ADAPTIVE DATA COMPRESSION WITH MEMORY

K. Frimpong-Ansah

A Thesis submitted for the
Degree of Doctor of Philosophy of the University of London
and the
Diploma of Membership of Imperial College

November 1985

Electrical Engineering Department
Imperial College of Science and Technology
University of London

ABSTRACT

In this thesis a class of adaptive coding schemes for speech and image compression which make use of previously coded data, is proposed, analysed and experimentally investigated.

The first and second chapters of this thesis are an introduction and a review of block coding techniques respectively.

The third chapter describes the basic coding scheme, whose concept is the cement for this thesis. This involves the representation of variable length blocks of data by previously coded and transmitted source symbols. The coordinate of the previously transmitted symbols and the block size in question form the information sent down the communication channel. Several variations on this scheme are presented and experimentally studied with artificially generated data, speech data and image data. Results show the algorithm to be capable of achieving good compression, requiring no prior knowledge of the statistics of the source to be coded.

The use of some of the particular properties of image and speech data, allow more efficient compression of these types of source, using variations on the above method. This avenue is extensively studied for speech data.

Chapter four develops the theory associated with the limits of the performance of the class of adaptive coding schemes proposed. It is shown that as some block size parameter is allowed to approach infinity, for the case of zero distortion, the coding rate approaches the Shannon entropy, plus a small factor associated with

adaptation. The theoretical properties are also discussed, but not quite as conclusively, in the case of coding with a fidelity criterion.

Chapter five is a review of scalar coding schemes, examples being PCM, DPCM, ADPCM and Multipath-search-coding.

Chapter six gives the results of studies in adaptive multipath search coding, where the adaptation information is derived from previously coded data.

TABLE OF CONTENTS

STATEMENT OF ORIGINALITY

The following is a brief list of the contributions which are, to the best of my knowledge, original to this thesis.

Chapter 3

1) The use of blocks of previously coded data, to approximate blocks of data to be coded, in the case of grey scale images and speech coding [Section 3.2 to 3.4].

2) The method 'b' of section 3.2 for sampling the set of previously coded symbols to find an approximation to a block of symbols being coded.

3) The Fourier transform coding of a baseband residual signal, for residual excited LPC [Section 3.6.3.1]

4) The concept of variable bit rates for the transmission of the excitation and the model parameters for residual excited LPC coding [Section 3.6.3].

Chapter 4

1) A discussion of the theoretical performance of the concept of coding source symbols by approximating these with blocks of previously coded data [Sections 4.3 to 4.4 and 4.5].

2) The proof of the theorem of section 4.5.1, on the probability of observing an outcome, within distance d , of any block of N symbols each belonging to the sample space of an ergodic source, as N is approaches infinity.

Chapter 6

1) The presentation of results for step size adaptation for the scalar coding of images where a small default step size is employed, except in a region of slope overload (where an edge is observed) otherwise a immediate switch to the default step size is effected [Section 6.3.1]

2) The concept of adaptive multipath search coding in blocks, with a linear prediction based graph (tree or trellis) weighting, where the prediction parameters are derived from previously coded symbols [Section 6.4]

3) The presentation of results for image coding employing a very simple codebook based convolutional coder as originally described by Stewart, Gray and Linde-(1982)]

4) The proposal and the presentation of results for an adaptive version of the convolutional coder described Stewart et al.[Section 6.6]

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisor, Dr A. G. Constantinides for his guidance and for his great forbearance during the course of the research and the preparation of this thesis. There were several occasions on which I am sure his patience must have been sorely tried, when trying to obtain some work from a wayward student. I suspect that Dr Constantinides is very relieved at the completion of this thesis.

I would like to thank my colleagues both present and past for the several hours of fruitful and unfruitful discussion, both social and academic, during my stay at College. Especially deserving of mention are Drs M. G. X. Fernando and G. M. X. Fernando, Mr. A. J. Singarajah, Mr. C. C. Meewella, Drs A. S. Gujral, C. N. Daskalakis, T. A. Lanfear, P. J. Wilson, R. A. Valenzuela, I. M. I. Habbab, Mr. M. Toutounjee, Mr. K. S. Chase, Dr A. S. Kwabwe, Mr. C. K. Patni, Mr. S. A. Hanna, Mr. G. M. Livadas and Mr. L. J. Wu.

Finally I would like to very sincerely thank my parents for the support and encouragement given to me during the course of my research.

INTRODUCTION

This chapter is a short and rather unconventional introduction. The style of the chapter has been directed, not altogether regrettably, by the diverseness of the topics covered in this thesis. The thesis investigates the use of a technique for solving a class of problems. This is appreciably different from most theses, which deal with a problem and where the conventionally adopted approach is the following. Granted a certain problem, several ways of tackling this are investigated, some proposed. The research culminates in the advocation of a 'good' technique for solving this problem. An example of a conventional thesis is one which is concerned with say image transmission over a certain class of channel for a certain restriction on rate, a particular one being coding for video conferencing, to find for this particular application, a technique that best achieves the aim, bearing in mind the particular conditions in which it is to be used. A thesis of this sort might be introduced with the background to video-conferencing, images and in particular, sequences of images.

Because the glue for this thesis is a technique for tackling some data compression problems and thus diverse applications are dealt with, the relevant chapters deal with the background to these problems. This deprives the thesis introduction of half its traditional content. Well then, it might be asked, what does this chapter contain? It concerns itself with firstly, why data might be compressed and secondly describes, briefly the structure and approach to this thesis.

1.1 Why data compression?

At being presented with yet another thesis on data compression, the reader justifiably expects some sort of reason from the author for the choice of this topic for research. So the question to be answered is, why do we want to do data (speech, image etc.) compression? It should be stated, to start with, that data compression should be avoided, whenever possible. If applicable, research effort should be directed to other studies which would render compression unnecessary. The following paragraphs are an effort to justify this opinion and why, considering this, another thesis on data compression has been written.

Data compression has been used for the following types of data: Speech, images and abstract data symbols, an example being ASCII characters. Depending on which type of data is being compressed, particular algorithms had been developed by various research workers. Almost all compression schemes have one or more of the following drawbacks.

1) In the case of compression with zero distortion `entropic coding', all the known schemes result in a variable transmission rate. The result is that very large buffers are usually required, in order to transmit the resulting code over a fixed rate channel. If the source statistics are not well known, the code could be very inefficient, much more so than if a straightforward, non-committal assumption on the statistics (uniform distribution) is used for coding. Adaptive methods tend to be wasteful of transmission bandwidth and result in poor performance when the source is of low redundancy.

2) Fixed rate systems always result in

distortion or noise. It ought to be mentioned that research workers tend to be remarkably tolerant of distortion and noise, especially when they have seen their test images or heard their test sentences a few hundred times.

3) The compression (coding and/or decoding) processes often require unrealistic quantities of processing power to implement. This is especially so for image compression.

4) When schemes are presented, which do not exhibit any of the above drawbacks, they almost invariably do not achieve much compression.

5) All compression schemes worsen the effects of channel errors on the source data. In some cases channel errors could be catastrophic, requiring some form of error detection and/or correction.

Having said all this though, there are some instances where these drawbacks, damning as they seem, may be ignored. On some occasions, there is just no choice and compression has to be employed. On other occasions large quantities of distortion are tolerable, this is especially so for speech data and moving pictures. And in some situations coding complexity, variable rate output and large coding delays are of no consequence, for example, when rate reduction is for storage and not transmission. Therefore research in data compression continues. When all is said and done though, for the research worker there is a highly important point to be made and this is that the work is interesting. There is a lot which may be done, and fruitfully, in the three areas of theory, computer simulation and practical construction; and this with finite resources in all senses. It is this last property of the subject which justifies another thesis on data compression.

1.2 Summary of thesis contents

Chapters 2 and 5 contain reviews of the better known methods for respectively, block coding and scalar coding. These are included, as usual in a thesis to give some perspective to the main body of the work reported here. The reviews are written as two separate chapters for consistency with the fact that the thesis is largely devided into two portions. The first part of the thesis is concerned with data compression, using previously coded data, for "block" coding. The second half is concerned with "scalar" (Tree and Trellis) coding.

Chapter 3 contains the bulk of the experimental work for this thesis. It is rather large chapter, which perhaps might have gained from being broken up into several smaller ones. The principle of coding data using other previously coded samples is presented.It is given the name MPPCD (the Matching of Patterns in Previously Coded Data). A discussion of some of the ways that coding may practically be achieved, using the above principle, is undertaken. In the remainder of the chapter, we present diverse applications of the coding scheme and discuss the results obtained. Below is a brief list of the coding applications.

1) Source coding with zero distortion: This is done using a variety of artificially generated data. It is shown that the method has some promise, although in each case particular, other well tailored coding schemes may be employed for the given source. The MPPCD scheme works reasonably well with all types of sources.

2) Source coding with a fidelity criterion: The MPPCD scheme is next used to code artificially generated first order auto-regressive data. The results are compared with those obtained by coding the same data via the Discrete Cosine Transform.

3) Image coding: The MPPCD scheme is used for image coding. Three methods are investigated. The first is a straightforward application of the MPPCD scheme on the one-dimensional signal obtained by the line by line scanning of an image. Secondly, the application of edge weighting to the above scheme, is investigated. Next, the results of an extension of the MPPCD scheme for coding two-dimensional data, is presented.

4) Speech coding: The MPPCD scheme is applied to speech coding. Initially, the speech waveform is directly coded using the MPPCD scheme. No attempt is made to employ some of the features of the speech signal. The results were judged using a signal to noise ratio achievement. The MPPCD scheme is next employed for speech coding, using the framework of Linear Predictive Coding (LPC). A variable rate transmission and a fixed rate transmission scheme are presented. All the speech coding methods are subjectively tested, using independent listeners.

At this point it is worth indicating that both the material on speech and image coding, which generally make up the contents of an introduction, (for example the physics of speech generation, sight and hearing) are presented when the MPPCD scheme is applied to image and speech coding.

Chapter 4 contains some theoretical results on the asymptotic performance of the MPPCD scheme. It is first shown, with the aid of the Shannon-McMillan-Brieman asymptotic equipartition theorem, that in the limit as some quantity, to be defined later, tends to infinity, the MPPCD scheme yeilds a coding rate which tends to the Shannon entropy for the source, plus some ϵ. This factor ϵ, is equivalent to the overhead information required when doing adaptive coding and may be made very small. Next we consider the theoretical properties of the MPPCD scheme when used for coding with a fidelity criterion. In that section, although we are unable to obtain a formal proof that the coding rate tends towards the rate-distortion function for a source, under some considerations, several interesting theoretical properties of the MPPCD scheme and in fact ergodic sources in general are discussed.

In chapter 6, some results on tree and trellis coding for speech and image signals, are presented. In that chaper, we concentrate on methods of 'colouring' trees or trellisses given some source statistics. Some ideas are presented for the adaptive tree or trellis coding of speech and image signals. These are based on deriving source statistics from previously coded data.

Chapter 7 is the concluding chapter of the thesis. As usual this chapter begins with a brief description of the thesis contents. A discussion of the results obtained in the use of the various methods presented in this thesis, is undertaken. Also as usual, a section entitled suggestions for further research is included. In this section, the flashes of inspiration, which could not be followed up for various reasons, are detailed.

CHAPTER 2 A REVIEW AND DISCUSSION OF BLOCK CODING

2.1 Introduction

In this chapter, a survey is presented of a class of schemes for achieving data compression, refered to as 'block coding'.

The aim of data compression is to find a transformation so that the following actions are effected. A sequence of source symbols are transformed into another sequence of symbols. The rate at which this resulting sequence may be transmitted should be as small as possible. In addition it is required that the sequence obtained after transformation may be used to generate an approximation sequence for the original data. This approximation sequence should be close to the original to within some prespecified error. When the transformation described above processes a sequence or block of data at a time, this process is refered to as block coding. The alternative to block coding is scalar coding, where the input symbols are taken, effectively, one at a time.

Two main block coding schemes have been reported in the literature. These are transform coding and recently vector quantisation. In this chapter, a review of the current transform coding techniques used in both image and speech coding is presented. First the question of which transformation to use is discussed. A very simple Fourier transform coding scheme is then described. Following this, the shortcomings of this method is discussed. We then present some of the improvements which have been reported in the literature.

An interpretation of the functioning of transform coding serves to introduce some related coding schemes. These are;

Sub-band coding, Linear predictive coding (LPC) and Vector quantisation. The performance of these methods when applied to speech have been the subject of several papers. As yet however, they have not yet been extensively applied to image coding. The bulk of this chapter contains general descriptions of the ideas behind the well known block coding schemes, without extensive reference to publications. The chapter however, ends with a detailed bibliography, where the references associated with the different methods are given.

2.2 Transform coding

2.2.1 Choice of transform

Interpreted most generally, transform coding is the following process. A transformation A_{mn}, operates on a sequence of source symbols

$$\underline{x}_n = \{\zeta_1, \zeta_2, \ldots, \zeta_n\}$$

such that the result of the transformation is the sequence

$$\underline{y}_m = \{\omega_1, \omega_2, \ldots, \omega_m\}$$

$\underline{y}_m$ is quantised and approximated by $\tilde{\underline{y}}_m$. The $\tilde{\underline{y}}_m$ values have channel symbols associated with them, these symbols are transmitted and it is presumed that there are no channel errors. At the receiver, the transformation B_{nm} is used to generate an approximation $\tilde{\underline{x}}_n$ of $\underline{x}_n$ from the $\tilde{\underline{y}}_m$ sequence.

$$\tilde{\underline{x}}_n = B_{nm} . \tilde{\underline{y}}_m$$

The following properties should hold for the transformation pair A_{mn} and B_{nm}:

a) $d_1(\underline{x}_n,\tilde{\underline{x}}_n)$ should monotonically increase with $d_2(\underline{y}_m,\tilde{\underline{y}}_m)$, where $d_1(.,.)$ and $d_2(,.,)$ are the distortion measures used in the original and transform domains respectively.

b) If $d_1(\underline{y}_m,\tilde{\underline{y}}_m)=0$ then $d_2(\underline{x}_n,\tilde{\underline{x}}_n)=0$

Normally A_{mn} and B_{nm} are linear transformation and may be represented by the nxn matrices A_{nn} and B_{nn} .

A good transformation kernel A_{mn} should enable $\tilde{\underline{y}}_m$ to be represented by a small number of channel symbols and still allow $\tilde{\underline{y}}_m$ to be close to $\underline{y}_m$.

It is known that the best encoding scheme for $\underline{y}_m$ achieves distortion and rate values specified by the source's rate-distortion function [Shannon-(1959)]. In general, it is difficult to find a good sequence of channel symbols to assign to each possible value of $\tilde{\underline{y}}_m$. Were the members of $\underline{y}_m$ independent, this sequence may be efficiently coded by coding each symbol separately. This is because there exist several efficient scalar coding schemes (Lloyd-Max quantisation and Huffman coding [Max-(1960) and Lloyd-(1982)]). We know that the minimum coding rate possible is defined by the infinite block size rate-distortion function for a source. We also know that for a source with independent outcomes, the infinite block size rate-distortion function is equal to the single letter rate-distortion function for this source [Gallagher-(1968), Berger(1971)]. Independence-inducing transforms therefore allow efficient coding since each of the resulting independent outcomes may be coded at close to the single letter rate-distortion function, by efficient scalar coding methods.

Our discussion will now be restricted to linear transformations. In choosing a linear transformation, we try to find one which will give a transformed sequence as near independent as possible. The best that can be done using linear transformations, is to require that the resulting sequence be uncorrelated. The transformation which achieves this is the Karhunen-Loeve transformation (KLT). (This is sometimes refered to as the Hotelling transform for sampled data systems [Ahmed and Rao-(1975)])

The KLT is the transformation K that satisfies the equation 2.1. D is a purely diagonal matrix with positive entries.

$$\underline{Y} = K\underline{X} \qquad 2.1$$

$$\text{where} \quad E(\underline{Y}.\underline{Y}^T) = D \qquad 2.2$$

Referring to $E(\underline{X}.\underline{X}^T)$ as R, it is easily shown that a transformation K which satisfies the condition 2.2, is the matrix of eigenvectors of R, where D is the diagonal matrix of eigenvalues of R. The optimum linear transformation, the KLT requires the knowledge of the covariance function for the source and its evaluation involves the computation of the eigenvectors of a matrix.

In practice the KLT is not often used for transform coding. The main drawbacks are, the unreliability of the covariance function estimation and the lack of fast algorithms for evaluating the eigenvectors of the transform and for calculating the transform of a data sequence. The alternative, but non-optimum, transforms have fast implementation algorithms which make them convenient to use. In fact it has been shown practically that there is little to be gained from using the KLT compared with the Discrete Cosine Transform for some sources with high inter-symbol correlation [Zelinski and Noll-(1977) and Wintz-(1972)].

2.2.2 A Fourier transform coding example

Figure 2.1 shows a block diagram of the transform coder to be described. Non-overlapping blocks of symbols from a source are fed to a Fourier transformer. Let us refer to a block of input data of length L as $\underline{X}$. The Fourier transformer generates a sequence of L complex numbers, (L/2)+1 (L is even say) of which are to be encoded and transmitted. Of these, two are purely real. Refer to the sequence of complex numbers to be coded as $\underline{Y}$. The coding scheme may be one of the following.

1) Choose beforehand which of the $\underline{Y}$ are to be transmitted. The choice is fixed and is made by considering the long term spectral character of the source, in conjunction with a weighting function to colour the noise resulting from quantisation. For each of the frequency components to be coded, a fixed number of bits is assigned. Which frequency components to code and the number of bits to assign is decided in this way. For each frequency component "ω" we assign $I\lfloor \log_2 |\frac{\lambda(\omega)c(\omega)}{D^{*}}| \rfloor$ bits. Here $I\lfloor(u)\rfloor$ is the smallest integer greater than u and zero if u<0. $\lambda(\omega)$ is the source's power spectral density and $c(\omega)$ is some weighting function used to colour the coding noise. D^{*} is the distortion limit such that one

expects average distortion less than D^*. When $c(\omega)$ is unity the assignment of bits in this way results in a noise spectrum which is approximately flat [Huang and Shulthiess-(1963)]. For each frequency component, non-linear quantisation may be used. Some workers have employed A-law or μ-law quantisation [Frangoulis-(1978)] and others have employed Lloyd-Max quantizers.

2) The frequency components to be coded are not chosen beforehand. For each block a different but fixed number N of frequency components are chosen for coding and transmission. The N chosen are dependent upon the instantaneous magnitude of the frequency components. This is referred to as adaptive transform coding. The advantage of this is that no apriori source statistics are required. A shortcoming however, is the neccessity to send extra symbols to inform the receiver of which frequency components are coded and the number of bits assigned to each of the frequency components coded. [Wintz (1972)]

At the receiver, the received spectral signal is simply inverse transformed to obtain an approximation to the coded source symbols. Alternatively some sort of interpolation may be done to approximate the frequency components not transmitted, before inverse Fourier transforming

A myriad of schemes have been reported in the literature, which are variations and improvements to the above method. [Tasto and Wintz-(1972) and Wintz-(1972)]. For image coding, the variations on the above method are not outstanding, with the exception of using overlappping blocks. For speech coding however several methods have been proposed which are significantly different from the above. The foregoing is a discussion of some of the inadequacies of transform coding as described in section 2.2.2.

2.2.3 Improvements to the basic scheme

For the scheme just described and the improved methods to be described, transforms other than the Fourier transform may be used. The improvements gained when using alternative transforms are in these directions.

a) A further decrease in correlation between the random variables obtained after doing the transform. This enables more efficient coding of the data, since the greater the decorrelation the transformation achieves, the larger the compaction of the data in the transform domain. The excellent review paper by Wintz-(1972) gives a comparison of the results of the performance of discrete Fourier transform (DFT), discrete cosine transform (DCT) and Karhunen-Loeve transform (KLT) coding schemes.

b) A decrease in the computational time for the implementation of the transformation. To this end, the results of the use of the Walsh-Hadamard, Haar and other easily implementable transforms have been reported in the literature. Frangoulis-(1978), Ahmadi-(1980), Zelinski and Noll-(1977 and 1979) give a figure comparing the performance of the KLT, the DCT, the DFT, the discrete sine transform (DST) and the Walsh-Hadamard transforms.

