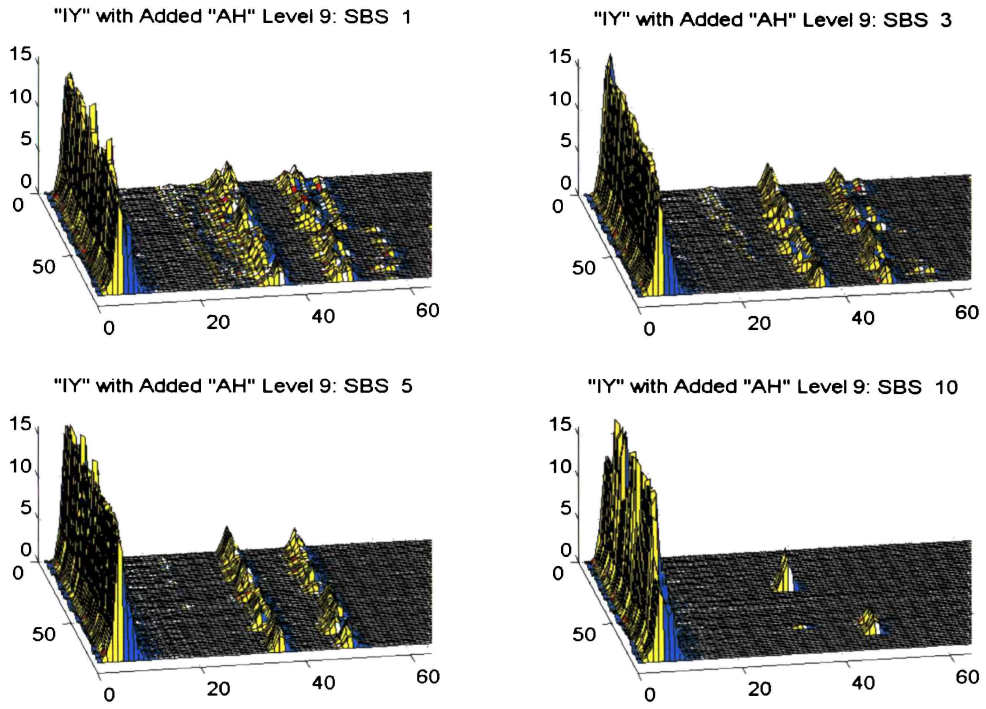


Figure 6.27 : Frequency Spectra Over Time for Vowel "IY" with Low Added Vowel Noise, and Varying SBS Thresholds

It is only at the highest Signal to Noise level (Figure 6.27) that the added vowel is not contributing components after SBS Threshold of three.

From this is can be seen that the addition of an isolated vowel is a worst case scenario for cocktail-party noise. The added frequencies exhibit strong frequency coherence, and, if added at sufficient amplitude (low S/N), can completely dominate the original vowel frequencies. Moderate to high SBS Thresholds serve only to eliminate the original frequencies, and hence it can be expected that a lower SBS Threshold of three (compared to six for white noise), would provide the higher recognition results. This is verified in the HTK experiments. Overall, cocktail party results score poorer in the HTK analysis than the white noise. As explained previously, this is to be expected as the lower SBS Thresholds do not eliminate as much of the noise as the mid to high values.

6.6 Sentence Analysis

The discussion so far has used a vowel to illustrate the noise suppression process A similar process occurs for the consonants and other phonetic features. The vowel was used in this instant as by its nature, it maintains a reasonably constant spectra over time, and permits easy visual inspection of the SBS frequency selection process.

To examine the spectra of actual speech, consider Figure 6.28, which contains the spectral plots of the TIMIT file SX85, "Continental drift is a geological theory".

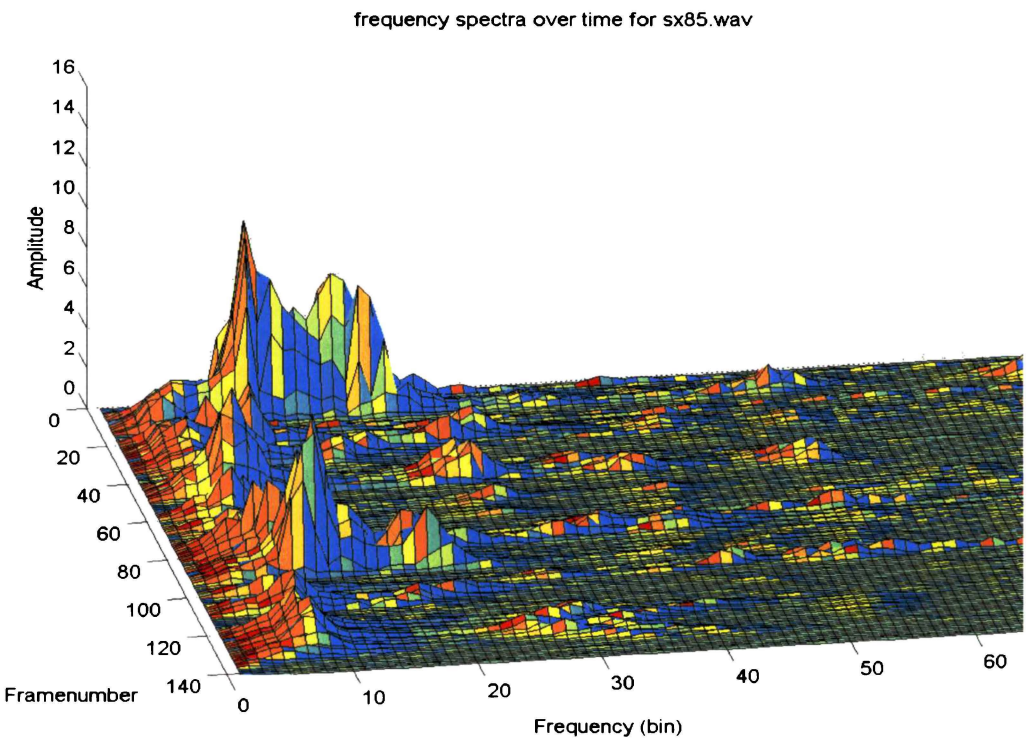


Figure 6.28 : Frequency Spectra Over Time for "SX85"

6.6.1 White Noise

White noise is added to this speech, at the nine signal-to-noise ratios stated throughout this thesis, and the lowest (6 dB), and highest (40 dB) levels are illustrated in the upper and lower plots (respectively) of Figure 6.29.

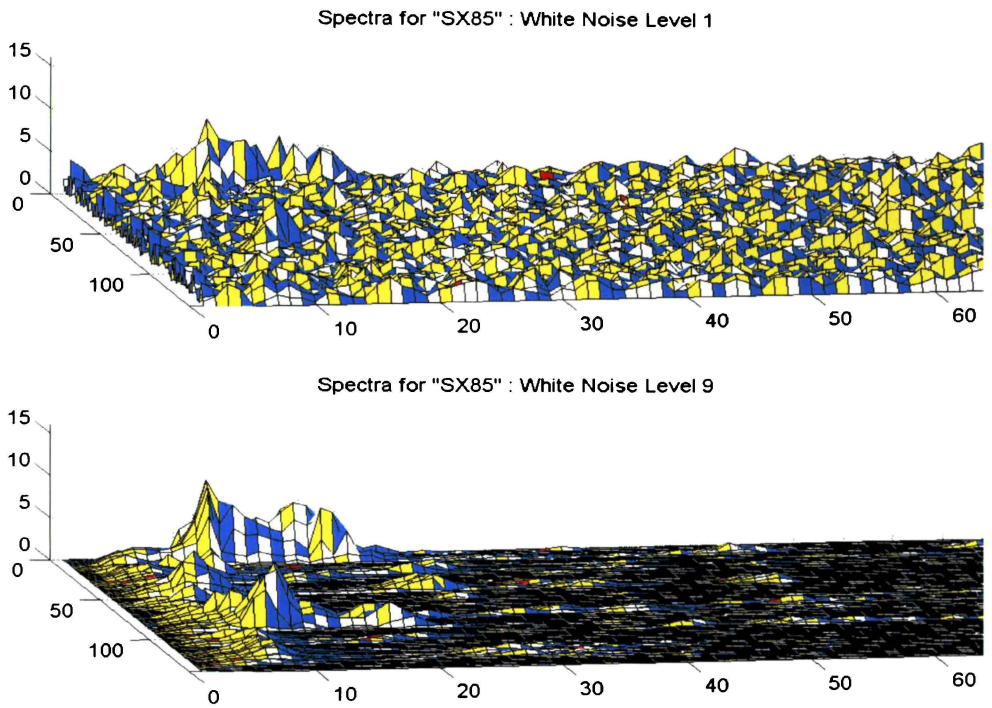


Figure 6.29 : Frequency Spectra Over Time for "SX85" for Two Values of SBS Threshold

SBS processing of this sentence corrupted with three levels of white noise are illustrated in Figure 6.30, Figure 6.31, and Figure 6.32 for low (6 dB), moderate (20 dB) and high (40 dB) S/N ratios respectively. For such a high level of added noise, it is extremely difficult to visually identify anything other than the most prominent features of the original sentence spectra (Figure 6.28), and hence the justification for using an isolated vowel for analysis purposes. Even for the very high values of SBS Threshold, the original sentence features are not obvious, though HTK recognition has peaked well before this point.

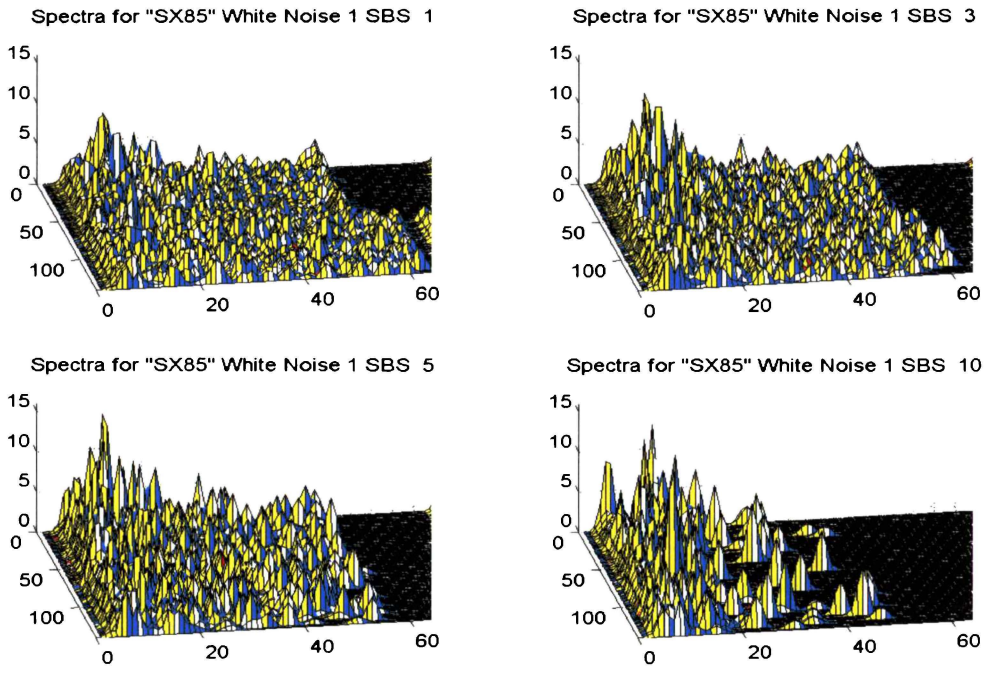


Figure 6.30 : Frequency Spectra Over Time for "SX85" with High Added White Noise, and Varying SBS Thresholds

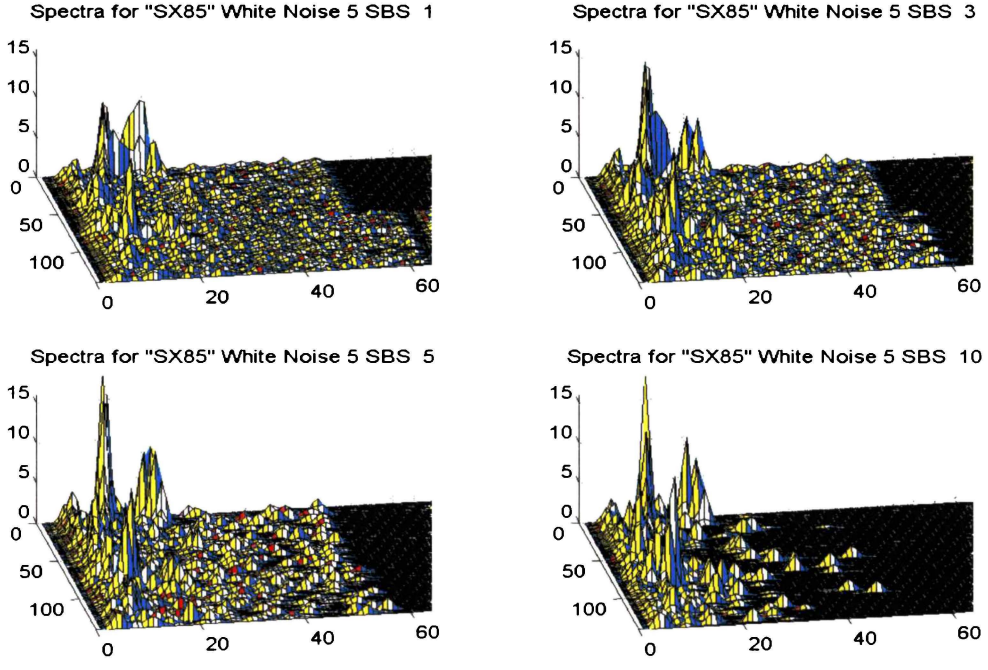


Figure 6.31 : Frequency Spectra Over Time for "SX85" with Medium Added White Noise, and Varying SBS Thresholds

For the more moderate added noise case, the SBS process becomes visually clearer. At SBS one, a multitude of the white noise frequency components are still evident. They are somewhat attenuated at SBS three, but it is at SBS five where a significant reduction in the white noise spectra can be observed. Further increasing the threshold to ten, begins to eliminate some of the original sentence frequencies, and would be expected to decrease a recognition score.

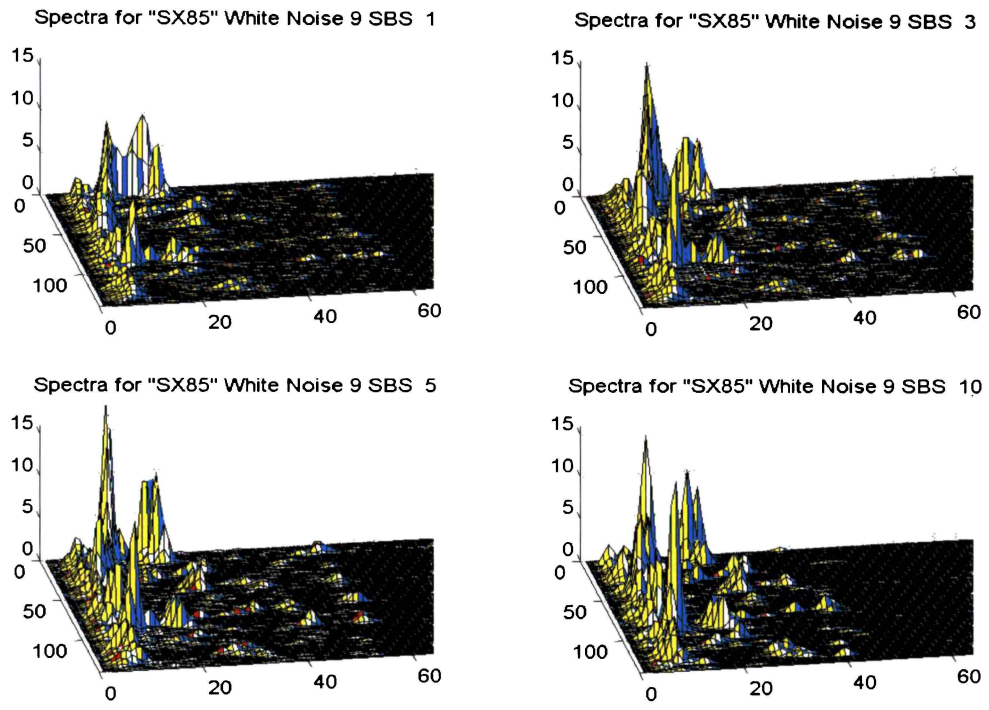


Figure 6.32 : Frequency Spectra Over Time for "SX85" with Low Added White Noise, and Varying SBS Thresholds

The process of SBS frequency elimination is perhaps more evident in Figure 6.32 (for the highest Signal to Noise situation) than in the preceding plots. By SBS five, most of the small amplitude, white noise components have been eliminated, and their energy transferred to the other valid (originating from the original speech) frequency bins. SBS ten again eliminates some of the original sentence frequencies.

An analysis of the spectra of cocktail party noise in this manner would not yield much in the way of additional understanding of the SBS process given that the cocktail party noise, by its nature, presents a confusing visual spectra.

To conclude, for the analysis of the white-noise corrupted, sentence spectra, processed with varying SBS Thresholds, a visual inspection (possible for the mid and high S/N ratios), indicates that at an SBS Threshold of approximately five, the spectra begins to closely resemble that of the original sentence, most of the white noise contribution having been eliminated. Lower thresholds do not eliminate these white noise components, higher thresholds eliminate valid original frequencies as well as the white noise. It is not surprising then, that the HTK results peak for an SBS Threshold of six (with five being very close), for these white noise situations.

6.7 Summary

6.7.1 Processing Parameters

6.7.1.1 SBS Threshold

HTK results indicate that recognition peaks for speech in the presence of white noise at an SBS Threshold of six. Visual inspection of both vowel and sentence spectra demonstrates that at this threshold (five in the actual plots, but with an almost identical effect to six), most of the white noise components have been eliminated, yet the spectra still retains the essential features of the original speech.

For speech corrupted by cocktail party noise, some of the added frequencies can have large amplitudes, and dominate the spectra for several frames. In such cases, the SBS Thresholds of five or six, can incorrectly select one of these added frequencies, and delete valid, desired frequency bins. For this form of added noise, recognition peaks at SBS three, a value that rejects much of the added signal, but not at the expense of the original waveform. As a result of the lower SBS Threshold (and hence the greater portion of noise in the reconstructed speech), the cocktail party noise recognition scores are lower than the white noise scores.

6.7.1.2 Amplitude, Phase, Frequency Bin Quantisation Levels

The two extreme quantisation levels (six bit logarithmic amplitude and frequency, eight bit phase, compared to three bit log amplitude and frequency, zero phase), produced recognition scores generally only a few percentage points apart. In situations near optimum recognition (SBS six for white noise, SBS three for Cocktail party), the lower bit level decreased recognition. In these cases, the addition of one bit phase increased the recognition score, and brought the curve near the eight bit phase line. The addition of another phase bit (to a total of two), did not produce a significant difference from the one bit trial.

It seems reasonable to finalise a quantisation scheme of three bit logarithmic amplitude and frequency bin, one bit phase, with the expectation that the decrease in recognition will be very small.

6.7.1.3 Data Overlap

Decreasing the data overlap from 100% down to 25% results in a slight decrease in recognition for cocktail party noise, and a slight increase for the lower S/N levels of added white noise. Reducing the data overlap has the significant advantage of lowering the overall bit rate, and so some degradation in speech recognition can be tolerated in view of the savings that can result. It is interesting to note that there was not a significant change in recognition scores between the 25% level and the 50% level. In view of the bit rate advantages, this 25% level will be further considered.

6.7.1.4 Differencing

Differencing follows a similar trend to the data overlap in that it seems to be beneficial for low Signal to White Noise ratios, but degrades all cocktail party noise. Given that the Differencing Algorithm also scored poorly on the previous intelligibility listening tests, no further investigation will be undertaken using Differencing techniques, and they will not form part of the final algorithm.

6.7.1.5 Unvoiced Modifier

The inclusion of the unvoiced modifier shifts the curves approximately two SBS Threshold values, i.e. SBS five moves towards SBS three, and the resultant curve falls mid-way between these two integral values. This modifier does not maximise HTK recognition, rather it forms a compromise or hybrid result. It appears to be more important for intelligibility purposes rather than recognition.

6.7.2 Conclusion

This chapter has produced a further elimination of some of the processing variables to determine a final, best form of low-bit rate coding scheme. SBS levels of three and six perform best for cocktail party and white noise respectively, there is a tolerable decrease in recognition for the implementation of three bit logarithmic frequency/amplitude – one bit phase coding, data overlap can be as low as 25%, no differencing algorithms are required, and the effect of the unvoiced modifier is to produce a recognition curve that is intermediate to the integral SBS curves and need not be further considered in the HTK experiments.

However, only Ghitza's SBS algorithm has been considered so far, and it needs to be compared to other processing techniques. The next chapter will evaluate (using the HTK tests) the performance of speech processed under different Seneff processing conditions, as well as an LPC10 algorithm, and compare it to the results obtained in these sections.

Chapter 7

7 Comparing HTK Results: SBS, Seneff and LPC

7.1 Seneff Implementation

Seneff's model is sufficiently internationally respected and utilised that the implementation of Stages I and II are contained in "The Auditory Toolbox", Version 2 developed by Malcolm Slaney (http://web.interval.com/papers/1998-010/). Some alterations were required to his code, specifically in altering the sampling rate and filter spacing, and modifying the outputs from Stage II to feed the synchrony and mean-rate detectors. The modified filter shapes are illustrated in Figure 7.1, and are of the form indicated by Seneff. Since the filter design, and the Stage II parameters were already implemented in Slaney's toolbox, the details are not reproduced here, and the reader is referred to either Slaney's web site, or the original Seneff reference.

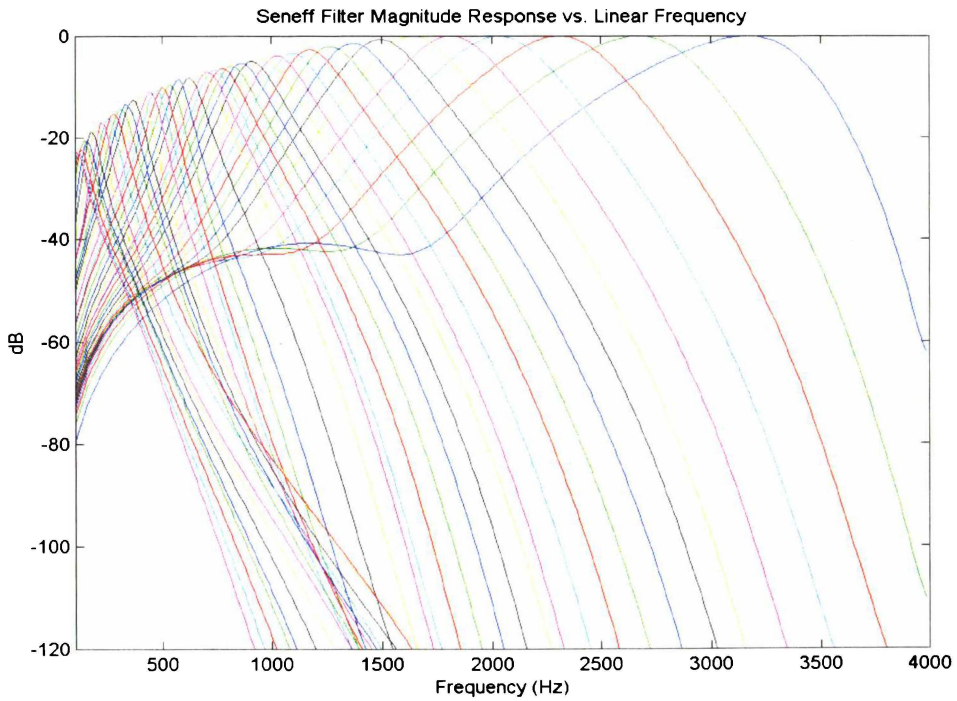


Figure 7.1 : Seneff Filter Magnitude Response

The Stage II output for the vowel IY is presented in Figure 7.2. This illustrates the output of the 40 channels, the lower frequency channel is at the top of the plot, the higher frequencies at the bottom (as per Seneff's (1988) convention). Each channel has been separated by an arbitrary vertical distance for display purposes.

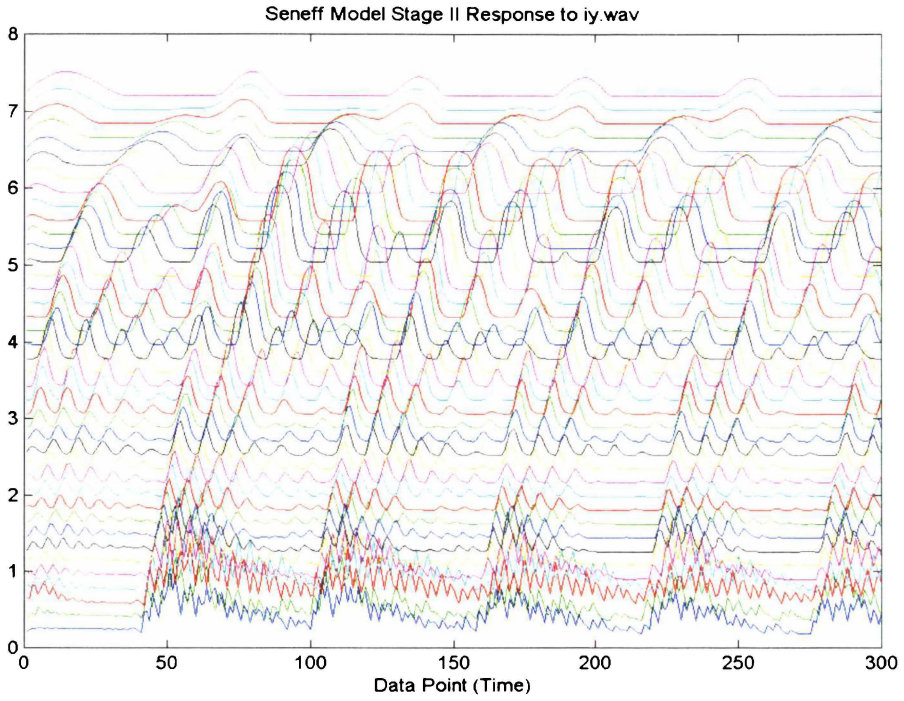


Figure 7.2 : Seneff Stage II Output for "IY"

The response of the synchrony detector to the vowel IY is shown in Figure 7.3. Due to the structure of the Seneff channels, the low frequencies appear to the right of the plot, i.e. the lowest frequency is channel 40. Converting these bin numbers to absolute frequencies (using Figure 7.1) yields the formant structure expected of this vowel (Section 6.6).

7.2 The Seneff Recognition Experiments

The Seneff experiments were performed in three parts, that vary according to the interpolation of the auditory nerve firing patterns. The first set of six results is obtained by taking a frame size of 256 points, and selecting the key frequencies (Section 3.3). The second set of results uses a frame size of 128 points, and the final set employs a frame size of 128 points, sliding 64 points for each frame (similar to that used for the default SBS

processing). Each set of results comprises six variations of the number of frequency components used in the reconstruction, and the quantisation levels of frequency, amplitude and phase. The three parts are arbitrarily labelled "2" (for **2**56 points), "*8*" (for 12**8** points), and "*f*" (for **f**ull data redundancy), and are summarised in Table 7.1.

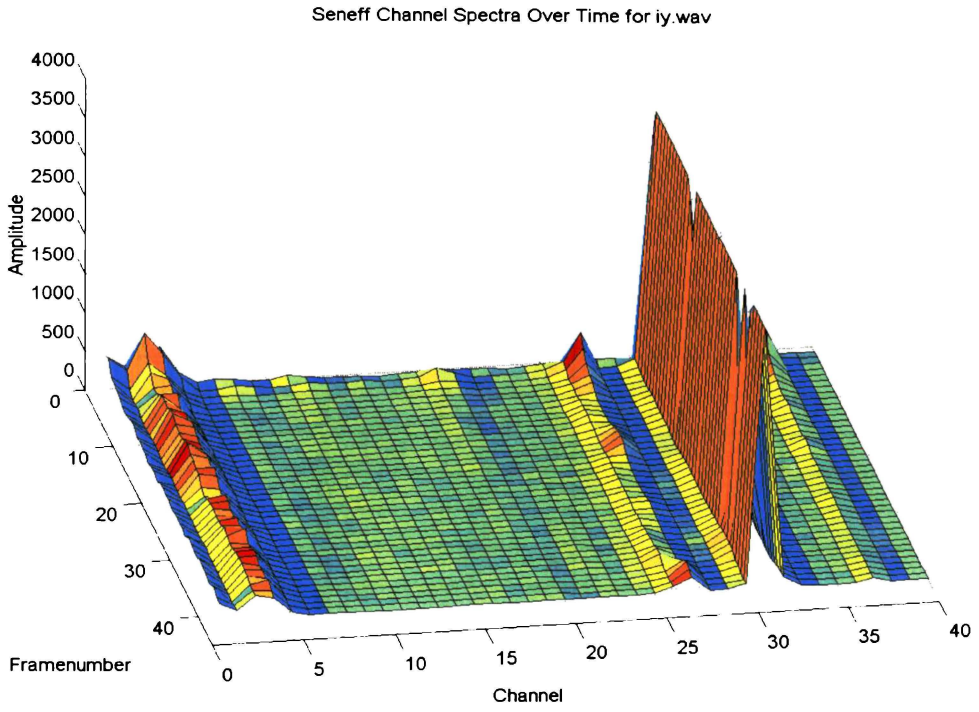


Figure 7.3 : Seneff Stage III Synchrony Output for "IY"

Each Seneff experiment takes about five times the CPU time of an SBS run, due to the computational complexity of the Seneff filters. As a result, the Seneff runs were limited to six variations of frequency number and quantisation levels. The "2" run was the initial run performed to minimise file size (3696 training and 5568 test files take up a significant amount of disk space). The sliding frame for the "*f*" run produces Seneff psd (Power Spectral Density) files that are four times the size of the 256 point "2" run, and each run requires well over 1 GB HDD capacity. A desktop PC is not capable of performing the required computation for a single Seneff run in reasonable time (approximately one month for a PII 400 MHz). DEC Alphas eventually had to be employed for this task, and even so, several thousand hours of CPU time were eventually required.

As indicated in Table 7.1, only the extreme quantisation levels (as discussed in the previous chapter) are employed. The justification for this is that the SBS processing provided considerable information concerning the relationship of the quantisation schemes to these two extremes, and given the considerable CPU time required for the Seneff filters, it is acceptable to interpolate the other quantisation cases from these runs. Additionally, only indicative values of the number of frequencies selected for reconstruction are tested, the SBS trends can be used to interpolate for the other cases. Note that the quantisation routines are constant between the algorithms, it is only the frequencies selected that varies between the Seneff and SBS processes.

Run-type & label	Number of Frequencies	Freq/amp Quantisation	Phase Quantisation
S2a	3	6	8
S2b	5	6	8
S2c	7	6	8
S2d	10	6	8
S2e	3	3	0
S2f	5	3	0
S8a	3	6	8
S8b	5	6	8
S8c	7	6	8
S8d	10	6	8
S8e	3	3	0
S8f	5	3	0
Sfa	3	6	8
Sfb	5	6	8
Sfc	7	6	8
Sfd	10	6	8
Sfe	3	3	0
Sff	5	3	0

Table 7.1 : Variation of Seneff Quantisation Parameters

For the Seneff files, frequency, amplitude and phase quantisation comparison involves the consideration of *SXa* vs *SXe* and *SXb* vs *SXf* (where "*X*" means any of the "*2*", "*8*", or "*f*" runs). The effect of altering the number of frequencies used to reconstruct the speech can be seen from viewing runs *SXa*, *SXb*, *SXc*, *SXd* and also *SXe* vs *SXf*.

The effects of the different Seneff frame sizes (and therefore frame interpolation) requires the comparison of any of the experiments a → f, for the different "*2*", "*8*" or "*f*" runs, for example *S2a* vs *S8a* vs *Sfa*.

7.3 Comparison of the Three Seneff Runs

A decision needs to be made concerning which frame method ("*2*", "*8*", or "*f*") produces the best overall recognition results. To determine this, the sums of the squares of the differences between perfect recognition and the actual recognition score are computed. For each form of Seneff frame interpolation (three types), for each quantisation variation (six types), for each form of added noise (four types), and for each of the signal to noise ratios (five or nine depending upon noise type), the recognition score is subtracted from 100%, squared, and added together. This is expressed by Equation 7.1 for white noise and one added speaker. For multiple added speakers, the sum limit of nine would be replaced with a limit of five. The factor of 10E-04 in Equation 7.1 takes into account that the figures are percentages.

$$\text{Total Squared Error} = \frac{1}{100^2}\sum_{1}^{9}(100 - \text{Recognition Score}_i)^2 \qquad \textbf{Equation 7.1}$$

7.3.1 White Noise

The result of applying Equation 7.1 to the white noise results, is illustrated below in Figure 7.4. The solid lines indicate the total squared error for all nine signal to noise ratios To determine whether there is a difference in performance depending upon the level of the Signal to Noise ratio, dotted lines indicate the error for lowest four ratios, the dotted asterisk lines for the higher five S/N ratios.

As can be seen, most of the error occurs (as expected) for the lower Signal to Noise ratios. For this instance of white noise, the *f*-run is overall is marginally better than the *2*-run, and considerably better than the *8*-run. The *f*-run will therefore be used in the following discussion concerning the Seneff response to speech corrupted by white noise.

A plot of the *f*-run Seneff recognition results for speech in the presence of white noise is provided in Figure 7.5. The results are approximately grouped into three pairs, determined by the number of frequency components used in the reconstruction. Consistent with the SBS results, the variation of frequency/amplitude/phase quantisation has a much smaller affect on recognition than the variation in the number of frequencies.

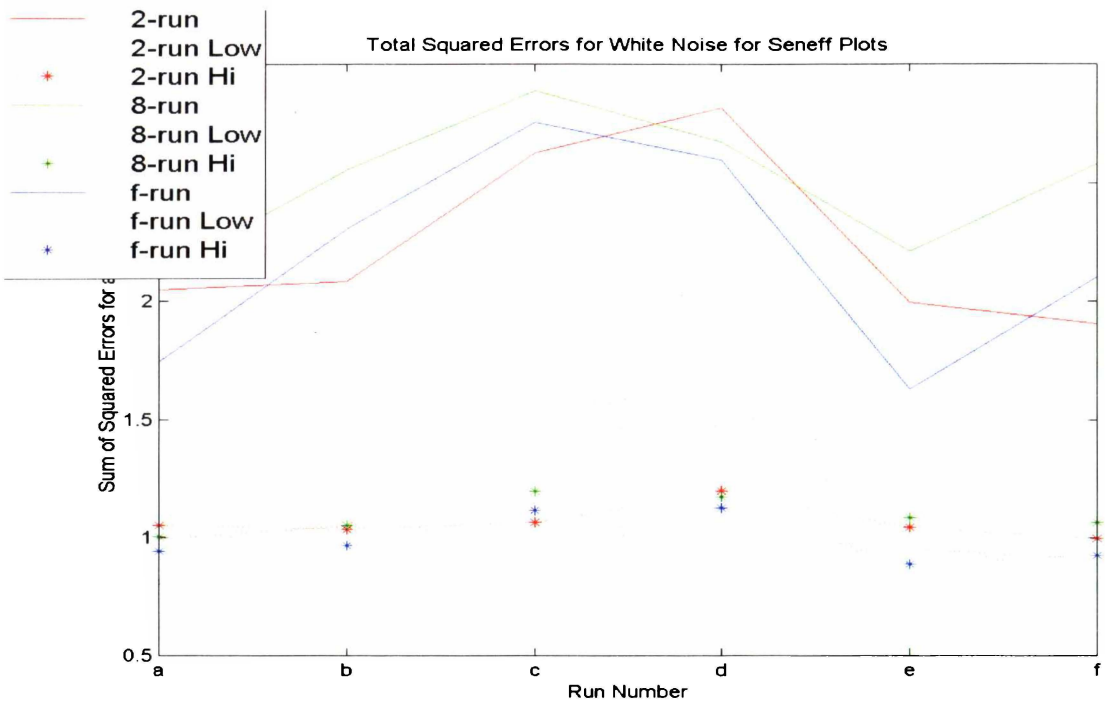


Figure 7.4 : Sum of Squared Errors for Three Seneff Runs in Presence of White Noise

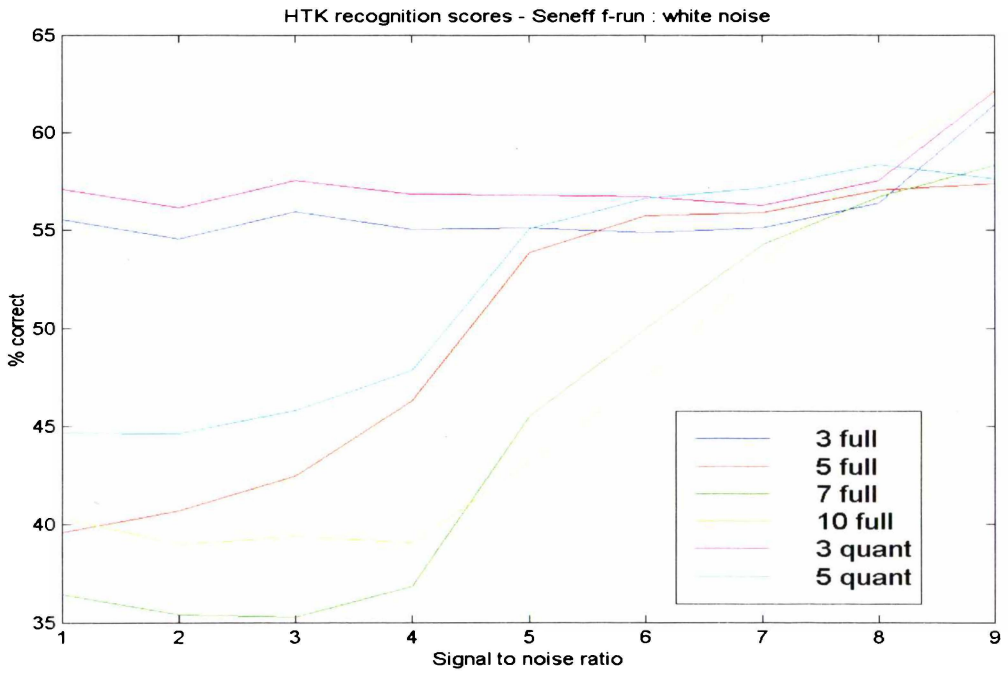


Figure 7.5 : Seneff *f*-run Recognition Results in the Presence of White Noise

In the presence of white noise then, three frequencies in the reconstruction provides the highest recognition score, followed by five, then seven and ten, with very little separation between these last two.

7.3.2 One Additional Speaker

Repeating the squared error analysis as discussed in 7.3.1, for the case of one added speaker, yields the results illustrated in Figure 7.6. Again the *f*-run has the lowest overall squared error (in other words, the best recognition results), so is again the test run used to analyse the response of the Seneff algorithms to noise of the form of one added speaker. Unlike the white noise experiments, this form of cocktail party noise seems to be evenly distributed amongst the nine signal to noise ratios. This is reflected in the fact that recognition does not dramatically increase as the cocktail party noise level is lowered.

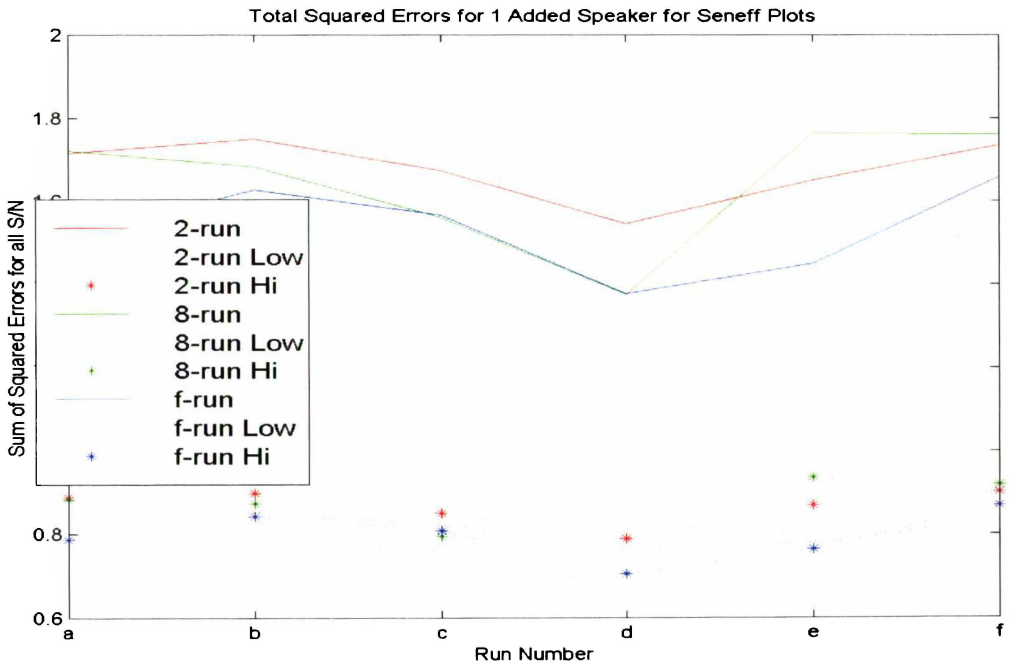


Figure 7.6 : Sum of Squared Errors for Three Seneff Runs in Presence of One Added Speaker

The *f*-run recognition results are illustrated below in Figure 7.7, and indicate only a four to five percentage point increase as the cocktail party Signal to Noise ratio is varied. Of interest is that recognition peaks for ten reconstructed frequencies, though the three frequency result is very close.