In order to improve the transform coding results, we may use overlapping blocks. This, in conjuction with the windowing of

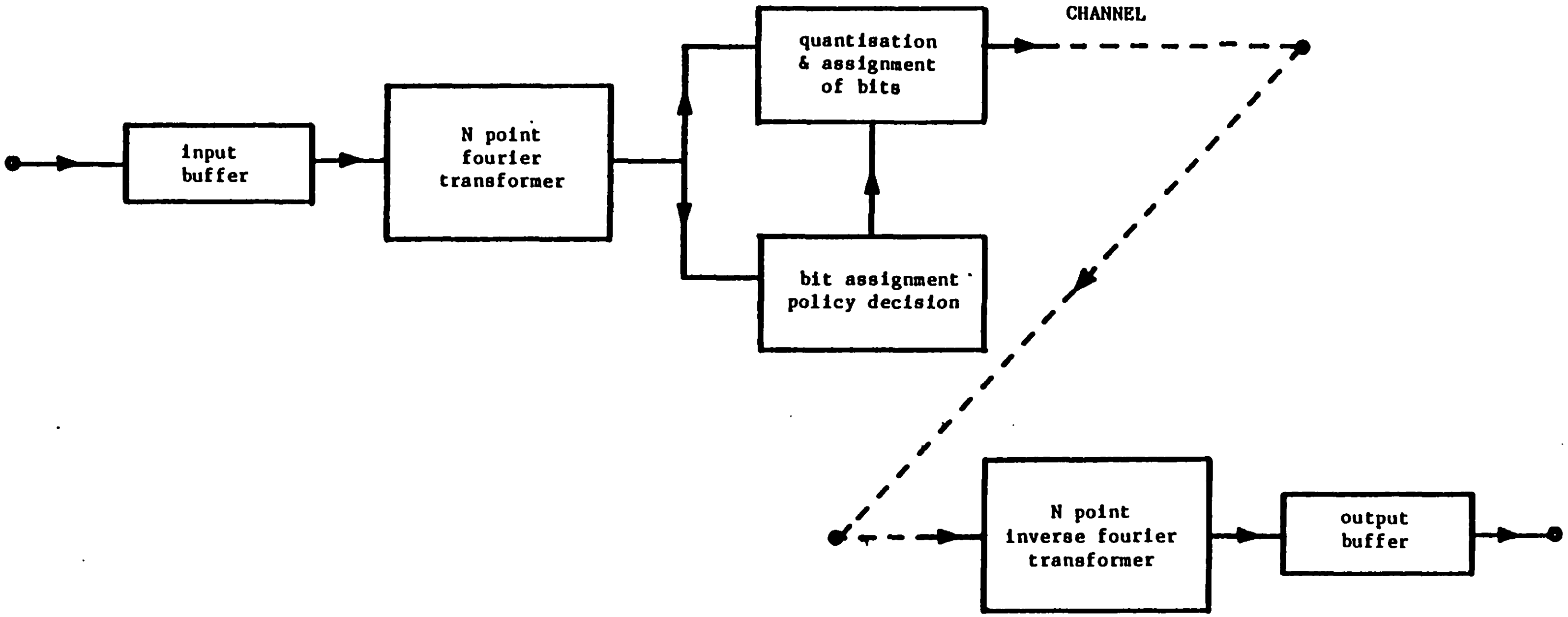

input buffer
N point fourier transformer
quantisation & assignment of bits
bit assignment policy decision
CHANNEL
N point inverse fourier transformer
output buffer

Figure 2.1 A basic Fourier transform coder.

blocks to be transformed enables the performance of a more reliable short term spectral analysis of the block to be coded. We may also improve the efficiency of the methods used to code frequency components. These two approaches are not mutually exclusive. To explain and justify the use of windowing we have to redefine what it is that we want to achieve.

The transformation procedure, using a transform defined by the kernel K, is shown by equation 2.4.

$$y_m(\omega) = \sum_{n=-\infty}^{\infty} h(n-m)x(n)K(\omega,n) \qquad 2.4$$

h(n-m) is some window function centered at m. x(n) is the doubly infinite sequence which is generated by the source and $y_m(\omega)$ is the frequency component at frequency ω, observed at time instant m.

For the basic transform coding scheme, h(n-m) is a rectangular window centered at time instant m. If the transform size is M, the window is of length M. The window centres, the various values of m, are integer multiples of M, so that non-overlapping blocks are transformed. It may be seen from equation 2.4 that for every value of $\omega = \omega_0$ say, $y_m(\omega_0)$ is a sequence with m, whose members are estimates of the amplitude value for that particular frequency component ω_0. The periodicity of the sampling of that particular frequency ω_0, for the case of the simple transform coding scheme described is 1/M. Obviously the best scheme as far as resolving the various frequency components is that which would take the whole time infinite sequence and transform this. The values of the frequency components would only need to be sampled and transmitted once. Since in reality this cannot be done, we window the data sequence with some windowing function of duration T, where T is finite. The problem of the best window shape to use was solved by Kaiser for the Fourier frequency domain. Consider the equation

$$y_m(\omega_0) = \sum_{n=-\infty}^{\infty} h(n-m)x(n)K(\omega_0,n) \qquad 2.5$$

For the single frequency ω_0, with value $y_m(\omega_0)$ at a time instant "m", it may be seen that $y_m(\omega_0)$ is the convolution of the time sequence $\{x(n)K(\omega_0,n)\}$ by the function h(.). It is required that for each i, $y_i(\omega_0)$ has as few contributions from other frequencies as possible. The best window for this, given any block length is the Kaiser window. [Rabiner and Gold-(1975) sections 3.8-3.16] The necessary sampling rate (frequency at which the magnitude of frequency component ω_0 should be sampled) is determined by the bandwidth of the filter whose impulse response is h(.). The greater the duration of the window, the sharper the cutoff of the filter whose impulse response is h(.). This is the rationale for the use of overlapping blocks and a window.

Now we shall consider the improvements which may be made in quantising the members of the transformed sequence Y. In the simple transform coding scheme described in section 2.2.2, one non-adaptive and one adaptive method were described. It should be recalled that the adaptive scheme had the advantage of following the changing spectral patterns better since for each block, the frequencies coded are chosen according to the local spectral characteristics. We shall now describe in detail schemes reported by Zelinski and Noll-(1977) and Tribolet and Crochiere-(1979). Figure 2.2 shows a block diagram for these two schemes.

Zelinski and Noll chose a set of non-uniformly spaced frequencies $f_1, f_2, \ldots$ The average power at frequencies around each chosen frequency value f_i, is evaluated. These values of average power are quantised and transmitted. These values for average power are also used to evaluate an estimate of the power spectral density over the whole of the frequency range at both the receiver and the transmitter. This estimate of the power spectral density is then used to design a bit allocation scheme. This bit allocation scheme is then used for quantising and coding the whole of the frequency band. Tribolet and Crochiere reported a scheme in which they do the spectral density function estimation by LPC analysis. The LPC analysis is done using an auto-correlation function derived via an inverse DCT of the square magnitude of the cosine tranformed sequence. For speech coding applications in particular, Tribolet and Crochiere weight the DCT spectral signal with a comblike frequency response which is supposed to represent the effects of the pseudo-periodic characteristic of voiced speech. They call this a vocoder driven adaptive transform coder. It was reported by Zelinski and Noll-(1979) and Tribolet and Crochiere that quantisation of the frequency components, using Lloyd-Max quantisers made no significant improvement over doing linear quantisation.

Other improvements to the transform coder are in the direction of finding frequency domain weighting functions which are supposed to improve the perceptive quality of the speech or image signal.

The 'short-term' Fourier transform is an alternative to the windowed and overlapped Fourier or cosine transforms. This involves the Fourier transformation of a reflectively doubled version of each

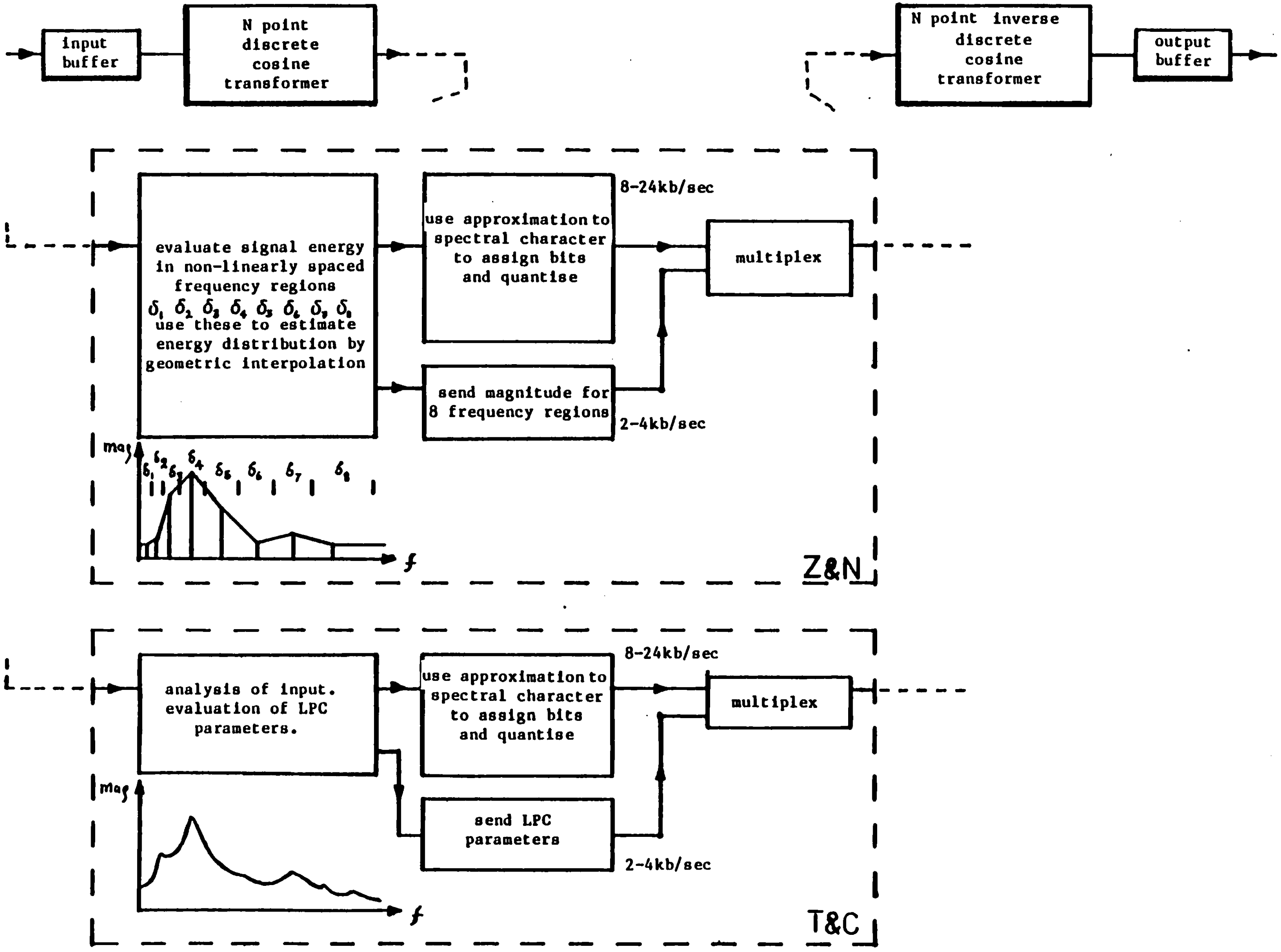

input buffer
N point discrete cosine transformer
N point inverse discrete cosine transformer
output buffer
evaluate signal energy in non-linearly spaced frequency regions
δ1 δ2 δ3 δ4 δ5 δ6 δ7 δ8
use these to estimate energy distribution by geometric interpolation
use approximation to spectral character to assign bits and quantise
8-24kb/sec
multiplex
send magnitude for 8 frequency regions
2-4kb/sec
mag
δ1 δ2 δ3 δ4 δ5 δ6 δ7 δ8
f
Z&N
analysis of input. evaluation of LPC parameters.
use approximation to spectral character to assign bits and quantise
8-24kb/sec
multiplex
send LPC parameters
2-4kb/sec
mag
f
T&C

Figure 2.2 The adaptive discrete cosine transform coders of Zelinski and Noll and Tribolet and Crochiere.

input data block. The Fourier components are consequently purely real, and are coded as the cosine transform components were described as being coded in the previous section. It has been reported that this procedure results in less edge effects, which effect is especially important for image coding.

2.3 Methods that follow from transform coding

The underlying effect of using a transformation will be discussed by considering the Fourier transform. Consider the sequence of source symbols $\{x_1,x_2,\ldots,x_N\}$. These are transformed so that we have $\{y_1,y_2,\ldots,y_N\}$. For the Fourier transform kernel, each of y_m is the output of a low-pass filter preceded by a modulator, which does frequency shifting. The following shows this. Consider the equation describing the transform

$$y_m(\omega) = \sum_{n=-\infty}^{\infty} h(n-m)x(n)e^{-jn\omega} \qquad 2.6$$

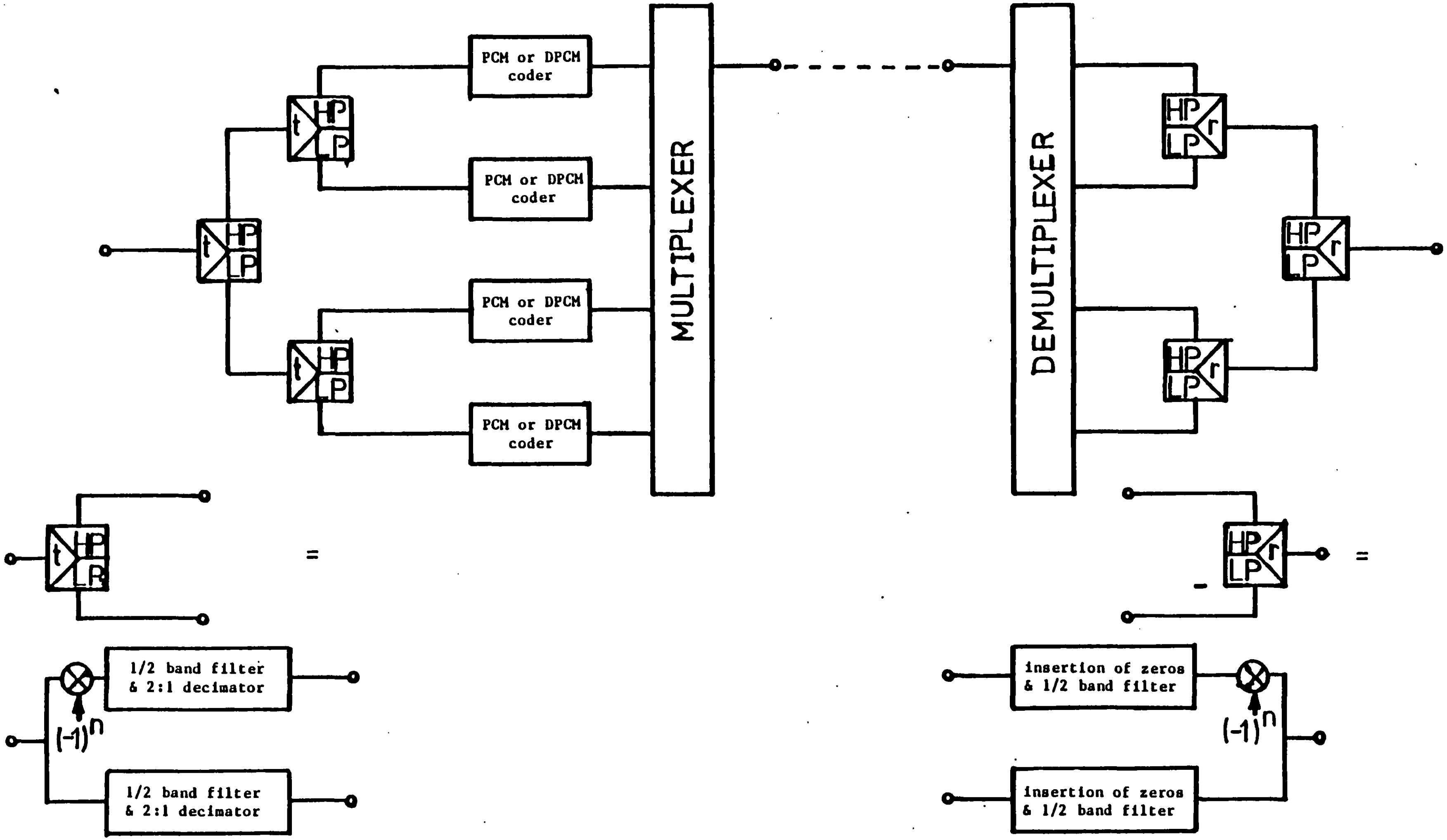
Then for each frequency ω , one obtains the sequence y_m say, if exp(-jωn) is replaced by z(n) where

$$y_m(\omega) = \sum_{n=-\infty}^{\infty} h(n-m)x(n)z(n) \qquad 2.7$$

The function x(n)z(n) is therefore convolved with the function h(.), to obtain the value y_m at instant m. Now it should be noticed that the multiplication of x(n) with z(n) is a modulation which effects a frequency shift by ω rads/sec. The h(.) is a low pass filter impulse response. (Note that with no windowing, that is with a rectangular window, the low pass filter is just an averaging process.) Thus each spectral component is the result of a modulation followed by a low-pass filter. Transform coding restricts one to using Finite impulse response (FIR) filters for the low-pass filter operation.

2.3.1 Sub-band coding

In sub-band coding, very long length filters may be used to effect the band-pass filtering operation. The use of IIR filters effectively means the use of longer analysis windows and the possibility of sharper frequency discrimination. Usually a small number of frequency bands is used, for example four or eight. A class of filters particularly suited to the job of band-pass analysis are the quadrature mirror filters. The basic building block is a "half-band coder". This divides an incoming sequence into a lower frequency band of half the original bandwidth and a higher frequency band of half the original bandwidth. The high frequency band is then modulated to the base-band. Figure 2.3 shows a tree of filters with half-band coders at each node. The individual signal from each of the channels is coded using any of the well known scalar quantisation schemes. Like the transform coder, a different number of bits are allocated to each of the bands. This is done according to the energy of the signal component in this band. The assignment scheme is the same as in transform coding. Alternative subband coders are ones which use frequency bands of different bandwidths.



t
HP
LP
PCM or DPCM coder
PCM or DPCM coder
PCM or DPCM coder
PCM or DPCM coder
MULTIPLEXER
DEMULTIPLEXER
HP
LP
r
=
1/2 band filter & 2:1 decimator
1/2 band filter & 2:1 decimator
$(-1)^n$
insertion of zeros & 1/2 band filter
insertion of zeros & 1/2 band filter
$(-1)^n$

Figure 2.3 A subband coder showing the tree structure obtained by using successive half-band coders.

2.3.2 Linear predictive coding (LPC)

The transform coding approach to frequency analysis uses FIR filters to do spectrum analysis. The short term spectrum is then represented by the transformed sequence. The members of this sequence are quantised and transmitted. An alternative is to find parameters which describe a smooth function in the frequency domain. This smooth function is an approximation to the short term spectrum of the data. This is what LPC attempts to do. Parameters to indicate a smooth function which approximates the power spectral density are obtained, quantised and have channel symbols assigned to these. The channel symbols are then transmitted. In addition, the residual or error signal associated with this approximation is coded and transmitted. In LPC, the source symbols are approximated as the output of a time varying infinite impulse response (IIR) filter. The coefficients of this filter are parameters which define the spectrum of the input signal. Suppose

$$x(1), x(2), \ldots, x(n)$$

is a block of the input signal. This is modelled to be the output of the filter H(z). Let

$$\epsilon(1), \epsilon(2), \ldots, \epsilon(n)$$

be the input sequence to this filter. The filter coefficients are calculated so that for each block the residual signal $\{ \epsilon(1), \epsilon(2), \ldots, \epsilon(n) \}$ has the smallest variance. The following is a description of the general formulation.

A source may modelled in the most general case as an "auto-regressive moving average" (ARMA) process. Let the source sequence be x(n), then the model is described thus:

$$x(n) = \sum_{i=1}^{k} a_i x(n-i) + \sum_{i=1}^{L} b_i \epsilon(n-i) + \epsilon(n) \qquad 2.8$$

The source is "identified" by the parameters $\{a_1, a_2, \ldots, a_k\}$ and $\{b_1, b_2, \ldots, b_L\}$ such that the variance of the error sequence $\{\epsilon(1), \epsilon(2), \ldots, \epsilon(n)\}$ is minimised. If the a_i are all zero the source is said to be a "moving-average" process. Alternatively if the b_i are zero, then

$$x(n) = \sum_{i=1}^{k} a_i x(n-i) + \epsilon(n)$$

$$(1 - \sum_{i=1}^{k} a_i z^{-i}) x(n) = \epsilon(n) \qquad 2.9$$

where z^{-i} is a time shift of i positions. Then

$$\begin{aligned} x(n) &= \frac{\epsilon(n)}{(1 - \sum_{i=1}^{k} a_i z^{-i})} \\ &= \frac{1}{A(z)} \epsilon(n) \\ &= H(z)\epsilon(n) \end{aligned} \qquad 2.10$$

Appendix 1 shows the procedure for evaluating the filter coefficients for the case when the b_i are zero. The model is termed an "auto-regressive" or "all-pole" model in this case. When an estimate is made of the auto-correlation or covariance function for the source beforehand, then an algorithm named after Levinson and Durbin may be used. An alternative is the Burg maximum entropy method which does not require a prior estimate of the auto-correlation or covariance functions. Both these methods of evaluating the all-pole filter coefficients are recursive. The filter coefficients for an N-th order autoregressive source are evaluated by first working out the best coefficient for the case when the source is modelled as a first order system. Next we use this to evaluate some of the coefficients for the case where we model the system as a second order auto-regressive source. We then use the coefficients obtained by doing this to work out the coefficients for the case where we use a third order model and so on. To evaluate the m coefficients for the case where we model a source as being of order m, we first evaluate the m-th coefficient. This in conjunction with the m-1 coefficients obtained by using a model of order m-1 is used to calculate the m coefficients of the m-th order model. In this stage by stage process, the m-th filter coefficient evaluated at the m-th stage, is referred to as the m-th reflection coefficient or partial correlation coefficient. To ensure the stability of the resulting all-pole filter all the reflection coefficients have to be of magnitude less than unity.

It has been observed that the frequency response of the all-pole filter model is very sensitive to the variation of its filter coefficients. For coding, it has been found that the quantisation and transmission of the reflection coefficients or some function of the reflection coefficients has resulted in less distortion of the spectrum. Non-linear quantisation of the reflection coefficients is generally done. This is because it has been observed that the filter response is sensitive to reflection coefficient error when these are near unity. Quantisation is done so that there is less error when a reflection coefficient is close to unity. Non-linear quantisation is generally done by linearly quantising some non-linear one-to-one function of the reflection coefficients. Two such functions which have been found to work well are given below. [Gray and Markel-(1976)]

$$g_i = \log \frac{1 + k_i}{1 - k_i}$$

$$h_i = \sin^{-1}(k_i - 1)$$

k_i are the reflection coefficients and
g_i are called the log-area-ratios

2.11

To recapitulate, LPC involves the use of an all all-pole filter to model a source. (ARMA models have seen little use because of the difficulty of evaluating their parameters) The filter gives the parameters of a smooth frequency function which approximates the spectrum of the source. The error associated with this spectral representation has an associated time domain sequence which is referredto as the residual signal. The residual signal, in addition to parameters describing the model for the source are quantised and transmitted. These are used by the receiver to generate an approximation to the input sequence.

The LPC Vocoder

A vocoder is a coder which does low bit rate speech coding by extracting, coding and transmitting parameters which describe the speech generation process. The linear prediction vocoder is one of several types of vocoder. The LPC vocoder does linear prediction analysis on a block of speech data, generally of duration 10 to 30 msecs. The Log-Area-Ratios as shown in equation 2.11 or the arcsines of the reflection coefficients are evaluated and quantised.

There are several ways of solving the problem of evaluating the filter parameters. To solve the problem by the use of the Levinson-Durbin algorithm, requires the evaluation of the auto-correlation or covariance function for the source. The approaches for doing this are:

1. Windowing the data and directly finding the auto-correlation function by presuming that the signal is zero outside the span of the window.

2. Performing a FFT of the input sequence, evaluating the square

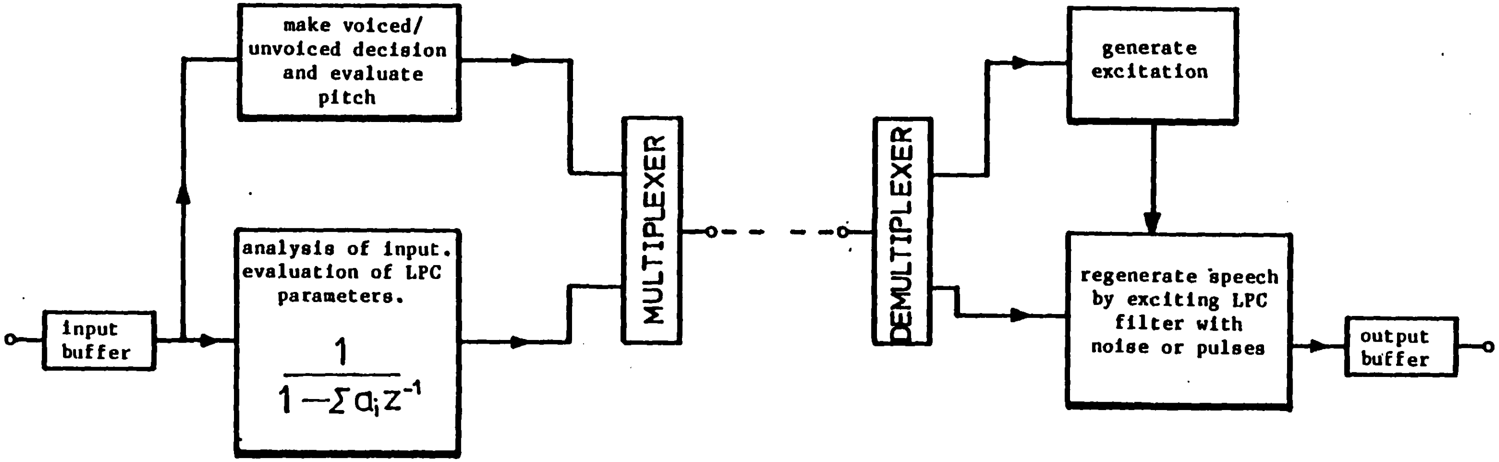


Figure 2.4 An LPC vocoder.

magnitude of the transformed signal for each frequency component, and computing the inverse transform of this. One then obtains an approximation to the auto-correlation function. This implies that the signal is periodic. The covariance function obtained is that of a periodic sequence, where outside the duration of the signal available, this sequence is repeated.

3. The use of a pitch synchronous system has been reported by Barnwell-(1980). Here use is made of the pseudo-periodic character of speech when an utterance is made. A pitch period's worth of signal is used to evaluate the covariance function by the FFT method described in 2. This is a reasonable course of action since the speech signal, on these occasions is semi-periodic.

4. The use of the Burg maximum entropy method. This avoids a direct evaluation of the auto-correlation or covariance functions. It has been reported to give better spectral estimation than any of the other methods.

Figure 2.4 shows a basic LPC vocoder. This requires a decision to be made concerning whether the block being considered is the result of a voiced or unvoiced utterance. If it is ascertained that the block is that of a voiced sound the pitch associated that utterance has to be evaluated. At the receiver, the residual or excitation for feeding the filter representing the all-pole process is derived as follows. When the block is unvoiced a pseudo-random sequence of an appropriate variance is used to excite the filter. When the block is voiced, a sequence of pulses of the appropriate frequency and variance is used to feed the all-pole filter used to model the source. This type of vocoder allows data transmission at between 2kbit/sec and 12kbit/sec. Several LPC vocoders with more efficient coding of the Log-area parameters, the pitch and gain have been reported in the literature. Using these method allow the transmission of digital speech at rates as low as 800 bits/sec.

The residual waveform may be encoded in a manner other than by modelling this as a pulse train or pseudo-random noise. Vocoders excited by alternative methods are referred to as residual excited vocoders (RELP vocoders) or voice excited vocoders (VELP vocoders)

RELP vocoders have in addition to the filter parameters (or reflection coefficients) a low pass filtered version of the residual signal coded and transmitted.

VELP vocoders are similar except that a low pass filtered version of the input data sequence is sent to the receiver.

In both these cases the signal transmitted in addition to the filter parameters is low pass filtered to between 500Hz and 1kHz. At the receiver this signal is processed to generate a full band signal which is then used to excite the modelling filter.

2.4 Vector quantisation

Vector quantisation is another name for block quantisation. This name is however by recent tradition used solely for block quantisation schemes which do not depend on the prior application of some independence inducing transformation.

Vector quantisation may be described as follows. Presume that a source generates a sequence $\Omega = \{x_{-\infty},\ldots,x_{-1},x_0,x_1,\ldots,x_\infty\}$.

Suppose that non-overlapping blocks are considered and the block $\{x_{n+1},\ldots,x_{n+L}\}$ is refered to as $\underline{X}_n^L$. To any values that the block $\underline{X}_n^L$ takes, we want to assign an approximation, $\tilde{\underline{X}}_n^L$. We constrain the alphabet of the set of possible approximation vectors to have only 1024 members say. Thus if $\underline{X}_n^L$ has say, $L=8$ members each of which may take one of 256 values, we have a total of 256^8 possible values for $\underline{X}_n^L$. Using vector quantisation, the compression ratio achieved for this example is 1:64 .

A major task in vector quantisation is choosing the members of the approximation set. These must be chosen such that the average distortion obtained by approximating the possible outcomes by members of the approximation set, is minimised. The minimisation is over all possible approximation sets of a given size. There is no known optimisation scheme, (except of course a complete search) which will solve the general problem for any source. A solution is generally evaluated by clustering. Some axioms proposed by Lloyd which help us to do reasonable clustering are given below. We shall then descibe a practical vector quantisation scheme.

Axiom 1.

Given a set of partitions, $S_1, S_2, \ldots.$ say
an optimal quantizer should have for each cluster i, a centroid m_i, so that the following is true. The distortion associated with representing the members of cluster i by the centroid m_i is minimised. This gives us a criterion for choosing the centroids of clusters given a certain partitioning scheme.

Axiom 2.

An optimal quantiser should have for a given set of centroids

$M_n = \{m_1, \ldots, m_n\}$, a partitioning scheme S such that the distortions $\{d_1, \ldots, d_n\}$ associated with representing the members of a cluster (defined by the partitioning scheme S) by their centroids $M_n$is minimised. This gives us a criterion for choosing a partitioning scheme, granted that a set of centroids has been defined already. The set of partitions may be found by evaluating the boundary between every pair of centroids m_i and m_j. The boundary is defined as the locus such that the following holds. All points to one side of the boundary will be approximated by one centroid m_i say, and all points to the other side of the boundary will be represented by the other centroid m_j. The boundary chosen is the one which gives the least average distortion. We shall refer to this boundary as B_{ij}. Granted a set of centroids M_n we define for each particular centroid, the quantity B'_i defined as follows.

$$B'_i = \bigcap_{\forall j \neq i} B_{ij} \qquad 2.12$$

This is a VORONOI cell associated with the particular centroid m_i. The set of all these Voronoi cells, one each for the centroids, $m_1, m_2, \ldots$ define the best partitioning scheme S given a set M_n of centroids.

These two axioms together define an optimum partitioning scheme.

The clustering scheme most employed for coding purposes is detailed briefly below. It was first applied to scalar quantities by Lloyd-(1982) and to vector quantities by Forgy-(1965). A set of cluster centroids M_{n1} is arbitrarily chosen, then a set of optimum boundaries $\{B'_1, B'_2, \ldots, B'_n\}$ defining a partitioning scheme S_{n1} is chosen in accordance with axiom-2 above. Denote the resulting average distortion by the quantity $D(S_{n1})$. Granted these partitions, we find the best centroids in accordance with the axiom-1 above. Refer to the resulting set of centroids as M_{n2}. Using M_{n2} find the best set of partitions, say S_{n2}. Continuation of this procedure will lead to at least a local minimum as far as distortion associated with partitioning is concerned.

An image coder based upon vector quantisation

A vector quantisation scheme based almost exactly on the method of clustering just described, has recently been reported by Gersho and Ramamurthi-(1982).

A large training sequence is used to define a set of cluster centres in the following manner. This approach was first suggested by Linde, Buzo and Gray-(1980). Suppose it is decided that there should be K groups or clusters. Each member is an N point sequence. A set of K cluster centres is arbitrarily chosen. The training sequence is then used to define new centroids and partitions as described in the above. The whole training sequence is used at every stage to define new centroids and then partition schemes. When it is observed that this recursive scheme has converged, the centroids form the set M_n . This set is referred to as either a library, codebook or a set of templates.

In the method of Gersho and Ramamurthi, each training sequence vector is first classified as containing an edge or is a "shade", that is a region with no edge. Two different codebooks are designed for the two different data types.

The encoding procedure is simply that of finding the nearest member of the codebook to a block under consideration. The channel symbols or coordinate associated with this particular member is transmitted. At the receiver the coded block is approximated by this member of the codebook. Various block sizes were tried by Gersho and Ramamurthi and bit rates of 0.5 to 1.5 bits/pixel were achieved with reasonable quality.

A speech coder based upon vector quantisation.

The first use of a vector quantiser for speech coding was reported by Smith in an abstract in 1963. The idea is remarkably like that followed today. A vector quantisation scheme for speech coding scheme was investigated by Ahmadi-(1980), this used a rather ad-hoc quantisation scheme. Intelligible speech was reported to have been obtained for very low data rates (around 1kbit/sec) using this method. The vector quantisation scheme described here was reported by Linde, Buzo, Gray Gray and Robodello-(1980). The following is a detailed description of how the codebook is designed. Of particular note is the way in which the codebook is initialised.

Set a stage counter M to 1 initially. Suppose it is decided that the codebook will contain K members. Set an initial centroid A_{11} of dimension L (the blocks considered are each of dimension L). Choose A_{11} to be the mean block of length L, by going through the whole training sequence. From A_{11} derive two centroids defined as follows,

$$A_1^1 = A_{11} + \epsilon, \qquad A_1^2 = A_{11} - \epsilon \qquad 2.13$$

ϵ is an arbitrary perturbation vector of dimension L. These two vector are used as initial vectors for clustering. From these two, a partition scheme is derived. This partition scheme is then used to define a optimum pair of centroids for this partition scheme. M is replaced by 2M and the optimum pair of centroids are refered to as A_{21} and A_{22} respectively.

From these two centroids we obtain 4 new centroids by perturbation by the vector ϵ of dimension L. These four are used to define a new partitioning scheme. This partitioning scheme, is then used to define a new set of 4 centroids by going through the training scheme. Refer to this new set of centroids as A_{4i}. From these four define a new set of partitions then centroids and so on. This is done until we obtain K centroids.

This set of centroids form the initial members of the codebook. We are now ready to apply the clustering algorithm as desribed in the previous section.

It has to be pointed out that for speech coding, vector quantisation has not been applied directly to the speech samples but upon the Log-Area-Ratios. Some frequency domain distortion function is used.

Linde, Buzo and Gray reported on the use of 10 filter coefficients for the all-pole filter derived after LPC analysis. They employed a codebook of 256 members, to approximate the LPC filter parameters. This resulted in a coding scheme with a reduced bit rate, from 6kbits/sec to 1.4kbits/sec. They claim the reduction in quality was small.

2.5 A bibliography of block quantisation schemes

Reviews

Flanagan et al.-(1979) give a thorough review of the accepted speech coding schemes. Holmes-(1982) gives a briefer but nevertheless very good review of current speech coding techniques. Jain-(1981) and Netravali and Limb-(1980) give very thorough reviews of image coding systems, which should be enough to give a strong grounding in the field of picture coding. Wintz-(1972) gives a more detailed description of transform coding schemes. Despite the age of this paper, few fundamentally new schemes for transform coding have been reported since this was written. Habibi-(1977) is also helpful in the area of transform coding. A very thorough book dealing with the whole field of speech and image compression is that by Jayant and Noll-(1984). It covers most of the topics described in this and chapter 5.

The following is a list of references dealing in more detail with particular block coding schemes.

Transform coding.

Huang and Shulthies-(1963) first reported the use of transform coding techniques for image compression. The papers by Andrews, Kane and Pratt-(1969), Anderson and Huang-(1971), Landau and Slepian-(1971), Pratt,Chen and Welch-(1974), Rao, Narashima and Revuluri-(1975), and by Bisherurwa and Coakly-(1981) describe transform coding schemes using the various orthogonal transforms; DFT, DCT, DST, Haar, Hadamard and Slant transforms. For image coding methods which include some classification, in order to better adapt the coding scheme to the short term statistics of the source, see the papers by Gimlett-(1975) and Wen-Hsuing Chen and Harrison-Smith-(1977). These use an activity index to classify blocks. Tasto and Wintz-(1971) use a more subjective classification scheme, with three classes. These are;

1) Blocks with a lot of detail.

2) Low intensity blocks with low detail and

3) High intensity blocks with low detail. The KLT is used to do the coding, the basis functions differ with each class. Ngan-(1982) paper is gives a comprehensive comparison of the WHT and DCT and uses a human visual characteristic for adaptive bit allocation. In addition classification according to activity is made. A hybrid technique using transform coding of the rows of a picture and differential pulse code modulation on the columns of the resulting after transfomation of the rows.

For speech coding the following papers by the following are worth reading: Campanella and Robinson-(1971), Shum, Elliot and Brown-(1973), Zelinski and Noll-(1977 and 1979) and Tribolet and Crochiere-(1979). The latter paper deals with transform coding and subband coding in a unified manner and makes very interesting reading. A further reference is the thesis by Frangoulis-(1978). This details a very thorough investigation of various methods for doing Walsh-Hadamard transform coding. A good presentation of subjective results is given in this work.

Subband coding

This was introduced by Crochiere in 1976. The concept though is very similar to that behind the channel vocoder. [Shroeder-(1966)]

Other references are the papers by Crochiere-(1977), Esteban and Galland-(1978) and Grauel-(1980).

Vocoders

The background for modelling the speech generation process is given in the definitive book by Fant-(1961). The vocoder works by making use of the speech model in deciding the important features to code. The excellent review paper by Shroeder-(1966) descibes the different types of vocoder. Other references are papers by Higgins-(1954) and Shroeder-(1962) which describe an auto-correlation vocoder. The concept of the voice excited vocoder is covered by the following authors: Shroeder and David-(1960) David, Shroeder, Logan and Prestigiacomo-(1972).

Linear predictive coding

For speech coding the references for this overlap with those for the vocoder. The most definitive work on this is the paper by Atal and Hanauer-(1971). Further work has been reported by Atal, in one of these papers he details the effects of applying linear predictive coding to the residual waveform [Atal-(1982)]. Viswanathan, Makhoul, Shwartz and Huggins-(1982) have reported a scheme for further bit rate reduction in a vocoder which does the following. Only filter coefficients which are observed to be significantly different from the previous filter coefficients are transmitted. Atal and Shroeder-(1978) have reported the effects of doing pole-zero analysis (using an ARMA model) of the speech waveform. Yegananarayana(1981) also describes a method of finding the ARMA model parameters for speech segments.

In image coding, relatively little work has been reported, concerning the use of a "space" varying filter model for the image data generation process. The papers by Jain and Ranganath-(1980) and Jain-(1981) report on the use of an auto-regressive model. The thesis by D Mitrakos-(1983) detail some interesting coding techniques which use a space varying auto-regressive model for modelling an image. Maragos, Schafer and Mersereau-(1984) recently published a thorough investigation of the use of an adaptive two-dimensional predictor for image coding which is very noteworthy.

Vector quantisation

The first mention of this in a coding context was in the abstract by Smith-(1963). Other work has been done at Stanford, and has been reported in the papers by Linde, Buzo and Gray-(1980), Buzo, Gray, Gray and Markel-(1980a and 1980b), Gray, Gray, Robodello and Shore-(1981) and Abut, Gray and Robodello(1980). The theory and some results on vector quantisation have also been presented by Gersho-(1982), Gersho and Ramamurthi-(1982) and Fischer and Dicharry-(1984). Ahmadi-(1980) and Wilson-(1983) have also investigated the application of vector quantisation to speech coding, They have investigated the clustering of speech according to a spectral distance measure, defined in a transform domain.