There is the same close grouping of the different quantisation levels of the three and five frequencies runs, the recognition scores often being within one percentage point of each other.

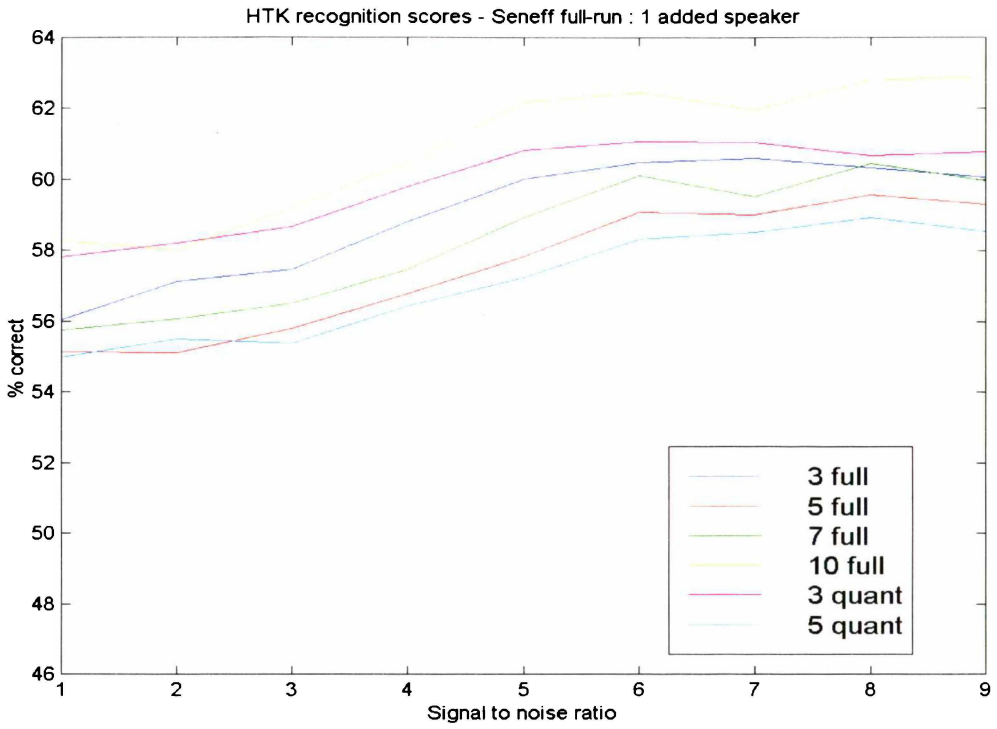


Figure 7.7 : Seneff *f*-run Recognition Results in the Presence of One Added Speaker

7.3.3 Multiple Added Speakers

For two and four added speakers, only the total squared error response is illustrated in Figure 7.8, as only five Signal to Noise ratios are used for these forms of cocktail party noise. Note that as a result of the reduced number of S/N values, the ordinate axis spans a different range than for the white and one added speaker noise case.

For the multiple added speakers, it is again the *f*-run that has the best recognition results, and so it is that run once again that is presented in Figure 7.9 and Figure 7.10 for the noise analysis.

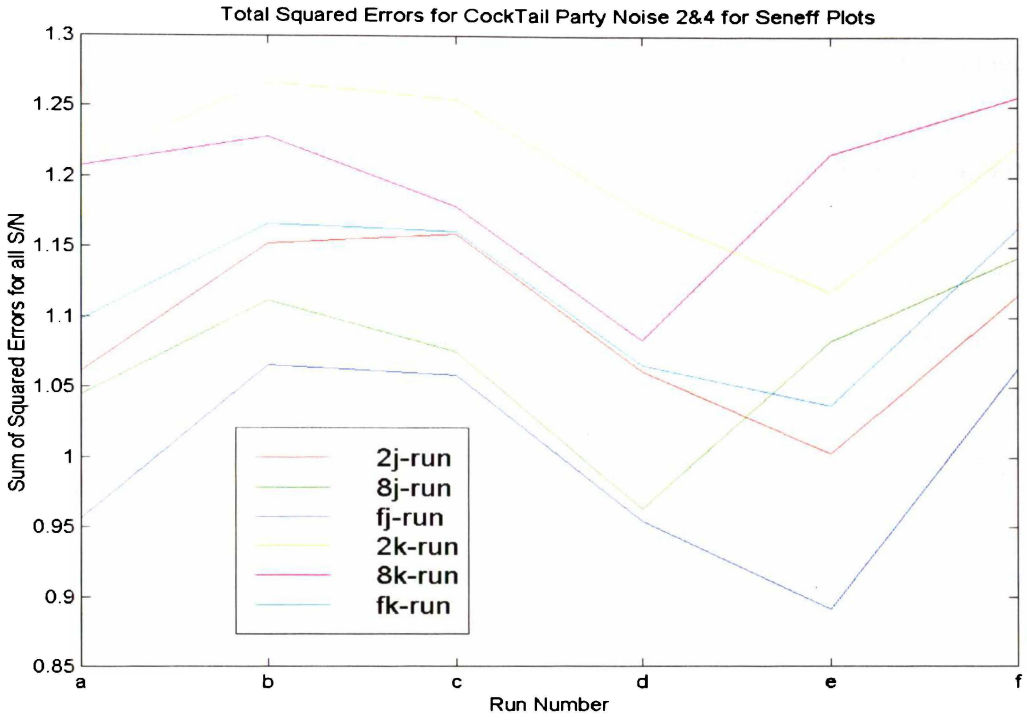


Figure 7.8 : Sum of Squared Errors for Three Seneff Runs in Presence of Two and Four Added Speakers

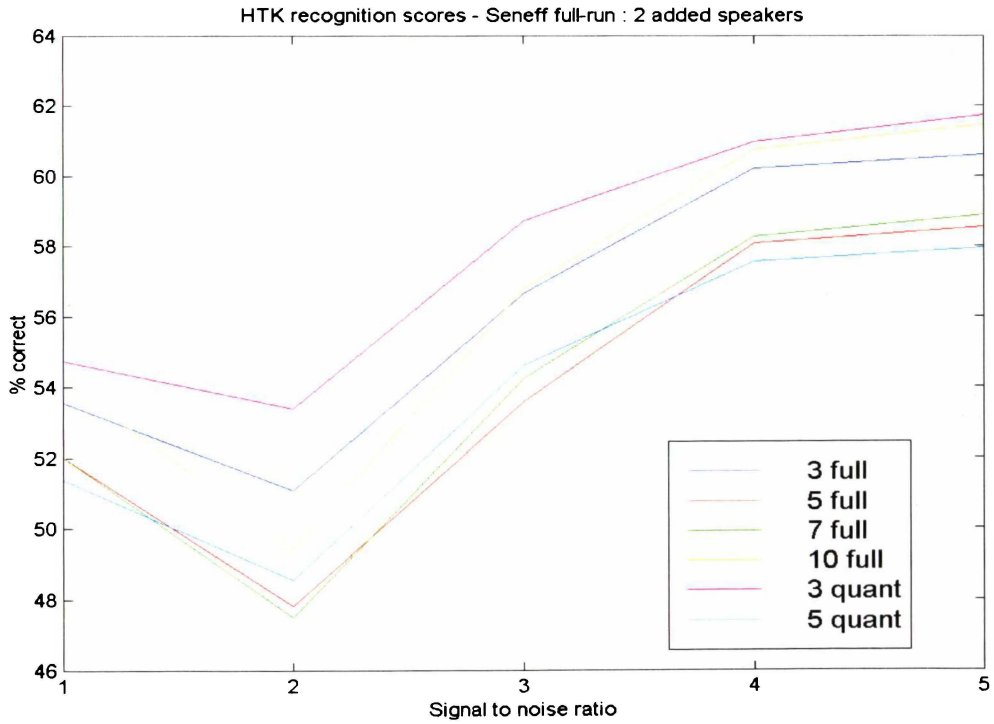


Figure 7.9 : Seneff *f*-run Recognition Results in the Presence of Two Added Speakers

Initial examination of Figure 7.9 shows that there is very little difference between the two five frequency runs and the seven frequency run (they are seldom separated by more than one percentage point). Again the three frequency components provide good recognition scores, but as per the one speaker case, so too does the ten frequency response.

Figure 7.10 illustrates a nearly identical trend to the previous plot. For both instances of multiple speakers, the entire range of results between the best and worst performing runs is only three to four percentage points. The two five frequency runs are extremely close, the three frequency results are somewhat more separated, but mirroring each other closely.

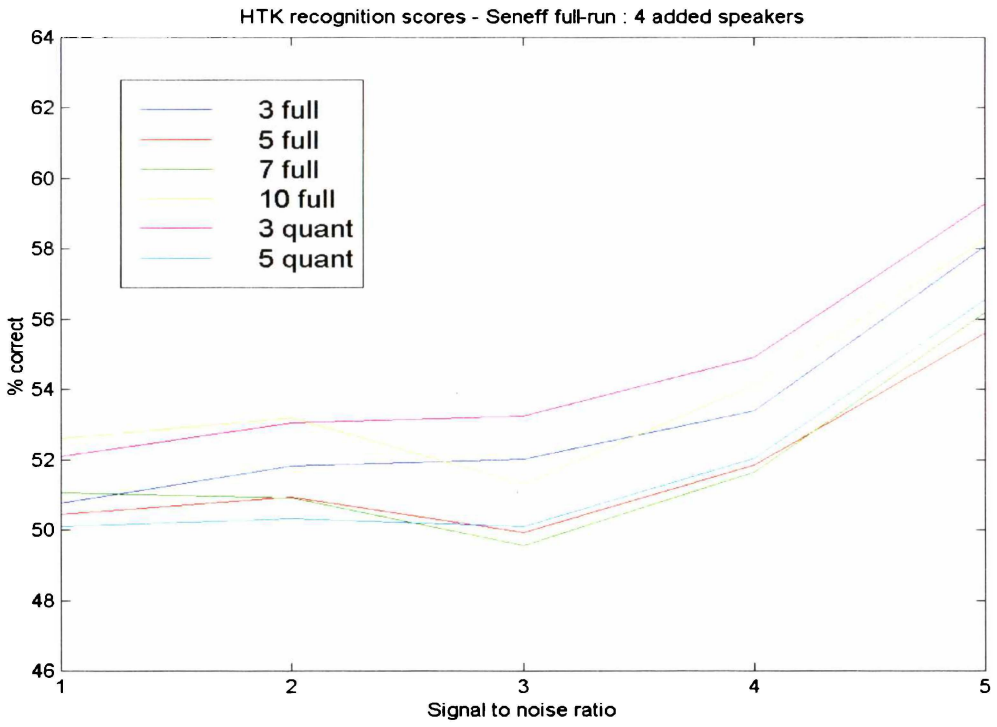


Figure 7.10 : Seneff *f*-run Recognition Results in the Presence of Four Added Speakers

7.3.4 Summary

For each form of added noise, the *f*-run produces the best recognition results. It appears that the 256 point frame is too coarse a frame division, and components of the speech that are important to HTK recognition are not reliably included at this frame size. The 128 point sliding frame produces a more accurate analysis, and has the added advantage of being the

same as that adopted for the Ghitza SBS processing, enabling direct comparison of the Seneff and SBS results.

The Seneff recognition scores are low, seldom reaching 60%, and then only for the lowest levels of added noise. Adding more frequencies (up to the simulation's maximum of ten), did not in general improve recognition. The conclusion from this is that the Seneff process is not amenable to the production of a reduced frequency representation of speech.

7.4 Comparing Seneff and SBS Results

A comparison may be made between Seneff and SBS processing by noting that an SBS threshold of ten corresponds to (on average) three frequencies used in the speech reconstruction, SBS six results in five frequencies, SBS three provides seven frequencies, and SBS one provides about ten frequencies. The equivalence between the Seneff runs and SBS runs (Table 6.1) is presented in Table 7.2.

Seneff Label	SBS Equivalent
SXa	G
SXb	Q
SXc	B
SXd	A
SXe	H
SXf	R

Table 7.2 : Equivalence of Seneff and SBS Experiments

A comparison between the Seneff and SBS results for each of the four forms of added noise are considered in the following sections.

7.4.1 White Noise

For clarity, the six divisions presented in Table 7.2 are plotted over two graphs. Figure 7.10 illustrates the four cases of six bit log amplitude/frequency eight bit phase coding and Figure 7.11 for the three bit log amplitude/frequency – no phase cases. Comments will be made in a later section concerning the overall magnitude of the results, but at this stage, what is of interest is the comparison between the full lines representing SBS coding, and the dotted lines obtained from the Seneff process.

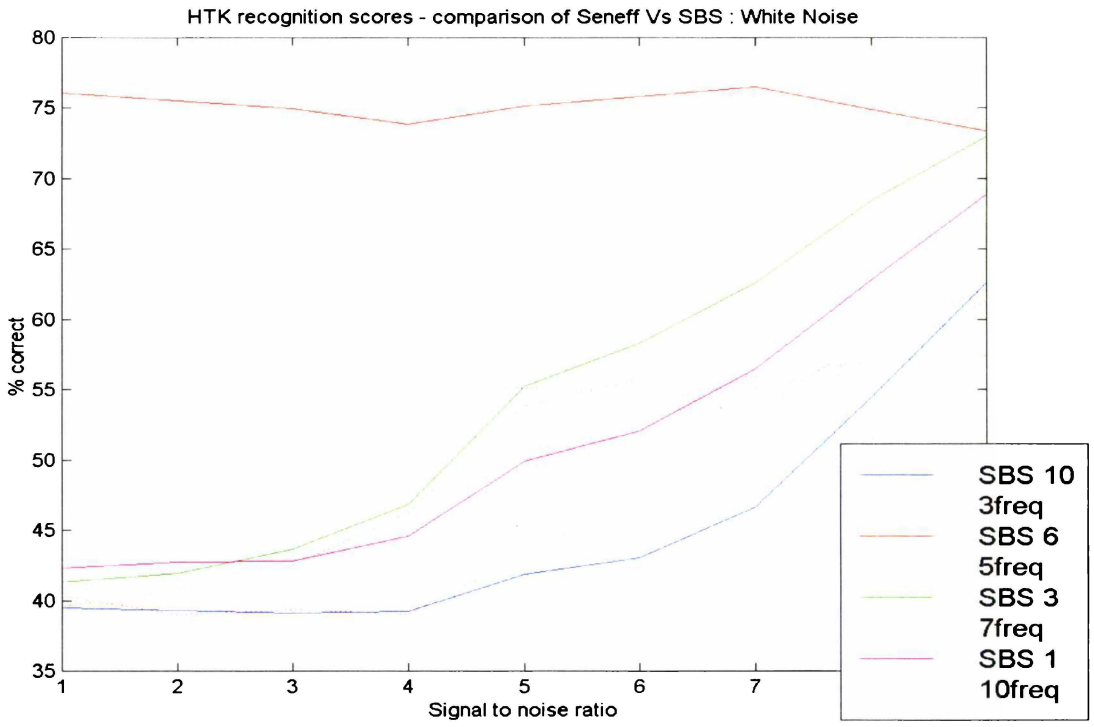


Figure 7.11 : Comparison of SBS and Seneff *f*-run Results for White Noise

From Figure 7.11, it is evident, that only for the case of SBS Threshold ten and three Seneff frequencies does the Seneff result better the SBS case. For the other three threshold values, particularly that for SBS six (the peak white noise SBS response), the SBS processing is considerably superior to the Seneff.

Figure 7.12 repeats this trend, with the three frequencies selected by the Seneff results outperforming the SBS equivalent threshold of ten, but SBS Threshold six is considerably more recognisable than Seneff five frequencies.

These results indicate that, in general, the frequencies selected by the Seneff algorithm, do not distinguish between white noise and valid speech as successfully as does the SBS process (except for the extreme value of SBS ten).

Figure 7.12 : Comparison of SBS and Seneff *f*-run Results for Quantised White Noise

7.4.2 Cocktail Party Noise – One Added Speaker

The plots illustrating the comparison between the Seneff and SBS process for speech corrupted by cocktail party noise in the form of one added speaker are shown below. Figure 7.13 shows a similar trend to those for white noise in that the SBS processing generally outperforms the Seneff equivalent. For the case of SBS Threshold ten compared to Seneff three frequencies, the relationship is closer than for the other thresholds, but in this instance, the SBS process produces the superior recognition results.

The view of the more heavily quantised cases (Figure 7.14), is again similar, a very large improvement for the SBS over the Seneff for five reconstructed frequencies, and a smaller, but still superior SBS performance for the three reconstructed frequencies.

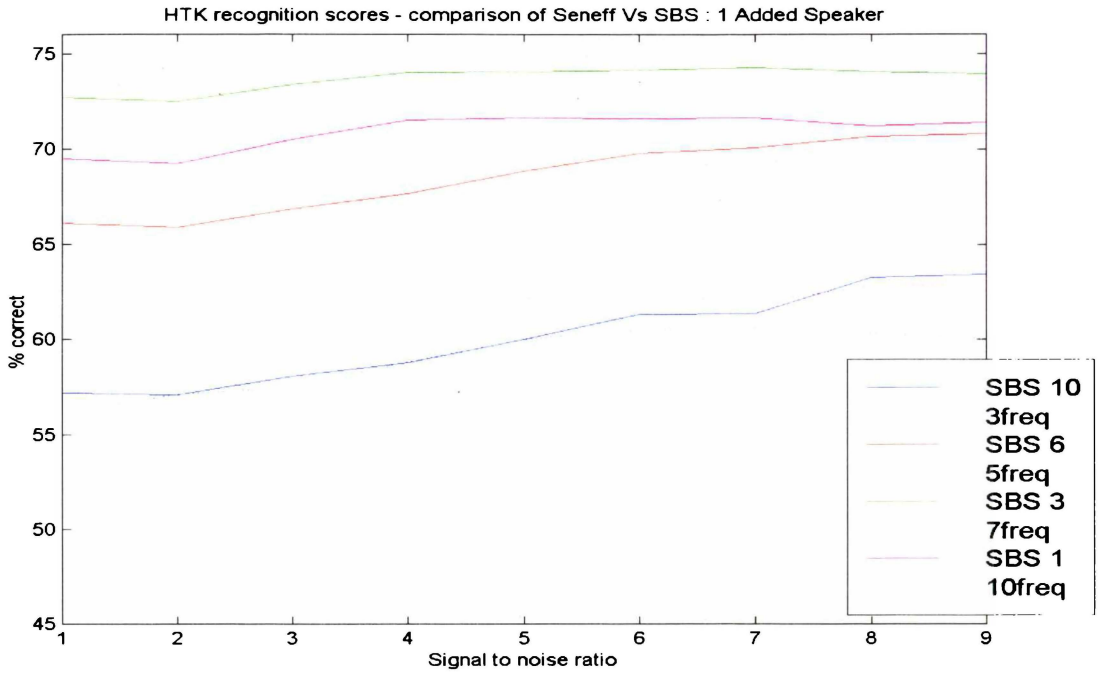


Figure 7.13 : Comparison of SBS and Seneff *f*-run Results for One Added Speaker

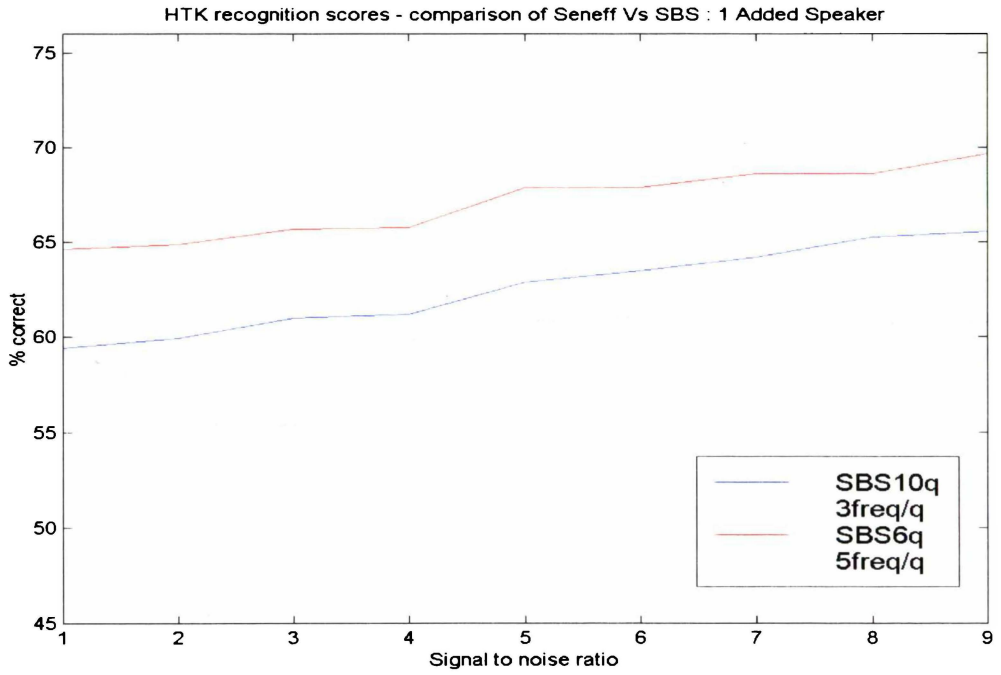


Figure 7.14 : Comparison of SBS and Seneff *f*-run Results for Quantised One Added Speaker

The closeness of the Seneff curves indicate that increasing the number of frequency components used in the speech reconstruction does little to enhance recognition. In effect, frequency selection by the Seneff process is not distinguishing between original speech and the added speech. The large difference between the SBS and Seneff plots demonstrate that the former is far more effective at eliminating this form of cocktail party noise.

7.4.3 Cocktail Party Noise – Two Added Speakers

The ordinate axis for these multiple added speaker results has been kept the same as for Figure 7.14 to aid comparison.

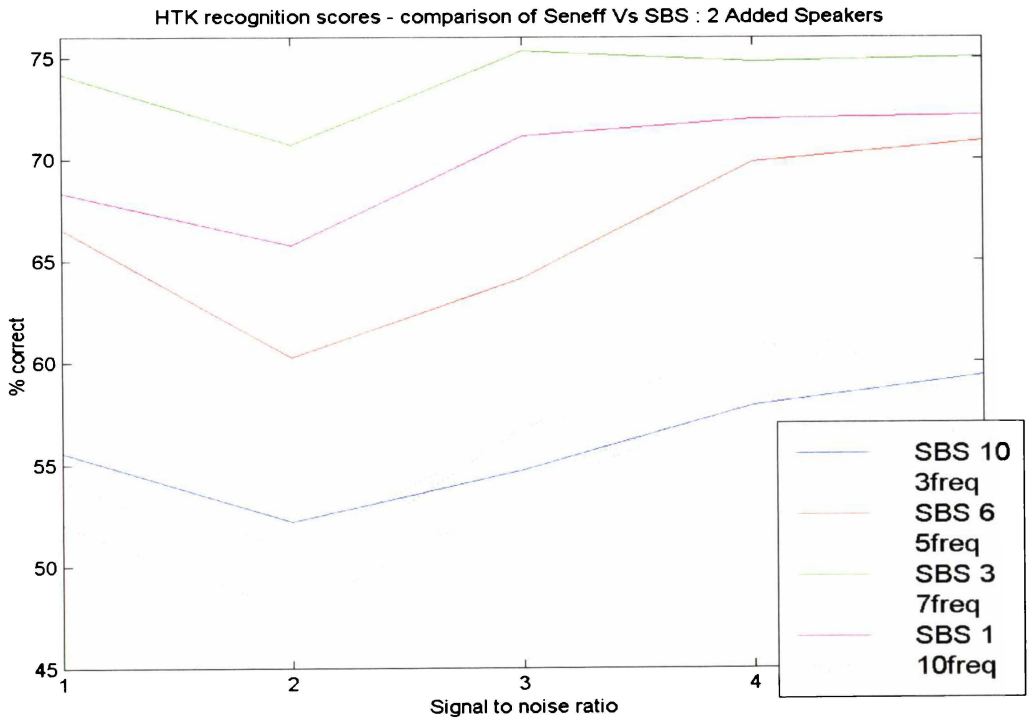


Figure 7.15 : Comparison of SBS and Seneff *f*-run Results for Two Added Speakers

As previous results have tended to indicate, there is little difference between the various forms of cocktail party noise, and hence it is not surprising that Figure 7.15 and Figure 7.16 show similar trends to their one added speaker equivalents discussed in the previous section. Again, the SBS is clearly superior in all instances except for the threshold value of ten, for both levels of amplitude/frequency/phase quantisations.

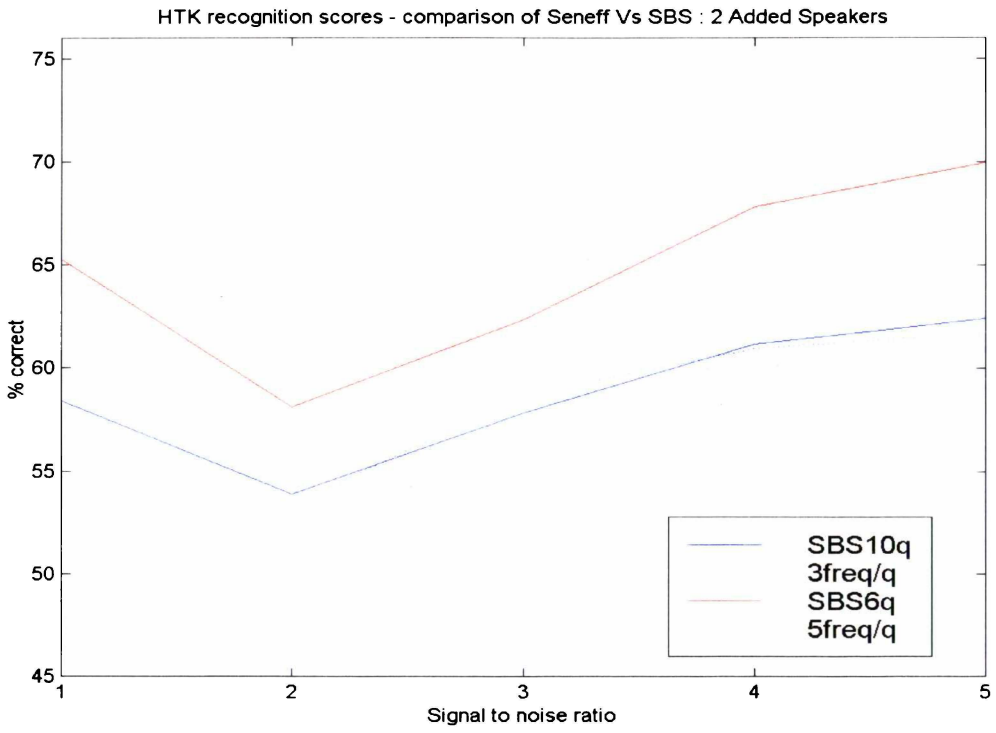


Figure 7.16 : Comparison of SBS and Seneff *f*-run Results for Quantised Two Added Speakers

7.4.4 Cocktail Party Noise – Four Added Speakers

Again the trend is that the SBS processing has a superior performance to the Seneff, but the results for SBS Threshold ten, and Seneff three are very close. For the levels of four added speaker noise, the SBS is still superior to the Seneff at this level (unlike the case of the white noise and two added speakers), but the difference is not large.

For both the two and four added speakers, the Seneff recognition scores vary little with a change in the number of selected frequencies, and are almost universally substantially inferior to the SBS results. The conclusion from this is that the frequencies selected from the Seneff process are not capable of rejecting noise of the form of added speakers.

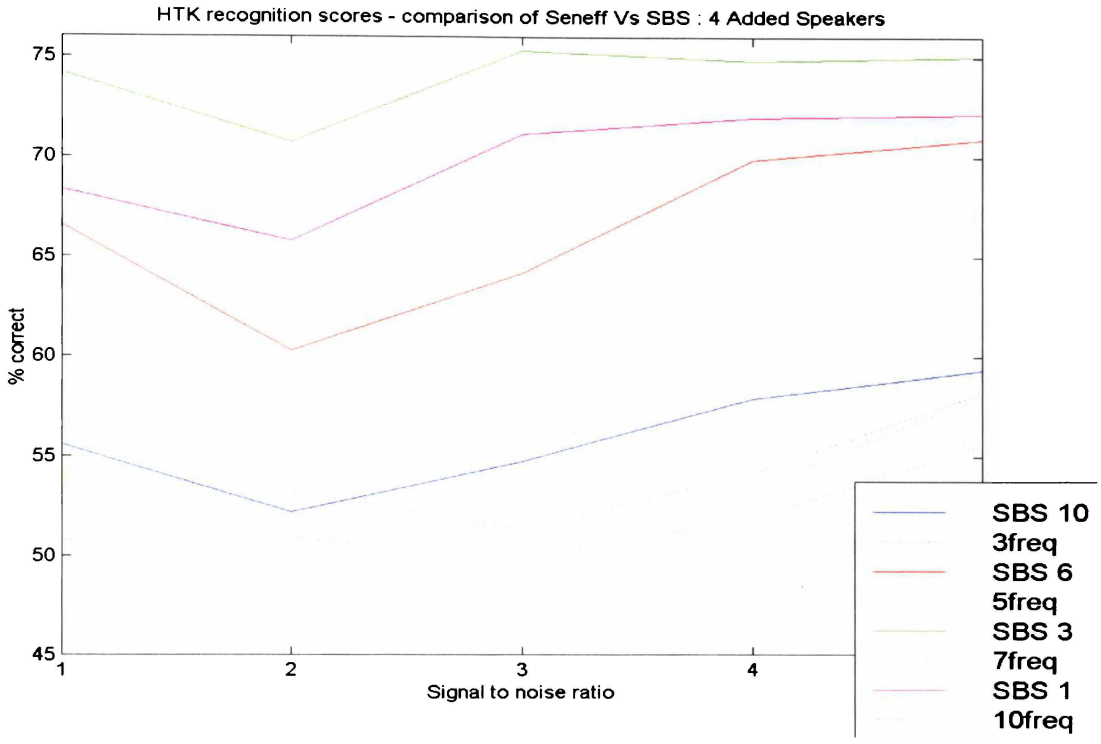


Figure 7.17 : Comparison of SBS and Seneff *f*-run Results for Four Added Speakers

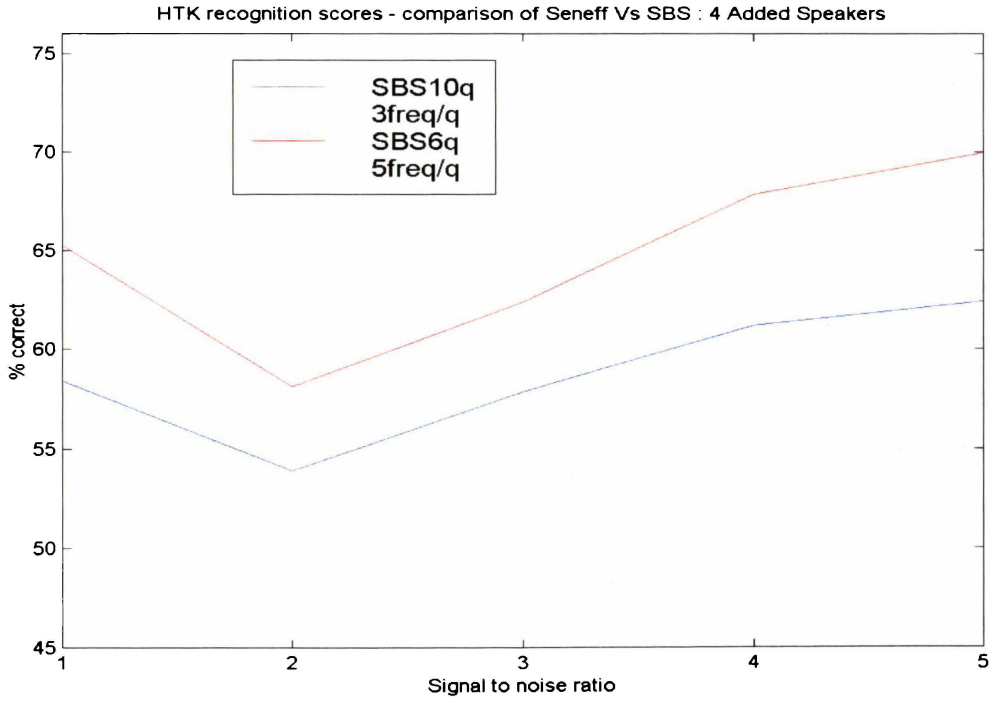


Figure 7.18 : Comparison of SBS and Seneff *f*-run Results for Quantised Four Added Speakers

7.4.5 Summary

The Seneff results indicate some very similar trends to the SBS results in that the variation in the number of frequencies selected for the reconstruction produces a larger variation in recognition scores than does a variation in the quantisation parameters.

Introducing more frequencies into the Seneff speech reconstruction does not necessarily increase the Seneff recognition performance. In fact, for all forms of added noise, the three frequency scheme produces nearly the best results.

Only for the extreme SBS threshold value (ten) does the Seneff equivalent perform better than the Ghitza algorithm. This is not a matter of the Seneff process suddenly becoming a much better algorithm at this level, rather, as the previous chapter indicated, the Ghitza scheme just worked poorly at this threshold level.

The substantial difference generally, between the SBS and Seneff results, demonstrates how poor the noise rejection performance of Seneff algorithm is.

7.5 Comparison of LPC, Seneff and SBS Results

The LPC data is generated by reducing a *.wav* file using LPC10 processing, using this compressed file to reconstruct a time domain speech file (16 bit integer format), and then converting this to the TIMIT format for input into the HTK stages. The system was trained with the same 3696 files of the SBS and Seneff process, and the same nine S/N levels of white noise, one added speaker, and five S/N levels of multiple added speakers were used for the test files.

7.5.1 White Noise

Figure 7.19 illustrates the best SBS and Seneff results, as well as the LPC white noise experiment. For the higher Signal to Noise ratios, the results begin to converge as would be expected, but for the higher noise levels, the SBS results are far superior. The Seneff trend

is quite flat, and the LPC is incapable of providing acceptable recognition performance for the high white noise levels.

It is clear that an SBS Threshold of six (and in fact most of the SBS thresholds) provides a considerably superior result to either of the other two algorithms for speech in the presence of white noise.

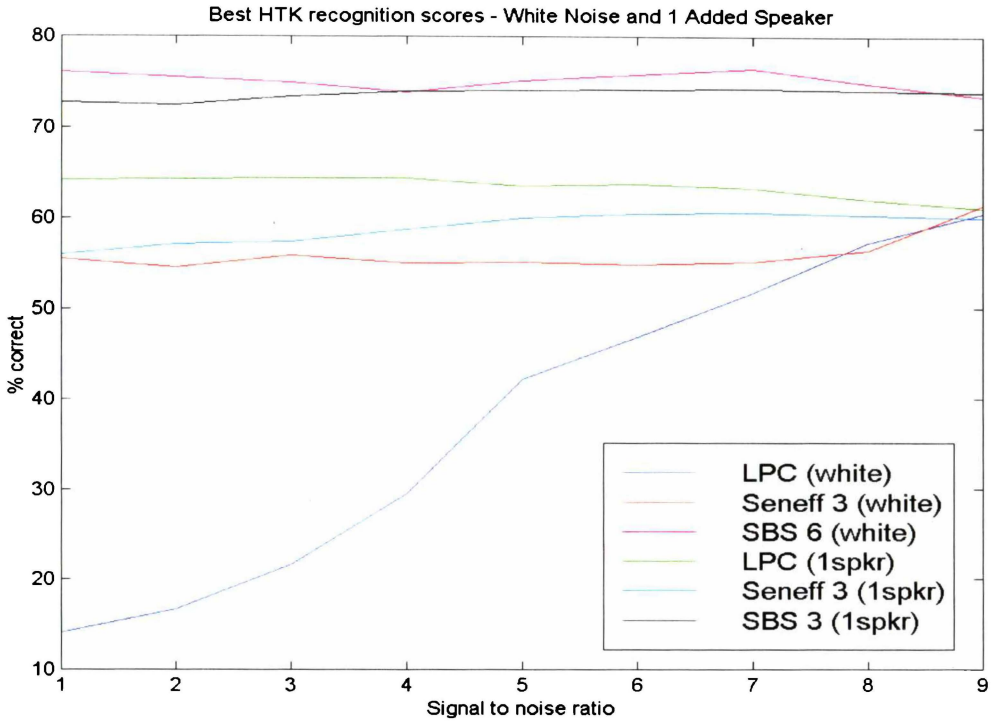


Figure 7.19 : Best Recognition Results for White Noise and One Added Speaker

7.5.2 Cocktail Party Noise

Figure 7.19 also plots the best SBS, Seneff and the appropriate LPC. In this instance, the LPC outperforms the best Seneff result, but again, the SBS produces the best results. The one added speaker cocktail party noise trend is repeated for the case of two added speakers and four added speakers (Figure 7.20). Again the LPC outperforms the Seneff, but a wide range of SBS thresholds provide considerably better performance than the LPC (though only SBS three is actually illustrated).

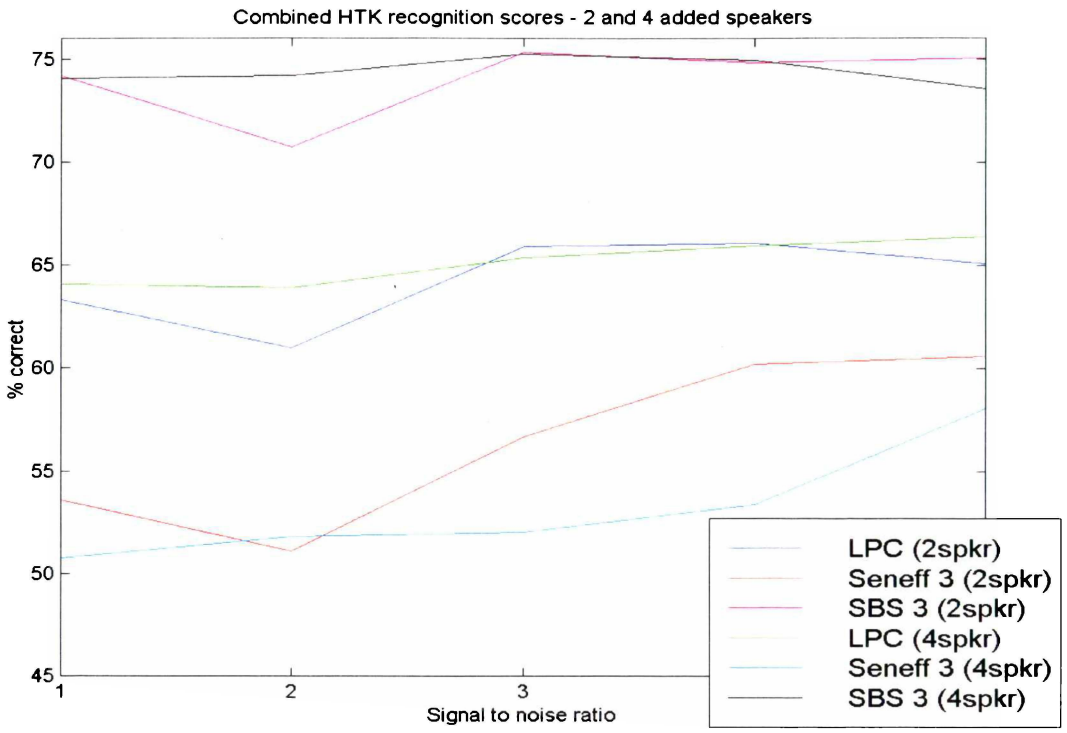


Figure 7.20 : Best Recognition Results for Two and Four Added Speakers

7.5.3 Summary

A computational model (Ghitza's SBS), a physiological model (Seneff), and a classical analytical model (LPC) have been compared in terms of their HTK recognition scores, over a total of 28 different levels and forms of added noise. In every situation, the SBS has markedly outperformed the other two algorithms.