Comparison of various methods

Some useful papers, assessing the relative merits of some coding schemes have been reported by the following: Tribolet, Noll, McDermott and Crochiere-(1979) have published the results of the comparison of adaptive transform coding, adaptive differential pulse code modulation and subband coding. Matsuyama and Gray-(1982) have reported the results of a comparison of vector quantisation based on LPC and tree coding using adaptive prediction coefficients.

CHAPTER 3 ADAPTIVE DATA COMPRESSION WITH MEMORY, THE BLOCK CODING APPROACH

3.1 Introduction

In this chapter we describe and investigate a class of coding schemes which relies upon the representation of data blocks by previously encoded blocks.

Most source coding schemes, to achieve good compression, need to be designed with due regard to the statistics of the source to be coded. Coding theorems have been proved for block coding, tree and trellis coding. These theorems show that as some parameter is allowed to go to infinity, these coding schemes may be designed to work arbitrarily close to the rate-distortion function of a source. [Shannon-(1959), Jelinek-(1969), Viterbi and Omura-(1974)] For these schemes to achieve their promise however, they need to be well designed and this requires a knowledge of the statistics of the source to be coded. For most of the sources of interest the design of a good coding scheme is not easy. This is because the statistics of these are generally unknown apriori. Alternatively, the local statistics of long sequences from the source may be observed to vary from a block of data to the next. A source that exhibits this property is referred to as one with time varying statistics. Knowledge of the overall statistics of a source with time varying statistics and coding according to these statistics does not necessarily result in the lowest bit rate that may be achieved for such a source [Viterbi and Omura-(1979) p526]. For such a source if the statistics are varying slowly enough, it may be profitable to employ a different coding scheme for each of the different statistical classes which this source may exhibit from block to block. To code such a source, apriori analysis is required;

parameters describing the different statistical classes which may occur, should be extracted. This is often impracticable. The solution is to employ a coding algorithm which will enable the reasonable compression of data and which will make few assumptions about the character of the statistics of the source to be coded. The common approach is to encode the data in blocks as follows. [For example Zelinski and Noll-(1977 and 1979) in adaptive transform coding] For each block the local statistics are evaluated. An appropriate coding strategy is employed for this block, bearing those statistics in mind. The receiver is sent symbols, identifying either the statistics of each block or symbols indicating the coding strategy used. In addition, the transmitter sends the symbols associated with the coding of the source in the manner chosen.

In this chapter an alternative scheme to that described above is studied. It shall be refered to as the MPPCD scheme. This stands for 'the Matching of Patterns in Previously Coded Data'. The scheme is described in section 3.2. This method of coding is not new. It has received however, very little attention and its application has been until now been limited to the compression of facsimile data. The results, for facsimile coding, have been reported by Arena and Zarone-(1978) and Pratt, Capitant, Chen, Hamilton and Wallis-(1980). It is our intention to generalise its field of application and report on its performance. The MPPCD scheme is thus applied to the coding of multilevel image data and speech data. We propose several new variations to the scheme and investigate their data compression abilities. This is done in the situation where no assumptions are made concerning the data to be coded. A comparison of the performance is made between, in the case of noiseless coding, this scheme, the Shannon entropy and Huffman coding. In the case of coding with error, a comparison is made between this scheme the rate-distortion bound for the source and Discrete Cosine Transform(DCT) coding.

The MPPCD scheme is then used in situations where we allow ourselves some knowledge of the type of source being coded. For image data this allows the consideration of alternative sampling schemes and two dimensional blocks instead of one dimensional sequences. In addition the effects of the use of distortion measures other than the simple mean square error is studied.

For speech coding, the scheme is used to improve the performance of the Linear Predictive Vocoder (LPC Vocoder).

The chapter concludes with a discussion of the merits and the shortcomings of the MPPCD scheme.

3.2 The basic coding scheme

The underlying principle of this scheme is as follows. A block is coded by matching the patterns of this block with those of previously coded data. It is referred to as the MPPCD scheme, where MPPCD stands for the Matching of Patterns in Previously Coded Data. In order to describe the operation of the scheme the following items have to be defined.

1. A distortion measure is chosen and a distortion limit d^* is set. d^* is a distortion value which should not be exceeded as the coding scheme proceeds.
2. Sequences of lengths L_1, $L_2, \ldots L_N$, are chosen where $L_1 < L_2 < \ldots < L_N$.
3. Choose a quantity C, where C is the number of levels that may be chosen so that the distortion which results from uniform quantisation to C levels is less than d^*.
4. The quantity N, the number of possible lengths is chosen so that $\log_2 N$ is much smaller than $\log_2 C$. An example is C=256, N=4 and $L_1=1$, $L_2=2$, $L_3=4$, $L_4=8$.

The coding scheme will be explained using the particular values of C, N and L given above. The general case is an easy extension. A flow chart for the scheme is given in figure 3.1.

Suppose that a pointer is set to i so that all data symbols $x_j : j<i$ have already been coded and hence both the receiver and transmitter know that these have been approximated by the symbols, $\tilde{x}_j : j<i$. We consider a block of data $\{x_i, x_{i+1}, \ldots, x_{i+7}\} = \underline{X}^8$ say. We attempt to encode this sequence by sampling the set of previously coded data $\tilde{x}_j : j<i$ for a sequence of symbols which are similar to

$\underline{X}^8$. This sampling is done in an orderly and predefined manner and we have exactly C tries. Associated with each of the C tries is a coordinate value. It should be noted that the set of approximations are known to both the receiver and the transmitter. If an approximation $\tilde{\underline{X}}^8$ is found in one of the C sampling experiments on the set of previously coded outcomes, such that $d(\underline{X}^8, \tilde{\underline{X}}^8) \leq d^*$ ($d(\underline{X}^8, \tilde{\underline{X}}^8)$ is the distortion between the two sequences $\underline{X}^8$ and $\tilde{\underline{X}}^8$), then the coordinate of this event in the set of previously coded symbols is transmitted using $\log_2 C$ bits. An additional $\log_2 N$ (in this case $\log_2 4$) bits are used to indicate the length of the block coded. At the receiver, the sequence $\underline{X}^8$ is approximated by $\tilde{\underline{X}}^8$. The counter is advanced by 8 positions and the coding scheme proceeds exactly as described so far. In addition, at the transmitter, the fact that the receiver will approximate $\underline{X}^8$ by $\tilde{\underline{X}}^8$ is noted.

If an approximation $\tilde{\underline{X}}^8$ which satisfies the distortion conditions is not found, we attempt to code a block of smaller size. Consider the block $\{x_i, x_{i+1}, x_{i+2}, x_{i+3}\} = \underline{X}^4$ say. We attempt to code this sequence by sampling the set of previously coded data $\tilde{x}_j : j<i$ for 4 symbols $\tilde{\underline{X}}^4$ say, so that $d(\underline{X}^4, \tilde{\underline{X}}^4) \leq d^*$. Exactly C sampling experiments are conducted. If an approximation is found which satisfies the distortion constraint, the coordinate of the particular event is transmitted, using $\log_2 C$ bits. An additional $\log_2 N$ bits are used to indicate the block size. At the receiver, $\underline{X}^4$ is approximated by $\tilde{\underline{X}}^4$. The algorithm counter is advanced by 4. The coding scheme as described so far, is continued.

Failure to find a sequence $\tilde{\underline{X}}^4$ which satisfies the distortion constraint, leads to an attempt to code $\{x_i ,x_{i+1}\}=\underline{X}^2$ say, employing the previously coded data. Success at this, means the receiver approximates $\underline{X}^2$ by $\tilde{\underline{X}}^2$ and the counter is increased by 2, the coding scheme proceeds as described so far. We next proceed by trying to code a block of length 8.

Upon failure to code a block of length 2, the symbol x_i is quantized to the closest of C levels and transmitted using $\log_2 C$ bits plus $\log_2 N$ bits to indicate the block size. The receiver approximates x_i by $\tilde{x}_i$, the nearest quantization level. The transmitter notes that the receiver has done this. The algorithm counter is advanced by 1 and coding continues as described before.

At this point it is useful to indicate the way in which the sequence of previously coded data is sampled in search of an approximation to a sequence of interest. Two examples of how this may be done are as follows.

a) Consider the sequence of approximations $\tilde{x}_j : j<i$. If one tries to code the block of data $\underline{X}^N$, of length N the following are the blocks of previously coded data which are candidates for approximating $\underline{X}^N$:

$\{\tilde{x}_{i-1},\ldots,\tilde{x}_{i-N}\},\{\tilde{x}_{i-2},\ldots,\tilde{x}_{i-N-1}\},\{\tilde{x}_{i-3},\ldots x_{i-N-2}\},\ldots,\{\tilde{x}_{i-C},\ldots,\tilde{x}_{i-C-N+1}\}$

These are overlapping blocks. Alternatively non-overlapping blocks may be considered:

$\{\tilde{x}_{i-1},\ldots,\tilde{x}_{i-N}\},\{\tilde{x}_{i-N-1},\ldots,\tilde{x}_{i-2N}\},\{\tilde{x}_{i-2N-1},\ldots,\tilde{x}_{i-3N}\},\ldots,\{\tilde{x}_{i-(C-1)N-1},\ldots,\tilde{x}_{i-CN}\}$

In the above two methods we almost inevitably have C sequences which are not all distinct. To get over this the following may be done.

b) We construct N libraries, each contains sequences of previously coded symbols, possibly distorted versions of $x_i : i<j$. The different libraries contain sequences of different lengths, $L_1, L_2, \ldots, L_N$. Sampling of the set of previously coded data C times, for a block of length L_m, actually means searching the m-th library. Suppose the counter for the algorithm is at position i. If the search of the m-th library (library containing sequences of length L_m) for a suitable approximation $\tilde{\underline{X}}^{L_m}$ proves unsuccessful, then for this library, the pointer value p=i is stored in memory. When the pointer value goes beyond $i+L_m$, then the sequence of approximations $\{\tilde{x}_i ,\ldots,\tilde{x}_{i+L_m}\}$ is included in this library. This sequence goes to the top of that library. The earliest in that library is removed. If the search of the m-th library is successful, this library is kept unaltered. In the receiver, the following happens. Upon the receipt of a symbol, m say, indicating the length of the block just coded, the pointer value q=i is stored, Note that the fact that a block of length L_m was coded implies that the coding of a block of longer length was not possible. This in turn means that the libraries containing sequences of length $L_{m+1}, \ldots, L_N$ should be altered. As the pointer value q goes past $i+L_k$, where $m+1<k<N$, the sequence $\{\tilde{x}_i ,\tilde{x}_{i+1}, \ldots, \tilde{x}_{i+L_k}\}$ is included in the set of sequences belonging to the k-th library. This sequence goes to the top of the k-th library and the earliest member of this library is removed. All members of this library are moved down one position.

To recapitulate then, there are N libraries of different lengths maintained at the transmitter. Replicas of these are available at the receiver. Each library is like a stack, whenever a sequence of a given length is not codable, using the library of sequences of this length, a subsequent approximation for this sequence, goes to the top of this stack, in both the transmitter and receiver. The earliest member of this library is removed.

The scheme for sampling the sequence of previously coded data as described in "a", is obviously inferior, as far as permitting efficient compression is concerned. The scheme as described in "b", is more complicated and requires more effort in implementation in addition to requiring large quantities of memory for the storage of the members of the N libraries. Most of the investigation undertaken in this chapter employ the scheme described in "a".

At this stage the concept of an elementary block size should be introduced. The coding process as described so far, may be implemented with each of the original source symbols replaced by blocks of these. The elementary block size is N if sequences of N original source symbols are used in place of one symbol in the scheme as described so far.

To conclude the description of the basic MPPCD scheme, we consider how the algorithm is initialised. This is actually rather obvious. There are two ways. The first is to have stored in memory a set of randomly generated data, duplicates of which are kept at both the receiver and transmitter. This sequence acts initially as the set of previously coded data. Alternatively, initially almost all generated data is transmitted. Sampling of previously coded data is done only as far as previously generated data exists.

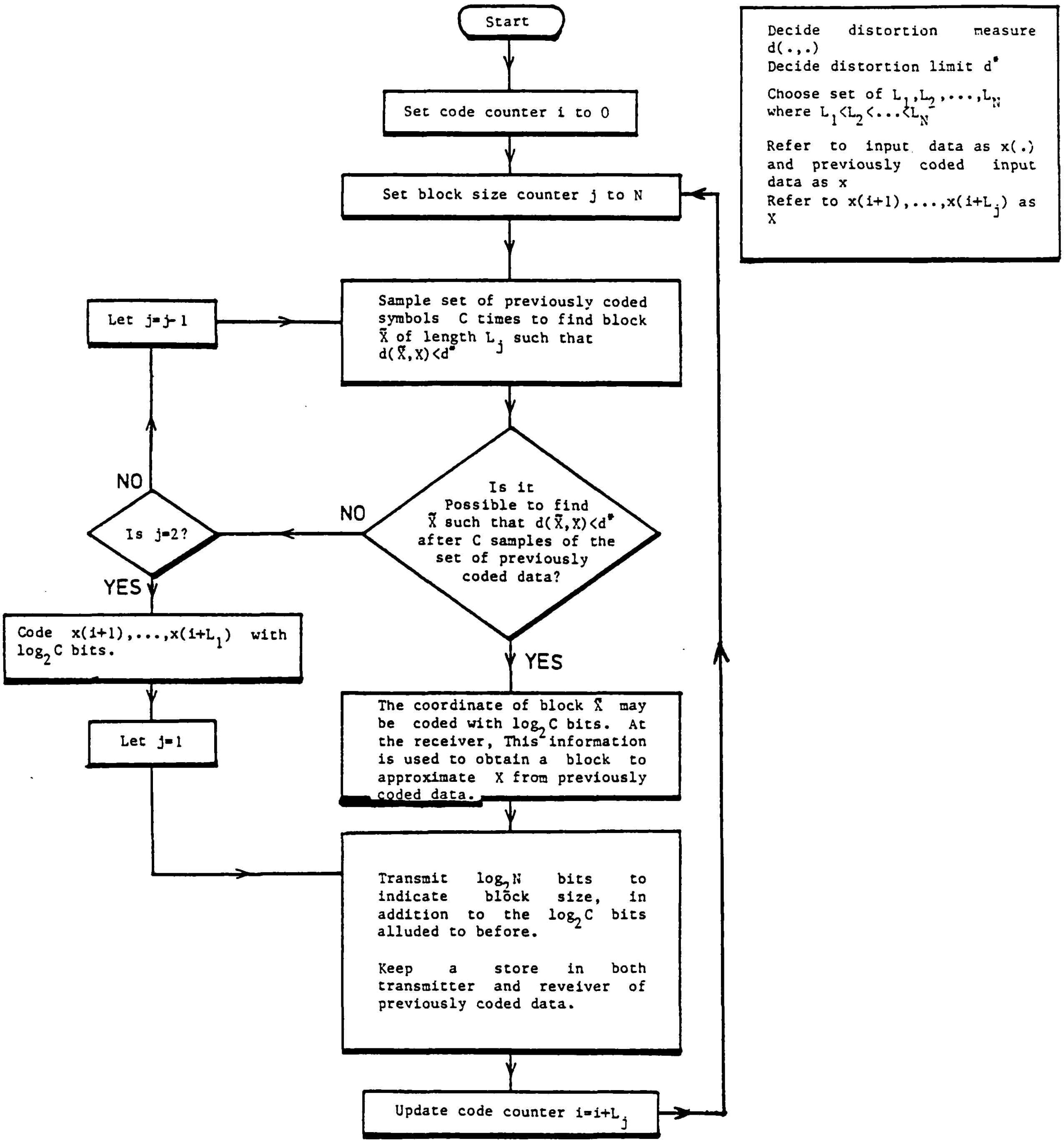

Start
Set code counter i to 0
Set block size counter j to N
Decide distortion measure d(.,.)
Decide distortion limit d^*
Choose set of $L_1, L_2, \ldots, L_N$ where $L_1 < L_2 < \ldots < L_N$
Refer to input data as x(.) and previously coded input data as x
Refer to $x(i+1), \ldots, x(i+L_j)$ as X
Sample set of previously coded symbols C times to find block $\tilde{X}$ of length L_j such that $d(\tilde{X}, X) < d^*$
Let $j = j-1$
NO
Is j=2?
YES
NO
Is it Possible to find $\tilde{X}$ such that $d(\tilde{X}, X) < d^*$ after C samples of the set of previously coded data?
YES
Code $x(i+1), \ldots, x(i+L_1)$ with $\log_2 C$ bits.
Let j=1
The coordinate of block $\tilde{X}$ may be coded with $\log_2 C$ bits. At the receiver, This information is used to obtain a block to approximate X from previously coded data.
Transmit $\log_2 N$ bits to indicate block size, in addition to the $\log_2 C$ bits alluded to before.
Keep a store in both transmitter and reveiver of previously coded data.
Update code counter $i = i + L_j$

Figure 3.1. Flow chart for basic MPPCD scheme

3.3 Performance with artificially generated data

3.3.1 Noiseless coding

The minimum coding rate achievable, in effecting the noiseless coding of a stationary statistical source is the Shannon entropy of the source [Shannon 1948a, theorem 3]. Suppose a source has a sample space Ω with C members $\omega_1, \omega_2, \ldots, \omega_C$. Let the probability of occurrence of ω_i be p_i. Then the Shannon entropy for this source is

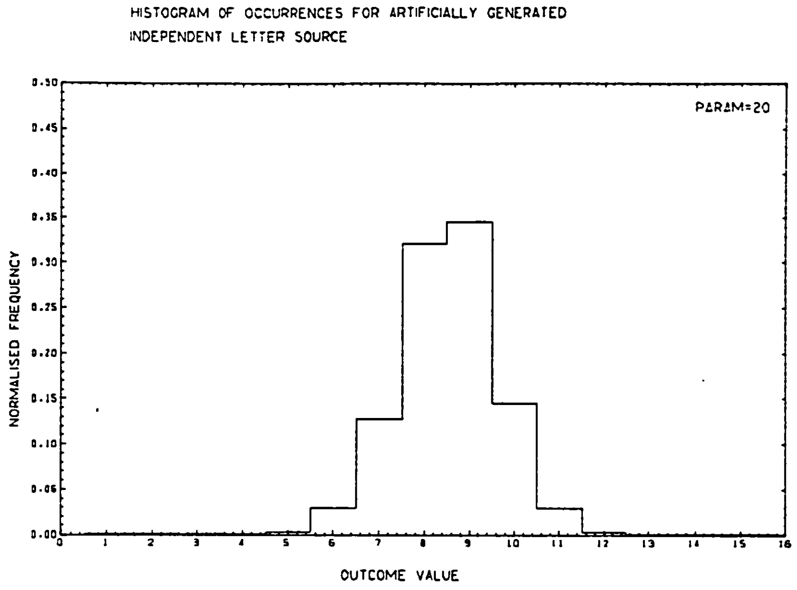
$$H(\Omega) = -\sum_i p_i \log_b p_i \qquad 3.1$$

The unit of the coding rate is "bits per symbol" if the base of the log is 2. Huffman coding, [Huffman-(1953)] is a scheme which, for any given block size, achieves the minimum possible coding rate.

The results achieved when using the MPPCD scheme are compared with the Shannon entropy and the Huffman coding rate for a variety of situations.

1) Independent letter source. An artificially generated sequence of random numbers are coded. The source sample space is the set of integers 1 to 16. The histogram and the coding results for this source are given in figure 3.2 and table 3.1. The MPPCD scheme as expected, is inferior to the Huffman coding scheme. It nevertheless achieves reasonable compression considering that no prior information about source statistics is used.

The compression efficiency of the scheme, for both the methods "a" and "b" of sampling the set of previous outcomes, improves as the elementary block size is increased.