Chapter 8

8 Determining the Optimal Parameter Settings

8.1 Statistical Analysis

To date, the comparison of the different algorithms has been rather subjective. The DRT and HTK algorithms have attempted to provide some objective meaning to intelligibility and recognition respectively, however, aside from the plotting of the Squared Errors, most of the analysis of the affects of (for example) *SBS Threshold*, or *Frame Overlap* has been done by subjective visual or auditory means. It would be desirable to obtain a more quantitative expression of the interaction of the variables in determining the intelligibility or recognizability of the speech. Preferably, such an analysis would yield information concerning the importance of the quantised variables to recognition score, an interpolation of recognition score over all the Signal to Noise levels of interest and be able to predict the maximum recognition score for the speech analysis/synthesis system.

Both the SBS and Seneff results are inherently nonlinear. This can easily be observed by noticing that the differencing routine (for example) is either turned on or off. An attempt to represent the system as numeric variables, and analysing via conventional linear multiple regression techniques, yields an R^2 of 0.191 for the SBS results, and the Seneff results are too ill-conditioned to obtain a result. This is not surprising as linear regression is based on the assumption that a linear relationship exists between the input variables and the output variables (which may be corrupted by Gaussian noise). In this situation, these assumptions are untrue, and obviously such a technique is of absolutely no use with regards analysing the dependence of the recognition score on the process variables. Neural networks are particularly well suited to such nonlinear systems, and so regression using Statistica's Neural Networks package was examined. An overview of neural networks (as related to this project) is presented in Appendix F.

8.1.1 SBS Neural Regression Results

Table 8.1 lists some of the various forms of networks employed in the search for a best solution. Due to a bug in the Statistica package, the four layer Multilayer Perceptrons (MLPs) were not included in this particular network search, however, a separate four layer MLP search did not obtain a superior result to the 8(:11) input, ten hidden unit three layer MLP structure listed below (network number 25).

	Type	Inputs	Hidden	Hidden(2)	Error	Perform.
01	Linear	1	-	-	10.06377	0.9994603
02	RBF	4	2	-	9.986137	0.9925747
03	RBF	1	3	-	9.838042	0.9771463
04	RBF	1	2	-	9.803552	0.9742619
05	Linear	7	-	-	9.272656	0.9194394
06	Linear	8	-	-	9.208336	0.9139563
07	MLP	5	6	-	3.596631	0.3582075
08	MLP	5	8	-	3.428644	0.3406294
09	MLP	5	9	-	3.295729	0.3275078
10	MLP	7	14	-	3.253547	0.3242212
11	GRNN	8	364	2	3.296451	0.328345
12	GRNN	8	364	2	3.282359	0.3270054
13	GRNN	8	364	2	3.281185	0.326852
14	GRNN	8	364	2	3.27634	0.3263974
15	GRNN	8	364	2	3.275192	0.3262761
16	GRNN	7	364	2	3.275192	0.3262761
17	MLP	4	6	-	3.20046	0.3188426
18	GRNN	6	364	2	3.192064	0.3174662
19	MLP	6	6	-	3.104429	0.3084081
20	MLP	6	7	-	3.011376	0.3000586
21	MLP	6	11	-	2.964317	0.2949917
22	MLP	7	6	-	2.961922	0.295262
23	MLP	7	10	-	2.873221	0.2864672
24	MLP	8	8	-	2.726693	0.2717439
25	MLP	8	10	-	2.705672	0.2697862

Table 8.1 : A Sample of Networks Trialled for the Evaluation of the SBS Data

In this table, the column:

Type Displays the network type,

Inputs Number of input variables used by the neural network (fewer inputs result in a faster algorithm)

Hidden The number of hidden units in the 2nd and 3rd layers of the network (linear networks have no hidden units - fewer hidden units reduce the network complexity)

Error The error of the network on the verification set (lower error is generally better)

Performance A measure of the success of the network

8.1.2 Statistics and Overlearning

To prevent overlearning (the system modelling a function in too complex a fashion, so that the noise is modelled along with the data), Statistica employs cross-verification, where some of the training cases are reserved and not used in the back-propagation training algorithm. Instead, these cases are used to keep an independent check on the progress of the network, and are referred to as the verification set. If the network is minimising the true error function, the error of both the training and verifications sets should drop. If the verification error is not reducing, this is a sign that the system may be overlearning, and that the number of hidden units/layers may need to be decreased.

However, the introduction of this verification set actually influences the network, and is therefore not truly independent To account for this, Statistica reserves a third set of data, the test set, to provide a completely independent result. It can only be used once (at the end of the design process), to check that the cross-verification error is not artifactual. If this test set is used to correct the training process, then it becomes another verification set. Statistica, by default, separates the data in the proportions 2:1:1 between the three subsets. For the SBS data set, which contains 728 cases, 364 cases form the training set, 182 the verification set, and another 182 cases forms the final test set. These sets are taken from random samples throughout the entire data set. Similarly, the Seneff data is divided into 252 training cases 126 verification and 126 test cases.

8.1.2.1 Regression Statistics

In regression problems, the purpose of the neural network is to learn a mapping from the input cases to the output value, and it is successful if it can make this prediction more accurately than some simple estimate, for example that arising from the calculation of a mean. This mean value can be used as the predicted value for all previously unseen cases, and the average expected error becomes the standard deviation of the training data output. The aim of the regression network then, is to produce an estimate which has a lower prediction error standard deviation than the training data standard deviation. The statistical results for the final MLP network employed to analyse the SBS run are listed in Table 8.2.

	Tr. SCORE	**Ve. SCORE**	**Te. SCORE**
Data Mean	65.85354	65.02544	63.92126
Data S.D.	9.141907	10.05661	10.186
Error Mean	0.002817	0.001631	-0.1192
Error S.D.	2.466712	2.713135	3.41143
Abs E. Mean	1.705819	1.917136	2.21283
S.D. Ratio	0.2698247	0.2697862	0.3349137
Correlation	0.9629164	0.9635424	0.9472207

Table 8.2 : Regression Statistics for 8:11-10-1:1 MLP Network

where

Data Mean is the average value of the target output

Data S.D. is the standard deviation of the target output

Error Mean is the average of the absolute difference between the target and actual output values

Error S.D. is the standard deviation of errors for the output variable

S.D. Ratio is the ratio of the Data and Error S.D.s.

Correlation is the standard Pearson-R correlation coefficient between the predicted and target values.

If the Training correlation score (**Tr**) is close to the Verification (**Ve**) and Test (**Te**) correlation scores, the user can be confident that over-learning has not occurred. Statistica claims it is not uncommon for the training error to be much lower than the other two, but if the verification and test errors are close, the results are still valid. The possibility of over-

learning may be reduced by acquiring more test cases, or using a different network A "Rule of Thumb" for the Statistica Neural Networks package is to use at least five, and preferably ten times as many cases as there are connections in the network. A rough estimation for the number of connections is the square of the number of inputs. For the SBS process, with eight inputs, there are 728 cases (which is greater than 8×8×10=640), and for the Seneff, with six inputs, there are 504 cases (which is greater than 6×6×10=360), so our system should be very amenable to this form of neural processing.

8.1.2.2 Sensitivity Analysis

The Sensitivity Analysis window of Statistica provides information concerning the relative importance of the variables used in the neural network, and effectively tests to see how the neural network would cope if each of its input variables were unavailable. The data set is submitted to the network repeatedly, with each variable in turn treated as missing, and the resulting network error is recorded. A large error indicates that the variable was important.

The window, part of which is illustrated in Table 8.3, shows the error when each variable is omitted for the SBS analysis, plus the ratio between the error with the variable omission and the total error, and ranks the variables in order of importance. If the ratio is less than unity, the network would perform better if that variable was completely omitted (and would be omitted in the configuration of the neural network used for this data analysis). Realistically, these variables are receiving a ranking according to their importance in a particular neural network, rather than a ranking according to their overall importance in the data – a different network may rank the variables differently. If several variables are correlated, the network training algorithm may arbitrarily chose some combination of them, and the sensitivities may reflect this. It is therefore advisable to run a sensitivity analysis over several different networks and draw conclusions only from consistent results.

The sensitivity of the SBS variables was analysed over 60 different forms of linear, RBF, GRNN and MLP networks. The three layer MLP consistently ranks SBS Threshold as the most important variable, followed by the form of noise, and the inclusion or omission of differencing. The Signal to Noise ratio is ranked next, followed by *Amplitude*. *Phase*, the Unvoiced Modifier and *Frame Overlap* are considered to be the least important. The four

layer MLP has an almost identical variable sensitivity to the three layer. GRNN is similar to the MLP, *Differencing* being the most important, followed by the *Threshold* and form of noise. The Signal to Noise ratio, and *Phase* are of medium importance, followed by the Unvoiced Modifier, *Frame Overlap*, and lastly, *Amplitude*.

	NOISE	**S_N**	**THRESH**	**AMP**	**PHASE**	**VUS**	**O_LAP**	**DIFF**
Rank	2	3	1	4	6	7	8	5
Error	8.19213	5.59570	9.65570	5.12618	4.47767	2.97293	2.62832	5.06790
Ratio	3.32564	2.27160	3.91978	2.08100	1.81773	1.20687	1.06698	2.05734
Rank	2	3	1	4	5	7	8	6
Error	9.05933	5.80680	10.62397	5.41363	5.03853	3.16367	2.76718	4.98435
Ratio	3.34827	2.14615	3.92655	2.00084	1.86221	1.16927	1.02273	1.84218

Table 8.3 : Sensitivity Analysis of SBS Results in a Three Layer 8:11-10-1:1 MLP Network.

A linear algorithm treats the amplitude and phase quantisation as being the most important, threshold values, *Frame Overlap* and The Unvoiced Modifier as the least. However, the best correlation score from a linear algorithm was only 0.452, and so can not really be seriously considered. The RBF considers the form of noise to be the most important, followed by the Differencing option, and *Amplitude*. *Frame Overlap* and the Signal to Noise ratio are the least important. The best correlation score for the RBF scheme was 0.600, and so it is not a serious contender as a regression algorithm.

This is summarised in below, where the sensitivity of each type of network is summarised using a three level ranking. The final column in the table corresponds to the best correlation factor obtained from that form of network (and by implication, which form of variable sensitivity seems to perform best).

Network Type	Noise type	S/N	Thresh.	Amp Quant	Phase Quant	VUS	Frame O/lap	Diff.	Corr
Linear	**	**	*	***	***	*	*	**	0.452
MLP3	***	**	***	**	*	*	*	***	0.964
MLP4	***	**	***	**	*	**	*	***	0.962
GRNN	***	**	***	*	**	*	*	***	0.948
RBF	***	*	**	***	**	*	*	***	0.600

Table 8.4 : Summary of SBS Network Variable Sensitivities

Where:

*	= not important
**	= somewhat important
***	= very important

The 0.964 correlation score of the three layer MLP was the best of the network options, (in that it was the highest score of any network that had not suffered overlearning), and perhaps gives a better indication of the variable sensitivities than the other network types. The correlation plot (a graphical indicator of how well the predicted score approaches the target score) is illustrated below in Figure 8.1. The match is generally very good, except for the low (sub 40%) scores.

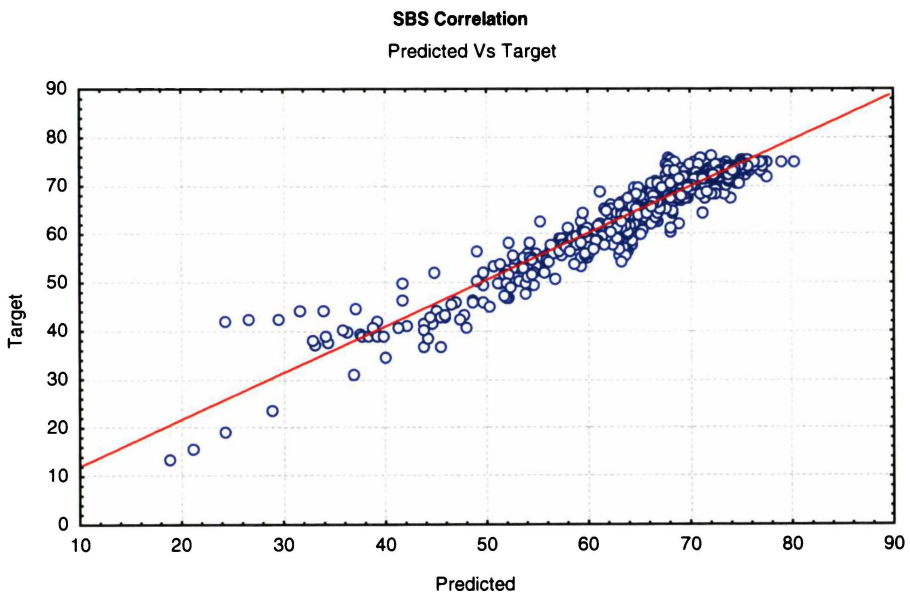


Figure 8.1 : SBS Correlation – Predicted Score Versus Target Score

8.1.3 Seneff Analysis

Again the three and four layer MLPs provide the best analysis of the Seneff data. The variable sensitivity for these algorithms is presented in Table 8.5, where run type refers to the "2", "8" or "f" run.

Network Type	Run type	Noise	S/N	#Freqs	Amp Quant	Phase Quant	Corr
Linear	*	**	**	*	***	***	0.789
MLP3	*	***	***	*	**	**	0.985
MLP4	**	***	***	**	*	*	0.963
GRNN	**	***	***	**	*	*	0.846
RBF	***	***	***	**	*	*	0.783

Table 8.5 : Summary of Seneff Network Variable Sensitivities

The four layer MLP correlation score stated above, was not the highest correlation four MLP score (which was 0.977), but was the best result that did not suffer from overlearning. The GRNN was particularly bad at overlearning, even at levels well below 0.840.

Again the MLP networks (both three and four layer) provide the best overall results, and these networks place the most importance on the form of added noise, and the level that noise is added at. Less important is the interpolation or run type and the number of frequencies used in the reconstruction. Amplitude and phase quantisation rank as the least important variables. Taking into account overlearning, and complexity of network, the best results were obtained from a three layer MPL, 6:11-12-1:1.

As indicated in Table 8.6, the correlation for this network when used on the Seneff data is approximately 0.985 – a very useable figure. Various combinations of training, verification and test data were used to verify this correlation, and the network consistently scored between 0.983 and 0.988, with 0.985 being a reasonable average. The plot of the correlation results (predicted score vs. target score) is illustrated in Figure 8.2, which indicates very good correlation results across the whole range of scores.

	Tr. SCORE	Ve. SCORE	Te. SCORE
Data Mean	53.42448	52.8369	54.21373
Data S.D.	5.983963	6.23737	5.271933
Error Mean	-0.01179	0.08678	-0.07073
Error S.D.	1.021663	1.051679	0.9725251
Abs E. Mean	0.7053085	0.714137	0.650305
S.D. Ratio	0.1707334	0.1686095	0.1844722
Correlation	0.9853292	0.9857077	0.9830531

Table 8.6 : Regression Statistics for a Three Layer 6:11-12-1:1 MLP Network

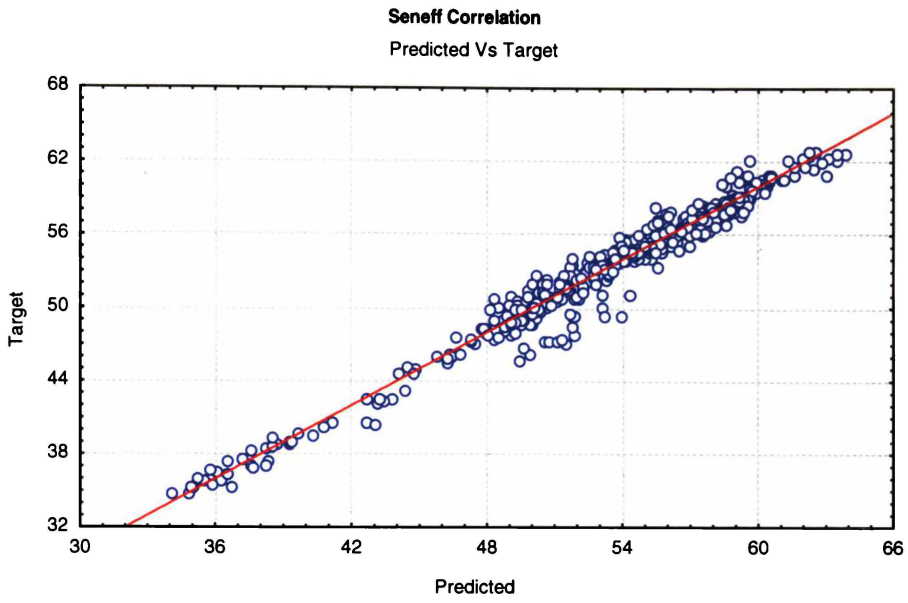


Figure 8.2 : Seneff Correlation – Predicted Score Versus Target Score

The network weightings are provided in Table 8.7, where the *INT* variable represents the form of frame size (i.e. the run type "*2*", "*8*" or "*f*"), the *NOI* variable indicates the form of added noise (white, one, two, or four speaker), *S_N* is the Signal to Noise Ratio, *FREQS* is the number of selected frequencies, and *AMP* and *PHASE* are the number of bits used in the respective quantisation (frequency bins have the same quantisation level as amplitude).

Threshold	-0.5187	-0.1216	-0.5457	-0.8242	0.3288	-0.7965	0.8108	0.1172	0.0500	1.0308	0.1360	3.0138	-2.7255
INT=2ru	0.4498	0.1686	-1.0654	1.6683	-0.1355	0.1509	0.0675	-0.5265	1.5906	0.1256	-0.0583	-1.3848	
INT=8ru	-0.2298	0.0437	-0.2283	0.4116	0.4561	0.5097	0.8255	-0.5633	0.7821	-1.4190	0.8018	0.1367	
INT=fru	0.0340	0.1638	0.8866	0.7700	0.2526	0.3219	0.3416	-0.5537	-0.4639	-0.5823	-1.4812	-0.2473	
NOI=whi	-0.3304	-1.2729	4.1092	-1.4964	-0.2131	-2.4655	1.0374	-1.8064	-2.3378	0.5782	0.6589	1.1852	
NOI=1sp	0.1302	0.9537	-2.1808	2.4557	0.3647	0.1813	-1.1022	0.5669	1.0355	-1.1202	0.1904	-2.0041	
NOI=2sp	0.2390	0.7131	-1.8428	1.6662	0.0229	3.010	-0.5314	-0.5210	0.7594	-0.1529	0.1527	-0.9021	
NOI=4sp	-0.0877	0.3793	-0.9057	1.5017	0.4867	1.2485	-0.1787	-0.1197	-0.1069	0.2011	-0.1708	-0.8386	
S_N	3.1109	-0.3963	3.5747	5.8517	-4.360	8.098	0.4083	-1.9540	-3.3675	1.8994	-0.1117	-5.2189	
FREQS	-2.5201	-0.9245	5.4350	-4.2120	2.1875	0.7182	0.1076	1.3423	-5.3266	-0.0822	-1.5788	5.3310	
AMP	-0.2917	0.7208	-0.234	-0.4601	0.0797	0.6030	0.3216	-0.5894	0.0031	-0.4452	1.2615	0.1828	
PHASE	0.1644	-0.1185	0.3462	0.4480	0.0381	-0.3261	-0.685	0.3390	-0.105		0.8707	-0.1577	

h1#01	h1#02	h1#03	h1#04	h1#05	h1#06	h1#07	h1#08	h1#09	h1#10	h1#11	h1#12
-2.8245	2.6910	-3.008	2.8341	-2.8181	-1.6054	2.0813	1.2309	-3.1121	0.9511	-0.1332	2.2145

Table 8.7 : Weightings for the MLP 6:11-12-1:1 Network Used to Analyse the Seneff Data

8.2 Towards The Best System

The regression analysis of the previous section enables us to predict the performance of both the Seneff and SBS algorithms for a wide range of values of the input variables (taking care not to extrapolate to areas outside the training range of these input variables).

It was stated in 8.1 that it would be desirable to obtain an indication of the importance of the simulation variables to the recognition score. A useful tool of Statistica's Neural Networks is its ability to plot a response curve. An example of such a curve showing (in this case) the relationship between Recognition Score and SBS Threshold is provided in Figure 8.3. This plot assigns the mean value to all the other contributing variables, and plots the output (Recognition Score) against the independent variable (SBS Threshold).

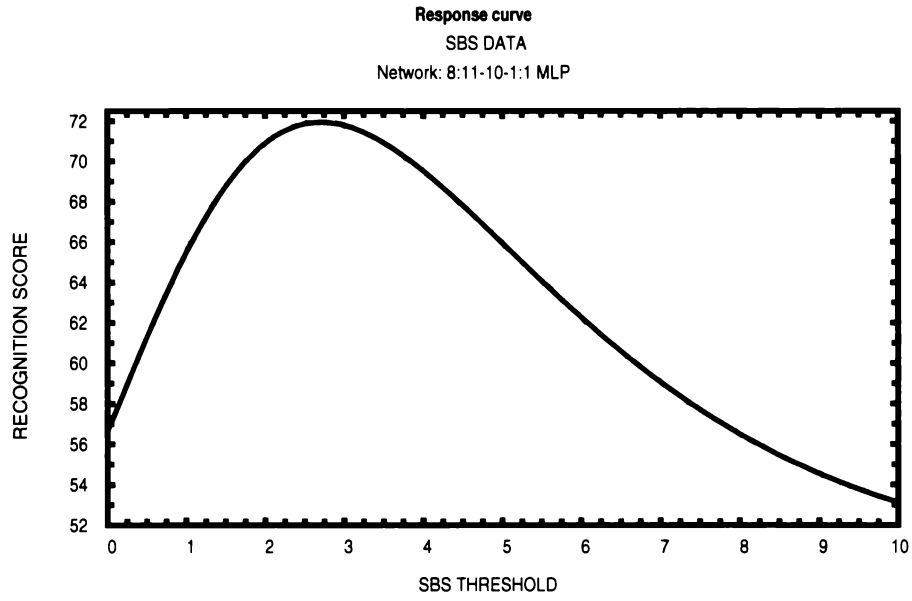


Figure 8.3 : Response Curve of SBS Data – SBS Threshold Vs Recognition

For this network, recognition peaks at approximately SBS Threshold three, which, while it optimises the response to cocktail party noise, is far from the optimum level for added white noise (Threshold six). That the network selects this Threshold peak of three is not surprising given that there are 234 cases of white noise compared to 494 cases of cocktail party noise in the data set (and the class frequencies were used to assign an average value for this nominal variable).

Similar plots of amplitude response show only a 0.2% difference in recognition score as the amplitude quantisation (and therefore frequency bin quantisation) is varied from six bit (log) to three bit (log). Altering the phase quantisation from zero bits to eight bits, increases the response curve by only 1.4%, and altering the data overlap from 25 to 100% provides a similar response change. There is a large change in Recognition Score with increasing Signal to Noise Ratio, and the curve continues to increase suggesting even better recognition as the S/N is increased above 40 dB.

The Seneff response curve of Number of Selected Frequencies (similar to SBS Threshold) versus Recognition Score (Figure 8.4), shows a local maxima for six selected frequencies (similar to the SBS equivalent, Figure 8.3). However, the peak recognition score of 51% is consistent with the earlier claims that the Seneff selected frequencies do little to reject added noise (as measured by the HTK process). Additionally, the noise response curve (not illustrated) indicates that the response of the Seneff network levels off at 40 dB, whereas the SBS equivalent shows recognition still increasing.

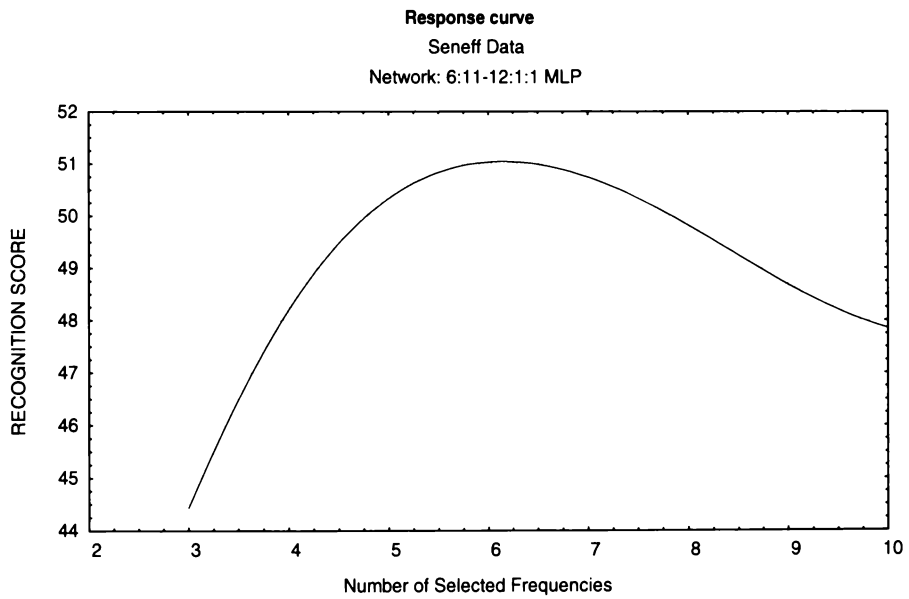


Figure 8.4 : Response Curve of Seneff Data – Number of Frequencies Vs Recognition

A "best" system can now be considered that fulfils the requirement of a low bit rate at high recognition as having

- 100 bandpass filters, all with incline 18 dB/octave, those whose CFs are below 1kHz have decline –18 dB/octave, the remainder a decline of –120 dB/octave.
- Frequency selection via an In-Synchrony-Bands-Spectrum algorithm
- Hamming window
- 128 point, 16 ms frame size (8 kHz sampling)
- Bandwidth 300-3300 Hz
- No frequency emphasis from differencing algorithms
- 25 % data overlap
- 1 bit phase representation (primarily for intelligibility purposes rather than recognition)
- 3 bit logarithmic amplitude quantisation
- 3 bit logarithmic frequency bin quantisation
- SBS Threshold of three (i.e. seven reconstructed frequencies) (alternatively, for white noise only, a Threshold of six i.e. five reconstructed frequencies)

The resulting bit rate for such a system is:

$$\begin{aligned} &\#\text{ frames per second} \times \#\text{ of frequency components} \\ &\times (\#\text{ bits for the frequency bin} + \#\text{ bits phase} + \#\text{ bits amplitude}) \\ &= 78.125 \times 7 \times (3 + 1 + 3) = 3{,}828 \text{ bits per second} \end{aligned} \qquad \textbf{Equation 8.1}$$

i.e. about 3% of the original bit rate. For white noise corrupted speech, this drops to 2,734 bits per second (approximately 2% of the original bit rate). Note that no Vector Quantisation or Delta compression, nor any other compression technique is involved here. Such compression algorithms can be applied to the already reduced information set to even further lower the bit rate well into the sub 1 kbit/s regime.

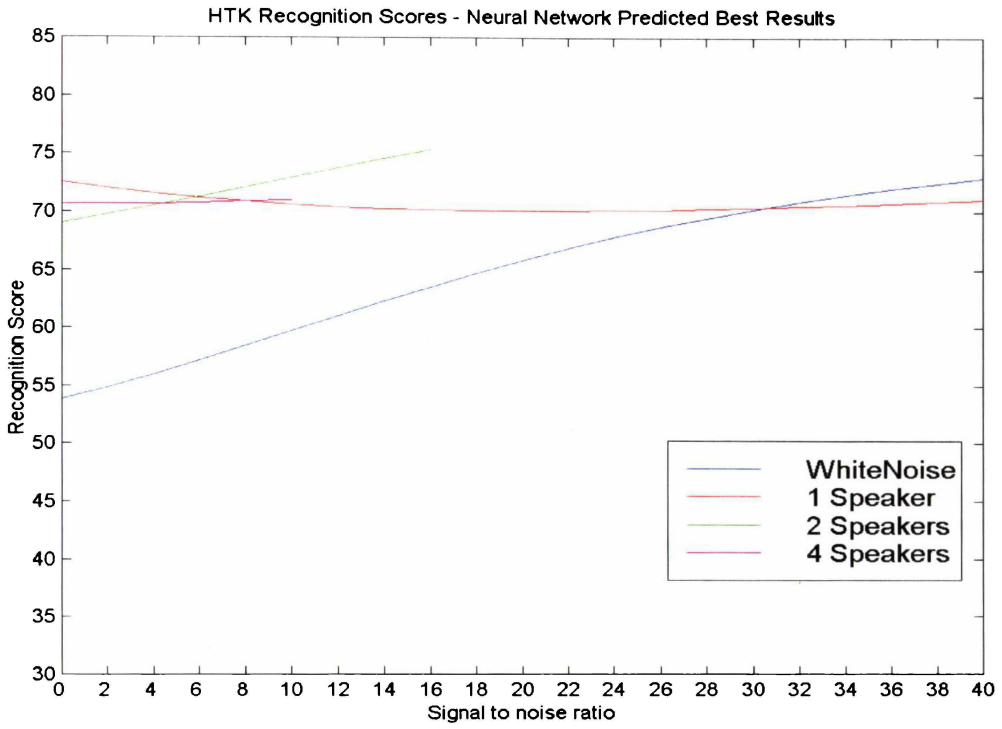


Figure 8.5 : Neural Network Predicted Best Recognition Response

A simulation of this system (from the Neural Network) yields Figure 8.5, and predicts the recognition score for all noise types, over the range 0 to 40 dB Signal to Noise Ratio. This interpolation, resulting from the neural multiple regression, can be expected to be considerably more accurate than a classical interpolation scheme. Note that the single speaker cocktail party noise and the white noise trends converge for high SN values. The multiple speaker curves would be expected to similarly level out and approach the same 70-75% recognition score.

8.3 Improved Systems

The essential shortcoming of the system as developed to this point is that the peak performance for the speech corrupted by white noise occurs for an SBS Threshold of six, but for the other forms of cocktail party noise (as well as for speech with no added noise), the peak performance occurs at SBS Threshold three. To obtain a single final system, the options are:

1. Use SBS Threshold six to maximise white noise response, tolerate the degraded cocktail party noise.
2. Use SBS Threshold three to maximise the cocktail party noise response, tolerate the (severely) degraded white noise response.
3. Use a compromise SBS Threshold, that maximises neither form of noise, but nor does it degrade any of the noise types as severely as options 1 or 2.
4. Develop an algorithm to automatically detect the noise type, and adjust the SBS Threshold accordingly.

Option 4, if possible to be implemented, offers the best solution.

8.3.1 White Noise Identification Using the VUS Algorithm

Possibilities for this automatic white noise identification include a modified use of the VUS algorithm. This algorithm was designed to detect unvoiced (broadband) speech, and use more frequency components (lower the SBS Threshold) for frames identified as unvoiced. This broadband nature is similar to the effect of adding white noise, and so the possibility exists that the VUS algorithm may be of some use in the automatic detection of added white noise.

It was originally envisaged that the *unvoiced modifier* that results from the VUS algorithm, would decrease the SBS Threshold, but as has been illustrated, to better handle the added white noise it is desirable to increase the Threshold. To investigate this, the SBS Threshold was set to three, amplitude and frequency quantisation to three bit logarithmic, phase to one bit, data overlap at 25%, and no differencing was employed. Experiments were then conducted with the *unvoiced modifier* variable taking the values, 1.0, 1.5, and 2.0. The results for white noise and one added speaker are illustrated in Figure 8.6. The *orig* line is the closest of the original 26 runs to these settings (run *u* as per Table 6.1).

As hoped, the white noise response is considerably better than the original result, with a significant improvement for an unvoiced modifier setting of 1.5. However, the one added speaker result shows a degraded result.

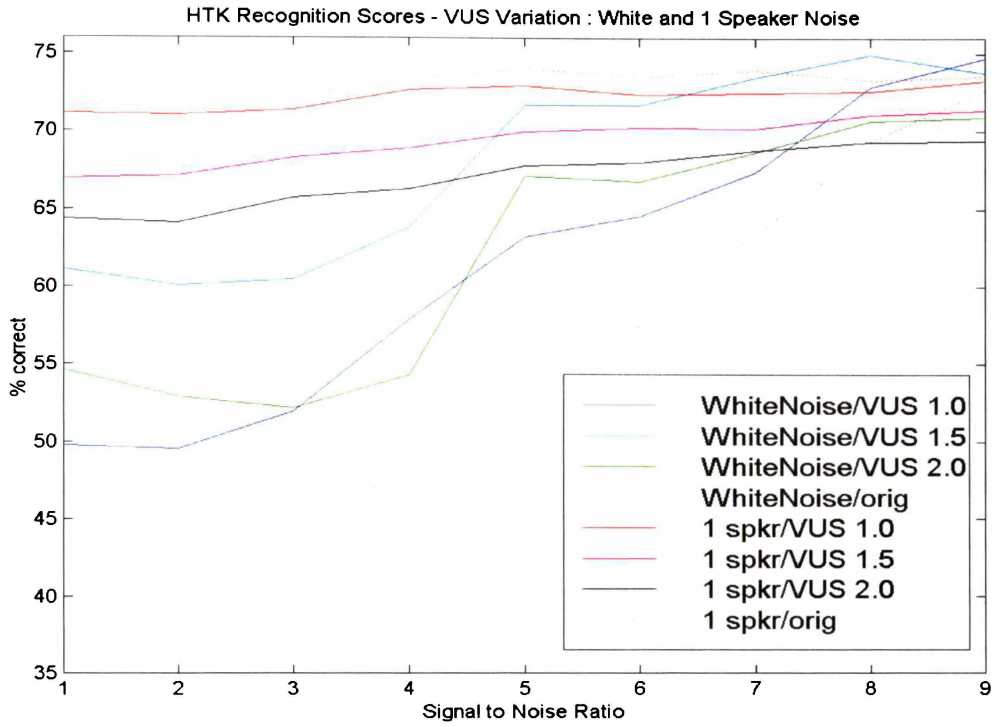


Figure 8.6 : White Noise and One Added Speaker Response to Large VUS

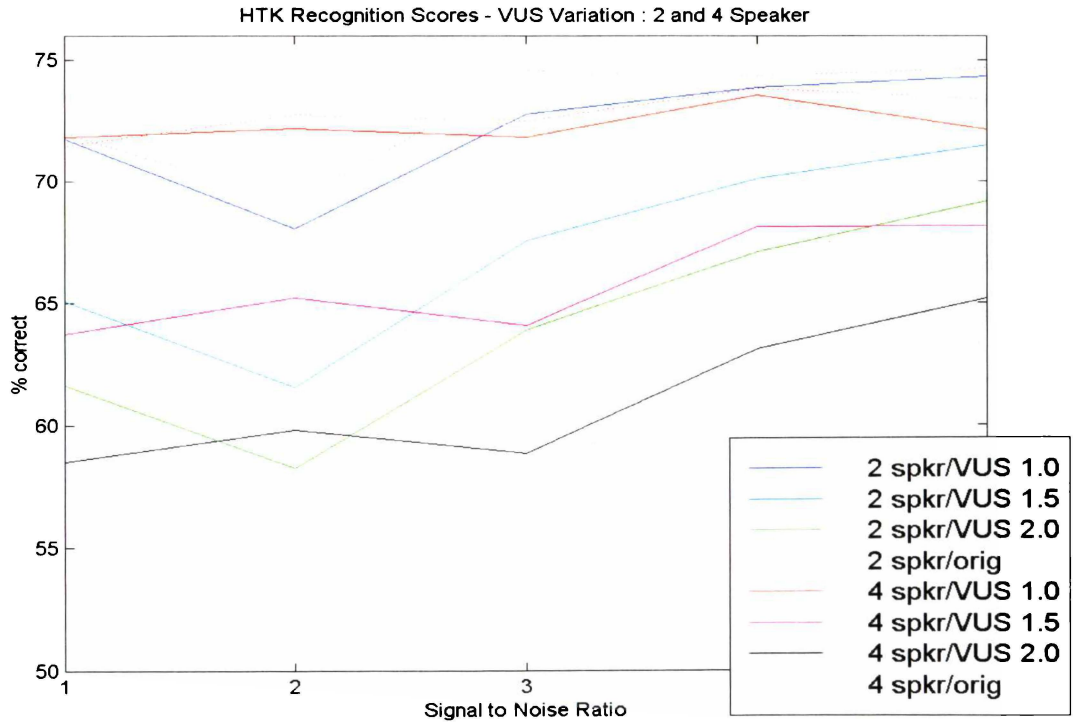


Figure 8.7 : Multiple Speaker Response to Large VUS

The multiple speaker response for these experiments appears in Figure 8.7. For both two and four added speakers, the recognition score drops as the VUS value is set greater than unity. The increase in Threshold has only improved the white noise responses – the same effect could have been achieved by just increasing the overall SBS Threshold. A better solution needs to be found.

8.3.2 White Noise Identification Using the SBS Spectra

The broadband nature of the white noise is such that at high frequencies it is to be expected that the white noise contribution will produce local SBS peaks (as discussed in Section 6.5.1). One result of these local peaks, is that the SBS amplitudes of the valid speech components will be reduced. An examination of these peaks for speech corrupted by the various forms of noise indicates that there is a significant decrease in the number of frames with SBS amplitudes greater than ten for white noise corrupted speech (as compared to cocktail party noise corrupted speech).

Further examination indicated that an SBS Threshold of 25 was particularly indicative of this trend. In fact, for most of the white noise corrupted files, less than half the number of frames had an SBS amplitude above 25. From repeated experimentation, an extremely good white noise detector was created by determining whether more than 40% of the frames in the speech file had an SBS amplitude of 25 or more. If this was the case, the speech was marked as not being significantly affected by white noise, if not, the speech was marked as being white noise corrupted. The performance of this white noise detector is detailed in Table 8.8.

In summary, 87.8% of the white noise corrupted files were correctly identified (though over half the failures were for the last SN value), 95.5% of the one speaker corrupted files were not identified as having significant levels of white noise, as were 97.0% of the two speaker corrupted files, and 95.6% of the four speaker corrupted files. This gives an overall success rate for the test files of 93.3%. Of the 3888 training files (theoretically noise free), 90.8% were correctly as not being significantly white noise corrupted.

Type of Added Noise	Signal-To Noise Ratio	% Correctly Identified
white	6	100
white	8	100
white	10.5	100
white	14	100
white	20	100
white	22.5	98.3
white	26	93.6
white	32	64.5
white	40	33.7
1 speaker	6	95.3
1 speaker	8	99.4
1 speaker	10.5	97.7
1 speaker	14	97.1
1 speaker	20	95.9
1 speaker	22.5	95.3
1 speaker	26	92.4
1 speaker	32	94.8
1 speaker	40	91.9
2 speakers	2.5	99.4
2 speakers	4.4	98.8
2 speakers	6.9	100
2 speakers	10.5	93.6
2 speakers	16.4	93.0
4 speakers	-1.5	96.5
4 speakers	2.4	97.7
4 speakers	-0.7	95.9
4 speakers	3.4	89.5
4 speakers	11.4	98.3
3888 Training files	358	90.8

Table 8.8 : Success of White Noise Detector

These are extremely encouraging results, and consequently this white noise detector was incorporated into the final simulation. If the detector indicated the presence of significant levels of white noise, the SBS Threshold was set to six, otherwise it was set to three. Using dynamic SBS Threshold allocation, the results for white noise and one added speaker are presented in Figure 8.8.

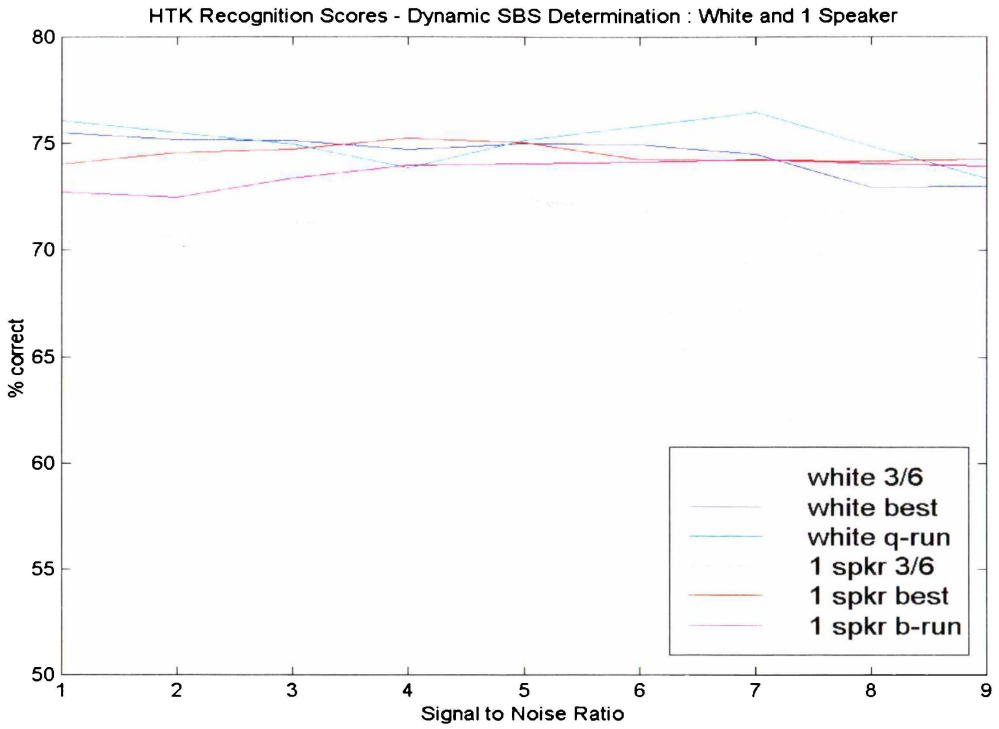


Figure 8.8 : Final White and One Added Speaker HTK Recognition Scores

The dotted blue and red lines (indicated by *white* (*1spkr*) *3/6* in the legend) are the recognition scores obtained using the dynamic SBS Threshold allocation and a training set that is similarly processed. The full blue and red lines (with legend title *white best* and *1 spkr best*) are the same dynamically allocated SBS test files, but compared to a training set using only SBS Threshold six training set (*q* run) if white noise was detected, otherwise the SBS Threshold three training set (*b* run) was used. Finally, the cyan line is the previous best result for white noise (from run *q*), and the magenta line the previous best result for the one speaker cocktail party noise (run *b*).

The dynamic SBS allocation is clearly a success, and for the first time, both white noise, and one speaker added noise results are at their maximum, in fact, better than any previous runs.

The results of multiple added speakers (Figure 8.9) again indicate extremely good performance with the dynamic SBS Threshold allocation. When the SBS Threshold three

training set (*b* run) is employed with this dynamic test-file Threshold setting, the response is superior to any previously obtained results (*b*-run).

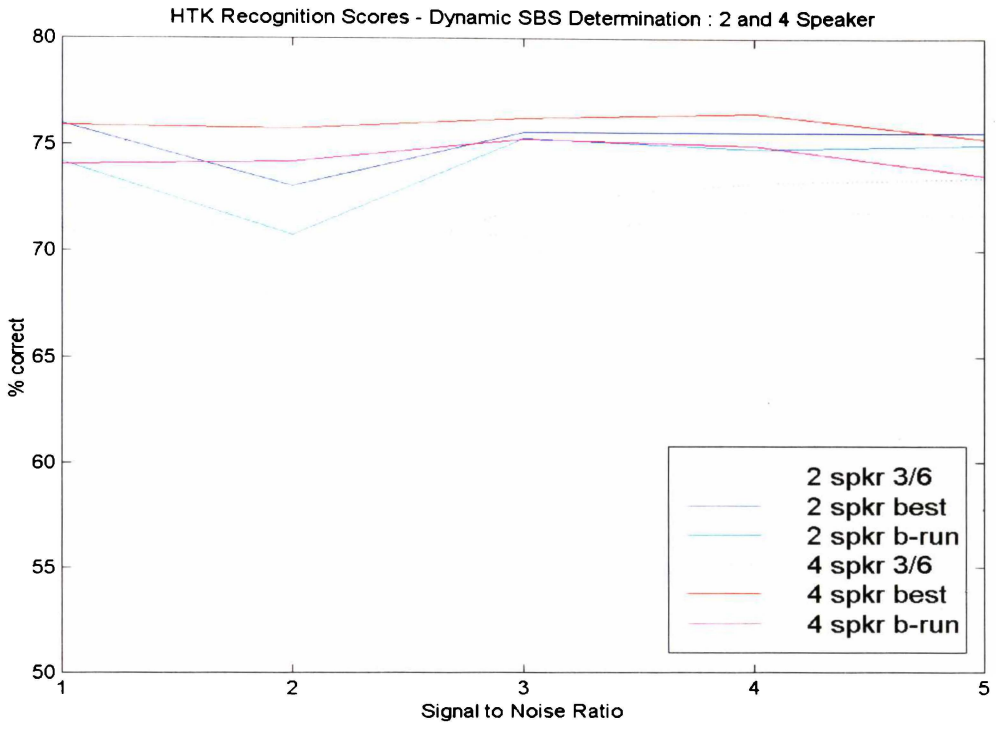


Figure 8.9 : Final Multiple Speaker HTK Recognition Scores

8.4 DRT Verification

To conclude the experiments, a DRT was undertaken on the final form of the simulation. Two male and two female speakers were recorded, and a program written to generate five variations of the DRT from their recorded words. Five simulation variations were processed for the four speakers, with five permutations of the DRT for each speaker. This resulted in a total of 100 DRT results. The Seneff files were listened to, but appeared (from informal tests) to be of much lower quality, and hence were not part of the DRT experiments.

The first series of DRT experiments (labelled *Orig* in Table 8.9), involved no SBS processing, only resampling from 44.1 kHz to 8 kHz, and then bandpass filtering with a passband of 300 - 3.3 kHz. The sub - 100% results for this run are indicative of the intelligibility loss that results due to the resampling/filtering as well as the recording and playback environments.

The next two tests (labelled *A* and *B*) used the final version of the simulation, including the energy normalisation and dynamic SBS Threshold allocation routines. Run *C* is a comparison with a high bit-rate system that will indicate the intelligibility loss resulting from the quantisation of the amplitude, phase, and frequency bins, as well as the reduced data overlap. The final run (*D*) uses only a single value for the SBS Threshold, and is included to compare the performance of the dynamic thresholding algorithm with the static threshold values that early simulations employed. This is summarised in Table 8.9.

Run Label	Simulation Parameters
Orig	300 – 3.3kHz band limited, sampling 8 kHz
A	3 bit amp/freq, 1 bit phase, 25% overlap, VUS=1, dynamic Threshold
B	3 bit amp/freq, 1 bit phase, 25% overlap, VUS=0.4, dynamic Threshold
C	6 bit amp/freq, 8 bit phase, 100% overlap, VUS=1, dynamic Threshold
D	3 bit amp/freq, 1 bit phase, 25% overlap, VUS=1, SBS Threshold = 6

Table 8.9 : DRT Experiment Parameters

The raw DRT results are presented in Table 8.10, showing a breakdown of the male and female responses. The DRT results for all speakers are plotted in Figure 8.10 .

	Intelligibility Result									
Word	**Run Orig**		**Run A**		**Run B**		**Run C**		**Run D**	
Type	**male**	**female**	**male**	**female**	**male**	**female**	**male**	**female**	**male**	**female**
Voicing	96.3	76.3	81.3	61.3	78.8	66.3	76.3	62.5	62.5	55.0
Nasality	98.8	100	100	95.0	98.8	97.5	98.8	98.8	98.8	91.3
Sustention	97.5	95.0	82.5	81.3	85.0	75.0	86.3	81.3	81.3	60.0
Sibilation	92.5	92.5	86.3	55.0	87.5	65.0	78.8	57.5	70.0	30.0
Graveness	90.0	85.0	65.0	52.5	78.8	45.0	68.8	46.3	58.8	37.5
Compactness	100	96.3	77.5	90.0	72.5	80.0	78.8	86.3	83.8	72.5

Table 8.10 : DRT Intelligibility Results Corrected for Guessing

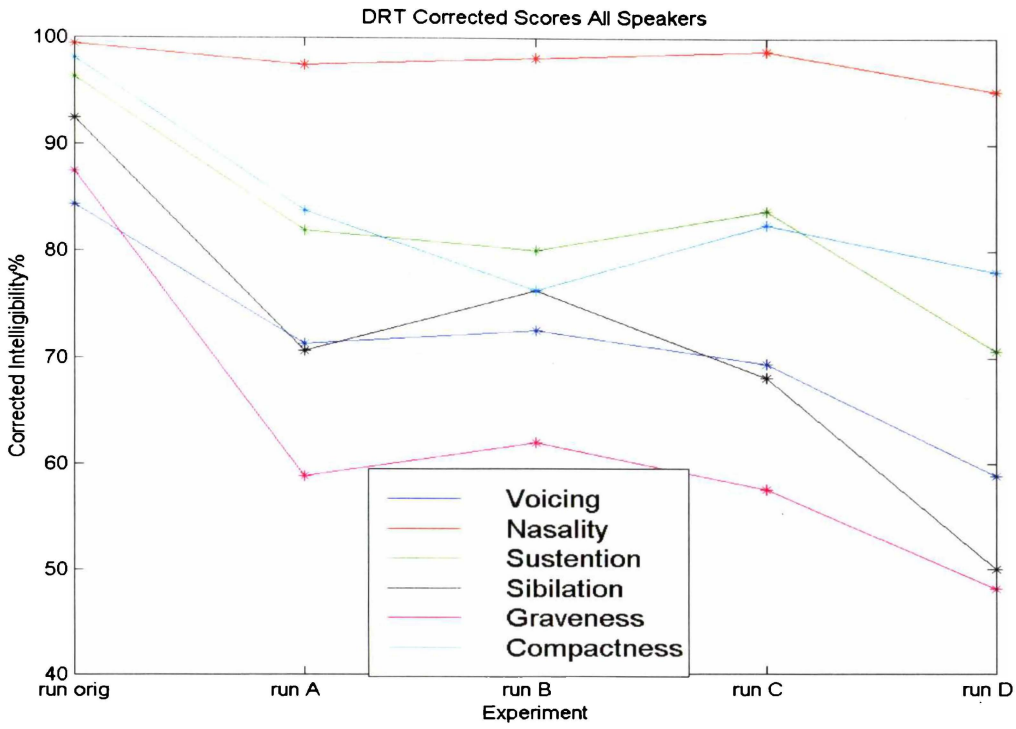


Figure 8.10 : DRT Results – Four Speakers

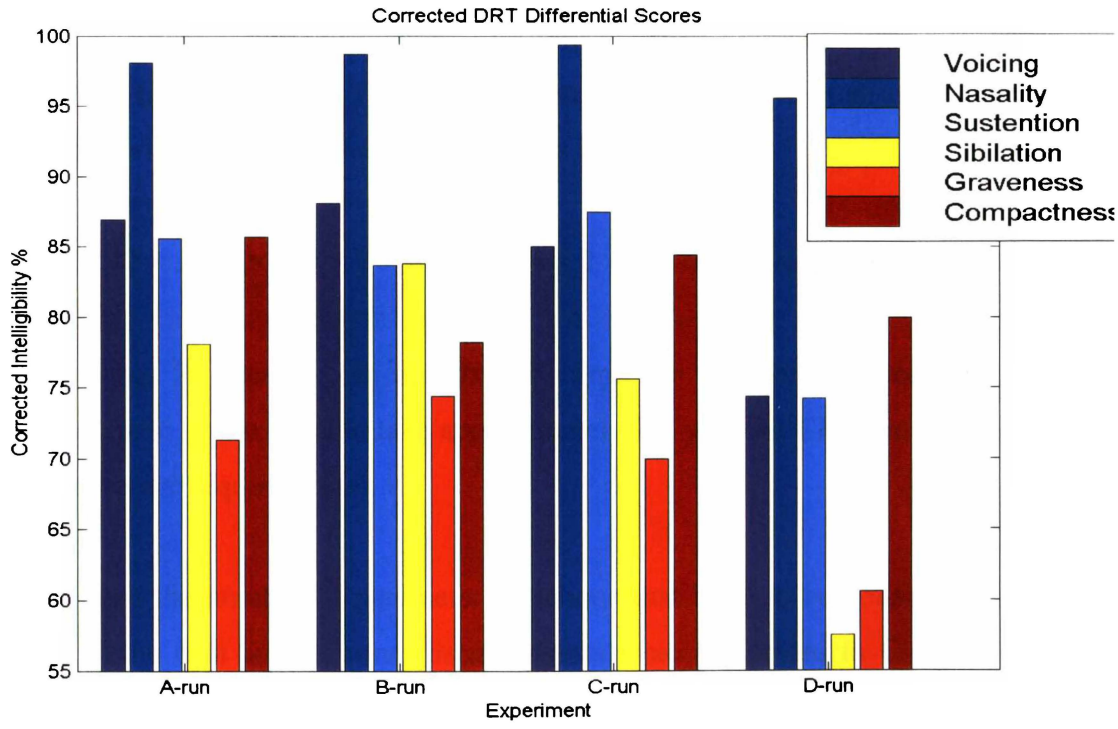


Figure 8.11 : Adjusted DRT Results

An immediate observation is that the results are considerably higher than those for the original DRT, as illustrated in Figure 5.1. To better view the results, they are replotted in Figure 8.11 with the intelligibility loss due to the testing procedure subtracted. The VUS modifier does increase the voicing score (run *B* compared to run *A*), as well as the scores for sustention, sibilation and graveness. Voicing is particularly improved for the female speakers.

Run *C* shows mixed results, though it is reasonable to conclude that the quantisation imposed on the speech generally does not decrease the overall intelligibility. Importantly, run *D* does show significantly lower intelligibility scores than runs *A* or *B*, and clearly indicates, that (with the exception of compactness) the dynamic SBS Threshold produces more intelligible speech.

Chapter 9

9 Summary and Conclusion

9.1 Parameter Elimination Methodology

An important consideration arising from this thesis is the methodology involved in the elimination of the number of parameters being considered. As an indication, amplitude quantisation can be either linear or non-linear (for example, logarithmic), and can theoretically vary from one bit to sixteen bits. Phase can vary from zero bits to sixteen bits, frequency bin representation (logarithmic and linear) from one bit to six bits, data-redundancy (twice frame overlap) from zero to 100%, the frame size could be either 128 or 256 points. Additionally, the speech could be preemphasised (non differenced or three forms of differencing), with a choice of (say) 10 VUS factors, use between one and 64 frequency components in the synthesis (indicated by SBS threshold), and employ either a logarithmic or linear bandpass filter structure. Finally, the speech could be processed using one of seven common windowing routines. Overall, this results in

$$2 \times 16 \times 17 \times 2 \times 6 \times 101 \times 2 \times 4 \times 10 \times 64 \times 2 \times 7 \approx 10^{10} \text{ combinations.} \quad \textbf{Equation 9.1}$$

If each test takes 20 seconds (an approximate time for the Ghitza processing), then the training set of 3696 phrases would take approximately 10^8 years of CPU time (not to mention considerable storage requirements).

To vastly reduce the number of parameter variations that have to be considered, numerical evaluations of the data and informal listening tests were performed in Chapters three and four. Two test vowels (AH and IY), and two test sentences (We were away a year ago, and Sally sells seashells by the seashore) were used as the initial test inputs. Examination of the numerical data determined the appropriate filter shape, and indicated that preemphasis could most efficiently be achieved by linear differencing. Numerical considerations also

determined a 16 ms, 128-point framesize, and crudely divided the data-redundancy issue to being characterised by 25, 50 or 100%. Informal listening tests then provided a fast means of eliminating further possibilities by subjectively detecting parameter variations that produced an obvious degradation in performance. Such tests indicated that little difference was noticed in SBS changes from seven to ten, (and so the SBS Thresholds of eight and nine were no longer considered), that amplitude could not be lowered below three bits without a serious loss of intelligibility, that three bit logarithmic frequency coding produced similar audible results to six bit linear, and that the Hamming was a good compromise window (that was also consistent with literature reports).

The next database was a selection of words from the MRT list. The testing with this database involved the variation of amplitude and phase representations, with the other parameters held constant. These tests provided further evidence of the lack of importance of the phase information, and also of the difficulty of reproducing fricatives with a reduced frequency set. At the conclusion of these informal tests, the system was constrained to the following parameter options:

Amplitude	Lower bound 3 bits
Phase	Suggestion that phase may be significantly reduced
Frequency quantisation	6 bit linear or 3 bit logarithmic
Data-redundancy	25%, 50%, 100%
Frame size	128 point (16 ms)
Preemphasis	Either non differenced or linear routine
SBS Threshold	0,1,3,4,5,6,7,10
Filter type	Height 18 – linear slope
Window type	Hamming
Other	Indication of very good noise immunity Indication of poor/stop fricative resolution

Table 9.1 : Reduced Simulation Parameter Values

These results were informative in that they provide a good qualitative indication of the performance of SBS processing, but a more quantitative measure was required to appropriately determine the relevance of the phase information, and to characterise the

performance of the system to the six different speech types. This was accomplished by the DRT test, the results of which indicated a failure of the system to successfully synthesise unvoiced speech.

To investigate the DRT results, the fourth database was a selection of voiced and unvoiced fricatives, stops and affricatives, that were subjected to intensive listening tests as well as numerical analysis. The numerical evaluation in particular was invaluable for determining how the SBS processing functioned. From this investigation, the loss of frame energy was identified as a major concern, and a routine was included in the simulation to compensate for this. Additionally, a Voiced/Unvoiced/Silence classifier was implemented to identify unvoiced fricatives, and synthesise these with more frequency components than for voiced speech. The simulation was now in its final form.

Amplitude	8,6,5,4,3, bit linear, 6,5,4,3,2 bit logarithmic
Phase	8, 6, 2, 1, 0, random bits
Frequency quantisation	2-6 bit linear, 2-5 bit logarithmic
Data-redundancy	25%, 50%, 100%
Frame size	128 point (16 ms)
Preemphasis	Yes, no
SBS Threshold	0,1,3,4,5,6,7,10
Filter type	Height 18 linear slope
Window type	Hamming

Table 9.2 : Reduced Simulation Parameter Values After DRT Testing

This provides

$$10 \times 6 \times 9 \times 3 \times 1 \times 2 \times 8 \times 1 \times 1 = 25{,}920 \text{ combinations} \qquad \textbf{Equation 9.2}$$

Using the Otago database narrowed the solution space down even further. This was the first multi-speaker New Zealand database employed, and yielded further qualitative information concerning the functioning of the simulation, particularly the values for the VUS modifier, amplitude and phase representations. The system was now ready for formal testing using the TIMIT database and the HTK ASR.

Amplitude	6 bit linear, 3 bit logarithmic
Phase	8 bit linear, 1 bit linear, 0 bit
Frequency quantisation	6 bit linear, 3 bit logarithmic
Data-redundancy	25%, 50%, 100%
Frame size	128 point (16 ms)
Preemphasis	Yes, no
SBS Threshold	0,1,3,4,5,6,7,10
Filter type	Height 18 linear slope
Window type	Hamming

Table 9.3 : Final Simulation Parameter Values

This provides

$$2 \times 3 \times 2 \times 3 \times 1 \times 2 \times 8 \times 1 \times 1 = 576 \text{ combinations} \qquad \textbf{Equation 9.3}$$

These 576 combinations would take approximately 1.3 years of continuous CPU time to process the TIMIT training data. However, not every combination needs to be considered. For example, the claim is that three bit logarithmic amplitude and frequency quantisation performs as well as a six bit linear code. Zero or one bit phase, should not exhibit inferior performance to 8 bit phase. It is sufficient to test these claims in isolation of most of the other variables. Similarly, the expectation is that the solution space minima will occur for an SBS Threshold of between three and seven. Threshold values of zero, one, two, and ten are included to both highlight simulation trends and compare the results with a "control", (i.e. every frequency selected in the case of SBS zero). The final testing scheme is that of Table 6.1, comprising 26 combinations, requiring 22.2 days of CPU for the training data. A final multi-speaker DRT is included to demonstrate the improvement of the new algorithm over the old one in terms of the six speech classifications.

In summary, informal testing on the simple vowels and sentences provided initial indications of simulation performance, and allowed the setting of some parameters, and the specification of upper and/or lower bounds for others. The first DRT identified specific failures of the

system; more informal testing provided the information necessary to correct these. Informal testing using a multi-speaker database further reduced the number of parameter variations to the extent that formal HTK testing was possible. These quantitative results were analysed both subjectively and objectively by a neural network, to determine the final optimal parameter set.

9.2 Results

The objective of this thesis was to investigate a reduced frequency representation of human speech, particularly in terms of its potential for low bit rate coding, and as a front end to ASR. The performance of the reconstructed speech has been measured by using informal listening tests, modified DRTs, HTK recognition results, and visual inspection of process variables and resulting spectra. Analyses of the results have been conducted using subjective conclusions from the above listed methods, as well as an inspection of squared error plots and neural network regression algorithms.

The original inspiration was Ghitza's In-Synchrony-Bands Spectrum model. Once DSP considerations were verified, intelligibility and recognition results identified some serious shortcomings in the model as presented by Ghitza. These flaws unacceptably degraded the performance of the system, and the SBS algorithm had to be substantially modified. These modifications include:

- Improvement on the intelligibility performance of unvoiced speech through the design and implementation of a Voiced-Unvoiced-Silence classifier
- Development of an algorithm to automatically amplify selected frequency bin energies
- The design and incorporation into the system of a white noise detection algorithm for dynamic SBS Threshold allocation

With these improvements to Ghitza's original model, factors for low bit rate coding retaining high intelligibility and recognizability had to be decided requiring:

- Determination of acceptable amplitude quantisation

- Determination of acceptable phase quantisation
- Determination of acceptable frequency bin quantisation
- Determination of acceptable frame-overlap
- Determination of appropriate unvoiced modifier values
- Determination of acceptability of Differencing
- Determination of best SBS Threshold for white and cocktail party noise

Loizou et al. (1999) have reproduced speech with a number of sine waves (between 2 and 16 spaced either logarithmically or by mel), and at various amplitude settings. The application for their study was to improve speech intelligibility in deaf individuals fitted with cochlear implants, given that it is not currently possible to provide fine spectral detail in such implants. They employed the TIMIT database for their testing, and evaluation was implemented by the subjects typing the sentences they had heard. Such a test makes no attempt to separate the paralinguistic information from the linguistic, and therefore yields little specific information on the deficiencies of the system.

This project has been an in-depth analysis of the frequency and amplitude quantisation that have been independently considered by Loizou et al., but has explored many more parameter variations. Additionally, the synchrony phenomenon has been exploited in this project to better determine what the speech synthesis frequencies should be. An application of the algorithm developed in this thesis would be, (as suggested by Loizou et al.), to cochlear implants – a feature not originally considered when this research commenced.

The final system not only has an inherent 97% reduction in the bit rate, but actually outperforms the original (non-SBS processed) speech as measured by the HTK Recognition System. As expected, the SBS processed speech is of significantly lower quality, but DRT results indicate that intelligibility remains high, higher than the original SBS model for the same bit rate.

9.3 Contributions

This thesis has made several contributions to research. These contributions include

- The development of a Voiced-Unvoiced-Silence classifier that uses the frame energy, and zero crossing values.
- The development of a white noise detector derived from the SBS spectra.
- The development of an energy redistribution algorithm to selectively amplify essential speech frequency bins, where energy has been lost from the system through windowing routines, selective frequency removal, or the elimination of phase.
- Code to automate HTK testing with a large training and test database
- Comparison of SBS, Seneff and LPC results using the HTK recognition system.
- Comparison of DRT results for various frequency domain parameter variation using the SBS model.
- Neural Network multiple regression analysis to determine variable sensitivity of the SBS model.
- Development of a low-bit rate speech analysis/synthesis system that performs extremely well in the presence of significant white and cocktail party noise.
- Initiation of a Waikato multi-speaker DRT database.
- The development of code to randomly generate DRTs, enabling informal Diagnostic Rhyme Testing to be performed without the listeners learning the testing patterns.
- The development of code to link Ghitza's SBS model to quantisation routines, VUS classifier, white noise detector, and frame interpolation software.
- The development of Matlab code to generate Stage III of Seneff's auditory model. This code complements Malcolm Slaney's Matlab toolbox.
- The investigation of the effects on speech of various forms of high-frequency preemphasis using numerical differentiation.
- The investigation of a computational bandpass filter model of the basilar membrane.
- The development of a sound experimental methodology for parameter elimination.
- An improvement over the reduced speech representation of Loizou et al. for cochlear implant applications.

9.4 Conclusions

In the introductory chapter, several questions were raised that this thesis answered. With respect to the broad categories of low bit rate coding and front end potential to ASR, it has been demonstrated that a synthesis system using only 2-3% of the original frequency domain information retains high intelligibility and exhibits superior HTK results to non-processed speech. These broad categories were broken down into specific questions in Chapter 1, the answers to which form a good summary of this thesis.

In general, seven frequency components provide highly intelligible speech, and maximise recognition performance in the absence of noise, or when other speakers are present. White noise levels are better suppressed when only five frequency components are used. The resulting bit rates are 3.8 kbit/s and 2.7 kbit/s respectively.

Ghitza's filter shapes perform better than the more physiologically accurate filters of Seneff, and better than any of the filter modifications trialled in Chapter 4. These filters additionally have the advantage of computational efficiency.

The DRT results of Chapter 5 indicated failures of the SBS processing to adequately preserve perceptually relevant phonetic information in the areas of voicing, nasality and graveness. The DRT results of the improved analysis/synthesis system (Figure 8.11) show considerable improvement in all three of these areas, as well as improvements in the sustention and sibilation scores. The HTK results indicate superior recognition to the original system, and as mentioned, are better than no processing at all. The noise immunity, (as measured by the HTK) particularly for white noise, is considerably better than both Seneff and LPC based systems.

The modified SBS has proven itself more suitable as a front end to an ASR than either the Seneff or the LPC models. Optimal quantisation levels of the amplitude, phase, and frequency representation have been found to maximise recognition performance, yet still retain high intelligibility. These levels were found both experimentally, and confirmed with neural network multiple regression variable sensitivity analyses. If intelligibility was no

longer an issue, it is possible that the quantisation levels for these factors could be further lowered, which may result in an even better HTK performance.

This thesis has more than achieved the goals as defined in the Introduction. An extremely noise immune system has been developed with potential use as a real-time front end system to an automatic speech recogniser, with the advantage that it can be transmitted at two or three percent of the original bit rate. The synthesised speech remains highly intelligible. The Voiced-Unvoiced-Silence classifier, and the White Noise detector may have a wider application than solely as a tool for this simulation.

9.5 Future Work

This thesis has been a thorough investigation of an SBS speech analysis/synthesis system, though there are several areas that warrant further investigation.

One immediate avenue for future work arises because informal DRTs were used in this thesis to obtain the intelligibility results. Formal testing by (say) Dynastat Inc., using professional recording and playback facilities, as well as trained listeners, would yield more reliable information, and perhaps better resolve the performance of the system with respect to the six categories of speech. If such testing indicated high intelligibility, then an additional avenue for future work would be the implementation of this algorithm into a cochlear implant.

A very important consideration that has not been addressed, is the maximum automatic speech recognition result that could be attained. This thesis has investigated recognition trends, and demonstrated that the final version of the SBS simulation outperforms a classical non-biological algorithm (LPC10), a more physiologically accurate model (Seneff), and original unprocessed speech. What was not obtained was an absolute recognition score using a modern recognition system. The assumption is that the reduced information set input into the recogniser helps in identifying essential parameters of the speech, and improves recognition. It would be interesting to use this SBS model as a front end to a modern automatic recognition system and compare its performance to contemporary recognition results.

The SBS system has been developed in Matlab. Matlab is an interpreted environment, and although careful vectorisation of the code, and the inclusion of pre-compiled MEX code has decreased the time it takes for the simulation to execute, it is still slow. To be more useful, the code could be ported back to C, optimised and compiled, and run as an ordinary executable. A hardware implementation of the filter system would also dramatically increase the execution speed, and between these two implementations, the simulation could be performed in real-time. Any application for cochlear implants would require such a speed increase.

Effects of masking have been incorporated into the parameter quantisation, but this has not been implemented in a frequency dependent fashion. The masking curve peaks at approximately 60 dB for frequencies centred around 500 Hz (Tempelaars, 1996), with an asymmetrical drop-off to 100 Hz on the low frequency side, and approximately 5 kHz on the high frequency side. This frequency dependence could be exploited to further lower the bit rate of the system. It is unknown what effect this would have on ASR results.

Finally, various compression systems (for example, Vector Quantisation) could be implemented on this reduced instruction set. Of interest would be the recognition score of this additionally compressed waveform compared to the quantised SBS system. Depending upon the results, the compression routine could be used either purely as a carrier of the SBS information, or incorporated into the front end of the automatic recogniser.

Appendix A. Mathematical Preliminaries

A.1 Overview

The auditory model simulations developed in this thesis require Digital Signal Processing (DSP) techniques to filter, interpolate, decimate, differentiate and transform data. For example, the simulation was originally designed to process speech sampled at 8 kHz from the Artisoft™ sound board, and filtered to telephone bandwidth. If a different sample rate is used, then sample rate conversion is necessary. If the new sample rate is to be higher, the process is generally called interpolation since we are creating samples of the original physical process from a reduced set of samples. The conversion to a lower rate is termed decimation (Crochiere, Rabiner, 1981). The conversion between arbitrary sample rates will, in general, require interpolation (and/or decimation), and the appropriate anti-aliasing filtering (to avoid the introduction of spurious frequencies).

The interpolation process may be carried out in either the time or frequency domain. Mathematical algorithms derived from time domain interpolation (such as those based on difference techniques or interpolating polynomials), are intuitively simple and straightforward to implement. However, the frequency domain characteristics of such techniques need to be considered for digital signal processing implementations, and alternative strategies based on FIR (Finite Impulse Response) filters are often necessary. These filter based techniques are referred to as frequency domain algorithms throughout this thesis.

The software implementation of Butterworth or Chebyshev filters require the use of the bilinear and the z transforms. Speech frequency analysis requires the use of Fourier transform techniques, and to be computationally efficient, the Fast Fourier Transform. Also, in speech processing, it is common to provide a high frequency gain to the speech data, and this is often implemented by numerical differentiation.

This Appendix provides an overview of the DSP algorithms required to implement these tasks. Additional details and derivations of the algorithms are presented in Appendices B, C and D.

A.2 Filtering

To cater for different sources of speech input, and the possibility of a different sample rate, a Butterworth IIR anti-aliasing filter was implemented in the simulation. On testing the performance of this filter with a large number of test speech frames from the TIMIT database (Section 3.7.2), it was decided to replace the Butterworth filter with a Chebyshev in order to obtain a sharper attenuation slope. The 1 dB ripples introduced by the Chebyshev function were not audibly noticeable (under test conditions), and would be masked in any event by the amplitude quantisation that the simulation imposes.

A tenth order Chebyshev filter was eventually used, with a 1 dB passband ripple, and a cutoff frequency of 3.3 kHz. This provides suitable attenuation characteristics and an acceptable level of phase shift (given that much of the phase information is deleted in the later quantisation experiments). No claim is made that this is the best form of anti-aliasing filter, merely that it fulfils the requirements of the auditory simulations, and is relatively straightforward to design and implement following the guidelines presented in Appendix A (which also includes the justification for employing a Chebyshev filter rather than a Butterworth, Elliptic or Bessel, and a discussion of FIR and IIR filters).

A.3 Interpolation

Interpolation produces intermediate values for discrete data. For example, if samples are at one millisecond intervals, half millisecond values may be obtained by suitable interpolation. In mathematical terms, interpolation techniques tend to fall into one of three categories—those based on forming an interpolating polynomial (Lagrangian Interpolation), those based on successive differences (principally Gaussian or Newtonian techniques), and those based on splining algorithms. Cubic splining is the most popular form of splining, where an attempt is made to match a series of tabular points with a third order polynomial in such a way that successive polynomials and their derivatives are continuous.

In digital signal processing, however, many techniques for decimation and interpolation for arbitrary sampling rate conversion are performed in the frequency domain by application of an appropriate FIR or IIR filter. Such techniques for interpolation involve the addition of *L*-

1 zeros (where L is the order of interpolation) between each tabulated time domain point, and the higher sampling rate is read from the filter output. Each method has advantages and disadvantages, and the particular method chosen depends on the application - the amount of error that can be tolerated and available computing time. Appendix C contains more detail on these techniques.

The simulation initially uses Newton Forward Difference interpolation formulae, the first three terms of which are:

$$P_1(s) = f_0 + s(f_1 - f_0)$$

Equation A.1

$$P_2(s) = f_0 + s(f_1 - f_0) + \frac{s(s-1)}{2}(f_2 - 2f_1 + f_0)$$

Equation A.2

$$P_3(s) = f_0 + s(f_1 - f_0) + \frac{s(s-1)}{2}(f_2 - 2f_1 + f_0) + \frac{s(s-1)(s-2)}{6}(f_3 - 3f_2 + 3f_1 - f_0)$$

Equation A.3

where

$$s = \frac{x - x_0}{h}$$

Equation A.4

Towards the end of a data array, forward difference interpolation will not be possible, and so the corresponding backward difference form is utilised:

$$P_1(s) = f_0 + s(f_0 - f_{-1})$$

Equation A.5

$$P_2(s) = f_0 + s(f_0 - f_{-1}) + \frac{s(s+1)}{2}(f_0 - 2f_{-1} + f_{-2})$$

Equation A.6

$$P_3(s) = f_0 + s(f_0 - f_{-1}) + \frac{s(s+1)}{2}(f_0 - 2f_{-1} + f_{-2}) + \frac{s(s+1)(s+2)}{6}(f_0 - 3f_{-1} + 3f_{-2} - f_{-3})$$

Equation A.7

For the same level of computation, central difference schemes are inherently more accurate than either the forward or backward techniques (Appendix C). The Stirling scheme is derived from taking a mean of Gaussian forward-difference and backward-difference formulae, and the first two terms are:

$$P_1 = f_0 + \frac{s}{2}(f_1 - f_{-1})$$

Equation A.8

$$P_2 = f_0 + \frac{s}{2}(f_1 - f_{-1}) + \frac{s^2}{2}(f_1 - 2f_0 + f_{-1})$$

Equation A.9

Examination of the numerical data from each of these schemes yields errors in accordance with that suggested by analytical investigations of the algorithms. No technique utilising anything higher than third order (forward or backward) approximations altered the audible nature of reconstructed speech, hence the difference techniques, being the simplest and least computationally intensive, have been retained. Third order forward and backward techniques yield similar errors to the second order central technique, the latter being preferred (except at the array end-points) as it is the more computationally efficient.

A.4 Differentiation

The simulation provides the user with the ability to apply high frequency emphasis to the input speech. For some variations of the cochlear filter functions (Chapters 2 and 3), such high frequency emphasis greatly improves the intelligibility of the synthesised speech. Common techniques for high frequency emphasis involve differencing the data, where differencing is effectively a first order forward numerical differentiation. Various methods of differentiation were investigated during the course of this thesis in an attempt to improve the results of simple differencing. These methods essentially involve differentiating some interpolating polynomial (for example, those developed for the sample rate conversion routines discussed above).

When experimenting with the effects of altering the incline and decline slopes of the bandpass filters used in the construction of the SBS spectrum (Section 4.9), it was noted that

a linear slope has inherently poor high frequency response, and much of the high frequency information was masked. In an attempt to recover some of this information, the input speech was subjected to various high frequency emphasis routines based on a form of "differencing". Differencing in its simplest form, merely involves the subtraction of data point f_0 from its succeeding data point f_1, i.e.

$$\Delta f_0 = f_1 - f_0 \qquad \textbf{Equation A.10}$$

This differencing is almost identical to the general case of a first order differentiation procedure, obtained from Newton's forward difference scheme,

$$f_1'(x) = \frac{\Delta f}{h} = \frac{f_1 - f_0}{h} \qquad \textbf{Equation A.11}$$

As is apparent, an ordinary differencing scheme varies from the first order differentiation only by a constant which is determined by the sample rate. However, the errors involved in this first order approximation are rather significant in purely numerical terms (Appendix C), so an investigation was performed to determine whether or not a technique with increased accuracy would improve the performance of the model.

The obvious extension was to take the generalised form of Newton's forward difference technique, and differentiate. The next two terms in the resulting expansion are:

$$f_2'(x_0) \approx \frac{-f_2 + 4f_1 - 3f_0}{2h} \qquad \textbf{Equation A.12}$$

$$f_3'(x_0) \approx \frac{2f_3 - 9f_2 + 18f_1 - 11f_0}{6h} \qquad \textbf{Equation A.13}$$

Again, Appendix C contains the derivation of these formulae, their errors, and the expression of the third and fourth order terms. Using Stirling's Central difference method and differentiating, the first two terms in the resulting expression are:

$$f'(x_0) \approx \frac{f_1 - f_{-1}}{2h} \qquad \textbf{Equation A.14}$$

$$f'(x_0) = \frac{-f_2 + 8f_1 - 8f_{-1} + f_{-2}}{12h}$$

Equation A.15

Compared to forward or backward differentiation techniques, the central difference algorithms offer improved accuracy for the same computation time. Consequently, the options presented in the simulation for high frequency emphasis are linear differencing, first order central differencing, and third order central differencing. The results are presented in Section 4.8.

A.5 Time Domain Decimation

Decimation (or integer ratio downsampling) involves the removal of excess data points from sampled speech input. For example, if the speech is sampled at 16 kHz, but is to be processed at 8 kHz, the process of decimation is simply the removal of every second data point. However, the process becomes rather more complicated for conversion between arbitrary sampling frequencies. An intuitive method, involving a combination of interpolation, filtering and decimation involves first interpolating the signal until the number of data points in the speech to be input to the simulation is some integer value of the sample rate used in the simulation processing. For example, if the speech is sampled at 20 kHz, but the processing requires a sample rate of 8 kHz, a simple method of conversion is to interpolate the input data to 24,000 samples/second, lowpass filter to the appropriate cutoff frequency, then decimate by retaining one sample in every three.

Speech input for this simulation originated from three sources. The first was speech sampled at 8kHz via a speech acquisition board on the PC, the second was from an 8 kHz *.wav* file. For low-bit rate coding, and conversion simplicity, the simulation was designed to operate at a default sampling rate of 8 kHz. The third source was from the TIMIT data base, sampled at 16 kHz. This makes a time domain sample rate conversion straightforward - lowpass filtering with a cutoff of 3.3 kHz (to ensure sufficient attenuation by 4 kHz) and then decimation by eliminating every second data point.

The intuitive simplicity and coding efficiency of linear interpolation routines make them a popular option in signal processing applications, however, such interpolation is

fundamentally a linear filtering process (and was experimentally observed to affect the synthesised speech). Whilst higher order interpolations in some way compensates for this, FIR and IIR digital interpolation filters often provide superior results.

A.6 Frequency Domain Sample Rate Conversion

In comparison to classical time domain techniques, FIR and IIR interpolation methods provide significant improvement in terms of their frequency domain response. Matlab simplifies the process of employing these filter interpolation techniques, and an overview is presented below (though the formal details and derivations have been left to Appendix C).

Matlab provides interpolation, decimation and resampling routines using FIR filters of arbitrary order, with a default to an eight order, lowpass, Chebyshev type I filter (for further details refer to Appendix B). The interpolation process inserts zeros between the original data values to expand the input vector to the correct length. Then the application of a symmetric FIR filter allows the original data to pass through unchanged and interpolates between so that the mean-square errors between the interpolated points and their ideal values are minimized.

A.7 Data Transformation

A.7.1 Speech Reconstruction Using Sinusoids

Implicit in the SBS model is the processing of speech data in the frequency domain, and in fact, the reason for using SBS in speech compression is that it can eliminate a considerable amount of information in this domain, with the claim that intelligible speech can still be reconstructed. SBS in essence, selects only a few "essential" frequencies, and then reconstructs the speech by a sum of weighted sinusoids.

The literature details many analysis/synthesis systems that endeavour to reproduce some physiological or analytical behaviour of the speech in terms of a sinusoidal model. Examples include the use of modulated sinusoids to capture formant structures and the modelling in terms of frequency modulated sine waves of the vocal cord excitation and vocal

tract system functions. McAulay & Quatieri (1986) have investigated the latter by assuming the speech waveform to be the output of passing a glottal excitation waveform through the vocal tract whose characteristics are modelled as a linear time-varying system.