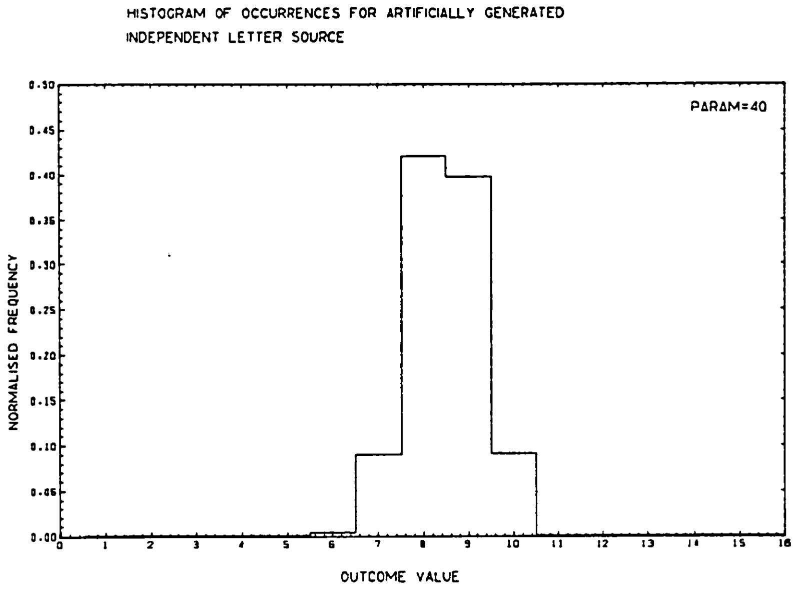


Figure 3.2. Histograms for independent letter sources coded by MPPCD scheme

Histogram parameter=20.0

Shannon Entropy:	2.2294 bits/symbol
Huffman Coding Rate:	2.3426 bits/symbol

MPPCD: (Sampling Scheme="a")

$L_1=1$, $L_2=2$, $L_3=4$, $L_4=6$;	Elementary block size=1, Rate= 3.6633 bits/symbol
	Elementary block size=2, Rate= 3.3431 bits/symbol
	Elementary block size=3, Rate= 3.1908 bits/symbol
$L_1=1$, $L_2=2$, $L_3=3$, $L_4=4$;	Elementary block size=1, Rate= 3.4671 bits/symbol
	Elementary block size=2, Rate= 3.2308 bits/symbol
	Elementary block size=3, Rate= 3.1908 bits/symbol

Same source as above but more data samples.

Shannnon Entropy:	2.1365 bits/symbol
Huffman Coding rate:	2.2422 bits/symbol.

MPPCD: (sampling Scheme="b")

$L_1=1$, $L_2=2$, $L_3=4$, $L_4=6$;	Elementary block size=1, Rate= 3.2546 bits/symbol
	Elementary block size=2, Rate= 3.0679 bits/symbol
$L_1=1$, $L_2=2$, $L_3=3$, $L_4=4$;	Elementary block size=1, Rate= 2.9897 bits/symbol
	Elementary block size=2, Rate= 2.9435 bits/symbol

Table 3.1 Results of coding independent letter source using the MPPCD scheme.

Table 3.1 (continued)

Histogram parameter=40.0

Shannon Entropy: 1.7758 bits/symbol
Huffman Coding Rate: 1.9151 bits/symbol

MPPCD: (Sampling Scheme="a")

$L_1=1$, $L_2=2$, $L_3=4$, $L_4=6$; Elementary block size=1, Rate= =2.9868 bits/symbol
Elementary block size=2, Rate= =2.6989 bits/symbol
$L_1=1$, $L_2=2$, $L_3=3$, $L_4=4$; Elementary block size=1, Rate= =2.7350 bits/symbol
Elementary block size=2, Rate= =2.4451 bits/symbol

Same source as above but more data samples.

Shannnon Entropy: 1.7607 bits/symbol
Huffman Coding rate: 1.8977 bits/symbol.

MPPCD: (sampling Scheme="b")

$L_1=1$, $L_2=2$, $L_3=4$, $L_4=6$; Elementary block size=1, Rate= 2.5603 bits/symbol
Elementary block size=2, Rate= 2.5680 bits/symbol
$L_1=1$, $L_2=2$, $L_3=3$, $L_4=4$; Elementary block size=1, Rate= 2.3325 bits/symbol
Elementary block size=2, Rate= 2.3329 bits/symbol

2) Non-independent source. An artificially generated sequence of non-independent random numbers are coded. The source is 1-st order Markov, where each outcome $x_k=\omega_i$ is dependent only upon the value of the immediately preceding outcome, $x_{k-1}=\omega_j$. The sample space of this source is the set of integers from 1 to 16. The symbols of the 1-st order Markov source are generated by picking as the k-th outcome a symbol from the i-th of C subsources, if $x_{k-1}=\omega_i$. The transition matrices with entries $p(x_k=\omega_i|x_{k-1}=\omega_j)$ ((i,j)th entry) for all i and j for two particular examples are shown in tables 3.2a and 3.2b. The overall outcome frequency histograms are shown in figure 3.3.

The coding rates for the MPPCD scheme, the single letter Shannon entropy values and the rate achieved when doing single letter Huffman coding are shown in table 3.3. It may be observed that the MPPCD scheme achieves better compression than the single letter Shannon entropy and the single letter Huffman coding rates. It should be noted, however that this is an easy source to code. With prior knowledge of the character of the source, we know that we can code these sources at under 2 and 3 bits respectively, for the two sources using differential encoding.

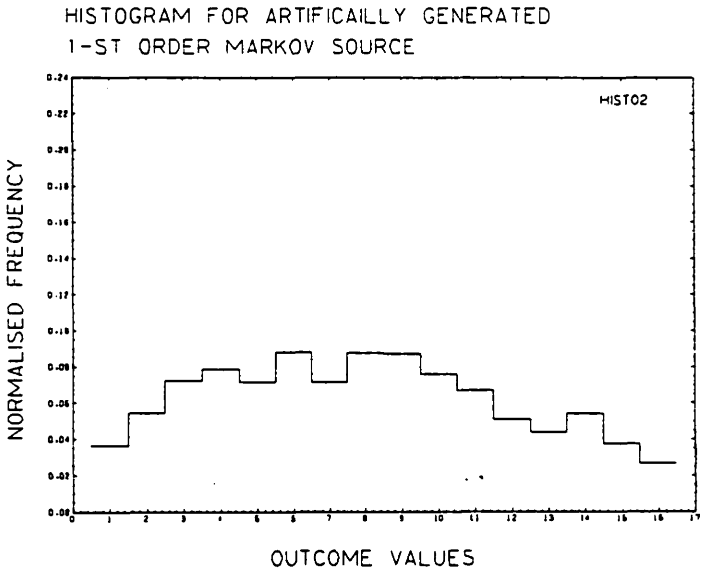
3) Sources with time varying statistics. An artificially generated source with the following characteristics is coded. The source is 1-st order Markov but where the transition probability matrix values are occasionally altered. The overall transition probability matrix is shown in table 3.4. The occasional variation of the transition matrix makes the design of a coder that exploits the basically Markov nature of the source, difficult.

.5818	.3594	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.4128	.2969	.3438	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.3438	.2500	.2301	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.4063	.4425	.3033	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.3274	.3033	.3000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.3934	.3188	.3742	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.3813	.3558	.2953	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.2699	.3758	.2934	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3289	.3054	.3526	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.4012	.3368	.3734	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3105	.3101	.3247	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3165	.3766	.3286	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.2987	.3357	.3507	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3357	.2910	.3692	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3582	.2846	.5233
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3462	.4767

.5000	.3203	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.5000	.3359	.3402	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.3438	.3222	.3531	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.3376	.3531	.3407	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.2938	.3565	.3430	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.3028	.2996	.3498	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.3574	.3004	.3414	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.3498	.3690	.3784	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.2897	.3739	.3125	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.2477	.3409	.3653	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3466	.3353	.2941	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.2994	.3588	.3089	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3471	.3770	.2947	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3141	.3892	.3290	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3161	.2987	.4971
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.3723	.5029

Table 3.2a Example of Markov source. This source is coded using the MPPCD scheme with sampling method "a".

Table 3.2b Example of Markov source. This source is coded using the MPPCD scheme with sampling method "b".



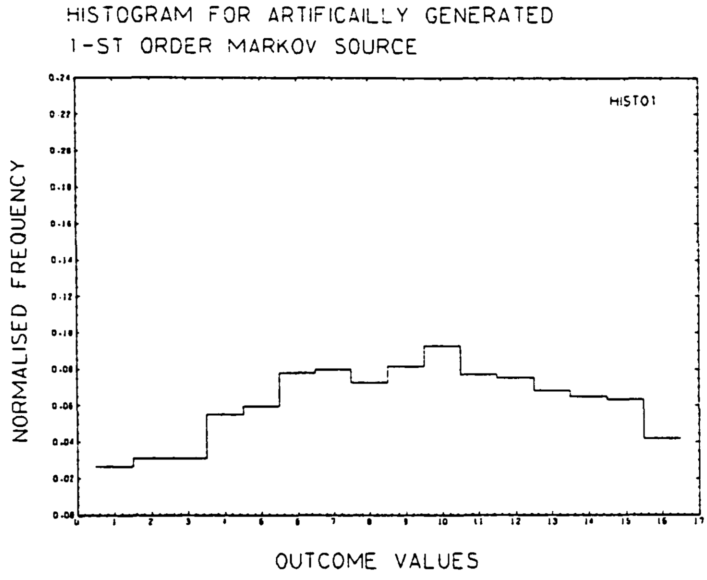


Figure 3.3. Histograms for two examples of Markov source used in MPPCD coding.

Single letter Shannon entropy: 3.9242 bits/symbol
Single letter Huffman coding rate: 3.959 bits/symbol

MPPCD: (Sampling Scheme="a")

L_1=1, L_2=2, L_3=4, L_4=6;	Elementary block size=1, Rate= =3.6535 bits/symbol
	Elementary block size=2, Rate= =3.3432 bits/symbol
	Elementary block size=3, Rate= =2.7255 bits/symbol
L_1=1, L_2=2, L_3=3, L_4=4;	Elementary block size=1, Rate= =3.4479 bits/symbol
	Elementary block size=2, Rate= =3.1798 bits/symbol
	Elementary block size=3, Rate= =2.6486 bits/symbol

Same source as above but more data samples.

Single letter Shannnon Entropy: 3.9401 bits/symbol
Single letter Huffman Coding rate: 3.98 bits/symbol.

MPPCD: (sampling Scheme="b")

L_1=1, L_2=2, L_3=4, L_4=6;	Elementary block size=1, Rate= 3.438 bits/symbol
	Elementary block size=2, Rate= 3.2083 bits/symbol
L_1=1, L_2=2, L_3=3, L_4=4;	Elementary block size=1, Rate= 3.2427 bits/symbol
	Elementary block size=2, Rate= 3.0739 bits/symbol

Table 3.3. Results of coding non-independent (1st order Markov) source using MPPCD scheme.

The results of the performance of the MPPCD scheme are given in table 3.5; along with these are the single letter Shannon entropy and the single letter Huffman coding rate.

It may be observed that the MPPCD scheme achieves a significantly lower coding rate than that obtained by single letter Huffman coding. Furthermore considering the complexity of the mechanism for generating the data it is unlikely that the underlying Markov character of the source could be discovered and advantage taken of this, in a normal coding environment.

The method has the following disadvantage though; if the period between instances when the local statistics change, is too short, its compression ability is greatly reduced. By too short, it is meant that this period is of a similar order of magnitude to the memory of the MPPCD coding scheme. For example, take the scheme whose results have been given in table 3.5. If the "elementary block size" is one, the size of the codebook and hence the memory of the coding system is 16. The period between instances of changes in statistics should be much greater than 16. This accounts for the poorer result when an elementary block size of 2 is used.

3.3.2 Coding with a fidelity criterion

The MPPCD scheme is used to code an artificially generated source, where we allow distortion. A distortion limit, or fidelity criterion, is set. We attempt to code a source at the smallest bit rate manageable and still make sure that the resulting distortion is

.4606	.0681	.0000	.0000	.1012	.0653	.0374	.0875	.0518	.0729	.0883	.0739	.1002	.0974	.0423	.0182
.0376	.3912	.0664	.0122	.0000	.0158	.0966	.0519	.0422	.0430	.0188	.0246	.0000	.0139	.0500	.0564
.0012	.0725	.3763	.2068	.0000	.0614	.0165	.0357	.1024	.0486	.0094	.03450	.0000	.0000	.0000	.0000
.0000	.0110	.1710	.3431	.0000	.0257	.0000	.0000	.0493	.0411	.0714	.0197	.0000	.0736	.0000	.0315
.0315	.0000	.0000	.0000	.3852	.0119	.0685	.0276	.0000	.0766	.0000	.0000	.0235	.0000	.0385	.0415
.0412	.0176	.0604	.0292	.0233	.4099	.0654	.0000	.0000	.0542	.0282	.0000	.1343	.0477	.0000	.0929
.0158	.0703	.0121	.0000	.0817	.0416	.3209	.0000	.0000	.0449	.0338	.0328	.0832	.0219	.0192	.0133
.0667	.0703	.0443	.0000	.0700	.0000	.0000	.4619	.0139	.0000	.0545	.0722	.0320	.0875	.0000	.1028
.0509	.0769	.1630	.0973	.0039	.0000	.0000	.0178	.4058	.0393	.0338	.0575	.1301	.1392	.0462	.0697
.0461	.0505	.0523	.0535	.1556	.0554	.0779	.0269	.0265	.3701	.0583	.0345	.0746	.0099	.0269	.0249
.0558	.0220	.0121	.0925	.0000	.0297	.0592	.0502	.0240	.0598	.3327	.0181	.0000	.0616	.1769	.0846
.0521	.0308	.0423	.0292	.0000	.0000	.0623	.0713	.0455	.0374	.0226	.4959	.0021	.0000	.0731	.1078
.0582	.0000	.0000	.0000	.0389	.1267	.1184	.0227	.0759	.0654	.0019	.0000	.3049	.0398	.0846	.0232
.0582	.0154	.0000	.0900	.0000	.0455	.0343	.0729	.0885	.0093	.0620	.0000	.0405	.3559	.0962	.0017
.0133	.0308	.0000	.0000	.0350	.0000	.0156	.0000	.0164	.0122	.0883	.0296	.0469	.0497	.3462	.0000
.0109	.0725	.0000	.0462	.1051	.1109	.0280	.1005	.0556	.0262	.0959	.1067	.0277	.0020	.0000	.3317

Table 3.4. Transition probability matrix for source with time varying statistics.

Single letter Shannnon Entropy: 3.9327 bits/symbol
Single letter Huffman Coding rate: 3.9557 bits/symbol.

MPPCD: (sampling Scheme="b")

L_1=1, L_2=2, L_3=3, L_4=4;; Elementary block size=1, Rate= 2.9853 bits/symbol
Elementary block size=2, Rate= 3.8350 bits/symbol

Table 3.5. Results of coding pseudo-Markov source (Markov source with time varying transition probability matrix) with MPPCD scheme.

less than the limit set.

To compare the performance of the scheme with the theoretically attainable limits, the rate-distortion (r-d) function is introduced. The minimum rate (in nats) theoretically attainable, for a given source with probability measure function q(x), so that distortion is less than some value d*, for different values of d*, defines the rate distortion function. Its formal definition is given thus,

$$R(d^*) = \inf_{p(y|x)} \int_R q(x) \int_R p(y|x) \ln \frac{p(y|x)}{\int_R q(x)p(y|x)dx} \, dydx \qquad 3.2$$

such that

$$d^* \leq \int_R q(x) \int_R p(y|x)d(x,y)dydx \qquad 3.3$$

and

$$1 = \int_R p(y|x)dy \quad \forall x \qquad 3.4$$

d(x,y) is a distortion measure which is chosen beforehand. For the tests conducted in this section the mean square error distortion measure is employed. The rate distortion function and the source coding theorem serve as the formal basis for the subject of data compression. Details of these may be found in the publications by Shannon-(1959), Gallagher-(1968), Berger-(1971), Viterbi and Omura-(1979).

A source whose rate-distortion function is easily bounded from above is used to test the MPPCD scheme. This is a 1-st order auto-regressive source, with a variance σ^2 and correlation coefficient ρ.

The upper bound on the rate-distortion function for this source is given by the rate-distortion function of the Gaussian with similar statistics. This upper bound is given below.

$$R(d^*,\rho) = \int_{-\pi}^{\pi} \max\left[0,\frac{1}{2}\ln\left(\frac{1-\rho^2}{d^*(1-2\rho cos\omega+\rho^2)}\right)\right] d\omega \qquad 3.5$$

The results of coding are presented in table 3.6. These results are compared to the results obtained by doing discrete cosine transform coding, in addition to the theoretically attainable rate. Discrete cosine transform coding is chosen as a good example of a practical source coding scheme for the following reason. This has been shown empirically to give good results, especially in the case where the source being coded may be modelled as a first order auto-regressive source. The papers by Ahmed, Natarajan and Rao-(1974) and Kitajima, Saito and Kuroba-(1977) show the closeness of the results of discrete cosine transform (DCT) coding to the theoretical results (obtained by doing finite length Karhunen-Loeve or Hotelling transform coding) for finite length blocks. It may be observed that the results of doing MPPCD coding are worse than those for DCT coding. This is to be expected since DCT transform coding is close to optimal for the source class considered. It is also to be noted that as the elementary block size is increased, the coding rate is reduced.

Most of the coding inefficiency of the MPPCD scheme may be attributed to the fact that extra bits are sent to indicate the block size. This is one of the reason for an increase in coding efficiency when the elementary block size is increased, this being

Autoregressive source generated as y(n)= y(n-1)+e(n). e(n) is a sequence of outcomes from an independent, Laplacian distributed random variable. σ^2=variance of y(n) sequence.

$\sigma^2=$	0.956
$\rho=$	0.8
d =	0.1, (-10db); R(d)$\leq$ 0.923

MPPCD: (sampling Scheme="b")

$L_1=1$, $L_2=2$, $L_3=4$, $L_4=6$; Elementary block size=1, Rate= 2.6483 bits/symbol
Elementary block size=2, Rate= 2.2979 bits/symbol
$L_1=1$, $L_2=2$, $L_3=3$, $L_4=4$; Elementary block size=1, Rate= 2.5417 bits/symbol
Elementary block size=2, Rate= 2.1117 bits/symbol

Coding results, not including the 2 bits for block size representation:

$L_1=1$, $L_2=2$, $L_3=3$, $L_4=4$; Elementary block size=1, Rate= 1.6945 bits/symbol
Elementary block size=2, Rate= 1.6894 bits/symbol

$\sigma^2=$	0.9562
$\rho=$	0.8
d =	0.02 (-17db); R(d)$\leq$ 2.084

MPPCD: (sampling scheme="b")

$L_1=1$, $L_2=2$, $L_3=3$, $L_4=4$; Elementary block size=1, Rate= =4.4475 bits/symbol
Elementary block size=2, Rate= =4.0729 bits/symbol

$\sigma^2=$	0.9562
$\rho=$	0.8
d =	0.04 (-14db); R(d)$\leq$ 1.584

MPPCD: (sampling scheme="b")

$L_1=1$, $L_2=2$, $L_3=3$, $L_4=4$; Elementary block size=1, Rate= =3.5463 bits/symbol
Elementary block size=2, Rate= =3.1669 bits/symbol

Table 3.6. Results for coding a 1-st order auto-regressive source with a fidelity criterion. Method is MPPCD scheme.

Rate in bits/symbol	Distortion	
4.0	6.53383E-03	21.8485dB
3.5	9.70505E-03	20.1300dB
3.0	2.02198E-02	16.9422dB
2.5	3.10698E-02	15.0767dB
2.0	6.38411E-02	11.9489dB
1.75	7.48229E-02	11.2596dB
1.5	9.73681E-02	10.1158dB
1.0	1.91715E-01	07.1734dB

Table 3.7. Results for coding a 1-st order auto-regressive source with a fidelity criterion. Method is Discrete-Cosine-Transform coding. Correlation coeff=0.8

because proportionately fewer bits are used to encode the block size coded. It is expected therefore that better compression may be achieved if more efficient methods are used to code the block lengths.

3.4 Application to speech coding

We shall now leave the artificially generated sources and consider the application of the MPPCD scheme to real data, in this case speech. For the results presented in this section three speech sentences are employed. The speech files are these:

1) A male speaker saying AN APPLE A DAY KEEPS THE DOCTOR AWAY (SR8KK).
2) A second male speaker saying A BIRD IN THE HAND IS WORTH TWO IN THE BUSH (KABITH).
3) A female speaker saying A BIRD IN THE HAND IS WORTH TWO IN THE BUSH (TABITH).

For all the above files the speech was low-pass filtered to 3.4kHz. and sampled at 8kHz. The first file was digitised to an accuracy of 12 bits per sample and the second and third to an accuracy of 10 bits per sample. Portions of these files are plotted in figures 3.4a, 3.4b and 3.4c.

For all investigation presented from here onwards, the set of previously coded symbols is sampled, as explained in method "a" of section 3.2. In this section no prior knowledge of the speech file to be coded is presumed. No attempt is made to take advantage of some of the known characteristics of speech.

The coding scheme used, is almost identical to that described in section 3.3.2. There is some difference though, therefore the scheme will be described again.

Presume that the coding pointer is at i, and all samples

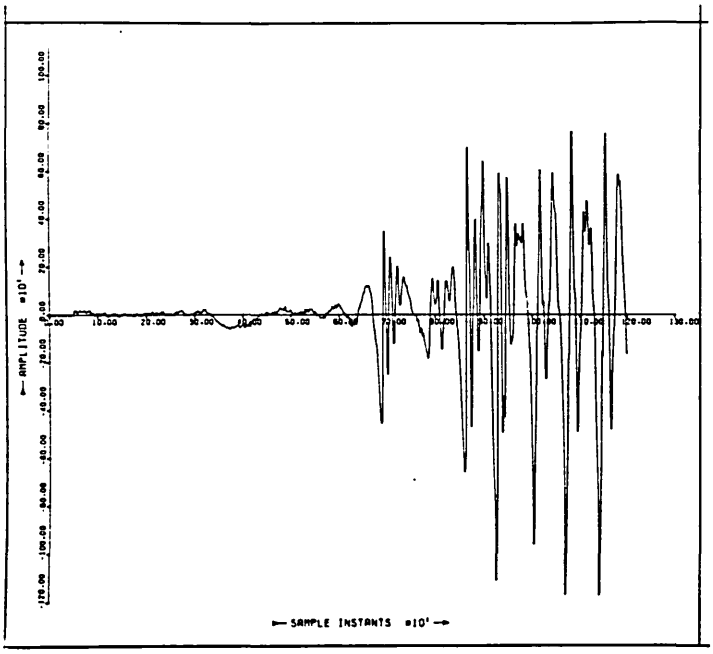


Figure 3.4a Portion of speech file:- SR8KK

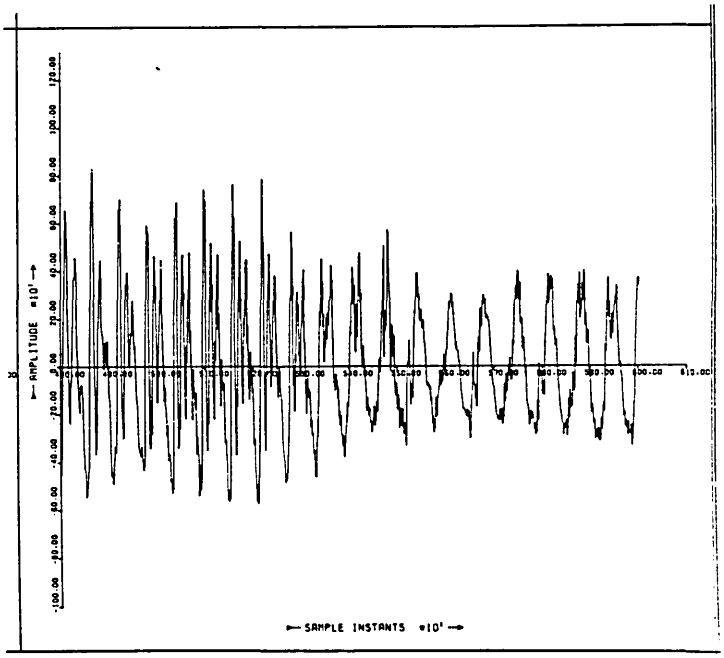


Figure 3.4b Portion of speech file:- KABITH

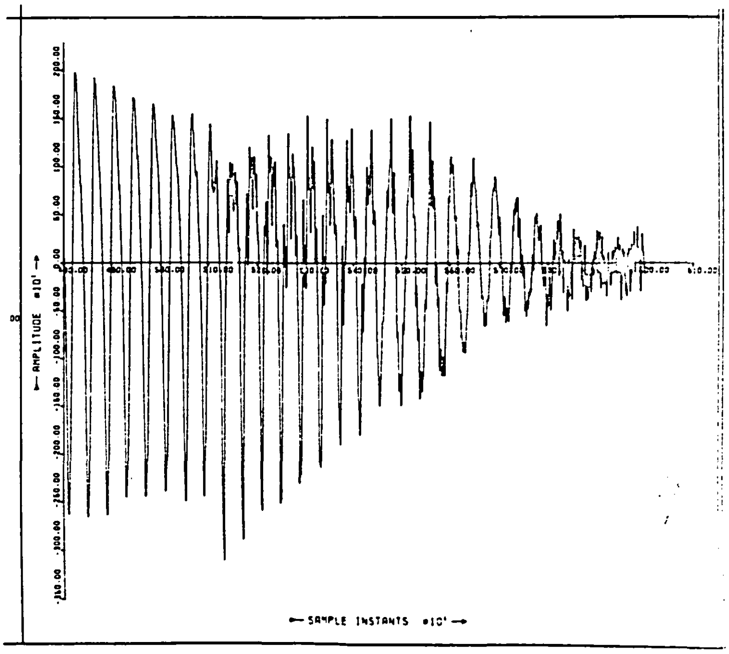


Figure 3.4c Portion of speech file:- TABITH

$x_j : j<i$ have been coded. The sequence of approximations $\tilde{x}_j : j<i$ to the previous symbols is known to both the transmitter and receiver. We shall consider coding sequences $\{x_i, x_{i+1}, \ldots, x_{i+L_N-1}\} = \underline{X}^{L_N}$ of length L_N. Two examples of what is done here, which show the differences between these and the scheme described in section 3.3.2 are as follows:

a) Normalisation with respect to the mean. Block lengths $L_1, \ldots, L_6$ are considered, with $L_6=64$, $L_5=32$, $L_4=16$, $L_3=8$, $L_2=4$ and $L_1=2$. Any block considered, $\underline{X}^{L_k}$, has its mean value extracted, quantised and coded separately using 6 bits. The quantised mean $\underline{M}^{L_k}$, is subtracted from $\underline{X}^{L_k}$. Let $\underline{Y}^{L_k} = \underline{X}^{L_k} - \underline{M}^{L_k}$. Then the job of coding the sequence $\underline{Y}^{L_k}$ is undertaken. This is effected by searching the set of previously coded outcomes for an approximation, $\tilde{\underline{X}}^{L_k}$; any candidate for the purpose of approximation has its mean subtracted, giving $\tilde{\underline{Y}}^{L_k} = \tilde{\underline{X}}^{L_k} - \tilde{\underline{M}}^{L_k}$. Coding is a success if

$d(\underline{Y}^{L_k} + \underline{M}^{L_k}, \tilde{\underline{Y}}^{L_k} + \underline{M}^{L_k}) \leq d^*$

If there is no success in coding $\underline{X}^{L_2}$, after having searched through a portion of the set of previous outcomes, the sequence $\underline{X}^{L_2}$ is quantised using 7 bits each for the two source symbols. On every occasion that a block is coded 17 bits are transmitted. The first three bits indicate the block length. If the block length is 64, 32, 16, 8 or 4, the next eight bits indicate the coordinate in the set of previous outcomes, where an approximation is to be found and the last six bits indicates the mean of the block in question. If the block length is two, the last fourteen bits indicate the quantised values of the symbols $\underline{X}^{L_1}$

b) Normalisation with respect to both mean and variance. The only difference between this and the strategy just descibed is that the blocks to be compared are normalised to have zero mean and unity variance. In addition $L_1=3$. On every occasion that a block is coded, 24 bits are transmitted. For blocks greater than L_1 in length, the first three bits indicates the block length. The next six bits indicate the block standard deviation, the next seven the block mean and the eight following these, the coordinate in the set of previous outcomes, where an approximation is to be found. If the block size is L_1, the next 21 bits are allocated, with seven bits each to the three symbols $\underline{X}^{L_1}$.

A flow chart for the schemes is given in figure 3.5. The distortion measure considered is the mean square error. The results for coding the speech files mentioned above are given in table 3.7. Figure 3.6 shows an example of the waveform distortion obtained by using the MPPCD scheme. The mean square error measure is possibly not the best error measure for speech coding. For this coding scheme, as described so far a single letter distortion measure is necessary and the mean square error measure is a convenient one to use. (It should be pointed out that the absolute error criterion, is also a convenient error measure to use. This is because for implementing the MPPCD scheme, with normalisation with respect to the mean, the whole coding scheme may be implemented without having to do any multiplications)

For comparison purposes, the same speech files are coded using the discrete cosine transform (DCT). The scheme used is an adaptive scheme similar to that described by Zelinski and Noll-(1977). The contents of this paper were explained in chapter 2, section 2.2.3.

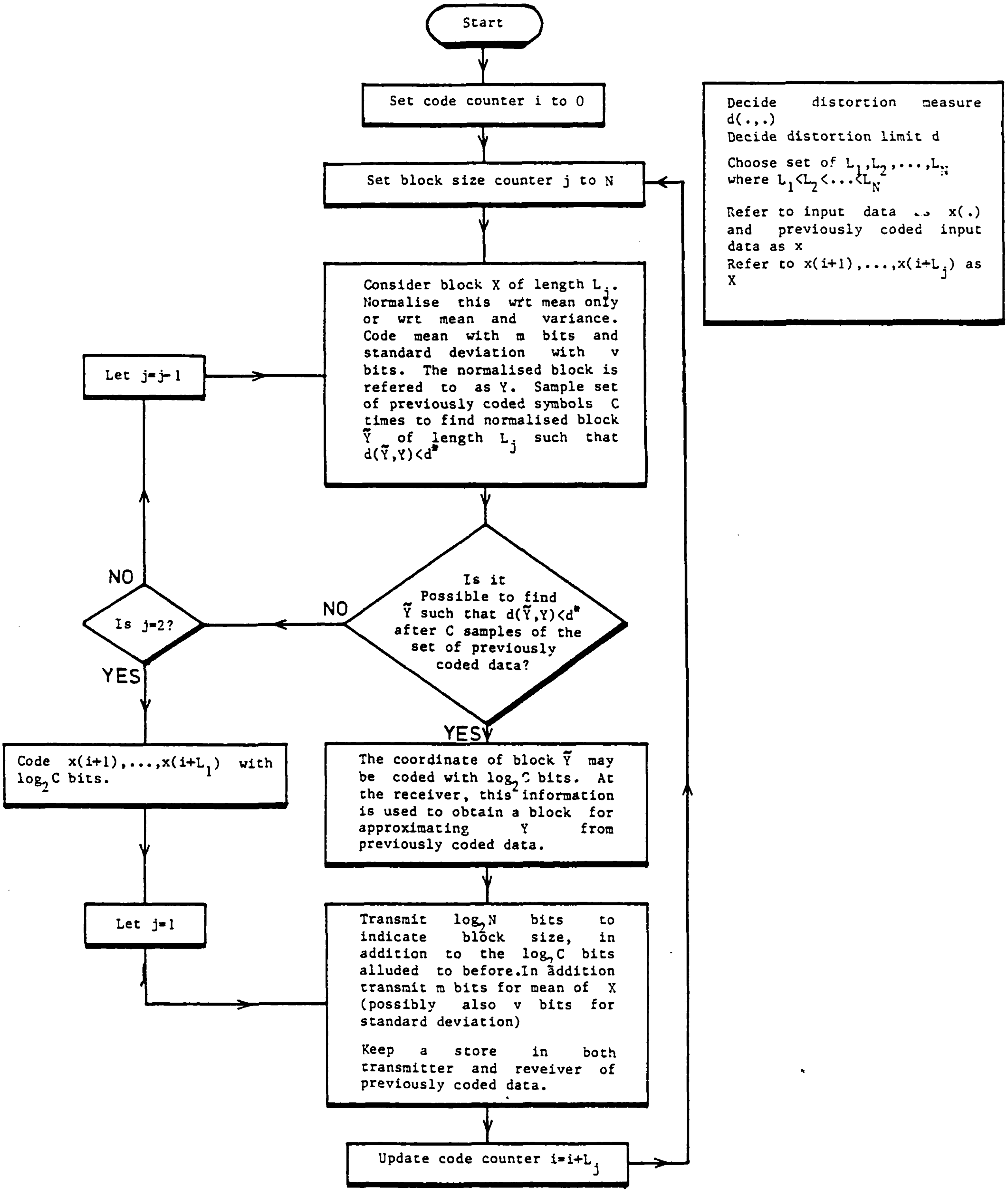

Start
Set code counter i to 0
Set block size counter j to N
Consider block X of length L_j. Normalise this wrt mean only or wrt mean and variance. Code mean with m bits and standard deviation with v bits. The normalised block is refered to as Y. Sample set of previously coded symbols C times to find normalised block $\tilde{Y}$ of length L_j such that $d(\tilde{Y},Y)<d^*$
Let j=j-1
NO
Is j=2?
NO
Is it Possible to find $\tilde{Y}$ such that $d(\tilde{Y},Y)<d^*$ after C samples of the set of previously coded data?
YES
YES
Code $x(i+1),\dots,x(i+L_1)$ with $\log_2 C$ bits.
The coordinate of block $\tilde{Y}$ may be coded with $\log_2 C$ bits. At the receiver, this information is used to obtain a block for approximating Y from previously coded data.
Let j=1
Transmit $\log_2 N$ bits to indicate block size, in addition to the $\log_2 C$ bits alluded to before.In addition transmit m bits for mean of X (possibly also v bits for standard deviation)
Keep a store in both transmitter and reveiver of previously coded data.
Update code counter $i=i+L_j$
Decide distortion measure d(.,.)
Decide distortion limit d
Choose set of $L_1,L_2,\dots,L_N$ where $L_1<L_2<\dots<L_N$
Refer to input data as x(.) and previously coded input data as x
Refer to $x(i+1),\dots,x(i+L_j)$ as X

Figure 3.5 Flow chart for MPPCD scheme applied tc speech coding with mean normalisation or mean and variance normalisation

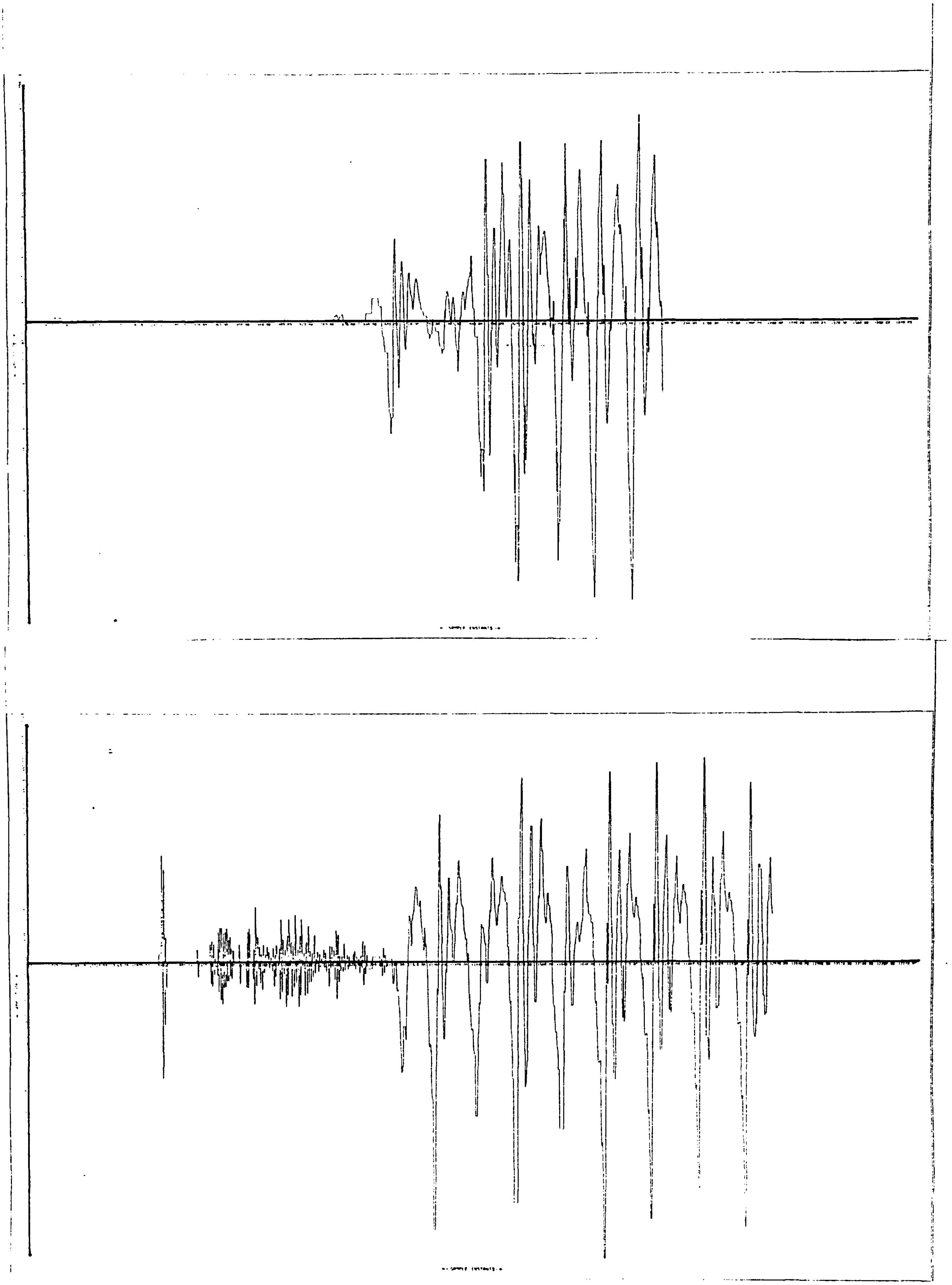


Figure 3.6. Plot of distorted waveform obtained by coding with MPPCD scheme.

Speech File	$L_1,L_2,L_3,L_4,$ L_5,L_6,L_7	Memory size	Mean bits	Variance bits	Bits/ block	Error limit	Rate	S/N ratio
SR8KK	32,16,12,8, 6,4,2	1024	6	0	19	0.2	11.0867kb/s	10.28dB
SR8KK	32,16,12,8, 6.4.2	1024	6	0	19	0.1	12.6016kb/s	12.613dB
SR8KK	32,16,12,8, 6.4.2	1024	6	0	19	0.05	14.464kb/s	15.405dB
SR8KK	32,16,12,8, 6.4.2	1024	6	0	19	0.025	15.984kb/s	17.56dB
SR8KK	/, /,48,24, 12,6,3	1024	7	7	27	0.1	9.058kb/s	10.57dB
SR8KK	/, /,48,24, 12,6,3	1024	7	7	27	0.05	11.257kb/s	13.228dB
SR8KK	/, /,48,24, 12,6,3	1024	7	7	27	0.025	13.876kb/s	15.65dB
KABITH	/, /,48,24, 12,6,3	1024	7	7	27	0.1	8.509kb/s	10.4105dB
KABITH	/, /,48,24, 12,6,3	1024	7	7	27	0.05	11.686kb/s	13.737dB
TABITH	/, /,48,24, 12,6,3	1024	7	7	27	0.1	9.353kb/s	10.4342dB
SR8KK	48,32,24,16, 12,6,3	512	6	6	24	0.1	8.143kb/s	10.64dB
SR8KK	48,32,24,16, 12,6,3	512	6	6	24	0.04	11.343kb/s	14.57dB
SR8KK	48,32,24,16, 12,6,3	512	6	6	24	0.025	13.0227kb/s	16.6dB
KABITH	48,32,24,16, 12,6,3	512	6	6	24	0.1	8.06kb/s	10.64dB
KABITH	48,32,24,16, 12,6,3	512	6	6	24	0.04	11.1917kb/s	14.588dB
KABITH	48,32,24,16, 12,6,3	512	6	6	24	0.025	12.976kb/s	16.394dB
TABITH	48,32,24,16, 12,6,3	512	6	6	24	0.1	8.5056kb/s	10.628dB
TABITH	48,32,24,16, 12,6,3	512	6	6	24	0.04	11.024kb/s	14.52dB
TABITH	48,32,24,16, 12,6,3	512	6	6	24	0.025	13.21kb/s	16.41dB

Table 3.8 Results of coding speech by the MPPCD scheme.