The general sinusoidal model represents the original speech signal $x[n]$ in terms of amplitude weighted frequency modulated sine waves (Smith, 1988). The synthesised speech $s[n]$, will closely resemble the original if a sufficient number of sinusoidal terms are employed. The frequency of the sine waves can most easily be assigned as the central frequency of each bin in the frequency domain, the spacing of which is determined by the number of samples in each window frame and the required bandwidth. This is discussed further in Chapter 3.

A.7.2 Frequency Domain Transformation

In order to perform frequency domain processing on speech data, the sampled data must first be transformed from the time domain. This requires that the speech be divided into appropriately sized frames and windowed prior to the application of some transform technique, for example, the Discrete Fourier Transform (DFT).

A time frame of 16 milliseconds was chosen to ensure that the pseudo-periodic nature of the speech is captured, and hence will be resolved in the frequency domain. This size frame contains 128 data points (assuming an 8 kHz sample rate), which enables the implementation of a Fast Fourier Transform (FFT), rather than the much slower DFT. A 128 point sample, spanning the frequency range 0-4 kHz, results in a spacing of frequency bins of 62.5 Hz, which will be the maximum error associated with the assigned frequency of the sine waves used in the speech reconstruction.

In order to prevent discontinuities at frame boundaries, the frames were initially overlapped by 50% at each end. The amount of frame overlap was varied from this maximum of 50% to 0%, the results of which are discussed in Chapter 4. However, 50% overlap, with linear interpolation to determine the valid time data points, effectively eliminates the discontinuities that result from the frame division (Section 4.6). Spectral leakage will be a significant problem in any such analysis, and so the form of windowing function implemented could have a large effect on the resynthesised speech.

A.8 Windowing

As mentioned in Section 4.3, the shape of the windowing function has a major influence on the characteristics of the reconstructed speech. A selection of windows were trialled in this simulation, including the Rectangular, Triangular, Hanning, Hamming, Blackman, Blackman-Harris and the Kaiser. Critical parameters of windowing functions include their coherent gain, spectral leakage, and height of the side-lobes and the side-lobe fall-off.

A.8.1 Coherent Gain

With the exception of the Rectangular window, all other windowing routines introduce some processing loss due to the window reducing the data to zero (or near zero) values near the boundaries. A measure of this effect is the coherent gain (sometimes given as coherent power gain, the square of the coherent gain), and is normalised so that the rectangular filter possesses a coherent gain of unity. All non-rectangular filters will have a sub-unity value of coherent gain, the lower the coherent gain figure, the more substantial the processing loss. The coherent gains for the windows used in this simulation are provided below in Table A.1 (Harris, 1978).

Window	Coherent Gain
Rectangular	1.0
Triangular	0.50
Hamming	0.54
Hanning	0.50
Blackman	0.42
Blackman-Harris	0.36
Kaiser	0.38

Table A.1 : Coherent Gains for Simulation Windows

These figures were experimentally verified by testing the simulation windowing routines with sine waves truncated by the frame size at both integral and non-integral periods. The

coherent gain then, is equivalent to the ratio of the sum of the amplitudes of the sampled values in the frame before windowing, to the sum of the amplitudes after windowing.

A.8.2 Spectral Leakage

Having eliminated the Triangular and Rectangular frames as appropriate window options, the coherent gain figures support the case for either the Hanning or the Hamming. However, the Blackman-Harris and Kaiser-Bessel functions have excellent side-lobe attenuation (92 dB and 82 dB respectively), and were investigated further to determine their effectiveness at suppressing spectral leakage in this simulation.

Spectral leakage results because the time domain data must in some way be truncated in order to implement the DFT. If a periodic function is truncated at any point other than at a multiple of the period, a sharp discontinuity in the time domain (which is equivalent to side-lobes in the frequency domain) results. These side-lobes are responsible for the additional frequency components which are termed spectral leakage. In fact, a given spectral component, say at $\omega = \omega_0$ will contribute output (i.e. will be observed) at another frequency, say at $\omega = \omega_a$ according to the gain of the window centred at ω_0 and measured at ω_a.

Leakage causes a bias in the amplitude and the position of a harmonic estimate and makes the detection of small signals in the presence of nearby large signals difficult. To reduce the effects of this bias, the window should exhibit low-amplitude side-lobes far from the central main lobe, and the transition to the low side-lobes should be very rapid. One indicator of how well a window suppresses leakage is the peak side-lobe level (relative to the main lobe); another is the asymptotic rate of fall-off of these side-lobes. These truncation functions are the windowing routines discussed below, and in Appendix D (summarised from Harris, 1978).

This leakage is a result of the inability of the analysis procedures to recognise harmonic components whose periods are not integer multiples of the basic frequency $1/T$. A pure sinusoid of a frequency which is somewhere between two frequencies i/T and $(i+1)/T$, would appear under Fourier transformation as a combination of integer frequency components (Beauchamp & Yuen, 1979). Under Fourier analysis, the power of the input sinusoid is

distributed over a set of neighbouring frequencies, thus "smearing" the spectrum. The shorter the analysis interval (i.e. window length), the worse the problem, whereas an infinitely long interval would eliminate leakage completely.

This convolving of the input spectrum with the transform of the observation window can be demonstrated by analysis of a pure sinusoid input, and was carried out early in the development of the simulation to investigate the exact effects that the windowing function produced.

A.8.3 Sinewave Experiments

The spectrum leaks into adjacent bands due to the sidelobes of the transform of the observation window. Windowing functions with large sidelobes, the rectangular window for example, will aggravate this leakage problem; other windows with smaller side-lobes will exhibit less leakage. To verify this experimentally for the simulation, all windows were tested with inputs of a pure sine wave at 500, 650, 800, 1250 and 1400 Hz. Frequency resolution during these experiments was 62.5 Hz (8 kHz sampling with frame size of 128 points); 500 Hz and 1250 Hz are integer multiples of the frequency resolution; the others are not. The 500 Hz sine wave should only contribute to frequency bin number eight, the 1250 to bin 20, whilst the others could expect to be represented over two bins.

Ideally, all frequency components other than the one or two bins containing the test sine wave should be zero. Leakage will be observed by the creation of additional frequency components when the time truncation interval is not chosen equal to a multiple of the period, i.e. for all tested frequencies other than 500 and 1250 Hz. A simple measure of this is to compare the energies in the bins surrounding the central peak to the sum of all the amplitudes in the frequency domain. A high level of leakage will be reflected in a low percentage of energy in these central bins. Note that it is necessary to consider the surrounding bins, and not just the energy contained in the central bin, as the non-zero frequency components are considerably broadened or smeared with respect to the desired impulse function. This is to be expected since the effect of time domain truncation is to convolve the frequency impulse function with the Fourier transform of the truncation function. In general, the more one reduces the leakage, the broader or more smeared the

results of the discrete Fourier transform appear (Brigham, 1974). The results are shown for the test sine waves below in Table A.2.

	Periods	Rectangular	Triangular	Hanning	Hamming	Blackman	Black-Harris	Kaiser
500(1)	8	100	93	100	100	92	85	90
500(3)		100	97	100	100	100	100	100
650(1)	10.4	51	90	96	91	98	95	97
650(3)		62	95	99	92	100	100	100
800(1)	12.8	58	91	97	94	98	94	97
800(3)		67	96	99	94	100	100	100
1250(1)	20	100	92	100	100	92	85	90
1250(3)		100	96	100	100	100	100	100
1400(1)	22.4	48	90	96	90	98	95	98
1400(3)		57	95	99	90	100	100	100

Table A.2 : Energy Distribution for Simulation Windows

It may be argued that this is not the most appropriate measure of energy distribution, and that, for example, root-mean-squared (rms) values should be evaluated. However, we are only interested in the comparative performance of the windows, rather than absolute values, and so the above method is adequate. An indication of the smearing of the central frequency bins can be seen by comparing the two rows of data for each frequency. The first row reflects the amplitudes of the one bin each side of the central peak, whilst the second (indicated by a three in parentheses), is the ratio of three amplitudes each side of the central peak to the total frame energy. The latter data is plotted below in Figure A.1.

For the non-integer sampling interval (the general case), the rectangular window was the least effective in reducing leakage (as indicated by the low energy retention percentage), followed by the Triangular and then the Hamming window. The distinction becomes less obvious from this point, but the data indicates that in general both the Kaiser and the Hanning are better performed than the Blackman-Harris, and the best performer for this criteria is the Blackman. Unfortunately, this is not the trend of the coherent gain performance, and so any window selected will be a compromise, and hence the Hamming is generally employed as the best compromise.

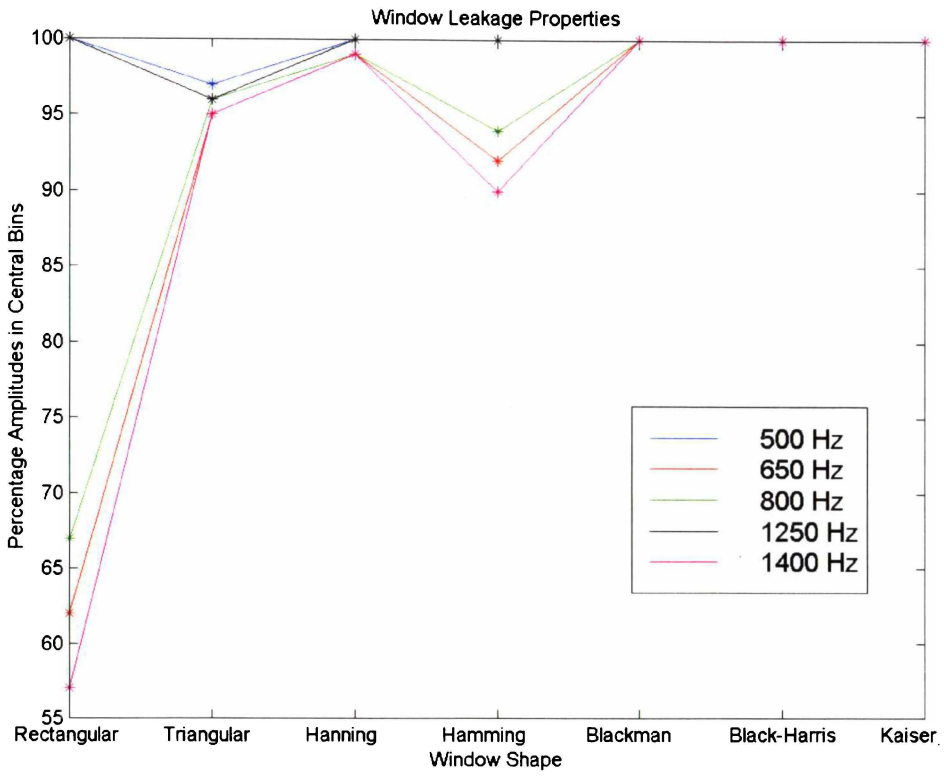


Figure A.1 : Window Leakage Properties

For the non-integer sampling interval (the general case), the rectangular window was the least effective in reducing leakage (as indicated by the low energy retention percentage), followed by the Triangular and then the Hamming window. The distinction becomes less obvious from this point, but the data indicates that in general both the Kaiser and the Hanning are better performed than the Blackman-Harris, and the best performer for this criteria is the Blackman. Unfortunately, this is not the trend of the coherent gain performance, and so any window selected will be a compromise, and hence the Hamming is generally employed as the best compromise.

Appendix B : Filtering

B.1 Overview

During the initial stages of this thesis, input speech was originating from at least two different sources, namely the Artisoft acquisition board, and the TIMIT database (Section 3.7.2). The former was sampled at 8 kHz, and became the default sampling rate for this thesis (given that the emphasis was on telephone bandwidth speech), the latter was sampled at 16 kHz. A sample rate conversion process was coded in C, and made to be very general, able to handle a variety of different sample rate conversions (including up-sampling if necessary). These conversion routines are discussed further in Appendix C. However, anti-aliasing filters were required for this sample rate conversion, and (at the time) there were no available libraries from which to obtain the required code. Consequently, much effort was given to researching the appropriate filter type, and how to implement these filters in software. This Appendix compares the properties of the filter types investigated, and provides the necessary details to facilitate their software design and implementation.

B.2 Digital Approximations of Analog Filters

Bandpass filtering is performed on the speech-data acquisition board used to record many of the test speech frames. However, for speech input that originates from alternative sources sampled at frequencies above 8kHz, some form of lowpass filtering will be necessary. Lower sample rates do not require filtering, but the value of the highest frequency bin will be reduced due to the Nyquist sampling theorem. There are many possible choices for the form the anti-aliasing filter may take; some alternatives and their merits are considered in the following discussion. Once a suitable form of filter has been selected, the digital approximation to that filter with the required amplitude attenuation, phase characteristics and computational complexity must be designed.

B.2.1 Overview of Analog Filter Types

It is sufficient to discuss all filter types in terms of a lowpass filter, as other filter types (bandpass, highpass etc.) are readily derived from them. The linear amplitude response of the ideal lowpass filter is shown below.

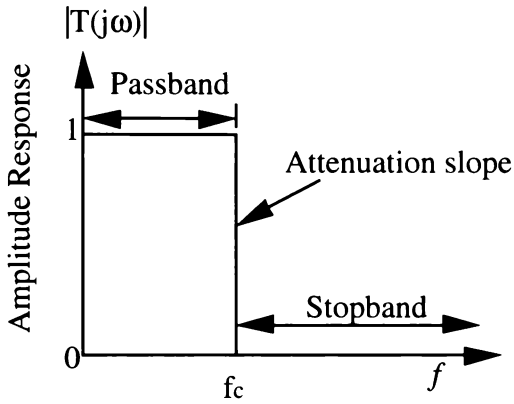


Figure B.1 : Frequency Response of an Ideal Low Pass Filter

This ideal filter is characterized by zero loss and ripple in the passband, an infinite attenuation slope at the cutoff frequency f_c (i.e. a zero transition region) , and an infinite attenuation in the stop band. In general, linear phase response is also assumed. However, such an ideal lowpass filter does not exist in practice, and we must implement suitable approximations. The best, and most common, are briefly described in the following sections (summarised from Moschytz & Horn, 1981). The plots shown are of fifth order filters designed with the same cutoff frequency[1]. The gain is expressed in a logarithmic format to illustrate ripple behaviour in the stop band (for some filters) that would not be evident with a linear response.

B.2.1.1 Maximally Flat or Butterworth Filters

The maximally flat (or Butterworth) approximation of an ideal lowpass filter is shown in Figure B.2. This filter incorporates a maximally flat response in the passband, achieved at the expense of phase linearity and attenuation slope steepness in the transition region from passband to stopband.

The Butterworth is a commonly used filter as it provides an excellent general-purpose approximation to the ideal filter response. Although other filters may have superior phase linearity, impulse response or steepness of attenuation, the Butterworth forms the best compromise. The Butterworth is also employed when no ripple is desired in either the pass or stop bands.

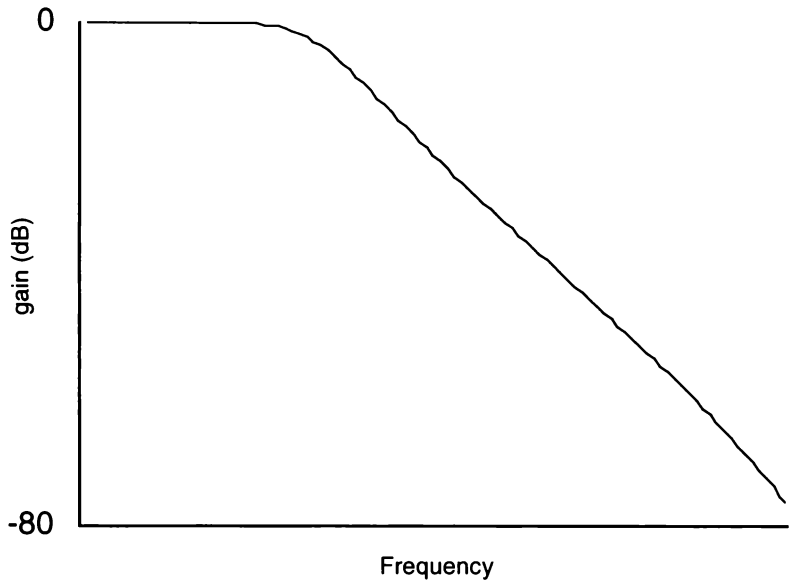


Figure B.2 : Frequency Response of a Butterworth Low Pass Filter

B.2.1.2 Equiripple or Chebyshev Filters

Steepness of attenuation slope, especially in the region of cutoff, is often more important than passband flatness or phase linearity. In such cases the Chebyshev response (Figure B.3) is often preferred over the Butterworth. The Chebyshev filter exhibits increased overshoot when driven by a step function, and is designed so that ripples in the passband never exceed a prescribed amount. To compensate for the lack of smooth response, there are advantages in a very much higher rate of cutoff around the edge of the passband.

---

[1] The order of a filter is given by the degree of the highest polynomial of s in the denominator of the filter's transfer function. This is covered in more detail in a later section in this chapter. (Sedra and Smith, 1987)

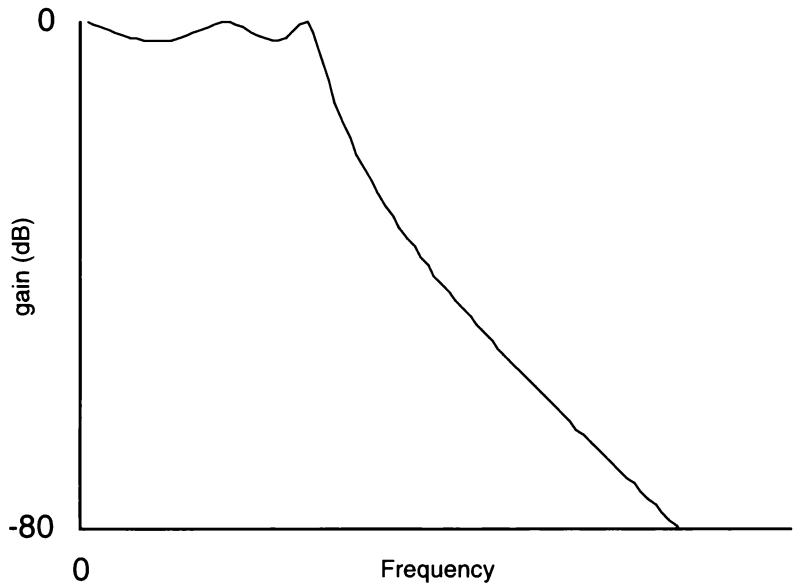


Figure B.3 : Frequency Response of a Chebyshev Low Pass filter

Note that both Butterworth and Chebyshev lowpass filters achieve infinite attenuation only at infinite frequency, (where all the zeros of transmission occur). At any other frequency, some component of the signal, even though greatly attenuated, will pass through the filter. If infinite attenuation at particular frequencies are required, the inverse Chebyshev response (Figure B.4) may be used. In this situation, there is no ripple in the passband, but it does exist in the stopband (where certain frequencies are infinitely attenuated).

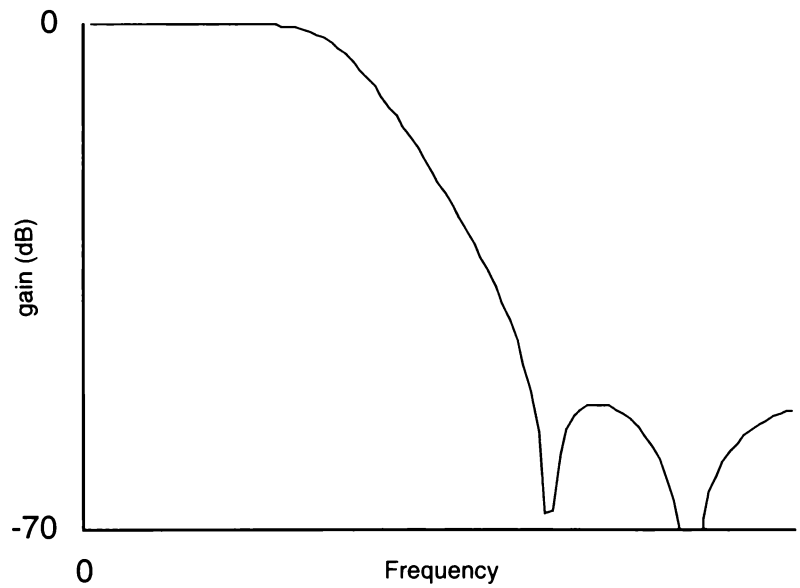


Figure B.4 : Frequency Response of an Inverse Chebyshev Low Pass Filter

B.2.1.3 Elliptic or Chebyshev-Cauer Filters

Elliptic or Chebyshev-Cauer Filters (Figure B.5) have a very rapid descent to the maximum attenuation, achieved at the expense of introducing ripple in both passband and stopband. Like the inverse Chebyshev filter, certain frequencies are infinitely attenuated. Elliptic filters are probably the most efficient in terms of component count[2] for approximating the amplitude response of an ideal filter. For a given filter order it is possible to produce filters more economically with either a very sharp cutoff or a very high attenuation in the stopband. On the other hand, the attenuation does not drop off smoothly to infinity outside the passband, but is maintained at a predetermined level. Note that Chebyshev and inverse Chebyshev filters are special cases of the more general Chebyshev-Cauer filter.

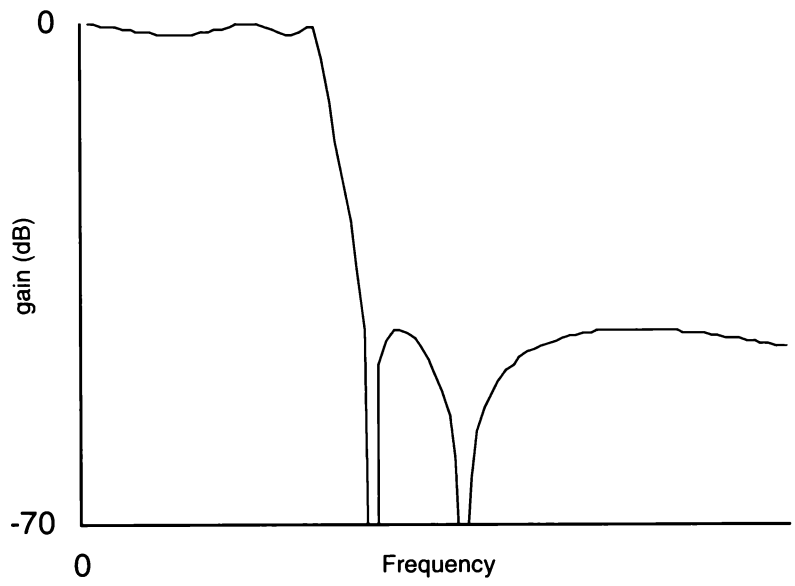


Figure B.5 : Frequency Response of an Elliptic Low Pass Filter

B.2.1.4 Optimum-Monotonic or Legendre Filters

In some applications, the attenuation slope of a Butterworth filter is inadequate and the ripple of a Chebyshev filter is undesirable. Designing a Chebyshev filter that will have a very small or zero ripple does not help because Chebyshev and Butterworth filters are of the

---

[2] In a hardware implementation, for a given number of active and passive components, an elliptic filter design will generally exhibit superior filter characteristics to any of the others discussed.

same family[3]. A solution may be to use a Legendre or optimum-monotonic filter whose amplitude response is shown in Figure B.6 together with the response of a Butterworth filter for comparison. The Legendre response sacrifices some flatness in the passband for a greater attenuation slope. A typical property of Butterworth and Legendre filters is their monotonic character, i.e. for any value of gain there is a unique frequency (unlike the Chebyshev and elliptic filters which exhibit ripple). The Legendre filter attempts to combine the best of the Butterworth and Chebyshev characteristics, where attenuation slope is made as steep as possible - with the restriction that the response remains monotonic.

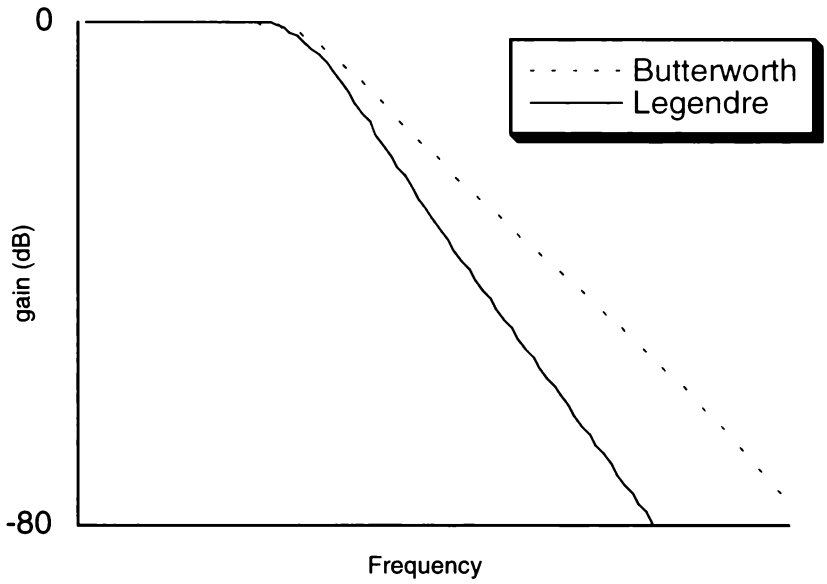


Figure B.6 : Comparison of the Frequency Response Between a Legendre and a Butterworth Low Pass Filter

B.2.1.5 Linear Phase or Bessel Filters

In the discussion so far, the emphasis has been on the amplitude response, however other factors must also be considered in filter design. When a rectangular pulse is passed through a Butterworth, Chebyshev or Legendre filter, overshoot or ringing will appear on the pulse at the output. If this is undesirable, one of the members of the Gaussian family of filters can be used, the most common of which is the Bessel filter (Figure B.7) (so named because Bessel polynomials occur in the denominator of the transfer functions). If ringing or overshoot is to

[3] A Chebyshev filter with zero ripple is a Butterworth filter

be avoided, the phase shift between the input and output of a filter must be a linear function of frequency, i.e. the group delay (the rate of change of the phase with respect to frequency) must be constant. The Bessel filter provides the best approximation to the ideal of perfect flatness of the group delay in the passband (Figure B.8), though is a markedly poorer approximation to the ideal of flatness in the passband and steepness of attenuation.

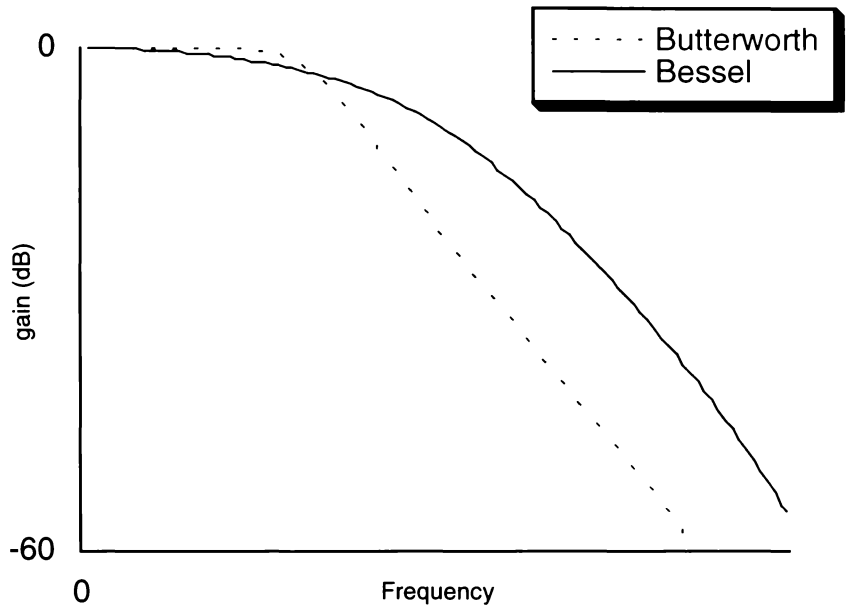


Figure B.7 : Frequency Response of a Bessel and a Butterworth Low Pass Filter

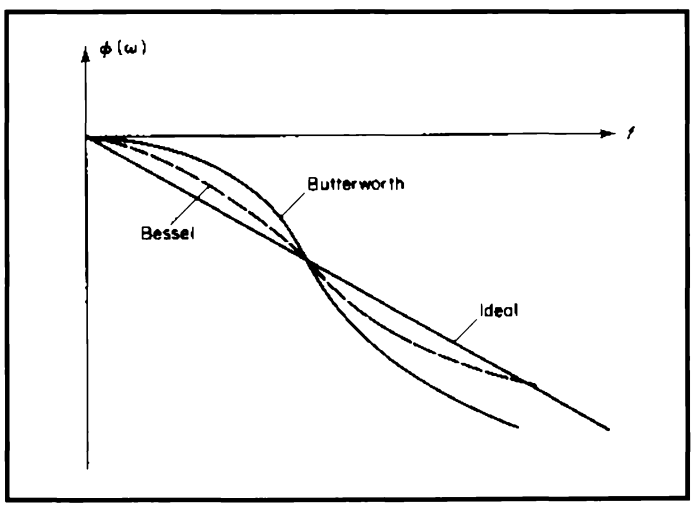


Figure B.8 : The Phase Response of the Ideal and Practical Bessel and Butterworth Low Pass Filters

B.2.2 Digital Filter Selection

To cater for speech input from an arbitrary source, the simulations in this thesis employ anti-aliasing lowpass digital filters (high frequencies must be eliminated when down-sampling to avoid aliasing effects). These filters require steep attenuation past the cutoff frequencies combined with maximally flat amplitude responses in the passbands. Since phase information is discarded during the compression experiments, phase linearity is not a requirement, and hence Bessel filtering offers no particular advantages. If ripples must be present in the filter design, these would be preferable in the passband. The quantisation of amplitude in later compression experiments introduces severe distortion of the passband, and hence a small amount of ripple is tolerable. However, ripple in the stopband may introduce unwanted aliasing effects, so the inverse Chebyshev filter is totally undesirable. The elliptic filter has the narrowest transition bandwidth of all filters (Childers and Durling, 1975), but also exhibits stopband ripple, and so would only be considered if the Chebyshev and Butterworth filters are inadequate. This has not proven to be the case, and either of these filters could be used.

In the following sections, design techniques for both the Butterworth and the Chebyshev lowpass filters are discussed (though a simple transformation will allow the design of highpass, bandpass and even band-stop filtering using the same technique). The disadvantage of the Butterworth is that a higher order filter is required to obtain the same attenuation as a Chebyshev, and therefore the filter of choice for most of the speech analysis is a fifth order implementation of the latter. It must be reiterated however, that such a choice was made with the knowledge that phase shift would not be a significant factor in the simulations, for if it was, the Butterworth would be preferred.

B.3 The Design of Butterworth and Chebyshev Lowpass Filters

This section discusses the design techniques required to construct an arbitrary order Butterworth or Chebyshev lowpass filter. Such filters are generally specified in terms of the magnitude squared frequency response, however linear systems are more easily understood from analysis of their transfer function. We begin with a discussion of the nature of the transfer function for a linear system, and show its relationship to the frequency response. A mathematical description of the Butterworth and Chebyshev filters is then presented, and

finally a transform technique to take the transfer function from the Laplace (s-domain) to the z domain where it may be implemented as a digital filter.

B.3.1 Preliminary Definitions

A filter with a single input $x(t)$ and a single output $y(t)$, consisting of linear, grouped elements, has x and y related by a linear, ordinary, integro-differential equation which may be Laplace-transformed to yield[4],

$$Y(s) = H(s)X(s) \qquad \textbf{Equation B.1}$$

where $s = \sigma + j\omega$ is the complex frequency, $Y(s)$ and $X(s)$ are the Laplace transforms of $y(t)$ and $x(t)$ respectively, and $H(s)$ is the transfer function of the system. When $s = j\omega$, the network function is complex and may be written in the form (Johnson, 1976)

$$H(j\omega) = |H(j\omega)|e^{j\phi(\omega)} \qquad \textbf{Equation B.2}$$

where $|H(j\omega)|$ is the amplitude or magnitude and $\phi(\omega)$ is the phase. A transfer function expresses the relationship between the output and the input of a system, and from it the behaviour of the system being modelled can be obtained.

B.3.2 The Transfer Function

A linear system, such as a filter, can be characterised by a complex transfer function expressed in Laplace (or s-domain) form as

$$H(s) = \frac{Y(s)}{X(s)} = \frac{\alpha_0 s^n + \alpha_1 s^{n-1} + \alpha_2 s^{n-2} + \cdots + \alpha_n}{\beta_0 s^m + \beta_1 s^{m-1} + \beta_2 s^{m-2} + \cdots + \beta_m} \qquad \textbf{Equation B.3}$$

---

[4] This assumes that t is a continuous-time variable, and that there is no initially-stored energy

where $\alpha_0, \alpha_1, \ldots, \alpha_n$ and $\beta_0, \beta_1, \ldots, \beta_m$ are constants. This can be factored into roots of the numerator and denominator polynomials

$$H(s) = K \frac{(s-a_1)(s-a_2)\cdots(s-a_n)}{(s-b_1)(s-b_2)\cdots(s-b_m)} \qquad \textbf{Equation B.4}$$

where $a_1, a_2, \ldots, a_n$ designate the roots of *Y(s)*, and $b_1, b_2, \ldots, b_m$ designate the roots of *X(s)*. When the complex frequency $s = j\omega$ assumes any of the values, $a_1, a_2, \ldots, a_n$, the system response is zero. When it assumes any value, $b_1, b_2, \ldots, b_m$, the response is infinite. These values of s are respectively, the zeros and poles of the transfer function (Beauchamp & Yuen, 1979).

Allpole filter networks are those whose transfer functions have no finite zeros. The typical n^{th} -order transfer function of such a filter has the form

$$H(s) = \frac{K}{D(s)} = \frac{K}{\prod_{j=1}^{n}(s-p_j)} \qquad \textbf{Equation B.5}$$

Thus, the numerator is a constant and the transfer function consists only of an n^{th} –order polynomial in the denominator. The n zeros of *H(s)* are said to be at infinity. Of the previously discussed basic filter types, the Butterworth, Chebyshev, Legendre and Bessel are examples of allpole filters.

The characteristics of allpole filters are determined by the location, in the s plane, of the n poles of *H(s)*. For Butterworth and Chebyshev filters, the poles of the Butterworth lie on a semi-circle in the left-half of the s-plane, whilst those of the Chebyshev lie on an ellipse which becomes narrower with increasing ripple.

B.3.2.1 Relationship Between Transfer Function and Amplitude Response

Often, filter specifications are described in terms of their amplitude response. To obtain a network with a given amplitude response, we must first obtain a valid transfer function. Since *H(s)* is a ratio of polynomials with real coefficients, we know that

$$H^*(j\omega) = H(-j\omega) \qquad \textbf{Equation B.6}$$

where H^* is the complex conjugate of H. Also, the given $|H(j\omega)|$ satisfies

$$|H_n(j\omega)|^2 = H(j\omega)H^*(j\omega) \qquad \textbf{Equation B.7}$$

as

$$|H_n(j\omega)|^2 = H(j\omega)H(-j\omega),$$

i.e.

$$H(s)H(-s)\big|_{s=j\omega} = |H_n(j\omega)|^2 \qquad \textbf{Equation B.8}$$

therefore,

$$H(s)H(-s) = |H_n(j\omega)|^2_{|\omega^2=-s^2} \qquad \textbf{Equation B.9}$$

In Equation B.9 we need to separate *H(s)* from a given *H(s)H(-s)*. Noting that if

$$H(s) = \frac{P(s)}{Q(s)} \qquad \textbf{Equation B.10}$$

then Equation B.9 can be rewritten in the form

$$\frac{P(s)}{Q(s)} \cdot \frac{P(-s)}{Q(-s)} = |H_n(j\omega)|^2_{|\omega^2=-s^2} \qquad \textbf{Equation B.11}$$

Thus for stability, *Q(s)* must be the Hurwitz factor[5] in *Q(s)Q(-s)* and if no right-half plane zeros are desired, *P(s)* must be the Hurwitz factor in *P(s)P(-s)*. Therefore we factor the numerator and denominator into, say 2^n factors, retain the *n* Hurwitz factors and reject the *n* non-Hurwitz factors. As an example, suppose the amplitude response of a system is described as

$$|H_n(j\omega)|^2 = \frac{4+\omega^2}{1+\omega^6} \qquad \textbf{Equation B.12}$$

[6]Replacing ω^2 by $-s^2$ yields

$$\begin{aligned} H(s)H(-s) &= \frac{4-s^2}{1-s^6} \\ &= \frac{(2-s)(2+s)}{(1-s)(1+s+s^2)(1+s)(1-s+s^2)} \end{aligned} \qquad \textbf{Equation B.13}$$

If no right-half plane zeros are desired, then we have the transfer function

$$H(s) = \frac{s+2}{(s+1)(s^2+s+1)} \qquad \textbf{Equation B.14}$$

B.3.3 The Design of the Butterworth Filter

The Butterworth filter of order *n* is described by the magnitude squared of its frequency response as

$$|H_n(j\omega)|^2 = \frac{1}{1+\left(\frac{\omega}{\omega_0}\right)^{2n}} \qquad \textbf{Equation B.15}$$

and therefore possesses the following properties (Ludeman, 1987):

---

[5] A Hurwitz polynomial $P(s) = a_n s^n + a_{n-1}s^{n-1} + \ldots + a_1 s + a_0$ is one with no right-half plane or multiple jω-axis zeros.

[6] Note that an amplitude function is always a function of ω^2 since it is an even function.

1 $|H_n(j\omega)|^2|_{\omega=0} = 1$ for all n (will not attenuate DC frequencies)

2 $|H_n(j\omega)|^2|_{\omega=\omega_0} = \frac{1}{2}$ for all finite n (the half-power or 3dB point)

3 $|H_n(j\omega)|^2$ is a monotonically decreasing function of ω (no ripples)

4 As n gets larger, $|H_n(j\omega)|^2$ approaches an ideal lowpass frequency response.

5 $|H_n(j\omega)|^2$ is called maximally flat at the origin as all derivatives exist and equal zero.

The transfer function of a second order Butterworth filter with unit DC gain is:

$$H(s) = \frac{p_1 p_2}{(s - p_1)(s - p_2)}$$ **Equation B.16**

where the poles p_1 and p_2 form a complex conjugate pair lying to the left of the vertical axis in the s-plane:

$$p_1 = \frac{-1+i}{\sqrt{2}}\omega_o, \qquad p_2 = \frac{-1-i}{\sqrt{2}}\omega_o$$ **Equation B.17**

ω_0 is the desired -3 dB cutoff frequency.

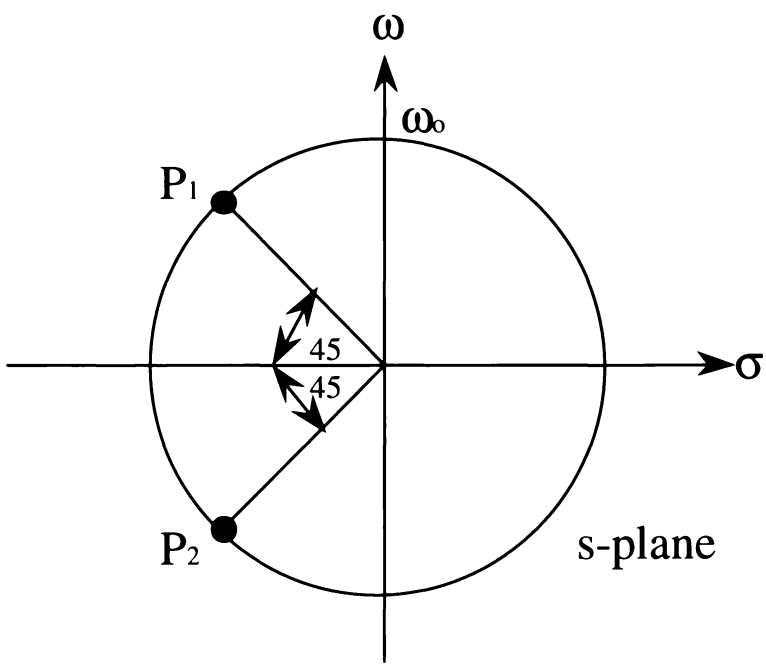


Figure B.9 : S-plane Representation of Pole Position for a Second Order Butterworth Low Pass Filter

Substituting these values into Equation B.16 above, the transfer function becomes

$$H(s) = \frac{{\omega_o}^2}{s^2 + \sqrt{2}\omega_o s + {\omega_o}^2}$$ **Equation B.18**

Similarly, for a third order Butterworth filter, with poles at $p_1 = -\omega_0$, $p_2 = -\omega_0\left(\frac{1 - j\sqrt{3}}{2}\right)$ and $p_3 = -\omega_0\left(\frac{1 + j\sqrt{3}}{2}\right)$, we obtain the transfer function

$$H(s) = \frac{-{\omega_0}^3}{(s + \omega_0)(s^2 + \omega_0 s + {\omega_0}^2)}$$ **Equation B.19**

This is often expressed in a normalised form where ω_0 is arbitrarily set to 1.

B.3.4 The Design of the Chebyshev Filter

The Chebyshev filter of order n is described by the magnitude squared of its frequency response as

$$|H_n(j\omega)|^2 = \frac{1}{\sqrt{1 + \varepsilon C_n^2(\omega)}}$$ **Equation B.20**

where

$$C_n(\omega) = \cos(n \cos^{-1} \omega)$$ **Equation B.21**

is the Chebyshev polynomial of the first kind of degree n and ε is a constant determined by the amount of permissible ripple. $|H(j\omega)|$ attains its maximum value of 1 at the zeros of $C_n(\omega)$ and its minimum value of $1/\sqrt{1+\varepsilon^2}$ at the points where $|C_n(\omega)|$ attains its maximum value of 1. Thus there are ripples in the passband, $0 \leq \omega \leq 1$, of ripple width:

$$RW = 1 - \frac{1}{\sqrt{1+\varepsilon^2}} \qquad \textbf{Equation B.22}$$

As previously mentioned, these ripples will be equal in magnitude. This constant ripple width is often expressed in dB at the passband minima, and is given by

$$\begin{aligned} RW_{dB} &= -20\log_{10}\left(\frac{1}{\sqrt{1+\varepsilon^2}}\right) \\ &= 10\log_{10}(1+\varepsilon^2) \end{aligned} \qquad \textbf{Equation B.23}$$

A second order 1 dB Chebyshev filter with unit DC gain, for example, requires $\varepsilon = 0.5088$, and in the s-plane, a normalized denominator polynomial $1.103 + 1.098s + s^2$. The poles are arranged in an ellipse (eccentricity ξ) whose major axis lies along the imaginary axis. The poles of a Chebyshev transfer function are simply those of a similar order Butterworth filter whose real component is multiplied by ξ (Moschytz & Horn, 1981). That is, given that each Butterworth pole pair has the form

$$p_B = -\sigma_B \pm j\omega_B \qquad \textbf{Equation B.24}$$

each Chebyshev pole p_C can be obtained from p_B by multiplying σ_B by ξ; thus

$$p_C = -\xi\sigma_B \pm j\omega_B \qquad \textbf{Equation B.25}$$

where the eccentricity ξ depends on the ripple factor ε as

$$\xi = \tanh\left(\frac{1}{n}\sinh^{-1}\frac{1}{\varepsilon}\right) \qquad \textbf{Equation B.26}$$

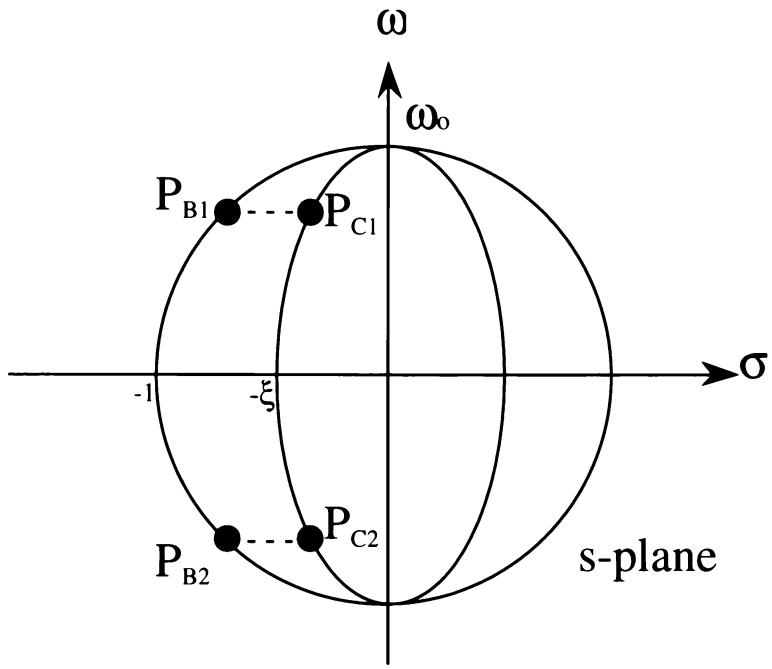


Figure B.10 : *S*-plane Representation Showing the Relationship Between Poles for a Second Order Butterworth and Chebyshev Low Pass Filter

To a first approximation, the roots can be assumed to have the same imaginary value as the corresponding Butterworth points.

B.3.5 The Bilinear Transform

We are forming digital approximations to these filters for implementation in software, however complex expressions in the $j\omega$ or s-domain cannot be directly realised. To digitally encode these filter designs, they must be expressed in the z-domain. What is required is a technique to map the transfer function $H(s)$ directly to $H(z)$ by some function $z = \zeta(s)$. This mapping function should possess the following properties (Gabel & Roberts, 1980):

1. $z = \zeta(s)$ should be a 1-to-1 mapping such that the inverse $s = \zeta^{-1}(z)$ exists.
2. The s-plane $j\omega$ axis should map to the z-plane unit circle so that $H\left(e^{j\theta}\right) = H(j\omega)$ for $e^{j\theta} = \zeta(j\omega)$.
3. The frequency points corresponding to DC should coincide, that is, $s = 0$ should map to $z = 1$: $1 = \zeta(0)$.

4. In order that stable causal models lead to stable causal discrete-time filters, the left half s-plane should correspond to the inside of the unit circle, so that poles will be correctly mapped.

A 1-to-1 function that maps circles and lines in one plane to circles and lines in another, as required by condition (2), is the **bilinear transform:**

$$z = \frac{a + bs}{c + ds} \qquad \textbf{Equation B.27}$$

To satisfy condition (3) we must have $a = c$, and to satisfy (4) we must have c and d of opposite sign ($|z| < 1$ if $s < 0$ for s real-values). To satisfy condition (2) we must have $|z| = 1$ for $s = j\omega$; that is,

$$\left|\frac{a + j\omega b}{c + j\omega d}\right| = 1 \qquad \textbf{Equation B.28}$$

A simple set of constants that satisfies all of these requirements is $a = b = c = 1$ and $d = -1$, yielding

$$z = \zeta(s) = \frac{1 + s}{1 - s} \qquad \textbf{Equation B.29}$$

with the inverse

$$s = \zeta^{-1}(z) = \frac{z - 1}{z + 1} \qquad \textbf{Equation B.30}$$

To relate the continuous and discrete-time frequency parameters[7],

---

[7] as $s = \dfrac{e^{j\theta/2}\left(e^{j\theta/2} - e^{-j\theta/2}\right)}{e^{j\theta/2}\left(e^{j\theta/2} + e^{-j\theta/2}\right)} = j\dfrac{\left(e^{j\theta/2} - e^{-j\theta/2}\right)/2j}{\left(e^{j\theta/2} + e^{-j\theta/2}\right)/2}$

$$s = j\omega = \frac{e^{j\theta} - 1}{e^{j\theta} + 1} = j\tan\left(\frac{\theta}{2}\right)$$ **Equation B.31**

Note that this is a pure imaginary number and so this transformation maps points on the z-plane unit circle onto points on the s-plane imaginary axis. This implies that for a continuous filter, frequency is measured along the imaginary axis in the s-plane, whereas for a discrete filter, frequency is measured along the circumference of the unit circle. This nonlinear relationship between the continuous (analog) and discrete (digital) frequencies can be seen by dividing Equation B.31 by j to obtain:

$$\omega = \tan\left(\frac{\theta}{2}\right)$$ **Equation B.32**

$$\theta = 2\tan^{-1}\omega$$ **Equation B.33**

It is evident that if the bilinear transformation is applied to an analog (s-plane) form of the transfer function with cutoff frequency ω, then Equation B.33 expresses the new digital (z-plane) cutoff frequency θ. This relationship between ω and θ represents a distortion or warping of the frequency axis, and so in order to get the proper digital frequency we must design an analog filter with an analog cutoff frequency given by Equation B.32. This is referred to as prewarping and compensates for the distortion that is later introduced in mapping the s-plane to the z-plane.

To now synthesize a discrete-time filter with the bilinear transform, it is necessary to first select the critical angles on the unit circle, then map these to the s-plane using Equation B.33, synthesize a continuous-time model using standard approximation techniques, and finally map the s-plane model to the z-plane filter using Equation B.31.

To illustrate, taking the example of the Butterworth filter, upon substituting Equation B.30 into Equation B.18

$$H(z)=\frac{\omega_A^2}{\frac{(z-1)^2}{(z+1)^2}+\sqrt{2}\omega_A\left(\frac{z-1}{z+1}\right)+\omega_A^2}$$
$$=\frac{z^2+2z+1}{z^2D+zk_1D+k_2D}$$

Equation B.34

or

$$H(z)=\frac{1+2z^{-1}+z^{-2}}{1+k_1z^{-1}+k_2z^{-2}}$$

Equation B.35

where:

$$D=1+\sqrt{2}\omega_A+\omega_A{}^2$$
$$k_1=\frac{2(-1+\omega_A{}^2)}{D}$$
$$k_2=\frac{1-\sqrt{2}\omega_A+\omega_A{}^2}{D}$$

As the transfer function simply expresses the ratio between the output ($Y(z)$) and the input ($X(z)$), an expression for $Y(z)$ can be obtained as

$$Y(z)=\frac{\omega_A{}^2}{D}\left[X(z)+2z^{-1}X(z)+z^{-2}X(z)\right]-k_1z^{-1}Y(z)-k_2z^{-2}Y(z)$$

Equation B.36

This has a direct equivalence with the time domain, noting that z maps to t, z-1 maps to t-1 etc., where t refers to the current sample, t-1, the previous sample and so-on.

$$y_t=\frac{\omega_A{}^2}{D}(x_t+2x_{t-1}+x_{t-2})-k_1y_{t-1}-k_2y_{t-2}$$

Equation B.37

A similar approach may be taken to yield the digital expression for the Chebyshev filter.

B.3.6 FIR / IIR Filtering

Equation B.3 is a differential equation applicable to the representation of continuous filters. In the discrete case, as in Equation B.37, a linear difference must be used. In general this will take the form:

$$y_i = \sum_{k=0}^{P} b_k y_{i-k} + \sum_{k=0}^{M} a_k x_{i-k} (i = 1,2,\ldots,N) \qquad \textbf{Equation B.38}$$

where P and M are positive integers and a_k and b_k are real coefficients. When $M = 0$ the filter is auto-regressive and the time duration of the filter impulse response is infinite. This is known as an IIR (infinite impulse response) filter. For $P = 0$ the duration of impulse response is limited and the filter is known as a FIR (finite impulse response) filter. The FIR, representing a summation of a limited number of inputs, has excellent phase characteristics but requires a large number of terms to obtain sharp attenuation. In comparison, the IIR, representing a summation of both input and output terms, requires relatively fewer terms than the FIR to obtain a similar attenuation, but this is at the expense of poorer phase performance.

Butterworth and Chebyshev filters are implemented in software in the form of Equation B.9 which is obtained from the z transform as in Equation B.37. Note that such filters are IIR. A large number of test speech frames were obtained from the TIMIT database (Section 3.7.2). The sample rate was 16 kHz, and so needed to be lowpass filtered to 8 kHz for the simulation. A Butterworth filter was initially trialled, but was later replaced by a Chebyshev in order to obtain a sharper attenuation slope. The ripples introduced by the Chebyshev function are not audibly noticeable (under our test conditions), and as mentioned earlier, are masked by the amplitude quantisation that the simulation imposes. A tenth order Chebyshev filter was used, with a 1 Db passband ripple, and a cutoff frequency of 3.3 kHz. This provides suitable attenuation characteristics and an acceptable level of phase shift (we delete the phase information in many of the later quantisation experiments anyway). No claim is made that this is the best form of anti-aliasing filter, merely that it fulfils the requirements of the following simulations, and is relatively straightforward to design and implement following the guidelines presented in this section.

Appendix C : Interpolation, Decimation, Frequency Conversion, Differentiation

C.1 Time Domain Interpolation

C.1.1 Overview

This section of the appendix contains details on the derivation and structure of the interpolation formulae summarised in Appendix A. The techniques based on what has become known as the classical interpolation formula (from numerical analysis) are derived in detail, and are included in the simulation for the user to take advantage of whenever interpolation is required. The FIR/IIR techniques are also discussed, though more in terms of frequency response, especially when the Nyquist sampling rate is only just satisfied. The relationship between time domain and frequency domain characteristics of the various methods are time consuming to derive individually, so only the linear case is treated in detail; the frequency responses of higher order interpolation formulae follow in a similar fashion.

C.1.2 Polynomial Interpolation

Consider the n + 1 distinct points $x_0, x_1, \ldots, x_n$ and let $f(x)$ be a real valued function which takes on the values $f(x_i)$ $(i = 0,1,\ldots,n)$ at these points. One method to interpolate this data is to construct a polynomial, of degree not exceeding n, which passes through the n + 1 points (x_i, f_i) $(i = 0,1,\ldots,n)$. The most direct approach is to use the general polynomial of degree n

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0 \qquad \textbf{Equation C.1}$$

where the a_i are arbitrary coefficients. Various methods exist for evaluating these coefficients - probably the most common is that due to Lagrange. Using this technique an n^{th} order polynomial through (x_0, f_0), ≡, (x_n, f_n) is written in the form:

$$P_n(x) = L_0(x)f_0(x) + L_1(x)f_1(x) + \cdots + L_n(x)f_n(x) \qquad \textbf{Equation C.2}$$

where L_i are polynomials of, at most, degree n. This polynomial will obviously have the same value of f at the tabular points $x_0, x_1, \ldots, x_n$ if

$$L_i(x_j) = \delta_{ij} \qquad \textbf{Equation C.3}$$

In general, $L_i(x)$ (the Lagrange interpolation coefficient) can be expressed as

$$L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \left(\frac{x - x_j}{x_i - x_j} \right) \qquad \textbf{Equation C.4}$$

to yield the interpolating polynomial

$$P_n(x) = \sum_{i=0}^{n} L_i(x) f(x_i) \qquad \textbf{Equation C.5}$$

Although this method will yield an exact interpolating polynomial, there are major difficulties in a practical implementation. Unlike the difference techniques discussed in the following section where the degree of the required approximating polynomial may be determined by computing terms until they no longer appear significant, for the Lagrange procedure, the polynomial degree must be chosen at the outset. Any change of degree involves completely new computation of all terms and for a polynomial of high degree it becomes large and unwieldy. Other methods were therefore considered for use in the simulations.

C.1.3 Interpolation Based Upon Differences

Interpolation based on weighted evaluation of forward, central and backward differences has existed for several centuries with much of the pioneering development originating from the work of Newton and Gauss. The difference between a number (say f_0) and its immediate successor (say f_1) is represented by Δf_0, and is referred to as the first difference. The same process may be repeated between these values of the first difference, for example, given Δf_0

and Δf_1, the difference between these is referred to as the second difference and is written in the form $\Delta^2 f_0$. This may be extended to any order *n*.

Various methods exist for interpolating the polynomial using these differences, and those incorporated in this thesis are outlined below. In the discussion that follows, *h* refers to the interval between tabular points, (the sample rate of the speech will not vary, therefore all the data will be equally spaced over the entire range), and the binomial $\binom{x}{r}$ is used, being equivalent to

$$\begin{aligned}
&\binom{x}{0} = 1 \\
&\binom{x}{r} = \frac{x(x-1)\ldots(x-r+1)}{r!} \; r \text{ a positive integer} \\
&\binom{x}{r} = 0 \; r \text{ a negative integer}
\end{aligned} \qquad \textbf{Equation C.6}$$

The data is considered over the range $x_0, x_1, \ldots, x_n$. To simplify the following expressions, a variable *s* can be introduced, and is defined by

$$s = \frac{x - x_0}{h} \qquad \textbf{Equation C.7}$$

C.1.3.1 Newton's Forward Difference Formula

The recursion form of the Newton Difference formula is

$$P_k(s) = P_{k-1}(s) + \binom{s}{k} \Delta^k f_0 \qquad \textbf{Equation C.8}$$

From this formula it is possible to construct a sequence of polynomials $P_0, P_1, \ldots, P_k$ of increasing degree, each of which interpolates at all the previous points, plus one additional point. Hence for the polynomial which interpolates at the *n* + 1 points $x_0, x_1, \ldots, x_n$ we have

$$P_n(s) = f_0 + \binom{s}{1}\Delta f_0 + \binom{s}{2}\Delta^2 f_0 + \cdots + \binom{s}{n}\Delta^n f_0$$

Equation C.9

For speech data, there is no justification in terms of required accuracy to extend this technique beyond the fourth order, and hence the user is given the option of selecting from first through fourth order approximation.

The first order approximation is linear:

$$\begin{aligned} P_n(s) &= f_0 + \binom{s}{1}\Delta f_0 \\ &= f_0 + s(f_1 - f_0) \end{aligned}$$

Equation C.10

with an associated error of

$$E_1 = \frac{h^2}{2} f''(\xi).$$

Equation C.11

Second order is a quadratic approximation:

$$\begin{aligned} P_2(s) &= f_0 + \binom{s}{1}\Delta f_0 + \binom{s}{2}\Delta^2 f_0 \\ &= f_0 + s(f_1 - f_0) + \frac{s(s-1)}{2}(f_2 - 2f_1 + f_0) \end{aligned}$$

Equation C.12

with error

$$E_2 = \frac{h^3}{6} f'''(\xi).$$

Equation C.13

Third and fourth order approximations (with associated errors) are, respectively:

$$\begin{aligned} P_n(s) &= f_0 + \binom{s}{1}\Delta f_0 + \binom{s}{2}\Delta^2 f_0 + \binom{s}{3}\Delta^3 f_0 \\ &= f_0 + s(f_1 - f_0) + \frac{s(s-1)}{2}(f_2 - 2f_1 + f_0) + \frac{s(s-1)(s-2)}{6}(f_3 - 3f_2 + 3f_1 - f_0) \\ E_3 &= \frac{h^4}{24} f^{(4)}(\xi), \end{aligned}$$

Equation C.14

$$P_n(s) = f_0 + \binom{s}{1}\Delta f_0 + \binom{s}{2}\Delta^2 f_0 + \binom{s}{3}\Delta^3 f_0 + \binom{s}{4}\Delta^4 f_0$$

$$= f_0 + s(f_1 - f_0) + \frac{s(s-1)}{2}(f_2 - 2f_1 + f_0) + \frac{s(s-1)(s-2)}{6}(f_3 - 3f_2 + 3f_1 - f_0) +$$

$$\frac{s(s-1)(s-2)(s-3)}{24}(f_4 - 4f_3 + 6f_2 - 4f_1 + f_0)$$

$$E_4 = \frac{h^5}{120} f^{(5)}(\xi).$$

Equation C.15

C.1.3.2 Newton's Backward Difference Formula

Newton's Forward Difference formulae exhibit the property that only information concerning the successive data points needs to be known. However, the number of points ahead of the position that we wish to interpolate from may be limited. For example, let I extend over the interval 1, 2, 3, ... , 9, 10 and assume it is desired to find, via interpolation based on a forward difference scheme, some value corresponding to 9.5. In this situation x_0 = 9 and x_1 = 10. Any interpolation scheme that requires knowledge of f_2 (i.e., data for x_2 = 11) or subsequent data points is meaningless, as that knowledge is not available. In this situation a backward difference technique is most useful, so the preceding information is utilised as opposed to the succeeding data implemented in the forward-difference method.

Newton's Backward Difference formula is such a scheme. It takes the recursive form

$$P_k(s) = P_{k-1}(s) + \binom{s+k-1}{k}\Delta^k f_{-k}$$

Equation C.16

Hence, as for the forward difference case, the interpolating formula which interpolates at the $n + 1$ points $x_0, x_{-1}, \ldots, x_{-n}$ is given by

$$P_n(s) = f_0 + \binom{s}{1}\Delta f_{-1} + \binom{s+1}{2}\Delta^2 f_{-2} + \ldots + \binom{s+n-1}{n}\Delta^n f_{-n}$$

Equation C.17

Again, the simulation allows the user to choose any scheme from first through fourth order. The explicit formulae, with associated errors are given below.

Equation C.18

$$P_1(s) = f_0 + s(f_0 - f_{-1})$$
$$E_1 = \frac{h^2}{2} f''(\xi)$$

Equation C.19

$$P_2(s) = f_0 + s(f_0 - f_{-1}) + \frac{s(s+1)}{2}(f_0 - 2f_{-1} + f_{-2})$$
$$E_2 = \frac{h^3}{6} f'''(\xi)$$

$$P_3(s) = f_0 + s(f_0 - f_{-1}) + \frac{s(s+1)}{2}(f_0 - 2f_{-1} + f_{-2}) + \frac{s(s+1)(s+2)}{6}(f_0 - 3f_{-1} + 3f_{-2} - f_{-3})$$
$$E_3 = \frac{h^4}{24} f^{(4)}(\xi)$$

Equation C.20

$$P_4(s) = f_0 + s(f_0 - f_{-1}) + \frac{s(s+1)}{2}(f_0 - 2f_{-1} + f_{-2}) + \frac{s(s+1)(s+2)}{6}(f_0 - 3f_{-1} + 3f_{-2} - f_{-3}) + \frac{s(s+1)(s+2)(s+3)}{24}(f_0 - 4f_{-1} + 6f_{-2} - 4f_{-3} + f_{-4})$$
$$E_4 = \frac{h^5}{120} f^{(5)}(\xi)$$

Equation C.21

If the user specifically selects a Newtonian difference approach to interpolation, then for speech samples at the beginning of the data file a forward difference technique is automatically employed, with the appropriate order input by the user. Similarly, at the end of the file, the simulation uses a backward technique of the same order. However, for the vast majority of the data points that lie between these two extremes, a considerable improvement in accuracy, with no extra cost in computation time, is obtained by use of a central difference scheme.

C.1.3.3 Central Difference Formula

If data on both sides of the interpolation point is known, then a central difference scheme can be implemented which is significantly more accurate than either of the two difference techniques previously discussed. In this case, preceding and succeeding samples are used to evaluate the non-tabular point. One method (and that actually utilised as the default in the speech simulations) is attributed to Stirling, and is derived from taking a mean of the Gaussian forward-difference and backward-difference formulae. The Gaussian schemes differ from the Newtonian ones only in which tabular points are employed in the interpolation formula.

Again, considering a maximum of a fourth order scheme, the Stirling approximation becomes

$$P_4(s) = f_0 + \binom{s}{1}\frac{\Delta f_{-1} + \Delta f_0}{2} + \frac{1}{2}\left[\binom{s}{2} + \binom{s+1}{2}\right]\Delta^2 f_{-1}$$
$$+ \binom{s+1}{3}\frac{\Delta^3 f_{-1} + \Delta^3 f_{-2}}{2} + \frac{1}{2}\left[\binom{s+1}{4} + \binom{s+2}{4}\right]\Delta^4 f_{-2}$$

Equation C.22

This is known as Stirling's formula, and the fourth order scheme is used as the default in any difference-based interpolation (though the user is free to select other options). Explicitly the first four terms of the Stirling formula are

$$P_1 = f_0 + \frac{s}{2}(f_1 - f_{-1})$$
$$E_1 = \frac{h^3}{12} f'''(\xi)$$

Equation C.23

$$P_2 = f_0 + \frac{s}{2}(f_1 - f_{-1}) + \frac{s^2}{2}(f_1 - 2f_0 + f_{-1})$$
$$E_2 = \frac{h^4}{24} f^{(4)}(\xi)$$

Equation C.24

$$P_3 = f_0 + \frac{s}{2}(f_1 - f_{-1}) + \frac{s^2}{2}(f_1 - 2f_0 + f_{-1}) + \frac{s(s+1)(s-1)}{12}(f_2 - 2f_1 + 2f_{-1} - f_{-2})$$

$$E_3 = \frac{h^5}{120} f^{(5)}(\xi)$$

Equation C.25

$$P_4 = f_0 + \frac{s}{2}(f_1 - f_{-1}) + \frac{s^2}{2}(f_1 - 2f_0 + f_{-1}) + \frac{s(s+1)(s-1)}{12}(f_2 - 2f_1 + 2f_{-1} - f_{-2}) + \frac{s^2(s+1)(s-1)}{24}(f_2 - 4f_1 + 6f_0 - 4f_{-1} + f_{-2})$$

$$E_4 = \frac{h^6}{620} f^{(6)}(\xi)$$

Equation C.26

Note that for this procedure the first order approximation involves only one term each side of the f_0, whilst the second order approximation contains only the extra term f_0. Hence for very little extra calculation, accuracy can be greatly improved. Similarly the third order approximation involves the two points each side of f_0, while the fourth contains these points as well as f_0. Recognising this fact, the user is given the option of employing either the second or fourth order Stirling approximations, as there are no great savings of computation by using first or third order, and the resultant improvement in accuracy more than compensates for this.

C.1.4 Cubic Splines

Interpolation by curve fitting a polynomial of appropriate degree to the tabular points was the first technique discussed. However, as mentioned, this form of interpolation requires a formula with the same number of elements as the number of points intersected. Interpolating 90 points for example, yields a polynomial of degree 90, which is unwieldy. Therefore, if such a scheme is to be implemented for speech processing, the question becomes what would be an appropriate order polynomial. In the simulation used for this thesis, the speech is segmented into 16 millisecond frames, each containing 128 data points. Hence, one obvious division is to use a 128 point frame. However, it is difficult to justify repeatedly evaluating a 128th order interpolating polynomial. Even though this polynomial may be an exact fit, there

are likely to be discontinuities between frames, resulting in degraded speech output. Another consideration is that the interpolation scheme does not need to be supremely accurate, since small deviations will not be audible; what is far more important is how quickly the interpolation can be performed.

So, if a 128 point frame is too large, what is appropriate? If we make a separate curve in each interval $[x_{i-1}, x_i]$, so that the curves meet with no jaggedness or irregularity, we could piece each of these polynomials together to interpolate the whole of the input data. Cubic splining involves such a scheme, patching together 3rd degree polynomials to form a "smooth" curve. Several different splining techniques exist, including the Free Cubic Spline Interpolation, where the second derivative is assumed to be zero at the endpoints (thus the cubic spline is "free"), and the Clamped Cubic Spline Interpolation, where the first derivative at the endpoints of the interval are defined by the user, (i.e. "clamped"). The cubic spline interpolant possesses the following properties:

1. It passes through every data point
2. It is continuous
3. Its first derivative is continuous.
4. Its second derivative is continuous.

Cubics that join adjacent data points are of the following form:

$$S[i](x) = A[i] + B[i](x - x[i]) + C[i](x - x[i])^2 + D[i](x - x[i])^3 \qquad \textbf{Equation C.27}$$

where i ranges between 1 and the number of data points minus 1, the $x[i]$'s are the x-coordinates of the input data, and $x[i] \le x \le x[i+1]$. The interpolated values of $f(x)$ are found by evaluating the i^{th} cubic polynomial at x. A method of Free Cubic Spline Interpolation was employed as a preprocessing option but was later removed as it did not exhibit any clear advantage over the differencing techniques, and was more computationally intensive. Cubic splining is a popular option where accuracy is required for little cost in terms of computation, however it was evident that the extra accuracy gained from this technique as compared to the differencing schemes would not be audible.

C.1.5 Selection of Interpolation Technique

Three interpolation schemes have been considered. Though inherently the least accurate of the three, difference techniques have been employed for preprocessing in this simulation. Investigation of numerical data from each of these schemes yields errors in accordance with that suggested by analytical investigations of the algorithms. No technique utilising anything higher than third order approximations altered the audible nature of reconstructed speech, hence the difference techniques, being simplest and least computationally intensive have been retained. Although the option exists for the user to select a linear (first order) interpolation scheme, this is discouraged, as audible differences do result. Its advantage, however, is that it is extremely fast. To match the accuracy of the Stirling approximations, third order forward and backward techniques yield similar errors to the second order central, whilst the fourth order Newtonian techniques still do not yield the accuracy of the fourth order Stirling, but the error differences are negligible.

In principle, difference formulae such as the Newtonian or Stirling, provide an answer which, within roundoff errors, could be found by other interpolating polynomials using the same sample points. Although such alternative schemes exist, such as those attributable to Gauss, Everett or Bessel, no significant advantage exists in their use, and as such these are not discussed. Any scheme employing differences provides some idea of the accuracy of the algorithm - unlike the Lagrangian method, which yields no such information. As a result, difference methods tend to be used in exploratory work, and the Lagrange methods in well-understood, routine work. (Hamming, 1962).

C.2 Differentiation

Appendix A indicates that an ordinary differencing scheme varies from first order differentiation only by a constant which is determined by the sample rate. Such differentiation schemes can be obtained by differentiating the generalised form of Newton's forward difference formula:

$$f'(x) \approx \frac{1}{h}\frac{dP}{ds} = \frac{1}{h}\left[\Delta f_0 + \frac{2s-1}{2}\Delta^{2} f_0 + \ldots + \frac{d}{ds}\binom{s}{n}\Delta^{n} f_0\right] \qquad \textbf{Equation C.28}$$

The first term of this expansion is:

$$f_1'(x) = \frac{\Delta f}{h} = \frac{f_1 - f_0}{h}$$
$$E_1 = -\frac{h}{2} f''(\xi)$$

Equation C.29

second order:

$$f_2'(x_0) \approx \frac{1}{h}\left(\Delta f_0 - \frac{1}{2}\Delta^2 f_0\right)$$
$$= \frac{-f_2 + 4f_1 - 3f_0}{2h}$$
$$E_2 = \frac{h^2}{3} f'''(\xi)$$

Equation C.30

third order:

$$f_3'(x_0) \approx \frac{1}{h}\left(\Delta f_0 - \frac{1}{2}\Delta^2 f_0 + \frac{1}{3}\Delta^3 f_0\right)$$
$$= \frac{2f_3 - 9f_2 + 18f_1 - 11f_0}{6h}$$
$$E_3 = \frac{-h^3}{4} f^{(4)}(\xi)$$

Equation C.31

fourth order:

$$f_4'(x_0) \approx \frac{1}{h}\left(\Delta f_0 - \frac{1}{2}\Delta^2 f_0 + \frac{1}{3}\Delta^3 f_0 - \frac{1}{4}\Delta^4 f_0\right)$$
$$= \frac{-3f_4 + 16f_3 - 36f_2 + 48f_1 - 25f_0}{12h}$$
$$E_4 = \frac{h^4}{5} f^{(5)}(\xi)$$

Equation C.32

Similarly taking Stirling's Central difference method and differentiating, all even orders of Δ disappear upon setting $x = x_0$. Hence we are left with

$$f'(x) \approx \frac{1}{h}\left[\frac{1}{2}(\Delta f_{-1} + \Delta f_0) + \frac{3s^2 - 1}{12}(\Delta^3 f_{-1} + \Delta^3 f_{-2})\right].$$

Equation C.33

A first order approximation then yields

$$f'(x_0) \approx \frac{f_1 - f_{-1}}{2h}$$
$$E = -\frac{h^2}{6} f'''(\xi)$$

Equation C.34

and taking the next term in the expansion

$$f'(x_0) = \frac{-f_2 + 8f_1 - 8f_{-1} + f_{-2}}{12h}$$
$$E = -\frac{h^4}{30} f^{(5)}(\xi)$$

Equation C.35

Numerical differentiation is an inherently unstable process, and even when very little error exists in the original data, great accuracy cannot normally be expected in the estimation of the derivative. One contributor to this is the fact that very little can be said about the accuracy at an interpolated point. This is intuitively evident upon consideration of Figure C.1. The interpolating polynomial will generally oscillate about the true curve. At an interpolated point the slope of the interpolating polynomial $p(x)$ may be quite different from the slope of the true function $f(x)$. Rounding and other errors will also result in the values of $f(x)$ at the interpolating points being rather inaccurate. In Figure C.1 the circles represent points on the curve while the x's represent the values of $f(x)$ actually used. From this figure it can be seen that although the displacement at the tabular points may be small, the error associated with the slope can be significant.

Given the minimal variation between different routines for speech processed using the original log slopes proposed by Ghitza (and supported by physiological models (Chapter 3)), and the inherent errors in numerical differencing, there seems little point in pursuing this topic into an investigation of its frequency domain properties, or even discussing alternative differentiation techniques. The differentiation routines supplied to the user of the simulation are computationally efficient, and seem to provide satisfactory high frequency emphasis when required.

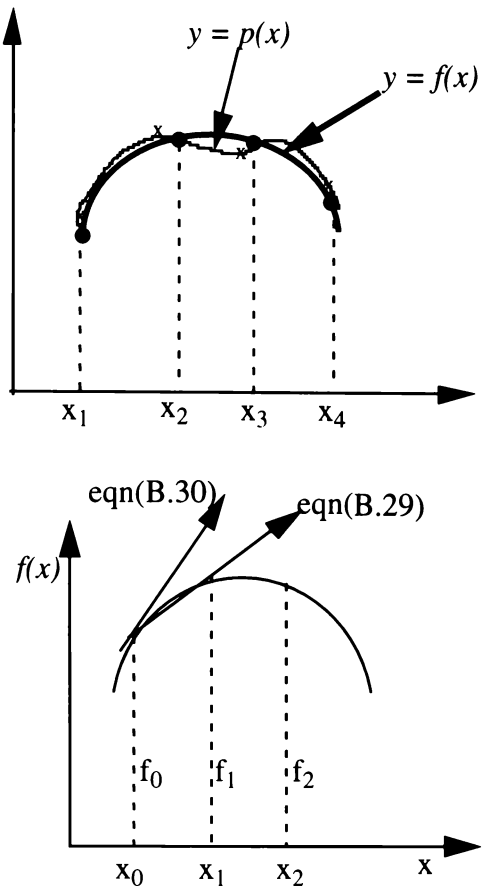


Figure C.1 : Demonstration of the Errors Inherent in Numerical Differentiation

C.3 Frequency Domain Sample Rate Conversion

These use some form of FIR filtering, and before the acquisition of Matlab, some of these routines were hand-coded in C. Consequently, substantial effort was placed into understanding these routines, and some of the details are included in this section of the Appendix.

As discussed in previous sections of this Appendix, tables of mathematical functions may be constructed to form interpolation techniques based either on Lagrangian or cubic polynomial fits, or a fit based upon differences. A temptation in signal processing applications is to make do with linear interpolation routines because of their intuitive and coding simplicity, however, as will be shown, such interpolation is fundamentally a linear filtering process (and was experimentally observed to be audibly noticeable). Whilst higher order interpolations in some way compensates for this, the last two decades have seen a large amount of research

into FIR and IIR digital filters for use as interpolation filters. In comparison to the classical techniques, these methods provide significant improvement in frequency domain response.

C.3.1 Classical Polynomial Interpolation

The simplest form of polynomial interpolation is linear, and involves only two consecutive samples of the original sequence (say $x(n)$) in the computation of an interpolated sampled. Specifically, the values interpolated between the two samples $x(0)$ and $x(1)$ (assuming a forward interpolation technique) lie on a straight line that connects these two sample points. Thus the equation relating the output $y(n)$, having sampling period $T' = T/L$, to the input sequence $x(n)$, having sampling period T, is

$$\begin{aligned} y(n) &= x(0) + \frac{x(1)-x(0)}{LT'-0}(nT'-0) \\ &= x(0)\left(1-\frac{n}{L}\right) + x(1)\left(\frac{n}{L}\right), \qquad 0 \le n \le L \end{aligned}$$

Equation C.36

In order to interpret linear interpolation as a linear filtering process, we must derive an impulse response for the linear interpolation filter. It can be shown, (Schafer and Rabiner, 1973) that the impulse response is of the form

$$\begin{aligned} h(n) &= 1-\frac{|n|}{L}, \quad |n| < L \\ &= 0 \qquad \text{otherwise} \end{aligned}$$

Equation C.37

and has length $N = 2L - 1$. It is also clear that

$$y(n) = x\left(\frac{n}{L}\right), \quad n = 0, \pm L, \pm 2L \ldots$$

Equation C.38

The system function corresponding to linear interpolation is

$$H(e^{j\omega T'}) = \frac{1}{L}\left\{\frac{\sin[\omega LT'/2]}{\sin[\omega T'/2]}\right\}^2 \qquad \textbf{Equation C.39}$$

An interpolation filter must remove the images of the signal spectrum that are centred at integer multiples of $2\pi/T$, while leaving the frequencies below π/T unaltered. Linear interpolation achieves significant attenuation (> 40 dB) in only a very small band of width $0.35\pi/T$ centred at $2\pi/T$, $4\pi/T$ etc. In classical numerical analysis, the inadequacies of linear interpolation are overcome by the use of higher order polynomials (Schafer and Rabiner, 1973). These higher order interpolation polynomials provide for greater attenuation in a narrow band around integer multiples of $2\pi/T$ because the zeros of the system function tend to be clustered about those frequencies. However, the frequency domain response for even an eighth order polynomial fit still leaves much to be desired, though these schemes are an acceptable solution provided the original sampling rate is many times the Nyquist rate. Noting that 3.3 kHz is the maximum frequency of interest in this simulation, this condition is only partially satisfied when sample rate conversion is undertaken from the TIMIT database (5 times the Nyquist rate).

This would be of concern if we were investigating the quality of the reconstructed speech, however it appears that in the intelligibility experiments undertaken by the simulation following sample rate conversion (discussed later), these frequencies are sufficiently attenuated so as not to be audibly evident. In general, especially where the quality of speech is important, the attenuation may not be sufficient to mask the audible contributions from these aliased frequency components, and hence some other form of interpolation technique may be required.

C.3.2 FIR / IIR Filter Interpolation

In practical situations, signals are often sampled at a rate only slightly higher than twice the Nyquist frequency in order to minimize the computation required for digital filtering and other signal processing procedures. In this case, the ideal interpolation filter for increasing the sampling rate has constant gain in the frequency range $0 \leq |\omega| < \pi/T$, and zero gain

elsewhere. For such signals we are interested in the best possible approximation to the ideal lowpass filter. For the situations where the sampling rate is considerably in excess of the Nyquist rate, what is required is a relatively narrow passband of constant gain and a number of stopbands of zero gain, with the frequency response being somewhat arbitrary elsewhere. This means that high-order polynomial interpolation filters may be quite satisfactory for signals that are sufficiently over sampled. However, it is generally possible to achieve significantly better interpolation filters using optimisation techniques, for example those based upon discrete Chebyshev approximations (Schafer and Rabiner, 1973). These result in frequency responses similar to those obtained from the classical interpolators, but are always superior in terms of attenuation response.

Many alternative means of interpolation have been proposed. Lagadec et al. (1981), chose to interpolate digitally to an extremely high sampling frequency using an FIR filter, and then choose the closest sample to the correct sampling instant. This relies on the principle that if the interpolation factor is so large that the difference between adjacent samples cannot exceed half the interval representing the least significant bit, then any sampling point can be represented by choosing the nearest sample to the desired time instant within the signal resolution.

This is by no means an efficient technique; a worst case analysis for speech input represented by 12 bits and a highest frequency signal of 8 kHz shows (Ramstad, 1984) that, in order to satisfy the above requirements, the sample rate must be in excess of 100 MHz! Even noting that this is a pessimistic calculation, this rate places excessive calculation and storage demands upon our system. Using linear interpolation, such that the resulting error is still less than the least significant bit, reduces this rate to 1.6 MHz - a rate that is still computationally intensive.

A great improvement over classical interpolation techniques can be provided by the implementation of FIR or IIR interpolation filters. Whilst a detailed analysis of the comparative advantages and disadvantages of these schemes is beyond the scope of this thesis, the techniques are discussed together with a suitable FIR implementation.