Speech file	Block size	Block period	Window	Rate	S/N ratio
SR8KK	128 non-overlapped	128	None	8kb/s	9.953dB
SR8KK	128 non-overlapped	128	None	12kb/s	12.946dB
SR8KK	128 non-overlapped	128	None	16kb/s	15.265dB
KABITH	128 non-overlapped	128	None	8kb/s	11.6618dB
KABITH	128 non-overlapped	128	None	12kb/s	11.6618dB
KABITH	128 non-overlapped	128	None	16kb/s	16.16dB
TABITH	128 non-overlapped	128	None	8kb/s	11.58dB
TABITH	128 non-overlapped	128	None	12kb/s	14.06dB
TABITH	128 non-overlapped	128	None	16kb/s	16.09dB
SR8KK	128 overlapping	96	Trapezoidal	8kb/s	9.556dB
SR8KK	128 overlapping	96	Trapezoidal	12kb/s	11.89dB
SR8KK	128 overlapping	96	Trapezoidal	16kb/s	14.35dB
KABITH	128 overlapping	96	Trapezoidal	8kb/s	10.865dB
KABITH	128 overlapping	96	Trapezoidal	12kb/s	13.11dB
KABITH	128 overlapping	96	Trapezoidal	16kb/s	14.625dB
TABITH	128 overlapping	96	Trapezoidal	8kb/s	10.45dB
TABITH	128 overlapping	96	Trapezoidal	12kb/s	12.98dB
TABITH	128 overlapping	96	Trapezoidal	16kb/s	14.78dB
SR8KK	128 overlapping	96	Tukey	8kb/s	9.166dB
SR8KK	128 overlapping	96	Tukey	12kb/s	11.65dB
SR8KK	128 overlapping	96	Tukey	16kb/s	13.84dB
KABITH	128 overlapping	96	Tukey	8kb/s	10.51dB
KABITH	128 overlapping	96	Tukey	12kb/s	12.76dB
KABITH	128 overlapping	96	Tukey	16kb/s	14.1dB
TABITH	128 overlapping	96	Tukey	8kb/s	10.134dB
TABITH	128 overlapping	96	Tukey	12kb/s	12.3486dB
TABITH	128 overlapping	96	Tukey	16kb/s	14.055dB

Table 3.9 Results of speech coding, employing adaptive Discrete Cosine Transform (Adaptation method by Zelinzki and Noll-(1977))

Decription of the windows:

Trapezoidal: The data is multiplied by a weighting function w(n) where

$w(n)=1;\ 33 \le n \le 96$

$w(n)=(n-0.5)/32;\ 1 \le n \le 32$

$w(n)=(128.5-n)/32;\ 97 \le n \le 128$

Tukey: The data is multiplied by function w(n) where

$w(n)=1;\ 33 \le n \le 96$

$w(n)=0.5[1-\cos\{(n-0.5)\pi/32\}];\ 1 \le n \le 32$

$w(n)=0.5[1-\cos\{(128.5-n)\pi/32\}];\ 97 \le n \le 128$

Windowed blocks of 128 data symbols are overlapped so that the last 32 symbols of a block overlap with the first 32 points of the next block.

We believe that this scheme is one of the best in terms of signal to noise ratio, of the well known waveform coding schemes (also noted by Fehn and Noll-(1982) section VII). It is therefore instructive to compare the MPPCD scheme's results with those attained by this. The transform coding results are given in table 3.8.

3.4.1 Discussion.

The results of the coding scheme in terms of signal to noise ratio are encouraging, showing lower distortion for the same rate compared with the DCT transform coding results. This is contrary to the findings of section 3.3.2 for the following reasons.

1) The normalisation with respect to the mean and variance allows previous blocks to be employed which would have been considered otherwise unsuitable for coding a present block. This consequently allows larger blocks to be coded using previous blocks with the drawback that some extra bits are required to code the mean and variance.

2) Some aspects in the character of speech, particularly its semi-periodic nature for voiced segments, means that the chances of finding a block with a similar shape, located one pitch period in the past, is large. This property is not exploited by the method of Zelinski and Noll for the transform coding of speech.

3) During silent blocks, due to the distortion measure used, the coding of large blocks is facilitated.

Listening tests showed the following properties of the coding scheme.

At all rates, the resulting speech was clearly noisy. The noise is easily perceived as a roughness of the coded speech. Comparison with transform coded speech indicates the following preferences. For rates greater than 12kb/s the transform coded speech is prefered, although transform coding achieved worse signal to noise ratio values. The results of the MPPCD scheme were distinctly noisier, although as expected the higher frequencies are better preserved. The transform coding results for non-overlapping blocks were considerably less preferable to those with overlapping blocks.

Interestingly the transform coding scheme performs poorly at 8kb/s, the speech has a burbly character, at times reminding one of birdsong. This is also noticable at 12kb/s, but whereas at this rate the result is acceptable, it is not so at 8kb/s. For the MPPCD scheme, despite the large improvements in signal to noise ratio with increasing bit rate there is surprisingly little reduction in perceived noise.

For lower bit rates, that is below 12kb/s, the MPPCD scheme is preferable to transform coding by the method of Zelinski and Noll. At high bit rates the MPPCD scheme as described thus far is not preferable.

3.5 Application to Image coding

In this section we present the results obtained when coding images using the MPPCD scheme. Two 128 by 128 images are used for all tests conducted. These are a portrait image and a picture of a telephone box; they are refered to as AFTAB and TELEBOX. These two images are shown in figure 3.7. They represent two types of image. AFTAB is an easy image to code, it contains relatively few features and edges. TELEBOX is a more difficult image to code, having more features and a lot of edges.

The coding scheme employed is exactly the same as that used for speech coding as described in section 3.4. No use is made of the 2-dimensional nature of the data. The input sequence considered is a 1-dimensional stream of symbols as would be generated by the line by line scanning of an image. Each line is scanned from left to right. Coded blocks are normalized with respect to the mean as described in section 3.4, with block lengths $L_1=2$, $L_2=4$, $L_3=6$, $L_4=8$, $L_5=12$, $L_6=16$ and $L_7=32$.

The coding results are shown in figure 3.8. For comparison purposes, these images are also coded using a non-adaptive discrete cosine transform coding algorithm. The bit allocation scheme for each frequency component is similar to that described in section 2.2.3. The results are shown in figure 3.9.

3.5.1 Discussion

Very satisfactory coding results are obtained using the MPPCD scheme with the image AFTAB. Although the signal to mean square

128x128 AFTAB

128x128 TELEBOX

Figure 3.7 Original pictures:- uncoded.
Displayed with 30 level grey scale resolution.

Rate = 1.327 bits/pix.
S/N = 29.002 dB
mem sz = 1024

(a)

Rate = 1.203 bits/pix.
S/N = 26.203 dB
mem sz = 1024

(b)

Rate = 1.07 bits/pix.
S/N = 23.232 dB
mem sz = 1024

(c)

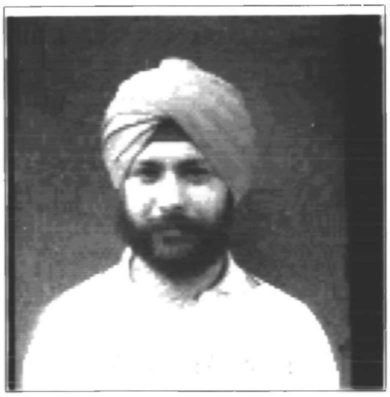
Rate = 1.69 bits/pix.
S/N = 25.33 dB
mem sz = 1024

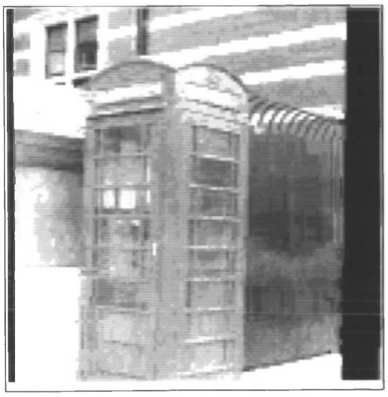
(d)

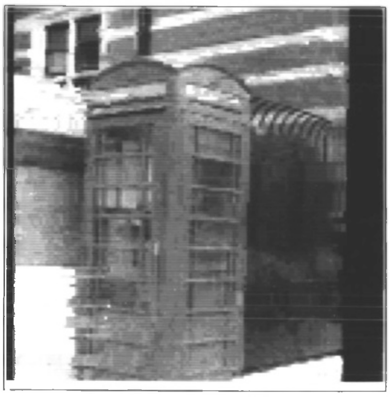
Rate = 1.465 bits/pix.
S/N = 22.859 dB
mem sz = 1024

(e)

Figure 3.8 Results of image coding via straightforward MPPCD scheme. Blocks are normalised with respect to the mean.







Rate = 2.00 bits/pix.
S/N = 30.88 dB
total = 32768 bits

(a)

Rate = 2.00 bits/pix.
S/N = 29.61 dB
total = 32768 bits

(b)

Rate = 1.50 bits/pix.
S/N = 29.48 dB
total = 24576 bits

(c)

Rate = 1.50 bits/pix.
S/N = 27.97 dB
total = 24576 bits

(d)

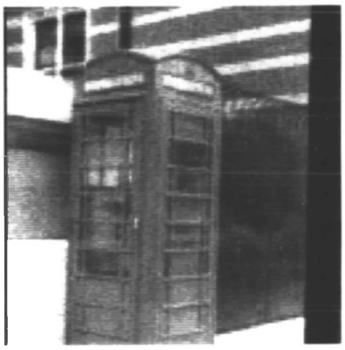
Rate = 1.00 bits/pix.
S/N = 27.83 dB
total = 16384 bits

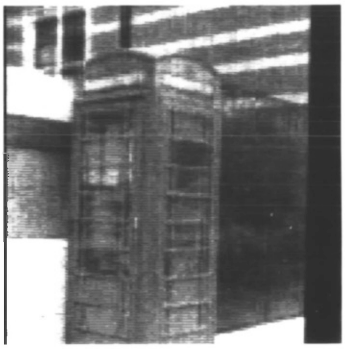
(e)

Rate = 1.00 bits/pix.
S/N = 25.70 dB
total = 16384 bits

(f)

Figure 3.9 Results for DCT transform coding of images. Block sizes are 16x16. The scheme is non-adaptive and 6x16x16 bits are sent initially, to indicate the standard deviation for each frequency pixel. This is used at the receiver to evaluate bits allocated to each frequency pixel.





noise ratio achieved is not as impressive as that obtained using the Discrete Cosine Transform, the results are subjectively just as good. The drawback of the method, in contrast to DCT coding, is that it produces images with jagged edges, where the transform method smoothes these. Detail however, is reasonably well preserved, as may noticed with the fencing at the middle right-hand-side of the picture TELEBOX.

3.5.2 Consideration of Image properties.

3.5.2.1 A description of images

Images are coded for broadly speaking two purposes, these are the following:

a) The storage or transmission of images for informal human use. For example, the transmission or storage of pictures for entertainment or the transmisson of images for conferencing.

b) The storage or transmission of images for formal use. By this, it is meant that some rather important information is to be derived from the image, either by humans or by machines. For example the storage or transmission of X-ray images or remotely sensed data. An example, of an occasion when a machine will use coded data is the storage of templates for the automatic interpretation of pictures by robots.

For each application, the distortion measure employed in coding an image should be different, being designed so as to introduce little or no distortion of the features which are to be extracted and employed by the user of the image.

All the work presented here presume that coding is for application "a" above, distortion is undesirable, but may be tolerated. All we are concerned with is that the picture "looks" reasonable, not terribly different from the original. In this case a knowledge of the human visual system may be useful when choosing an error measure. We shall therefore briefly review some of the important properties of the visual system. Firstly, however the description of an image in language appropriate for our purposes will be given.

A digitised image is represented by a 2-dimensional block of numbers, $\{x_{ij}\}$. There is no obvious way in which this block of numbers $\{x_{ij}\}$ may be sequenced for serial transmission, serial processing or serial storage. A process that represents the 2-dimensional block of numbers as a 1-dimensional sequence of numbers, is called a "scanning" scheme. The user of image data chooses a scanning scheme to suit his needs.

An image is composed of features. For example an object or objects and a background (the background may also be considered as just another one of the objects of an image). The features are distinguished from each other, in a manner of speaking, by a uniformity of texture within the body of a feature. The objects are separated by boundaries. A boundary is characterised by a rapid change in average pixel value or area texture.

When looking at an image, the pixels are not seen as individual entities with different intensities. The tendency is to interpret the image as a set of objects separated by boundaries. A coding scheme, therefore should preserve the boundaries of an object and within this object, the texture should not be altered, changes in pixel values are allowed except that these changes should not destroy the visibility of the boundaries. In addition pixel values within the boundaries of an object, may be altered, provided that the texture as it is perceived, is unchanged.

The variation in intensity, which may be perceived, is dependent on the brightness of the region. This is tested by the ability to notice a spot of brightness $I+\Delta I$ in a background region of intensity I. It has been noticed that the value ΔI at which the spot is just noticable is proportional to I so that $\Delta I/I$ is almost constant for a region of I near the middle of the range of intensities perceived. The response is shown in figure 3.11. The constancy of $\Delta I/I$ is called Weber's law[Gonzalez and Wintz-(1977)]. At each end of the range of intensities, the value $\Delta I/I$ is larger. Thus when coding, one may allow more distortion in regions where the average intensity is high and also in regions where the average intensity is very low. In the middle of the range of intensities, less distortion or noise should be allowed.

The spatial frequencies which may be percieved are dependent upon the contrast of the signal (see Rozenfeld and Kac-(1982) p56). A distortion measure may be used which allows more low and high frequency noise, when the contrast associated with the texture within a region falls below a given threshold.

Some of the properties of the visual system have been used to design a distortion measure to be used with the MPPCD scheme. Before describing this, the following point about the MPPCD scheme ought to be discussed.

The scheme is based upon coding blocks of data of sizes L_1, $L_2,\dots,L_N$. Until now it has been presumed that these blocks were 1-dimensional. There is some difficulty with extending the concept to 2-dimensional blocks. Suppose one is at coordinate (i,j) in an image and one considers the largest 2-dimensional block size L_k. The block of this size is coded by finding a block of similar size in the set of previously coded data, such that these two blocks are similar to within some distortion constraint. The problem is that the 2-dimensional block sizes which may be considered should be

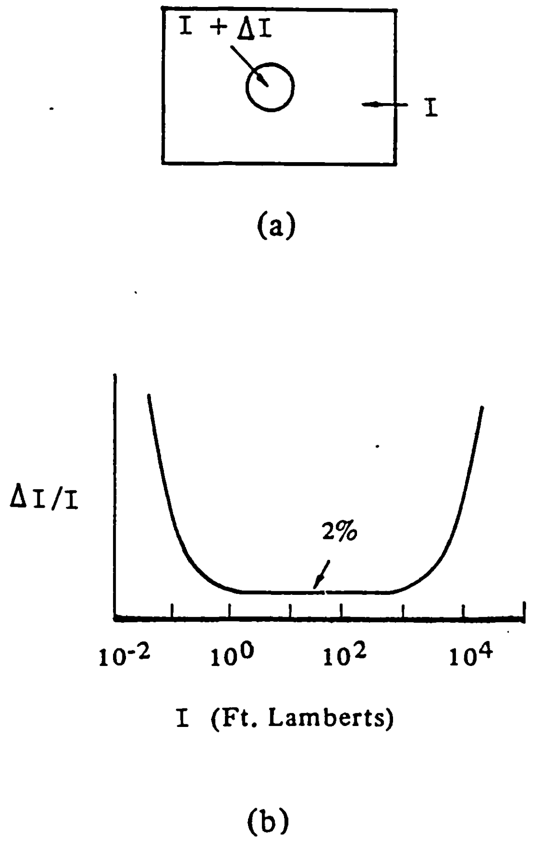


Figure 3.11 Weber's law
Contrast sensitivity with a constant background.

chosen so that the image may be thoroughly filled by these blocks of varying sizes.

The problem of defining a scheme and appropriate block sizes so as to be able to code 2-dimensional blocks, is deferred until later. In the mean time, the image is coded as if it is a one dimensional sequence. Previously, the scanning scheme had been that shown in figure 3.12a. This is the standard scanning scheme used for TV and was used because it was supposed to be the scanning scheme most likely to have been used if one is presented with a 1-dimensional sequence representing an image. There are better scanning schemes, these avoid the sharp discontinuities that result from going from the right side of a line to the left side of the next line.

The trajectory obtained in using a scanning scheme to represent the points of a larger dimensional space by a 1-dimensional sequence is called a "space filling curve". Two examples are given in figures 3.12b and 3.12c. The PEANO or HILBERT scan of figure 3.12c produces a 1-dimensional sequence with the following distinction. For any length of resulting one-dimensional sequence, the points of this sequence represent an N-dimensional space with the smallest maximum span in any direction. This scan in effect generates long 1-dimensional sequences associated with compact N-dimensional spaces. The advantage of this, as far as the MPPCD scheme is concerned, is this. A long 1-dimensional sequence to be coded, most likely corresponds to a 2-dimensional region as square as may be attained with any scanning scheme. The texture is thus more likely to be uniform over this sequence. The scan of

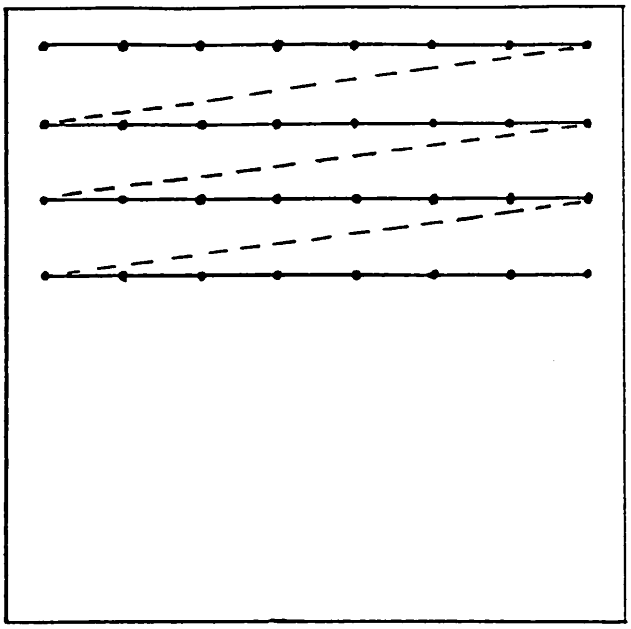


Figure 3.12a

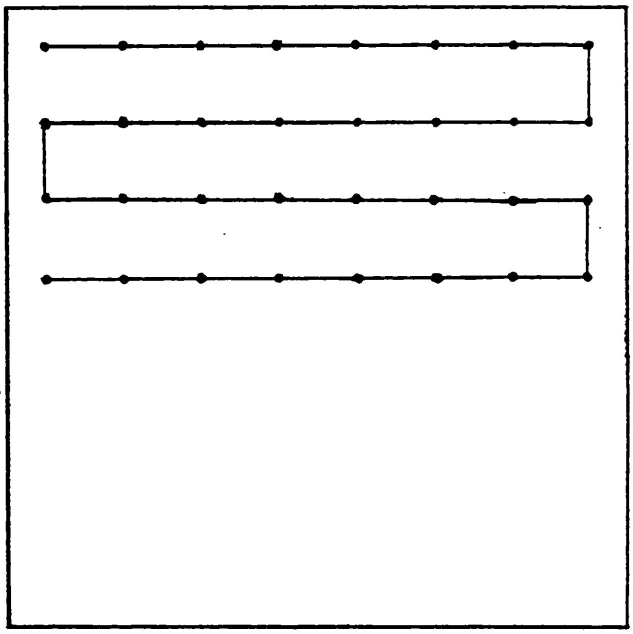


Figure 3.12b

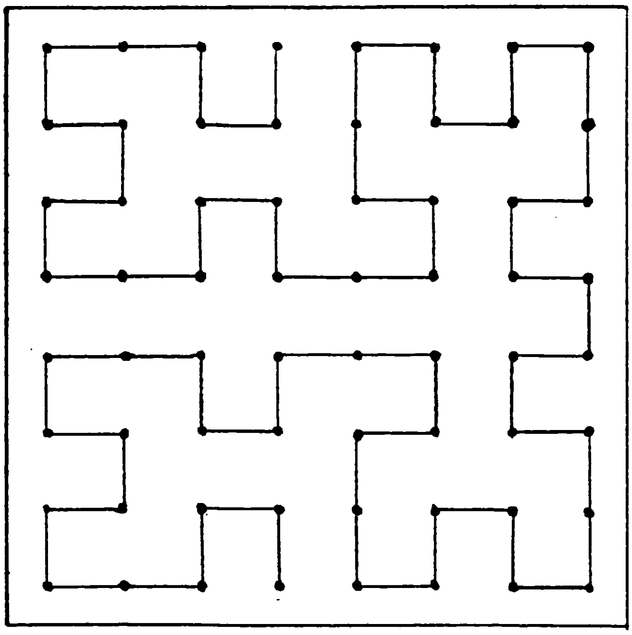


Figure 3.12c

Figure 3.12 Scanning schemes.

figure 3.12b however, was employed in the work reported hereinafter.