A continuous-time signal $\hat{x}(t)$ is sampled to produce the sequence

$$x(n) = \hat{x}(nT), \quad -\infty < n < \infty \qquad \textbf{Equation C.40}$$

where T is the sampling period. The Fourier transform of the sequence $x(n)$ is related to the Fourier transform of $\hat{x}(t)$ by

$$X(e^{j\omega T}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} \hat{X}\left(\omega + k\frac{2\pi}{T}\right) \qquad \textbf{Equation C.41}$$

If $\hat{x}$ is bandlimited, i.e. $\hat{X}(\omega) = 0$ for $|\omega| \geq \Omega$, and if $T \leq \pi/\Omega$, then

$$X(e^{j\omega T}) = \frac{1}{T}\hat{X}(\omega), \quad -\frac{\pi}{T} \leq \omega \leq \frac{\pi}{T} \qquad \textbf{Equation C.42}$$

Hence the original time signal can be obtained uniquely from the samples $x(n)$ through the interpolation formula

$$\hat{x}(t) = \sum_{\kappa=-\infty}^{\infty} x(k) \frac{\sin\left[\frac{\pi}{T}(t - kT)\right]}{\frac{\pi}{T}(t - kT)} \qquad \textbf{Equation C.43}$$

We are concerned with a sequence $y(n)$ corresponding to sampling $\hat{x}(t)$ at a different sampling rate, i.e. $y(n) = \hat{x}(nT')$. Evaluating Equation C.43 for $t = nT'$, a direct relationship is obtained between y(n) and x(n), which is, however, impossible to evaluate as Equation C.43 is of infinite duration. Rather than truncating this to an arbitrary order, it is more reasonable to design finite duration interpolators.

C.3.2.1 Sample Rate Reduction by an Integer Factor

First consider sample rate reduction, and suppose that the desired sampling period is $T' = MT$ (where M is an integer). So the new sequence is

$$y(n) = x(Mn) \qquad \textbf{Equation C.44}$$

That is, the sequence $x(n)$ is "sampled" by retaining only one out of each group of M consecutive samples. These will uniquely determine $\hat{x}(t)$ if and only if $T' < \pi/\Omega$. Aliasing will occur in the process of digital sampling rate reduction unless the original sampling period satisfied

$$T \leq \frac{\pi}{M\Omega} \qquad \textbf{Equation C.45}$$

where Ω is the Nyquist frequency of $\hat{x}(t)$. If this inequality is satisfied however, then (Schafer & Rabiner, 1973)

$$\begin{aligned} Y(e^{j\omega T'}) &= \frac{1}{M} X(e^{j\omega T}) \\ &= \frac{1}{T'} \hat{X}(\omega) \qquad -\frac{\pi}{T'} < \omega \frac{\pi}{T'} \end{aligned} \qquad \textbf{Equation C.46}$$

If the original sampling period does not satisfy Equation C.45, then aliasing distortion can only be avoided by passing the sequence $x(n)$ through an ideal lowpass filter with unit gain and cutoff frequency π/T'.

C.3.2.2 Sample Rate Increase by an Integer Factor

A similar analysis holds for sampling rate increase by an integer factor, L, to obtain the new sampling period $T' = T/L$. Since the sequence $x(n)$ provides samples of the desired sequence only at intervals of L, the remaining samples must be filled in by interpolation. This is accomplished by the addition of L - 1 zero-valued samples between each value of the original sequence, which is then filtered with an ideal lowpass filter. This new sequence will be

periodic with period $2\pi/T = 2\pi/LT'$, rather than $2\pi/T$ as is the case in general for sequences associated with a sampling period T'. To obtain a sequence $y(n)$ such that

$$y(n) = \hat{x}(nT') \qquad \textbf{Equation C.47}$$

then we must ensure that

$$Y\left(e^{j\omega T'}\right) = \frac{1}{T'}\hat{X}(\omega), \qquad -\frac{\pi}{T'} \le \omega \le \frac{\pi}{T'} \qquad \textbf{Equation C.48}$$

The other images of $(1/T)\ \hat{X}(\omega)$ centred at multiples of $2\pi/T$ must be removed by a digital lowpass filter that rejects all frequencies in the range $\pi/T < |\omega| < \pi/T'$. Furthermore, to ensure that the amplitude is correct for sampling interval T', the gain of the filter must be L. That is, (Schafer & Rabiner, 1973)

$$Y(e^{j\omega T'}) = \frac{1}{T} H(e^{j\omega T'})\hat{X}(\omega) \qquad \textbf{Equation C.49}$$

where $H(e^{j\omega T'})$ is periodic with period $2\pi/T'$ and

$$\begin{aligned} H(e^{j\omega T'}) &= L, \quad |\omega| \le \frac{\pi}{T} \\ &= 0, \quad \frac{\pi}{T} < |\omega| < \frac{\pi}{T'} \end{aligned} \qquad \textbf{Equation C.50}$$

C.3.3 Implementation of FIR / IIR Interpolation

It is impossible to realize the ideal lowpass filter required above, so some digital approximation must be employed. Ideally the interpolator will have zero phase, or at most a linear phase delay corresponding to an integer number of samples. This cannot be achieved by IIR filters, though FIR filters can have zero interpolation error due to phase nonlinearity, and the error due to ripples in the pass and stop bands can be made arbitrarily small.

An IIR filter generally requires less computation than a corresponding FIR filter, so IIR filters are commonly employed in DSP applications. However, this is not necessarily the

case in interpolation/decimation problems. In reducing the sample rate, it is not necessary to evaluate every term of the filter output, as only some integer (L) multiple of the samples are required. However, an IIR filter requires that the computations be carried out at the original sampling rate in order to realize the poles of the system function, even though L-1 out of L samples of $\hat{y}(n)$ are discarded. FIR filters do not suffer this limitation, and this, combined with the property exhibited by zero-phase filters, namely

$$h(n) = h(-n) \qquad \textbf{Equation C.51}$$

results in a dramatic reduction in the number of calculations required.

In the case of sampling rate increase, only one out of every L samples is non-zero, again dramatically decreasing the number of computations an FIR filter requires. If an IIR filter is used, relatively little saving is achieved, as once again every term must be calculated. When both decimation and interpolation are employed in sample rate conversion, it becomes obvious that an FIR filter can make use of savings in both processes, whereas an IIR cannot.

The exact form that the design of the FIR filter will take is application dependent. As mentioned previously, some practical situations involve signals sampled at a rate only slightly higher than twice the Nyquist frequency, whilst other situations may not be so limited. Each case places its own requirements on the behaviour of the filter in the passband, stopband and the transition region. Linear programming techniques or discrete Chebyshev approximations may be used to design filters that are optimal in the sense of having the narrowest transition bands for a given pass and stop band ripple. In general such techniques achieve significantly better results than can be obtained by high-order polynomial interpolation filters.

Appendix D: Windowing

D.1 Overview

The windowing options presented to the user of this simulation are the Rectangular, Bartlet, Hanning, Hamming, Kaiser and a variety of Blackman windows. These were introduced and briefly discussed in Appendix A, which included some experiments performed with the differing window types. This appendix provides additional details of these windows, including how they are formed, and an outline of their time and frequency domain properties.

In the definitions that follow, it is assumed that for the Finite and Discrete Fourier Transforms, that the sequences have a zero value for all values of n outside the stated range.

D.2 Rectangular Window

The rectangular (or Dirichlet) window is effectively no window at all. The window is unity over the observation interval and can be thought of as a gating sequence applied to the data so that they are of finite extent. The window for a finite Fourier transform is defined as

$$w(n) = 1.0, \qquad n = -\frac{N}{2}, \ldots, -1, 0, 1, \ldots, \frac{N}{2} \qquad \textbf{Equation D.1}$$

and is shown in Figure D.1a. The same window for a DFT is defined as

$$w(n) = 1.0, \qquad n = 0, 1, \ldots, N-1. \qquad \textbf{Equation D.2}$$

The spectral window for the DFT sequence is given by

$$W(\theta) = \exp\left[-j\left(\frac{N-1}{2}\right)\theta\right]\left[\frac{\sin\left(\frac{N}{2}\theta\right)}{\sin\left(\frac{1}{2}\theta\right)}\right] \qquad \textbf{Equation D.3}$$

The transform of this window is the Dirichlet kernel (Figure D.1b), which exhibits a DFT main-lobe width of two bins and a first side-lobe level approximately 13 dB lower than the main-lobe peak. The side-lobes fall off at 6.0 dB per octave, which is the expected rate for a function with a discontinuity.

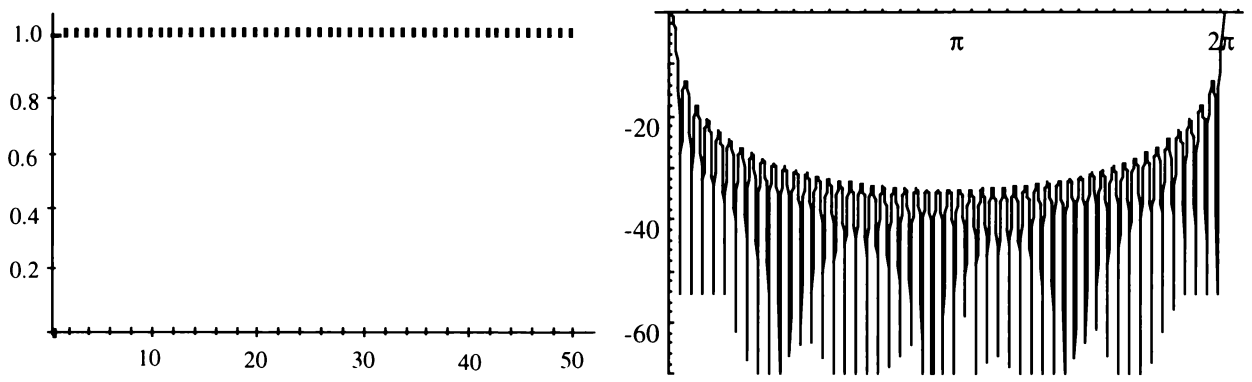


Figure D.1 : Rectangular Window (a) Time Domain (b) Log Magnitude of Transform

D.3 Triangle (Fejer, Bartlet) Window

The triangle window for a finite Fourier transform is defined as

$$W(n) = 1.0 - \frac{|n|}{N/2}, \qquad n = -\frac{N}{2},\ldots,-1,0,1,\ldots,\frac{N}{2} \qquad \textbf{Equation D.4}$$

and is shown in Figure D.2a. The same window for a DFT is defined as

$$W(n) = \begin{cases} \dfrac{n}{N/2} & (n = 0,1,\ldots,\dfrac{N}{2}) \\ W(N-n) & (n = \dfrac{N}{2},\ldots,N-1) \end{cases} \qquad \textbf{Equation D.5}$$

and the spectral window corresponding to this sequence is given by

$$W(\theta)=\frac{2}{N}\exp\left[-j\left(\frac{N}{2}-1\right)\theta\right]\left[\frac{\sin\left(\frac{N}{4}\theta\right)}{\sin\left(\frac{1}{2}\theta\right)}\right]^2 \qquad \textbf{Equation D.6}$$

The transform of this window is the squared Dirichlet kernel (Figure D.2b). Its main-lobe width (between zero crossings) and attenuation of the first side-lobe (compared to the main-lobe peak) are both twice that of the rectangular window.

As the discontinuity of the window resides in the first derivative, rather than in the function itself, the side-lobes exhibit a -12 dB per octave fall off. The triangle is the simplest window which exhibits a non negative transform.

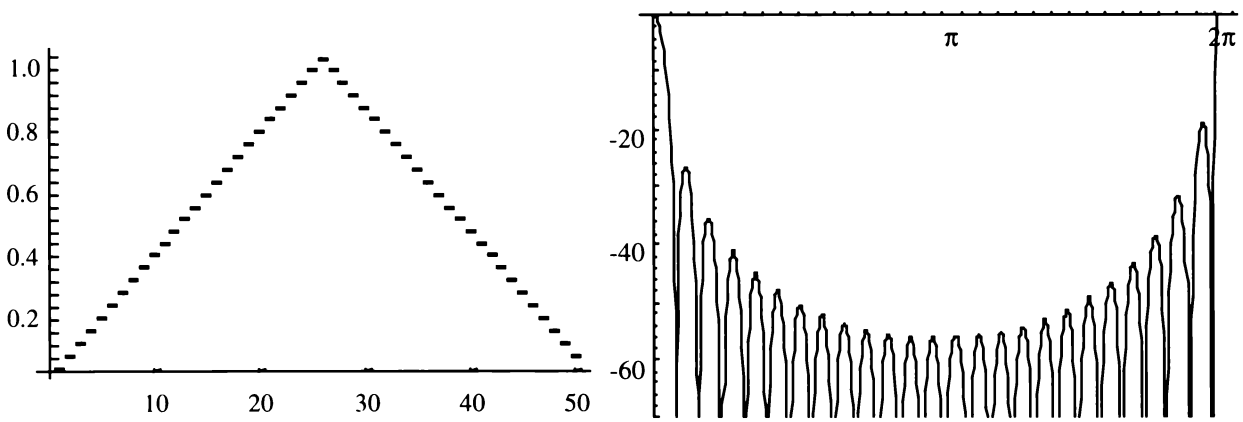


Figure D.2 : Triangular Window (a) Time Domain (b) Log Magnitude of Transform

D.4 Cos$^{\alpha}$(x) Window

This is a family of windows where α is normally some integer. They are easily generated, and exhibit the readily identified properties of the transform of the cosine function. The window for a finite Fourier transform is defined as

$$w(n)=\cos^{\alpha}\left(\frac{n}{N}\pi\right) \qquad n=-\frac{N}{2},\ldots,-1,0,1,\ldots,\frac{N}{2} \qquad \textbf{Equation D.7}$$

and for a DFT as:

$$w(n) = \sin^{\alpha}\left(\frac{n}{N}\pi\right) \qquad n = 0,1,\ldots,N-1 \qquad \textbf{Equation D.8}$$

where the change in origin results in a sine rather than a cosine wave. The most common values for α are the integers 1 through to 4, with 2 being the most well known (as the Hanning window). Explicitly, the Hanning window has the form for finite and discrete Fourier transforms (respectively) as:

$$\begin{aligned} w(n) &= \cos^2\left(\frac{n}{N}\pi\right) \\ &= 0.5\left[1.0+\cos\left[\frac{2n}{N}\pi\right]\right] \qquad n = -\frac{N}{2},\ldots,-1,0,1,\ldots,\frac{N}{2} \end{aligned} \qquad \textbf{Equation D.9}$$

and

$$\begin{aligned} w(n) &= \sin^2\left(\frac{n}{N}\pi\right) \\ &= 0.5\left[1.0-\cos\left[\frac{2n}{N}\pi\right]\right] \qquad n = 0,1,\ldots,N-1 \end{aligned} \qquad \textbf{Equation D.10}$$

The Hanning window is shown in Figure D.3a. This window and its first derivative are continuous, and the discontinuity resides in the second derivative. As a result, the transform falls off at $1/\omega^3$ or at -18 dB per octave. The sampled Hanning window can be written as the sum of the sequences:

$$w(n) = 0.5+0.5\cos\left[\frac{2n}{N}\pi\right], \quad n = -\frac{N}{2},\ldots,-1,0,1,\ldots,\frac{N}{2}-1 \qquad \textbf{Equation D.11}$$

and each sequence has the DFT (Figure D.3b):

$$W(\theta) = 0.5D(\theta) + 0.25\left[D\left(\theta - \frac{2\pi}{N}\right) + D\left(\theta + \frac{2\pi}{N}\right)\right] \qquad \textbf{Equation D.12}$$

where

$$D(\theta) = \exp\left(j\frac{\theta}{2}\right)\frac{\sin\left(\frac{N}{2}\theta\right)}{\sin\left(\frac{1}{2}\theta\right)} \qquad \textbf{Equation D.13}$$

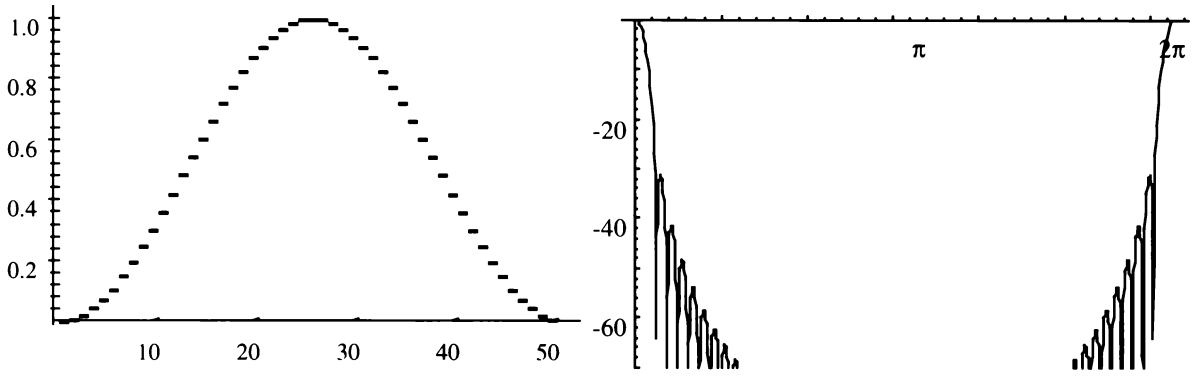


Figure D.3 : Hanning Window (a) Time Domain (b) Log Magnitude of Transform

The Dirichlet kernel at the origin is the transform of the constant 0.5 samples and the pair of translated kernels is the transform of the single cycle of cosine samples. Note that the translated kernels are located on the first zeros of the centre kernel, and are half the size of the centre kernel. Also, side-lobes of the translated kernel are about half the size and of opposite phase to the side-lobes of the central kernel. The summation of the three kernel's side-lobes being in phase opposition tends to cancel the side-lobe structure (Harris, 1978) and suggests a constructive technique to define new windows, the most well-known of which are the Hamming and Blackman windows.

D.5 Hamming Window

The Hamming window can be treated as a modified Hanning window, where an attempt is made to provide exact cancellation of the sidelobes resulting from the summation of the three kernels (Harris, 1978). This can be accomplished by adjusting the relative size of the kernels to achieve a more desirable form of cancellation:

$$w(n) = \alpha + (1-\alpha)\cos\left[\frac{2\pi}{N}n\right]$$
$$W(\theta) = \alpha D(\theta) + 0.5(1-\alpha)\left[D\left(\theta - \frac{2\pi}{N}\right) + D\left(\theta + \frac{2\pi}{N}\right)\right] \qquad \textbf{Equation D.14}$$

Perfect cancellation of the first side-lobe (at $\theta = 2.5$ [$2\pi/N$]) occurs when $\alpha = 25/46$. If 25/46 is taken to 2D (0.54), then the new zero occurs at $\theta \approx 2.6[2\pi/N]$ and a marked improvement in side-lobe level is realized. This is the Hamming window and is identified by:

$$W(n) = \begin{cases} 0.54 + 0.46\cos\left[\frac{2\pi}{N}n\right] & \left(n = -\frac{N}{2},\ldots,-1,0,1,\ldots,\frac{N}{2}\right) \\ 0.54 - 0.46\cos\left[\frac{2\pi}{N}n\right] & (n = 0,1,2,\ldots,N-1.) \end{cases} \qquad \textbf{Equation D.15}$$

The coefficients of the Hamming window are nearly the set which achieve minimum side-lobe levels. If a better approximation to 25/46 is made, i.e. 0.53856, then the side-lobe level is -43 dB and the resultant window is a special case of the Blackman-Harris windows, discussed below. The Hamming window and transform are shown in Figure D.4a and Figure D.4b. Note the deep attenuation at the missing side-lobe position and that the small discontinuity at the boundary of the widow has resulted in a $1/\omega$ (6.0 dB per octave) rate of fall-off. The better side-lobe cancellation does result in a much lower initial side-lobe level of -42 dB.

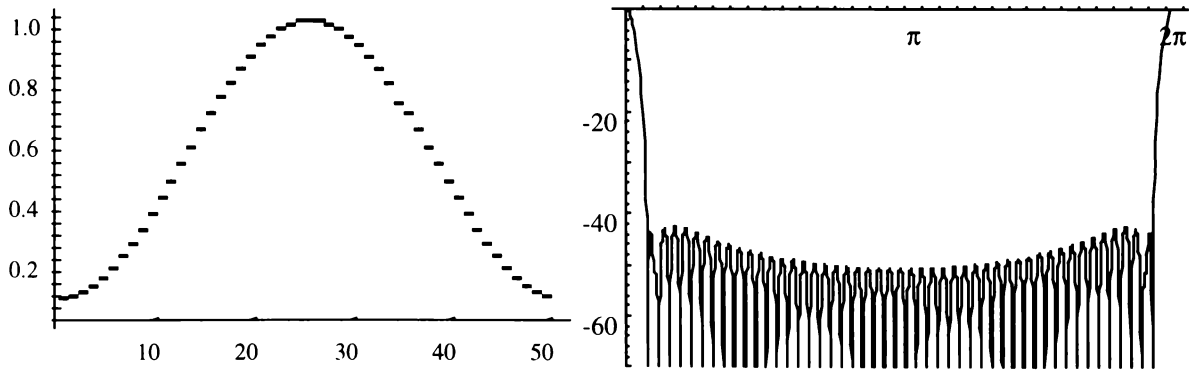


Figure D.4 : Hamming Window (a) Time Domain (b) Log Magnitude of Transform

D.6 Blackman Window

The Hanning and Hamming windows are examples of windows constructed as the summation of shifted Dirichlet kernels. This data window is defined for the finite Fourier transform and the DFT in Equations D.16 and D.17 respectively; Equation D.18 is the resultant spectral window for the DFT given as a summation of the Dirichlet kernels $D(\theta)$ defined by $W(\theta)$ in Equation D.3.

$$W(n)=\sum_{m=0}^{N/2} a_m \cos\left[\frac{2\pi}{N}mn\right] \qquad n=-\frac{N}{2},\ldots,-1,0,1,\ldots,\frac{N}{2} \qquad \textbf{Equation D.16}$$

$$W(n)=\sum_{m=0}^{N/2} (-1)^m a_m \cos\left[\frac{2\pi}{N}mn\right] \quad n=0,1,\ldots,N-1 \qquad \textbf{Equation D.17}$$

$$W(\theta)=\sum_{m=0}^{N/2} (-1)^m \frac{a_m}{2}\left[D\left(\theta-\frac{2\pi}{N}m\right)+D\left(\theta+\frac{2\pi}{N}m\right)\right] \qquad \textbf{Equation D.18}$$

subject to the constraint

$$\sum_{m=0}^{N/2} a_m = 1.0 \qquad \textbf{Equation D.19}$$

We can see that Hanning and Hamming windows are of this form with a_0 and a_1 being non zero. Their spectral windows are summations of three-shifted kernels.

Such windows can be constructed with any K non zero coefficients to achieve a ($2K$ - 1) summation of kernels. If K is kept small, windows with a narrow main lobe can be obtained. Blackman examined this construct for $K = 3$ and found the values of the non zero coefficients which place zeros at $\theta = 3.5\ (2\pi/N)$ and at $\theta = 4.5\ (2\pi/N)$, the position of the third and the fourth side-lobes, respectively, of the central Dirichlet kernel. These exact values and their two place approximations are

$$a_0 = \frac{7938}{18608} \approx 0.42$$
$$a_1 = \frac{9240}{18608} \approx 0.50 \qquad \textbf{Equation D.20}$$
$$a_2 = \frac{1430}{18608} \approx 0.08$$

The form known as the "exact Blackman window" does not truncate these coefficients, but if the two place approximations are employed, the window is then known simply as the "Blackman window". The Blackman window is defined for the finite transform in the following equation and the window is shown in Figure D.5a.

$$W(n) = 0.42 + 0.50\cos\left[\frac{2\pi}{N}n\right] + 0.08\cos\left[\frac{2\pi}{N}2n\right] \qquad n = -\frac{N}{2},\ldots,-1,0,1,\ldots,\frac{N}{2} \quad \textbf{Equation D.21}$$

The exact Blackman window is shown in Figure D.6a. The side-lobe level is 51 dB down for the exact Blackman window and is 58 dB down for the Blackman window. The coefficients of the Blackman window sum to zero (0.42, -0.50, 0.08) at the boundaries while the exact coefficients do not, thus the Blackman window and its first derivative are continuous at the boundary and the window falls off at $1/\omega^3$ or 18 dB per octave. The exact terms (like the Hamming window) have a discontinuity at the boundary and falls off at $1/\omega$.

Families of 3 and 4 term windows in which main-lobe width is traded for side-lobe level have been constructed and are referred to as the Blackman-Harris windows. It has been found that the minimum 3 term and 4 term windows can achieve a side-lobe level of -67 dB and -92 dB respectively. These windows are defined for the DFT by:

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi}{N}n\right) + a_2 \cos\left(\frac{2\pi}{N}2n\right) - a_3 \cos\left(\frac{2\pi}{N}3n\right) \quad n = 0,1,2,\ldots,N-1$$

EquationD.22

The coefficients for these two windows are presented in Table D.1 and the windows are plotted in Figure D.7a and Figure D.8a. The minimum 4-term Blackman-Harris window is presented to the user as a windowing option in the simulation to take advantage of the high side-lobe level attenuation, and the Blackman is offered because of the higher rate of side-lobe fall off, and improved coherent gain.

	Minimum 3-term	**Minimum 4-term**
$\mathbf{a_0}$	0.42323	0.35875
$\mathbf{a_1}$	0.49755	0.48829
$\mathbf{a_2}$	0.07922	0.14128
$\mathbf{a_3}$		0.01168

Table D.1 :Coefficients for Minimum 3-term and 4-term Blackman-Harris windows

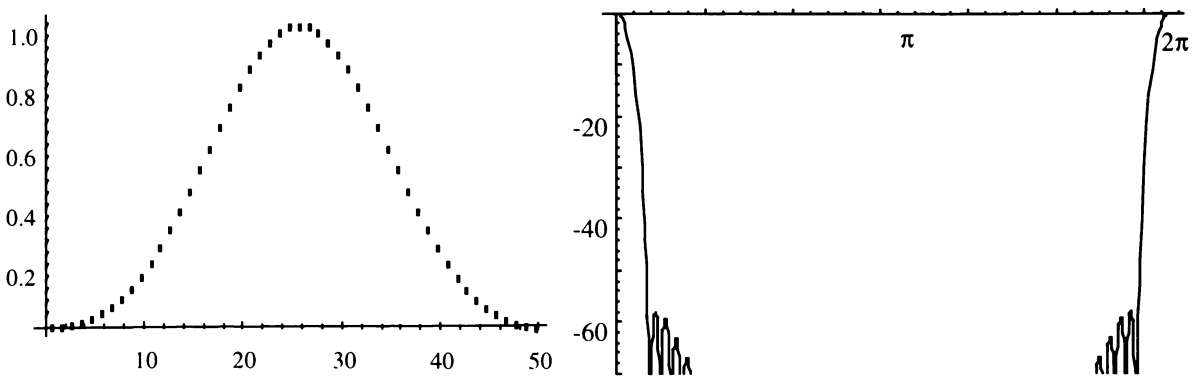


Figure D.5 : Blackman Window (a) Time Domain (b) Log Magnitude of Transform

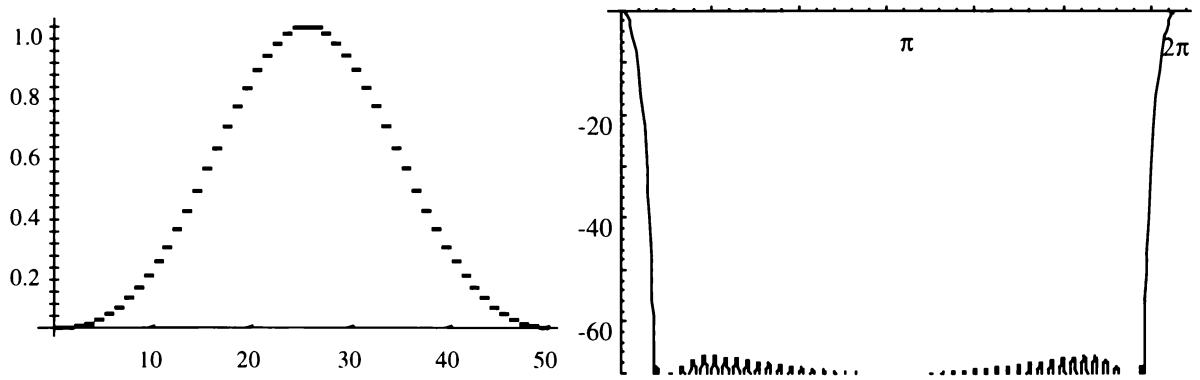


Figure D.6 : Exact Blackman Window (a) Time Domain (b) Log Magnitude of Transform

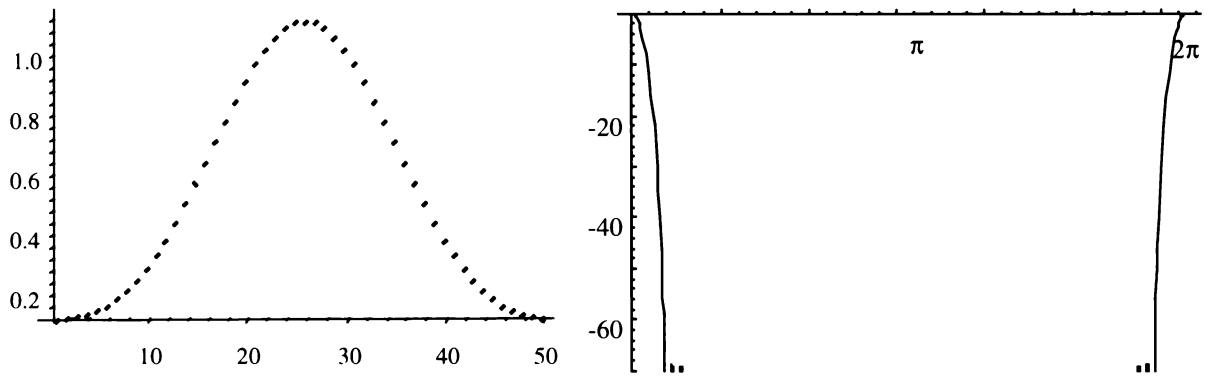


Figure D.7 : Minimum 3-term Blackman-Harris Window (a) Time Domain (b) Log Magnitude of Transform

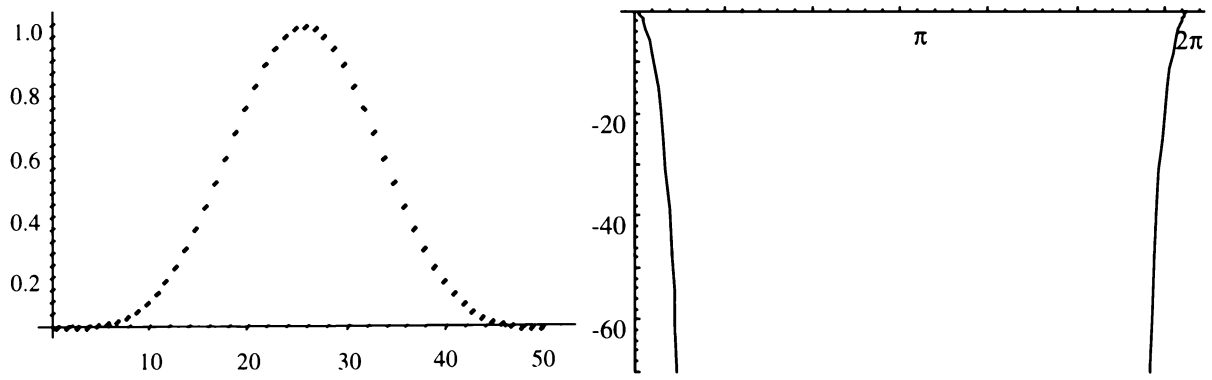


Figure D.8 : Minimum 4-term Blackman-Harris Window (a) Time Domain (b) Log Magnitude of Transform

D.7 Kaiser Window

Ideally, both the window and its transform should be narrow. In reality, this situation is not possible and so some compromise must be reached. In a continuous domain, the prolate spheroidal functions come close to achieving this ideal. Kaiser proposed a simple approximation to these functions in terms of the zero-order modified Bessel function of the first kind. This Kaiser-Bessel window is defined by

$$w(n) = \frac{I_0\left[\pi\alpha\sqrt{1.0-\left(\frac{n}{N/2}\right)^2}\right]}{I_0[\pi\alpha]} \qquad 0 \le |n| \le \frac{N}{2}$$

Equation D.23

where

$$I_0(X) = \sum_{k=0}^{\infty}\left[\frac{\left(\frac{x}{2}\right)^k}{k!}\right]^2$$

Equation D.24

The transform is approximately that of

$$W(\theta) \approx \frac{N}{I_0(\alpha\pi)} \frac{\sinh\left[\sqrt{\alpha^2\pi^2-(N\theta/2)^2}\right]}{\sqrt{\alpha^2\pi^2-(N\theta/2)^2}}$$

Equation D.25

Increasing α from 2.0 to 3.5 greatly attenuates the sidelobes (from -46 dB to -82 dB) - at the expense of broadening the main-lobe width and reducing coherent gain (from 0.49 to 0.37). A 4-sample approximation to these windows is presented as an option in the simulation (Figure D.9a & Figure D.9b). This windowing function is very similar to the $\alpha = 3.0$ Kaiser-Bessel, and produces an identical highest side-lobe level (-69 dB), side-lobe fall off rate (-6 dB/octave) and coherent gain (0.40). This window provides a compromise between the best features of the Hanning, Hamming, and Blackman options previously discussed.

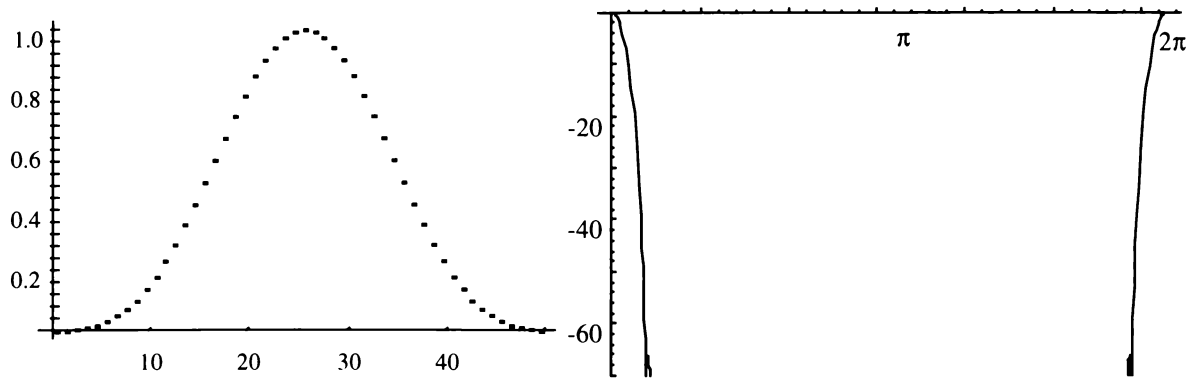


Figure D.9 : 4-Sample Kaiser-Bessel Window (a) Time Domain (b) Log Magnitude of Transform

Appendix E : DRT Corpus

VEAL	FEEL
VEE	BEE
WEED	REED
BEAN	PEEN
SHEET	CHEAT
PEAK	TEAK
GIN	CHIN
VILL	BILL
BID	DID
DINT	TINT
THICK	TICK
FIN	THIN
ZOO	SUE
FOO	POOH
MOON	NOON
DUNE	TUNE
SHOOES	CHOOSE
POOL	TOOL
VOLE	FOAL
THOSE	DOZE
BOWL	DOLE
GOAT	COAT
THOUGH	DOUGH
FORE	THOR
ZED	SAID
THEN	DEN
MET	NET
DENSE	TENSE

MEAT	BEAT
ZEE	THEE
YIELD	WIELD
NEED	DEED
CHEEP	KEEP
KEY	TEA
MITT	BIT
JILT	GILT
HIT	FIT
NIP	DIP
SING	THING
GILL	DILL
MOOT	BOOT
JUICE	GOOSE
COOP	POOP
NEWS	DUES
CHEW	COO
YOU	RUE
MOAN	BONE
JOE	GO
GHOST	BOAST
NOTE	DOTE
SOLE	THOLE
SHOW	SO
MEND	BEND
JEST	GUEST
KEG	PEG
NECK	DECK

FENCE	PENCE		CHAIR	CARE
PENT	TENT		YEN	WREN
VAST	FAST		MAD	BAD
THAN	DAN		JAB	GAB
BANK	DANK		GAT	BAT
GAFF	CALF		NAB	DAB
SHAD	CHAD		SANK	THANK
FAD	THAD		SHAG	SAG
VAULT	FAULT		MOSS	BOSS
THONG	TONG		JAWS	GAUZE
FOUGHT	THOUGHT		YAWL	WALL
DAUNT	TAUNT		GNAW	DAW
SHAW	CHAW		SAW	THAW
BONG	DONG		CAUGHT	THOUGHT
JOCK	CHOCK		MOM	BOMB
VON	BON		JOT	GOT
WAD	ROD		HOP	FOP
BOND	POND		KNOCK	DOCK
VOX	BOX		CHOP	COP
POT	TOT		GOT	DOT

Table E.1 : The DRT Corpus

Appendix F : Neural Networks

F.1 Overview

Neural networks grew out of an attempt to create a system that would function in a similar manner to the human brain. In principle, each neuron in a human brain is a specialised, massively interconnected (several thousand interconnections per neuron) cell, capable of propagating an electrochemical signal through its dendrites, axons and synapses. When a neuron is activated, it fires an electrochemical signal along the axon, which crosses the synapses to link with other neurons. Those neurons may fire if the total received signal is above some threshold, determined by the efficacy of the synapses. It is postulated that human learning consists primarily of altering the strength of these synaptic connections. A very simplified explanation then, is that the brain functions by possessing a very large number of basic processing units, each of which determines, from some weighted sum of its inputs, whether or not to "fire", i.e. produce some binary signal.

The artificial form of the neural network receives a number of weighted inputs (corresponding to synaptic efficacy), sums these inputs, and then if the sum is greater than some threshold, produces an output, which may be binary as in the human brain, or something more complex. The neurons themselves may be one of three different forms: input (corresponding to sensory functions), output (corresponding to motor nerves, e.g. in the hand), or hidden - implying that they perform some internal role in the network. All these forms of neurons need to be connected together.

A simple network, including those used in Statistica, have a feedforward structure. There is a signal flow from the inputs, through any hidden layers, eventually reaching the output unit, with no connections back from later to earlier neurons. Figure F.1 illustrates the feedforward structure finally used in the analysis of the SBS data, this particular network employing eleven input, ten hidden, and one output unit. The input layer introduces the values of the input variables, the hidden and output neurons are connected to all of the units in the preceding layer. The system is trained by using known results, and often employing a process called back

propagation (Remelhart et al., 1986) which uses the data to adjust the network's weights and thresholds to minimise the error in its predictions on the training set.

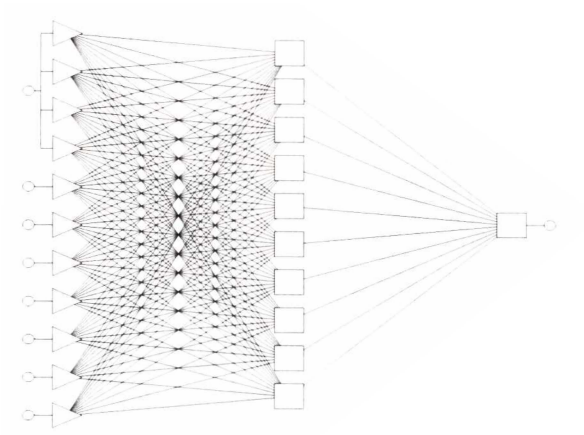


Figure F.1 : Illustration of a Three Layer (8:11-10-1:1) Neural Network

Some of the variables in the SBS data are non-numeric by nature, i.e. *Differencing* – either turned on or off, *Noise* one of [white, one speaker, two speakers, four speakers]. These nominal variables are well handled by Statistica's Neural Networks, provided they do not possess a large number of possible values. This system typically requires at least 100 cases in order to sufficiently analyse the system, a requirement easily met by both the Seneff and SBS results.

F.2 Multilayer Perceptrons

The system described so far is an architecture termed "Multilayer Perceptron" or MLP, and a vital component of such an architecture is obviously determining how many hidden layers to use, and the number of such units in each layer. An often used starting point is to use one hidden layer, with a number of units equal to half the sum of the number of I/O units. Known input cases are used to adjust the weights and thresholds of the system to reduce the error (commonly a sum squared error) between the known result, and the result predicted by the network. With such a nonlinear system, it is not possible to analytically determine the configuration that most reduces this error, and so intelligent guesses (based on an analysis of the

error profiles) need to be made. Each new weighting system is compared to the previous network, and replaces it if it provides a lower error.

One of the most successful networks for the SBS data was a three layer MLP, with the Statistica code 8:11-10-1:1, illustrated in Figure F.1. The 8:11 represents eight input variables, with eleven connections to the ten hidden layers (i.e. one input is used four times). The hidden layers are then connected to a single output. The weightings from the inputs and hidden layers for this three layer MLP network are provided below in Table F.1. In this table, the four values of the variable *Noise*, are independently listed, and the ten hidden layers have a weighting to the output unit. The Threshold value in the first row, is Statistica's threshold value for the relevant hidden unit. (the SBS Threshold is the THRESH variable).

Threshold	0.8591	-0.0224	-1.2024	0.5384	0.1501	0.5587	-0.1109	-0.1275	-0.8862	0.5664	-0.1725
NOI=whi	2.3800	0.9920	1.2964	1.0970	-1.2593	2.6217	-1.2794	1.4071	-0.3693	0.9510	
NOI=1sp	-0.0542	-0.4249	0.5511	0.6795	-0.3249	-0.0990	0.9970	-0.6326	0.5500	0.8842	
NOI=2sp	0.0664	-0.3968	-1.3038	-0.0958	0.7911	-0.047	-0.586	0.7411	0.5747	-0.5755	
NOI=4sp	0.1458	-0.5651	-1.4063	-1.3463	0.0875	-0.1096	-0.4214	-0.5538	0.1957	0.1419	
S_N	-2.1753	-1.9834	1.6861	1.0095	2.3159	-1.8486	2.8619	-0.3126	1.3791	0.3294	
THRESH	-3.8108	-0.279	-0.4139	-0.0309	0.2670	-7.1643	0.3051	-1.7581	4.6830	0.9223	
AMP	-0.391	-0.8206	-0.0480	-0.5886	1.0201	-1.0078	-0.4752	0.2479	-0.5182	-0.387	
PHASE	0.5070	0.2002	-0.1014	-1.1399	-0.5367	0.9786	0.0568	-0.1659	-1.3246	0.0630	
VUS	0.5895	1.0788	1.4461	-0.1727	1.5916	0.4350	0.8424	-0.2335	2.4906	-0.4613	
O_LAP	0.2839	-0.3084	-0.5217	-0.1467	0.3518	0.2689	-0.6038	-0.5006	0.6640	-0.2129	
DIFF	0.3736	-0.6811	-0.1054	0.4849	0.5235	0.4441	-0.2245	-0.8732	0.027	0.8229	
h1#01											2.5463
h1#02											-1.0055
h1#03											-0.9423
h1#04											0.1578
h1#05											-1.4600
h1#06											-2.1782
h1#07											1.1628
h1#08											0.5306
h1#09											2.1236
h1#10											-0.3365

Table F.1 : Weightings for the MLP 8:11-10-1:1 Network Used to Analyse the SBS Data

Statistica's Neural Networks provides an Automatic Network Designer that tries various forms of network architecture in addition to three and four layer MLPs, including the Linear, Radial Basis Function Networks (RBF), and Generalised Regression Neural Networks (GRNN). These forms of networks are briefly discussed in the following sections.

F.3 Radial Basis Function Networks

A very common neural network architecture is the Radial Basis Function Network. Where a Multiplayer Perceptron performs a weighted sum of the inputs from which is subtracted a threshold (referred to as a linear Post Synaptic Potential – or linear PSP), a radial basis function network uses a radial PSP function. In this approach, each unit measures the square of the distance of the input vector from the weight vector, and multiplies it by the threshold.

The radial approach is very localised, (compared to the linear approach which is active over the entire pattern space), and therefore tends to need more units than MLPs. However, MLPs are prone to unjustified extrapolations if data unlike the training data is used, whereas an RBF will always have a near-zero response. Additionally, an MLP may need two or more hidden layers to solve a problem, but the RBF only requires one. The RBF performed very poorly with the SBS data, and so was not extensively examined as a regression option.

F.4 Linear Models

A linear network only has two layers – an input layer and an output layer. The major popularity of neural networks arises from their ability to model nonlinear problems, which obviously the linear network is not capable of doing. However, the linear model does provide a good benchmark with which to judge the performance of the more complex techniques. In operation, this form of network multiplies the inputs by a weights matrix, adds the threshold vector, and passes the value to the output. Understandably, the linear networks were the worst performers for the SBS and Seneff data.

F.5 Generalised Regression Neural Networks (GRNN)

This form of network is commonly used where there is a single numeric output (as is the case for the speech data analysed here), and, as the name suggests, is used for regression problems. GRNN uses a kernel-based approximation whereby the presence of a case indicates some probability density at that point, with progressively decaying evidence in the immediate vicinity. Close to a case there is a high confidence of some probability density, which diminishes with distance from the case point. GRNN applies a Gaussian kernel function to each training case. The output is estimated using a weighted average of the outputs of the training cases, where the weighting is related to the distance of the point from the point being estimated (points nearby contribute most heavily to the estimate).

The first layer of this form of network contains the radial units which store a large number of cluster centres (not very much smaller than the number of cases in the training set). The second layer of a GRNN contains units to estimate the weighted average. Each output has a special unit in this layer which forms the weighted sum for the corresponding output. To obtain the weighted average from the weighted sum, the sum must be divided by the sum of the weighting factors. A single unit in the second layer performs this sum of the weighting factors, and the output layer performs the actual divisions. The second hidden layer therefore, always has exactly one more unit than the output layer, and in regression problems, this implies that a GRNN will typically have two units in the second hidden layer.

A GRNN trains almost instantly, but tends to be large and slow. Similarly to an RBF network, a GRNN does not extrapolate. The GRNN is typically employed when there is a relatively low number of cases (Statistica indicates 500 or less). For the SBS data, whilst the GRNN networks provided quite high verification correlation scores (often up to 0.945), invariably there was a major discrepancy between the training and verification results, indicating that overlearning had occurred (discussed further in 8.5.2). Only for correlation scores below 0.9 did the training and verification results become reasonably close, and at such levels, the 3 and 4 layer MLPs perform considerably better.

Appendix G : Ghitza Filter Centre Frequencies

Filter#	Log freq	Frequency(Hz)	Filter#	Log freq	Frequency(Hz)
1	2.318063	208.00	2	2.331033	214.31
3	2.344003	220.80	4	2.356972	227.50
5	2.369942	234.39	6	2.382912	241.50
7	2.395881	248.82	8	2.408851	256.36
9	2.421821	264.13	10	2.434790	272.14
11	2.447760	280.39	12	2.460730	288.89
13	2.473699	297.65	14	2.486669	306.67
15	2.499639	315.96	16	2.512608	325.54
17	2.525578	335.41	18	2.538548	345.58
19	2.551517	356.06	20	2.564487	366.85
21	2.577457	377.97	22	2.590426	389.43
23	2.603396	401.23	24	2.616366	413.40
25	2.629335	425.93	26	2.642305	438.84
27	2.655275	452.14	28	2.668244	465.85
29	2.681214	479.97	30	2.694184	494.52
31	2.707153	509.51	32	2.720123	524.96
33	2.733093	540.87	34	2.746062	557.27
35	2.759032	574.16	36	2.772002	591.56
37	2.784971	609.50	38	2.797941	627.97
39	2.810911	647.01	40	2.823880	666.62
41	2.836850	686.83	42	2.849820	707.65
43	2.862789	729.10	44	2.875759	751.21
45	2.888729	773.98	46	2.901698	797.44
47	2.914668	821.61	48	2.927638	846.52
49	2.940607	872.18	50	2.953577	898.62
51	2.966546	925.86	52	2.979516	953.93
53	2.992486	982.85	54	3.005455	1012.64

Filter#	Log freq	Frequency(Hz)	Filter#	Log freq	Frequency(Hz)
55	3.018425	1043.34	56	3.031395	1074.97
57	3.044364	1107.55	58	3.057334	1141.13
59	3.070304	1175.72	60	3.083273	1211.36
61	3.096243	1248.08	62	3.109213	1285.92
63	3.122182	1324.90	64	3.135152	1365.06
65	3.148122	1406.44	66	3.161091	1449.08
67	3.174061	1493.00	68	3.187031	1538.26
69	3.200000	1584.89	70	3.212970	1632.94
71	3.225940	1682.44	72	3.238909	1733.44
73	3.251879	1785.99	74	3.264849	1840.13
75	3.277818	1895.91	76	3.290788	1953.39
77	3.303758	2012.60	78	3.316727	2073.61
79	3.329697	2136.47	80	3.342667	2201.24
81	3.355636	2267.97	82	3.368606	2336.72
83	3.381576	2407.55	84	3.394545	2480.54
85	3.407515	2555.73	86	3.420485	2633.21
87	3.433454	2713.03	88	3.446424	2795.27
89	3.459394	2880.01	90	3.472363	2967.31
91	3.485333	3057.26	92	3.498303	3149.94
93	3.511272	3245.43	94	3.524242	3343.81
95	3.537212	3445.18	96	3.550181	3549.62
97	3.563151	3657.22	98	3.576121	3768.08
99	3.589090	3882.31	100	3.602060	4000.00

Appendix H : Glossary

ASR Automatic Speech Recogniser.

Bark A non-linear subdivision of frequencies and bandwidths based upon human auditory perception. First proposed by Zwicker in 1961.

CF Characteristic Frequency.

CVC Consonant-Vowel-Consonant. A form of word construction.

DFT Discrete Fourier Transform.

DRT Diagnostic Rhyme Test. A two option, objective listening test developed by Voiers, and used extensively in this thesis.

DSP Digital Signal Processing.

FAAF Four Alternative Auditory Feature Test. An objective speech evaluation system constructed as a compromise between the FRT and MRT.

FFT Fast Fourier Transform.

FRT Fairbanks Rhyme Test. An early objective speech evaluation system.

GRNN Generalised Regression Neural Network. A form of neural network.

GSD Generalized Synchrony Detector. Used to produce a synchrony spectrum output from Seneff's auditory model.

HMM Hidden Markov Models.

HTK Hidden Markov Model Toolkit. A Speech Recognition System developed by Cambridge University Engineering Department.

IFFT Inverse Fast Fourier Transform.

LPC Linear Predictive Coder. A 10 coefficient version, LPC10 was obtained from the US Department of Defence for use in this thesis.

Mel The mel is a unit of pitch. One thousand mels is the pitch of a 1 kHz pure tone for which the loudness level is 40 phons. The pitch of a sound subjectively judged to be n times that of a 1-mel tone is n mels.

MLP Multilayer Perceptron. A form of Neural Network.

MIT Massachusetts Institute of Technology.

MOS Mean Opinion Score. A five level subjective rating of speech quality.

MRT Modified Rhyme Test. A six option objective speech evaluation system.

Phon The unit of loudness level.

RBF Radial Basis Function Network. A form of neural network.

SBS Synchrony Bands Spectrum. Name given to Ghitza's frequency selection system.

SNR Signal to Noise Ratio. Usually expressed in dB. Also denoted by S/N. It is normally 10 times the log10 ratio of the signal power to noise power. If voltages are considered rather than power, the expression is 20 times the log10 ratio.

Sone The sone is a unit of loudness. One sone is the loudness of a sound for which the loudness level is 40 phons. The loudness of a sound that is judged by a subject to be *n* times that of a 1-sone tone is *n* sones.

SRI Stanford Research Institute.

SPL Sound Pressure Level. Given by 20 times log10 of the ratio of the pressure of the measured sound to a reference pressure (typically 20 $\mu N/m^2$ for sounds in gases).

TI Texas Instruments.

TIMIT Name given to a popular North American Speech corpus jointly developed by MIT, SRI and TI.

VUS Voiced-Unvoiced-Silence. Abbreviation normally used to label a classifier designed to distinguish speech frames as belonging to one of these three categories.

References

Ainsworth, W.A. *Mechanisms of Speech Recognition*. Pergamon Press, 1976.

Ainsworth, W., and Meyer, G. Speech Analysis by Means of a Physiologically-Based Model of the Cochlear Nerve and Cochlear Nucleus. In *Visual Representations of Speech Signals*, edited by Beet, S. and Crawford, M. Wiley, 1993.

Allen, J.B. Cochlear Modeling. In *Mathematical Modeling of the Hearing Process*, edited by Holmes, M.H. and Rubenfeld, L.A. Springer-Verlag, 1980.

Allen, J.B. Cochlear Modeling. *IEEE ASSP Magazine* **2** (1): 3-29, 1985.

Alwan, A., Narayanan, S., Strope, B., and Shen, A. Speech Production and Perception Models and their Applications to Synthesis, Recognition, and Coding. 367-371, 1995.

Ambikairajah, E., Black, N.D., and Linngard, R. Digital Filter Simulation of the Basilar Membrane. *Computer Speech and Language*, 3: 105-118, 1989.

Atal, B. S., and Rabiner, L.R. A Pattern Recognition Approach to Voiced-Unvoiced-Silence Classification with Applications to Speech Recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 24 (3): 201-212, 1976.

Atkinson, K.E. *An Introduction to Numerical Analysis*. John Wiley & Sons, 1978.

Bateman, A., and Yates, W. *Digital Signal Processing Design*. Pitman, 1988.

Beauchamp, K.G., and Yuen, C.K. *Digital Methods for Signal Analysis*. George Allen & Unwin, 1979.

Beet, S.W. Automatic Speech Recognition Using a Reduced Auditory Representation and Position-Tolerant Discrimination. *Computer Speech and Language*, 4: 17-33, 1990.

Bellanger, M.G., Daguet, J.L., and Lepagnol, G.P. Interpolation, Extrapolation, and Reduction of Computation Speed in Digital Filters. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-22, (4): 231-235, 1974.

Bellanger, M.G., Bonnerot, G., and Coudreuse, M. Digital Filtering by Polyphase Network: Application to Sample-Rate Alteration and Filter Banks. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-24, (2): 109-114, 1976.

Beauchamp, K.G., and Yuen, C.K. *Digital Methods for Signal Analysis*. G.Allen & Unwin, 1979.

Billa, J., and El-Jaroudi, A. An Analysis of the Effect of Basilar Membrane Nonlinearities on Noise Suppression. *Journal of the Acoustical Society of America*, 103, (5): 2691-2705, 1998.

Boothroyd, A., and Nittrouer, S. Mathematical Treatment of Context Effects in Phoneme and Word Recognition. *Journal of the Acoustical Society of America*, 84, (1): 101-114, 1988.

Bose, N.K. Digital Filters Theory and Applications. Elsevier, 1965.

Bozic, S.M. *Digital and Kalman Filtering*. Edward Arnold, 1979.

Brigham, E. O. *The Fast Fourier Transform*. Prentice-Hall, 1974.

Burkill, J.C. *A First Course in Mathematical Analysis*. Cambridge University Press, 1974.

Calistri, R.J., and Kallman, H.J. Programs to Administer and Score the Diagnostic Rhyme Test. *Behavior Research Methods, Instruments and Computers*, 18 (1): 57-58, 1986.

Carlson, R., Fant, G., and Granstrom, B. Two Formant Models, Pitch and Vowel Perception. In *Auditory Analysis and Perception of Speech*, edited by Fant, G and Tatham M.A.A.: 55-82, 1975.

Carnegie, D.A., Holmes, G., and Smith L. Implementation of an Auditory Model. *Proceedings of the Third Australian International Conference on Speech Science and Technology*, Melbourne, 1990.

Childers D., and Durling, A. *Digital Filtering and Signal Processing*. West Publishing, 1975.

Clark, G.M. Electrical Stimulation of the Auditory Nerve: The Coding of Frequency, The Perception of Pitch and the Development of Cochlear Implant Speech Processing Strategies for Profoundly Deaf People. *Clinical and Experimental Pharmacology and Physiology*, 23: 766-776, 1996.

Cole, D.R., and Moody, M.P. Enhancement of Degraded Speech for Forensic Application. Unpublished.

Cohen, J.R. Application of an Auditory Model to Speech Recognition. *Journal of the Acoustical Society of America*, 85 (6): 2623-2629, 1989.

Cooke, M.P. An Explicit Time-Frequency Characterization of Synchrony in an Auditory Model. *Computer Speech and Language*, 6: 153-173, 1992.

Cosi, P. Auditory Modelling for Speech Analysis and Recognition. In *Visual Representations of Speech Signals*, edited by Cooke, M. and Beet, S. Chapter 18, 1993.

Cowan, R.S.C., Alcantara, J.I., Blamey, P.J., and Clark, G.M. Preliminary Evaluation of a Multichannel Electrotactile Speech Processor. *Journal of the Acoustical Society of America*, 83, (6): 2328-2338, 1988.

Crochiere, R.E., and Rabiner, L.R. Optimum FIR Digital Filter Implementations for Decimation, Interpolation, and Narrow-Band Filtering. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-23, (5): 444-456, 1975.

Crochiere, R.E., and Rabiner, L.R. Further Considerations in the Design of Decimators and Interpolators. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-24, (4): 296-311, 1976.

Crochiere, R.E., and Rabiner, L.R. Interpolation and Decimation of Digital Signals-A Tutorial Review. *Proceedings of the IEEE*, 69, (3): 301-331, 1981.

Deisher, M.E., and Spanias, A. A HMM-Based Speech Enhancement System Using Harmonic Modelling. Proceedings of the 1997 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, Part 2, 1997.

Delgutte, B., and Kiang, N.Y.S. Speech Coding in Auditory Nerve: I-V. *Journal of the Acoustical Society of America*, 75 (3): 866-918, 1984.

Deng, L., and Geisler, C.D. Responses of Auditory-Nerve Fibers to Nasal Consonant-Vowel Syllables. *Journal of the Acoustical Society of America*, 82 (6): 1977-1988, 1987.

Deng, L., Geisler, C.D. and Greenberg, S. Responses of Auditory-Nerve Fibers to Multiple-Tone Complexes. *Journal of the Acoustical Society of America*, 82, (6): 1989-2000, 1987.

Dermody, P., Raicevich, G., and Katsch, R. Comparative Evaluations of Auditory Representations of Speech. In *Visual Representations of Speech Signals*, edited by Cooke, M., Beet, S., and Crawford, M. Chapter 21, Wiley, 1993

Di Benedetto, M. Vowel Representation: Some Observations on Temporal and Spectral Properties of the First Formant Frequency. *Journal of the Acoustical Society of America*, 86, (1): 55-56, 1989.

Duggirala, V., Studebaker, G.A., Pavlovic, C.V., and Sherbecoe, R.L. Frequency Importance Functions for a Feature Recognition Test Material. *Journal of the Acoustical Society of America*, 83, (6): 2372-2382, 1988.

Duifhuis, H. Current Developments in Peripheral Auditory Frequency Analysis. In *Working Models of Human Perception*, edited by Elsendoor, B.A.G., and Bouma, H. Academic Press, 1989.

Eggermont, J.J. Analog Modelling of Cochlear Adaption. *Kybernetik*, 14: 117-126, 1973.

Fairbanks, G. Test of Phonemic Differentiation: The Rhyme Test. *Journal of the Acoustical Society of America*, 30: 596-600, 1958.

Forsythe, G.E., Malcolm, M.A., and Moler, C.B. *Computer Methods for Mathematical Computations*. Prentice-Hall, 1977.

Foster, J.R., and Haggard, M.P. The Four Alternative Auditory Feature Test (FAAF)-Linguistic and Psychometric Properties of the Material with Normative Data in Noise. *British Journal of Audiology*, 21: 165-174, 1987.

French, N.R., and Steinberg, J.C. Factors Governing the Intelligibility of Speech Sounds. *Journal of the Acoustical Society of America*, 19, (1): 90-119, 1947.

Gabel, R.A., and Roberts, R.A. *Signals and Linear Systems* (2nd ed). John Wiley & Sons, 1980.

Gatehouse, S. Apparent Auditory Deprivation Effects of Late Onset: The Role of Presentation Level. *Journal of the Acoustical Society of America*, 86, (6): 2103-2106, 1989.

Ghitza, O. Auditory Nerve Representation as a Front end for Speech Recognition in a Noisy Environment. *Computer Speech and Language*, **1** (2): 109-130, 1986.

Ghitza, O. Robustness Against Noise: The Role of Timing-Synchrony Measurement. *International Conference on Acoustics, Speech and Signal Processing,* ICASSP **4**: 2372-2375, 1987.

Ghitza, O. Auditory Nerve Representation Criteria for Speech Analysis/Synthesis. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-35 (6): 736-740, 1987.

Ghitza, O. Temporal Non-place Information in the Auditory-Nerve Firing Patterns as a Front end for Speech Recognition in a Noisy Environment. *Journal of Phonetics*, 16: 109-123, 1988.

Ghitza, O. Adequacy of Auditory Models to Predict Human Internal Representation of Speech Sounds. *Journal of the Acoustical Society of America*, 93 (4): 2160-2171, 1993.

Ghitza, O. Processing of Spoken CVCs in the Auditory Periphery. I. Psychophysics. *Journal of the Acoustical Society of America*, 94 (5): 2507-2516, 1993.

Ghitza, O. Auditory Models and Human Performance in Tasks Related to Speech Coding and Speech Recognition. *IEEE Transactions on speech and Audio Processing*, 2 (1): 115-132, 1994.

Gold, B., and Rabiner, L. Parallel Processing Techniques for Estimating Pitch Periods of Speech in the Time Domain. *Journal of the Acoustical Society of America*, 46 (2):442-448, 1969.

Goldstein, J.L., and Srulovicz, P. Auditory-Nerve Spike Intervals as an Adequate Basis for Aural Spectrum Analysis. In *Psychophysics and Physiology of Hearing*, edited by Evans, E.F. and Wilson, J.P.: 337, 1977.

Goodman, D.J., and Carey, M.J. Nine Digital Filters for Decimation and Interpolation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-25, (2): 121-126, 1977.

Grant, K.W., and Seitz, P.F. The Recognition of Isolated Words and Words in Sentences: Individual Variability in the Use of Sentence Context. *Journal of the Acoustical Society of America*, 107, (2): 1000-1011, 2000.

Grant, P.M. Speech Recognition Techniques, Electronics and Communication *Engineering Journal*: 37-48, Feb 1991.

Greenberg, S. The Ears Have It: The Auditory Basis of Speech Perception. *Proceedings of the ICPhS* 3: 34-41, Stockholm, 1995.

Greenspan, S.L., Bennett, R.W., and Syrdal. A.K. Evaluation of the Diagnostic Rhyme Test. International Journal of Speech Technology, 2(3), 1998.

Greenwood, D.D. Critical Bandwidth and the Frequency Coordinates of the Basilar Membrane. *Journal of the Acoustical Society of America*, 33, (10): 1344-1356, 1961.

Griffith, J.D. Rhyming Minimal Contrasts: A Simplified Diagnostic Articulation Test. *Journal of the Acoustical Society of America*, 42, (1): 236-241, 1967.

Hamming, R.W. Numerical Methods for Scientists and Engineers. McGraw-Hill, 1962.

Hamming, R.W. *Digital Filters*. Prentice-Hall, 1977.

Hansen, M., and Kollmeier, B. Continuous Assessment of Time-Varying Speech Quality. *Journal of the Acoustical Society of America*, 106, (5): 2888-2899, 1999.

Harris, R.J. On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform. *Proceedings of the IEEE*, 66, (1): 51-82, 1978.

Hick, W.E. On the Rate of Gain of Information. *Quarterly Journal of Experimental Psychology*, 4: 11-26, 1952.

Hodgson, M. Experimental Investigation of the Acoustical Characteristics of University Classrooms. *Journal of the Acoustical Society of America*, 106, (4): 1810-1819, 1999.

Holton, T. A Computation Approach to Recognition of Speech Features Using Models of Auditory Signal Processing. Proceedings of ICPhS, 3: 50-57, Stockholm, 1995.

Hosking, R.J., Joyce, D.C., and Turner, J.C. *First Steps in Numerical Analysis.* Hodder and Stoughton, 1978.

House, A.S., Williams, C.E., Hecker, M.H.L., and Kryter, K.D. Articulation-Testing Methods: Consonantal Differentiation with a Closed-Response Set. *Journal of the Acoustical Society of America*, 37, (1): 158-166, 1965.

Hughes, P.M. Formant Based Speech Synthesis. In Speech and Language Processing, edited by Wheddon, C., and Linggard, R. Chapman and Hall, 1990.

Jakobson, R., Fant, C.G.M., and Halle, M. Preliminaries to Speech Analysis: The Distinctive Features and Their Correlates. *Technical Report* No. 13, Acoustic Laboratory, Massachusetts Institute of Technology, 1952.

Jankowski, C.R., Vo, H.H., and Lippmann, R.P. A Comparison of Signal Processing Front Ends for Automatic Word Recognition. *IEEE Transactions on Speech and Audio Processing*, 3 (4): 286-293, 1995.

Jenison, R.L., Greenberg, S., Kluender, K.R., and Rhode, W.S. A Composite Model of the Auditory Periphery for the Processing of Speech Based on the Filter Response Functions of Single Auditory-Nerve Fibers. *Journal of the Acoustical Society of America*, 90 (2): 773-786, 1991.

Johnson, D.E. *Introduction to Filter Theory.* Prentice-Hall, 1976.

Kaiser, J.F. Nonrecursive Digital Filter Design Using the I_0 – sinh Window Function. *Proceedings of the 1974 IEEE Symposium on Circuits and Systems:* 20-23, 1974.

Kajita, S., and Itakura, F., Subband-Autocorrelation Analysis and its Application for Speech Recognition. 2 193-196, 1994.

Kang, J. Comparison of Speech Intelligibility Between English and Chinese. *Journal of the Acoustical Society of America*, 103, (2): 1213-1216, 1998.

Kewley-Port, D., and Bishnu, S.A. Perceptual Differences Between Vowels Located in a Limited Phonetic Space. *Journal of the Acoustical Society of America*, 85, (4): 1726-1731, 1989.

Kewley-Port, D., and Zheng, Y. Auditory Models of Formant Frequency Discrimination for Isolated Words. *Journal of the Acoustical Society of America*, 103, (3): 1654-1666, 1998.

Khanna, S.M., and Leonard, D.G.B. Basilar Membrane Response Measured in Damaged Cochleas of Cats. In *Mathematical Modeling of the Hearing Process*, edited by Holmes, M.H., and Rubenfeld, L.A. Springer-Verlag, 1980.

Kiang, N.Y.S., Watanabe, T., Thomas, E.C., and Clark, L.F. *Discharge Patterns on Single Fibers in the Cat's Auditory Nerve*. MIT Press, 1965.

Kiang, N.Y.S. A Survey of Recent Developments. In The Study of Auditory Physiology. Annals of Otology, Rhinology, and Laryngology, 77: 656-675, 1968.

Kitawaki, H., Nagabuschi, H., and Itoh, K. Objective Quality Evaluation for Low-Bit-Rate Speech Coding Systems. *IEEE Journal on Selected Areas in Communication*, 6: 242-248, 1988.

Klatt, D.H. Speech Processing Strategies Based on Auditory Models. *The Representation of Speech in the Peripheral Auditory System*, edited by Carlson, R., and Granstrom, B. 1982.

Kluender, K.R., and Lotto, A.J. Virtues and Perils of an Empiricist Approach to Speech Perception. *Journal of the Acoustical Society of America*, 105, (1): 503-511, 1999.

Kreiman, J., and Gerratt, B.R. Validity of Rating Scale Measures of Voice Quality. *Journal of the Acoustical Society of America*, 104, (3): 1598-1607, 1998.

Laflamme, C., Salami, R., Matmti, R., and Adoul, J. Harmoinc-Stochastic Excitation (HSX) Speech Coding Below 4 kbit/s. Proceedings of the 1997 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, 1996.

Lagadec, R., Pelloni, D., and Weiss, D. A 2-Channel, 16-Bit Digital Sampling Frequency Converter for Professional Digital Audio. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 1981.

Lazzaro, J., and Wawrzynek, J. Speech Recognition Experiments with Silicon Auditory Models. *Analog Integrated Circuits and Signal Processing*, 13: 37-51, 1997.

Licklider, J.C.R., and Pollack, I. Effects of Differentiation, Integration, and Infinite Peak Clipping Upon the Intelligibility of Sound. *Journal of the Acoustical Society of America*, 20 (1): 42-51, 1948.

Lippman, R.P. Speech Recognition by Machines and Humans. *Speech Communication*, 22: 1-15, 1997.

Littler, T.S. *The Physics of the Ear*. Pergamon Press, 1965.

Loizou, P.C., Dorman, M., and Tu, Z. On the Number of Channels Needed to Understand Speech. *Journal of the Acoustical Society of America*, 106, (4): 2097-2103, 1999.

Ludeman, L.C. *Fundamentals of Digital Signal Processing*. John Wiley & Sons, 1987.

Lyon, R.F. Computational Models of Neural Auditory Processing. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing: 36.1.1-36.1.4, 1984.

Lyon, R.F. A Computational Model of Filtering, Detection and Compression in the Cochlea. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*: 1282-1285, 1985.

Lyon, R.F., and Mead, C. An Analog Electronic Cochlea. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36, (7): 1119-1134, 1988.

Mannell, R.H. The Effects of Phase Information on the Intelligibility of Channel Vocoded Speech. *Proceedings of the Fourth Australian International Conference on Speech Science and Technology*, Melbourne, 1992.

Maurice, R.D.A. Convolution and Fourier Transforms for Communications Engineers. Pentech Press, 1976.

McAulay, R.J., and Quatieri, T.F. Speech Analysis/Synthesis Based on a Sinusoidal Representation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34 (4): 744-754, 1986.

McAulay, R., Park, T., Quatieri, T., and Sabin, M. Sine-Wave Amplitude Coding at Low Data Rates. In *Advances in Speech Coding*, edited by Atal, B.S., Cuperman, V., and Gersho, A.: 203-214, 1991. XXX publisher required

McAulay, R.J., and Quatieri, T.F. Sinusoidal Coding. In *Speech Coding and Synthesis*, edited by Kleijn, W.B., and Paliwal, K.K. 1995.

Mendel, L.L., Hamill, B.W., Hendrix, J.E., Crepeau, L.J., Pelton, J.D., Miley, M.D., and Kadlec, E.E. Speech Intelligibility Assessment in a Helium Environment. II. The Speech Intelligibility Index. *Journal of the Acoustical Society of America*, 104, (3): 1609-1615, 1998.

Miller, G.A., Heise, G.A., and Lichten, W. The Intelligibility of Speech as a Function of the Context of the Test Materials. *Journal of Experimental Psychology*, 41: 329-335, 1951.

Miller, G.A., and Nicely, P. An Analysis of Perceptual Confusions Among Some English Consonants. *Journal of the Acoustical Society of America*, 27: 338-352, 1955.

Moore, B.C.J. *Introduction to the Psychology of Hearing*. MacMillan Press, 1977.

Moore, B.C.J., and Glasberg, B.R. Suggested Formulae for Calculating Auditory-Filter Bandwidths and Excitation Patterns. *Journal of the Acoustical Society of America*, 74, (3): 750-753, 1983.

Morton, K. Expectations for Assessment Techniques Applied to Speech Synthesis. *Proceedings of the Institute of Acoustics*, 13 (2): 1991.

Moschytz, G.S., and Horn, P. *Active Filter Design Handbook*. Wiley, 1981.

Nabelek, A.K., and Letowski, T.R. Similarities of Vowels in Nonreverberant and Reverberant Fields. *Journal of the Acoustical Society of America*, 83, (5): 1891-1899, 1988.

Nearey, T.M. Speech Perception as Pattern Recognition. *Journal of the Acoustical Society of America*, 101, (6): 3241-3254, 1997.

O'Shaughnessy, D. *Speech Communication-Human and Machine*. Addison-Wesley, 1987.

Ozawa, K., and Logan, J.S. Perceptual Evaluation of Two Speech Coding Methods by Native and Non-native Speakers of English. *Computer Speech and Language* 3: 53-59, 1989.

Papoulis, A. *Signal Analysis*. McGraw-Hill, 1977.

Parsons, T. *Voice and Speech Processing*, McGraw-Hill, 1987.

Peled A., and Liu, B. Digital Signal Processing-Theory, Design, and Implementation. Wiley, 1976.

Pelton, G.E. *Voice Processing*. McGraw-Hill, 1993.

Quatieri, T.F., and McAulay, R.J. Speech Transformations Based on a Sinusoidal Representation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-34, (6): 1449-1464, 1986.

Ramstad, T.A. Digital Methods for Conversion Between Arbitrary Sampling Frequencies. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-32, (3): 577-591, 1984.

Rabiner, L.R., Cooley, J.W., Helms, H.D., Jackson, L.B., Kaiser, J.F., Rader, C.M., Schafer, R.W., Steiglitz, K., and Weinstein, C.J. Terminology in Digital Signal Processing. *IEEE Transaction on Audio and Electroacoustics*, AU-20: 322-337, 1972.

Robert, A., and Eriksson, J.L. A Composite Model of the Auditory Periphery for Simulating Responses to Complex Sounds. *Journal of the Acoustical Society of America*, 106, (4): 1852-1864, 1999.

Rose, J.E., Brugge, J.F., Anderson, D.J., and Hind, J.E. Phase-Locked Response to Low-Frequency Tones in Single Auditory Nerve Fibers of the Squirrel Monkey. *Journal of Neurophysiology*, 30: 769-793, 1967.

Rose, J.E., Gross, N.B., Geisler, D., and Hind, J.E. Some Neural Mechanisms in the Inferior Colliculus of the Cat which may be Relevant to Localization of a Sound Source. *Journal of Neurophysiology*, 29: 288-314, 1966.

Rosenberg, M.E. *Sound and Hearing*. Edward Arnold, 1982.

Rothhauser, E.H. IEEE Recommended Practice for Speech Quality Measurements. *IEEE Transactions on Audio and Electroacoustics*, 17, (3): 225-246, 1969.

Sachs, M.B., and Young, E.D. Encoding of Steady State Vowels in the Auditory Nerve: Representation in Terms of Discharge Rate. *Journal of the Acoustical Society of America*, 66 (2): 470, 1979.

Sachs, M.B., and Young, E.D. Effects of Nonlinearities on Speech Encoding in the Auditory Nerve. Journal of the Acoustical Society of America, 68, 858-875, 1980.

Sachs, M.B., Young, E.D., and Miller, M.I. Encoding of Spech Features in the Auditory Nerve. In The Representation of Speech in the Peripheral Auditory System, edited by Carlson, R., and Granström, B. Elsevier Biomedical Press, 1982.

Schafer, R.W., and Rabiner, L.R. A Digital Signal Processing Approach to Interpolation. *Proceedings of the IEEE*, 61, (6): 692-702, 1973.

Schwartz, M., and Shaw, L. Signal Processing: Discrete Spectral Analysis, Detection, and Estimation. McGraw-Hill, 1975.

Sedra, A.S., and Smith, K.C. *Microelectronic Circuits* (2nd ed.). CBS College Publishing, 1987.

Sekuler, R., and Blake, R. *Perception.* Alfred A. Knopf, 1985.

Seneff, S. Pitch and Spectral Estimation of Speech Based on Auditory Synchrony Model. *Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing:* 36.2.1-36.2.4, 1984.

Seneff, S. A Computational Model for the Peripheral Auditory System: Application to Speech Recognition Research. *Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing*: 1983-1986, 1986.

Seneff, S. A Joint Synchrony/Mean-rate Model of Auditory Speech Processing. In *Speech Analysis*, Academic Press, 1988.

Sheffield, A.E., Krishnamurthy, A., Davidson, S., and Feth, L. Comparing Singled- and Two-Channel Telephone Speech Enhancement of Elderly Hearing-Impaired Listeners. Journal of the Acoustical Society of America, 107(5), 2000.

Simmons, R.B., Epley, J.M., Lummis, R.C., Guttman, N., Frishkopf, L.S., Harmon, L.D., and Zwicker, E. Auditory Nerve: Electrical Stimulation in Man. *Science*, 148: 104-106, 1965.

Shamma, S.A. Speech Processing in the Auditory System I: The Representation of Speech Sounds in the Responses of the Auditory Nerve. *Journal of the Acoustical Society of America*, 78, (5): 1612-1621, 1985.

Shamma, S.A. Speech Processing in the Auditory System II: Lateral Inhibition and the Central Processing of Speech Evoked Activity in the Auditory Nerve. *Journal of the Acoustical Society of America*, 78, (5): 1622-1632, 1985.

Shively, R.R. On Multistage Finite Impulse Response (FIR) Filters with Decimation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-23, (4): 353-357, 1975.

Slaney, M. Lyon's Cochlear Model. *Apple Computer Technical Report* #13, 1988.

Smith, C. Perception of Vocoder Speech Processed by Pattern Matching. *Journal of the Acoustical Society of America*, 46, (2): 1562-1571, 1969.

Southcott, C.B., Boyd, I., Coleman, A.E., and Hammett, P.G. Low Bit Rate Speech Coding for Practical Applications. In *Speech and Language Processing*, edited by Wheddon, C., and Linggard, R. Chapman and Hall, 1990.

Stearns, S.D. *Digital Signal Analysis*. Hayden Book Co. 1975.

Tempelaars, S. *Signal Processing, Speech and Music*. Swets and Zeitlinger, 1996.

Tchorz, J., and Kollmeier, B. A Model of Auditory Perception as Front End for Automatic Speech Recognition. *Journal of the Acoustical Society of America*, 106, (4): 2040-2050, 1999.

Tong, Y.C., Lim, H.H., and Clarke, G.M. Synthetic Vowel Studies on Cochlear Implant Patients. *Journal of the Acoustical Society of America*, 84, (3): 876-887, 1988.

Tremain, T.E. The Government Standard Linear Predictive Coding Algorithm: LPC-10. *Speech Technology*: 40-49, April, 1982.

Van den Enden, A.W.M., and Verhoeckx, N.A.M. Digital Signal Processing: Theoretical Background. *Philips Technical Review*, 42, (4): 110-144, December 1985.

Van Santen, J.P.H. Perceptual Experiments for Diagnostic Testing of Text-to-Speech Systems. *Computer Speech and Language*, 7: 49-100, 1993.

Van Summers, W. F1 Structure Provides Information for Final-Consonant Voicing. *Journal of the Acoustical Society of America*, 84, (2): 485-492, 1988.

Voiers, W.D. Diagnostic Evaluation of Speech Intelligibility. In *Speech Intelligibility and Speaker Recognition, Vol 2, Benchmark Papers in Acoustics*, edited by Hawley, M.E. Dowden, Hutchinson and Ross, 1977.

Voiers, W.D. Evaluating Processed Speech using the Diagnostic Rhyme Test. *Speech Technology:* 30-39, 1983.

Voiers, W.D., Cohen, M.E., and Mickunas, J. Evaluation of Speech Processing Devices. In *Intelligibility, Quality, Speaker Recognizability, Final Report,* OAS, 1965.

von Békèsy, G. *Experiments in Hearing*. McGraw-Hill, 1960.

Wang, S., Sekey, A., Gersho, A. An Objective Measure for Predicting Subjective Quality of Speech Coders. *IEEE Journal on Selected Areas in Communication*, 10 (5): 819-829, 1992.

Watanbe, T. and Hayashi, S. An Objective Measure Based on an Auditory Model for Assessing Low-Rate Coded Speech. *IEICE Transactions on Information Systems*, 78, (6): 751-757, 1995.

Welch, P.D. The Use of the Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging Over Short, Modified Periodgrams. *IEEE Transactions on Audio Electroacoustics*, 15: 70-73, 1967.

Wever, E.G., and Bray, C.W. The Perception of Low Tones and the Resonance-Volley Theory. *Journal of Psychology*, 3: 110-114, 1937.

Wiegrebe, L., Hirsch, H.S., Patterson, R.D., and Fastl, H. Temporal Dynamics of Pitch Strength in Regular-Interval Noises: Effect of Listening Region and an Auditory Model. *Journal of the Acoustical Society of America*, 107, (6): 3343-3350, 2000.

Witten, I.H. *Principles of Computer Speech*. Academic Press, 1982.

Wong, W., and Mori, S. The Magical "Wave" Seven, Plus or Minus Two. *Journal of the Acoustical Society of America*, 104, (1): 390-398, 1998.

Yost, W.A., and Nielsen, D.W. *Fundamentals of Hearing*. Holt, Rinehart and Winston, 1977.

Young, E.D., and Sachs, M.B. Representation of Steady-State Vowels in the Temporal Aspects of the Discharge Patterns of Populations of Auditory-Nerve Bribers. *Journal of the Acoustical Society of America*, 66: 1381-1403, 1979.

Young, S.J. HTK: *Hidden Markov Model Toolkit V1.2 Installation Guide*. Cambridge University Engineering Department – Speech Group, 1990.

Internet References

http://ket.otago.ac.nz/hyspeech/corpusinfo.html (accessed 1997 and 2000)