Next we consider error measures which may allow the allocation of different quantities of error to "edge" and "plane" regions of an image. To do this a gradient operator is applied to the one dimensional block of length L_k to be coded. The gradient at position i, g_i is approximated thus,

$$g_i = C.|x_{i+1} - x_i| \qquad 3.6$$

The error at i is then weighted by a function $f(g_i)$ at each point i. The results of an example where white noise is added to a row of an image are shown in figures 3.14a and 3.14b. To add more noise to edges, the noise added to the image signal is $e_i\sqrt{g_i}$. To add more noiśe to plane areas, the noise added to the image signal is $e_i \cdot 1/\sqrt{g_i+1}$. In each case e_i is a Gaussian noise source of zero mean. It was concluded that it is more desirable to have more noise in the region of edges. The effect is to distort the edge, however the edge appears to be observed as a region of fast variation in intensity and it seems to be relatively irrelevant how this fast variation in signal intensity is achieved. Thus distortion is more tolerable in edge regions.

The results of coding an image with the points just discussed considered, are shown in figure 3.15. When comparing a block $\underline{X}_i^{L_m}$ and a previously coded block $\underline{\tilde{X}}_j^{L_m}$: j<i, the gradient sequence $\underline{g}_i^{L_k}$ for the sequence $\underline{X}_i^{L_m}$ is computed. In ascertaining the error between the two sequences $\underline{\tilde{X}}_i^{L_m}$ and $\underline{X}_i^{L_m}$, the actual error at each point k is multiplied by

$$\frac{1}{\sqrt{g_k+1}}$$

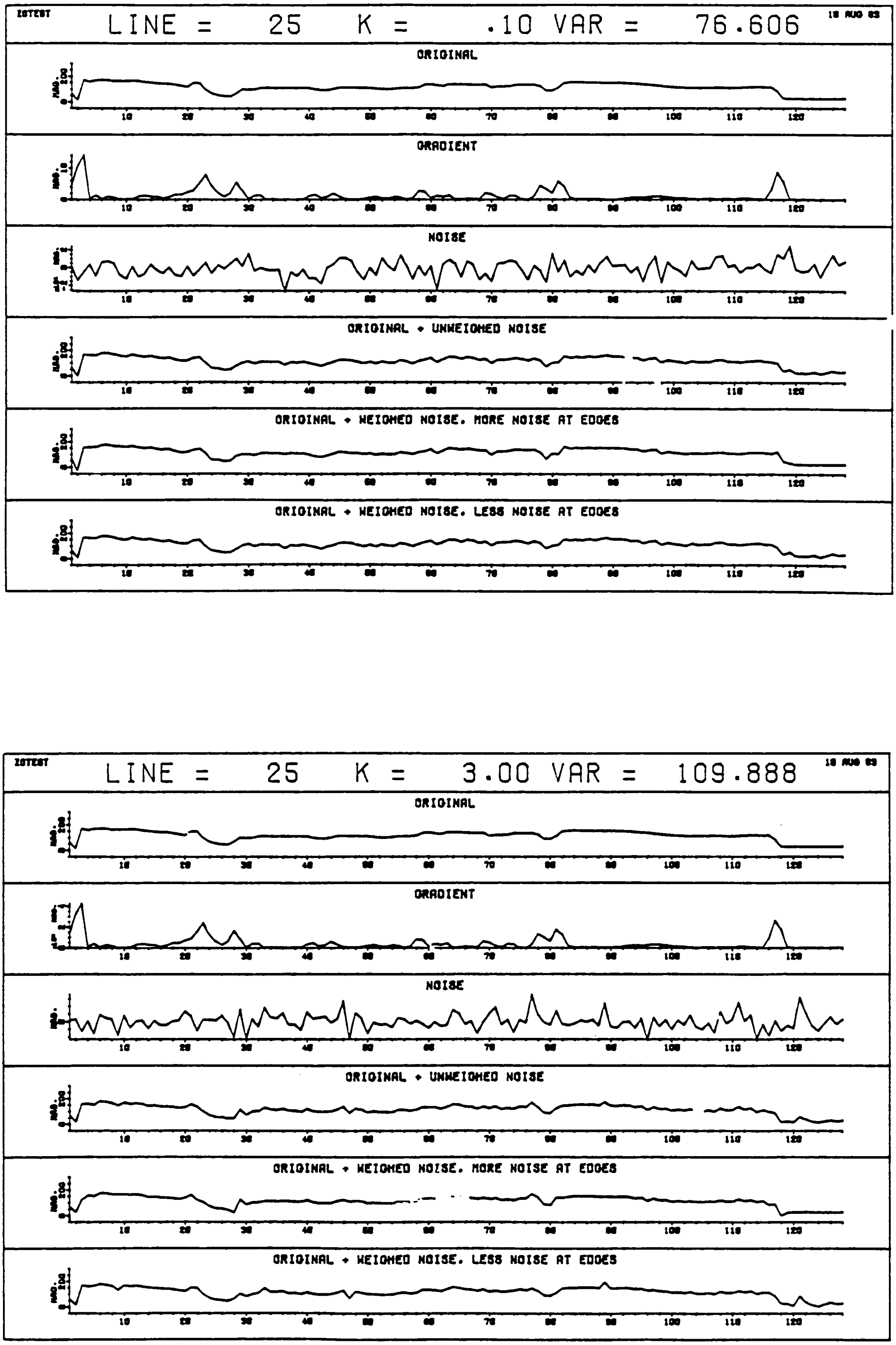


Figure 3.14 Results of adding noise to row of image.

Original (Low contrast)

(a)

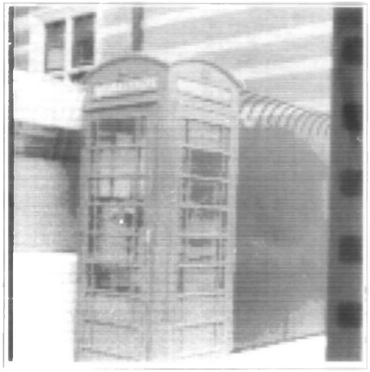
Rate = 1.728 bits/pix.
S/N = 26.433 dB

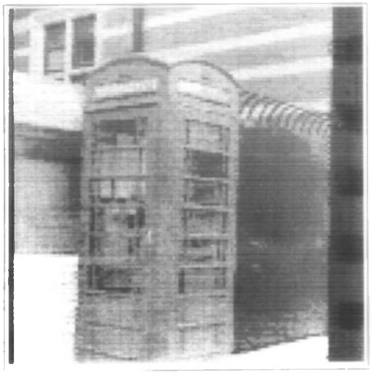
(b)

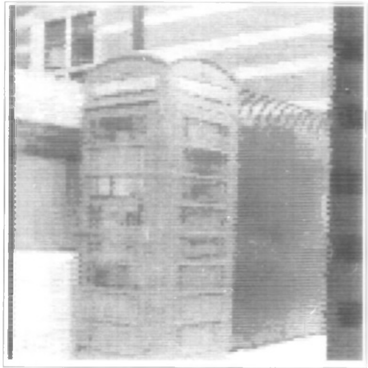
Rate = 1.18 bits/pix.
S/N = 22.59 dB

(c)

Figure 3.15 Results of image coding using 1-D processing (scanning scheme of figure 3.12b). Edge weighting is applied to force more noise at edges.







This ensures that the error observed at edge regions are weighted lower so that this is in effect tolerated.

In addition it should be mentioned that an attempt is made to incorporate Weber's law in designing the error measure, by dividing the error signal by the observed mean of the region before comparing this with the set distortion limit. At high overall values of intensity, more error is allowed.

3.5.3 Discussion of the results

A reduction in coding rate has been achieved, for a signal to noise ratio of about 23db when noise weighting is applied, for the image TELEBOX. (Compare figures 3.8e and 3.15c). Comparing the above two figures, it may be observed that edges have fewer instances of great distortion. This may be seen in the improvement of the bottom left hand edge of the telephone box in figure 3.15c. As a result of allowing more noise at edges however, all edges show some jaggedness. While figures 3.8d and 3.15b are about the same subjectively, figure 3.15c appears subjectively preferable to figure 3.8e.

3.5.4 Consideration of 2-dimensional blocks.

The MPPCD is implemented using 2-dimensional blocks of sizes 8 by 8, 8 by 4, 8 by 2, 8 by 1, 4 by 1 and 4 by 1 considered in order as shown in figure 3.16. This is in fact, almost like implementing the MPPCD scheme using 1-dimensional blocks. The difference is that now a single outcome is equivalent to a column of length 8. Thus we try effectively to code blocks of length 8, then 4, then 2, then 1. At "e" of figure 3.16 the situation changes somewhat; in order to consider smaller blocks we require a column of length smaller than 8. In this case we next try to code a 4 by 1 block in the top left hand corner. When coding of this block is unsuccessful, these 4 symbols are transmitted using 34 bits. 7 bit each for the 4 symbols and 6 bits to indicate the block size. These 34 bits are transmitted in 2 groups of 17 bits each. Each group has 14 bits to represent two pixels and 3 bits to represent the block size. If the top left hand side block of four are representable by previously coded symbols, then 17 bits are transmitted. 8 bits are used to indicate the coordinate of the approximate previous outcome, 6 bits are used to code the mean of the present block and 3 bits are used to code the block size. Whatever the outcome of trying to represent the top left hand 4 by 1 block, the next set of symbols to be transmitted is the bottom left hand 4 by 1 block. If the attempt to represent this by previously coded symbols is unsuccessful, 34 bits are sent, directly coding the set of 4 symbol described before. If approximation by previously coded symbols is possible, 17 bits are sent. The coding scheme proceeds in all other ways as described in previous sections.

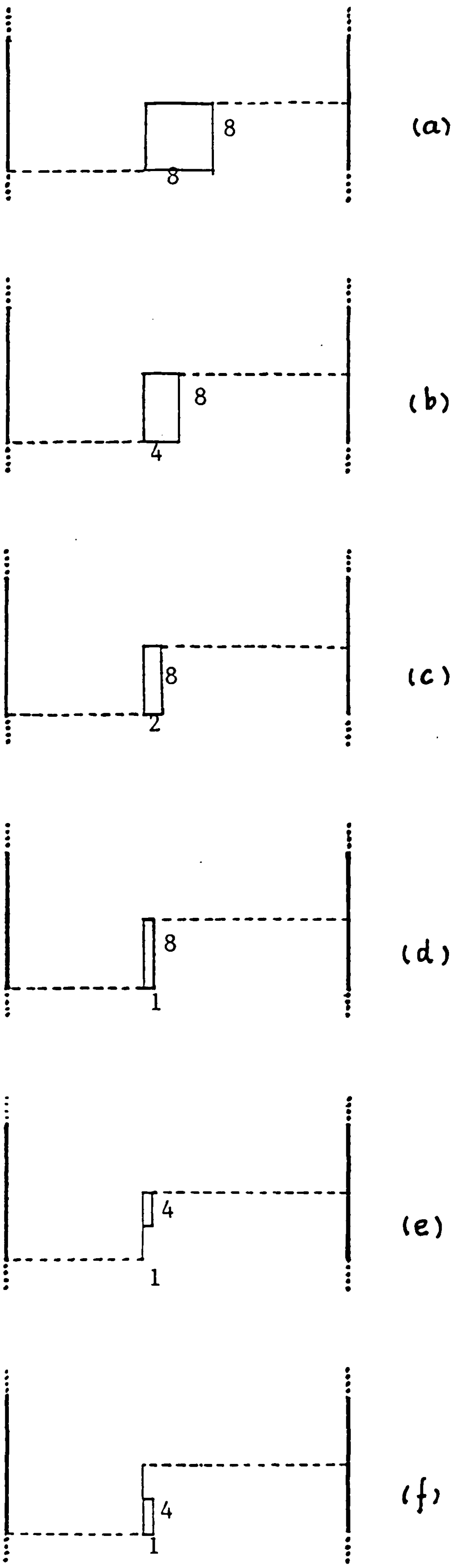


Figure 3.16 Block types for pseudo-2-D coding using MPPCD scheme.

Original (Low contrast)

(a)

Rate = 1.255 bits/pix
S/N = 28.127dB
mem sz = 512

(b)

Rate = 1.037 bits/pix.
S/N = 24.977 dB
mem sz = 512

(c)

Rate = 0.807 bits/pix.
S/N = 22.712 dB
mem sz = 512

(d)

Figure 3.17 (a-d) Image coding using MPPCD scheme. Blocks are rectangular. No edge weighting is employed.

Original (Low contrast)

(e)

Rate = 2.056 bits/pix.
S/N = 29.347 dB

(f)

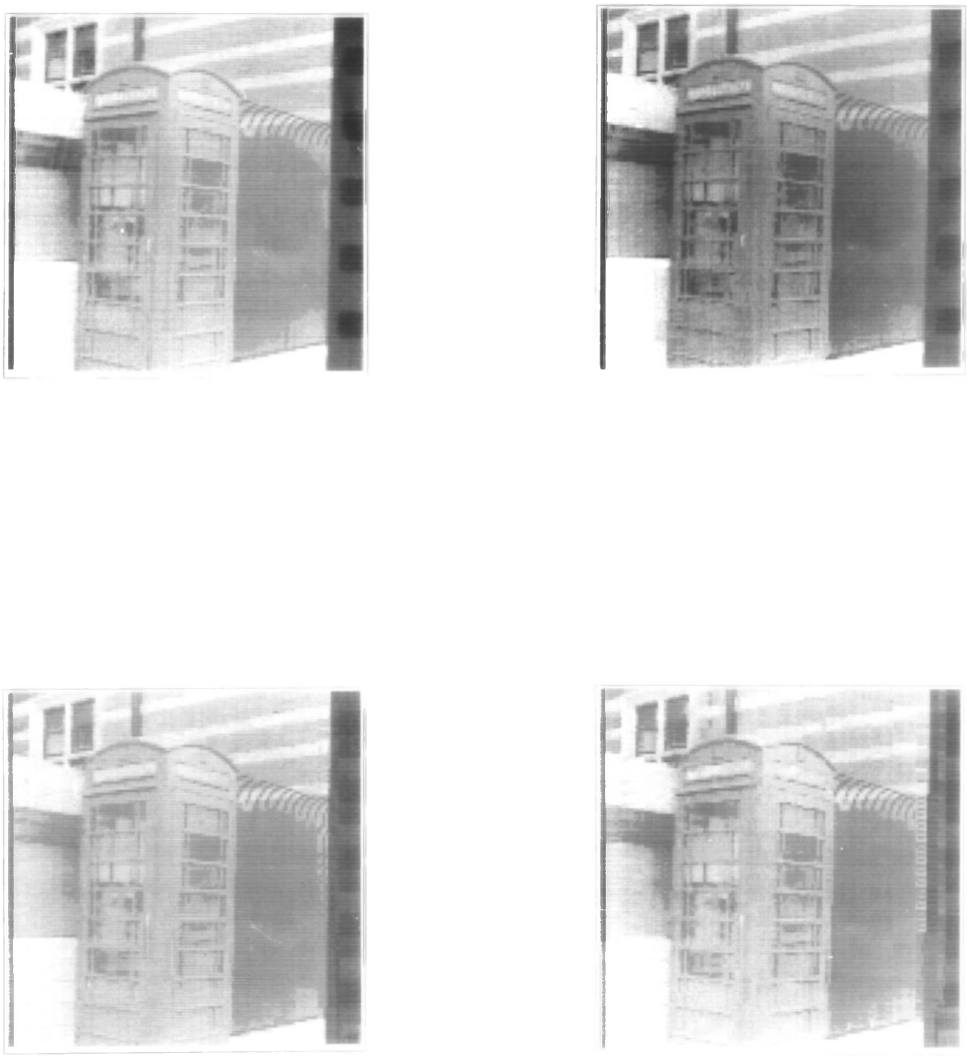
Rate = 1.75 bits/pix.
S/N = 25.25 dB

(g)

Rate = 1.383 bits/pix.
S/N = 21.46 dB

(h)

Figure 3.17 (e-h) Image coding using MPPCD scheme. Blocks are rectangular No edge weighting is applied.



Rate = 1.144 bits/pix.
S/N = 30.085 dB

(i)

Rate = 0.977 bits/pix.
S/N = 27.734 dB

(j)

Figure 3.17i,j Results of image coding with MPPCD scheme. Rectangular blocks are used. When checking the set of previously coded data, the best block is found, and if the associated error satisfies the distortion constraint, this is used to approximate the present block.

The error signal between blocks is weighted so that different quantities of error are tolerated at edges and plane areas. This requires the implementation of an edge detector. An edge detector is an operator which returns a large output signal when it is centered on an edge or boundary. There are numerous papers on the topic of the design of edge detectors; we shall not attempt to add to this field. In the following a very brief review of the types of edge detectors available is presented and the one which we elected to use and the reasons for this, given.

Well known simple edge detectors are the Roberts, Zobel and Prewitt operators. These are difference edge detectors which have two components Δx and Δy. These components return high outputs for edges in orthogonal directions. The signal presented by the overall operator is then

$$\sqrt{\Delta x^2 + \Delta y^2} \quad \text{or} \quad |\Delta x| + |\Delta y| \quad \text{or} \quad \max\left(|\Delta x|, |\Delta y|\right)$$

A detector is termed "isotropic" if its output is invariant with the angle of orientation of an edge. The two components of the edge detectors for the Roberts operator are

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

For the Prewitt operator these are

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

and for the Zobel

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

Other operators, "template match operators" have been reported in the literature. These are designed to detect edges in one compass direction only. The combination of these template match operators, each of which is designed for one of several directions, allows one to effect isotropic edge detection.

Another type of edge detector is the Laplacian. This is an approximation to the 2nd diferential of a signal. The impulse response of the Laplacian is

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & -4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

This operator gives a zero output when the signal is a ramp. These edge detectors are described by Rozenberg and Kac in section 10.2.1 of their book [Rozenfeld and Kac-(1980)], by Gonzalez and Wintz-(1977) in section 4.4 and 7.1.2.1. A thorough comparison of these edge detectors was reported by Abdou and Pratt-(1977). The last reference shows that the 3 by 3 operators perform better than the 2 by 2 (Roberts) operator and there is very little to choose between the Prewitt and Zobel operators.

The detectors mentioned are only able to detect boundaries characterised by abrupt changes in pixel intensity. Complex detectors are required if general boundaries between two areas of different texture are to be detected and if the operation of the edge detector is to be reliable when the image region under consideration is noisy. Examples of these are the Heuckel and Rozenfeld edge detectors. These were compared thoroughly by Fram and Deutsch-(1973).

The latter operators are not simple to implement, and since the main thrust of the work reported here is concerned with coding, it was decided use the Zobel operator.

The MPPCD scheme was tried using 2D blocks and Zobel filtering to detect edges. A block $\underline{X}_{ij}^{L}$ to be coded, has a gradient block $\underline{G}_{ij}^{L}$ constructed for this. In comparing $\underline{X}_{ij}^{L}$ with a previously coded block, the error at each pixel value is multiplied by

$$\frac{1}{\sqrt{1+g_{kl}}}$$

Where g_{kl} is a member of the block $\underline{G}_{ij}^{L}$. Thus edge points (k,l) with large gradient values g_{kl} have their errors weighted so that errors at these points are tolerated. The results are shown in figure 3.18.

3.5.4.1 Discussion

With the consideration of 2-dimensional blocks, similar results to those obtained using one dimensional blocks, have been obtained using a smaller memory size. It is difficult to say that the use of edge weighting results in a subjectively improved image. The edges appear more ragged, although looking at the plane area, at the top right-hand-side of the picture TELEBOX, one observes a smoother region, compared with the non-edge weighted coding result. It is expected that a thorough study of which mapping between the edge business factor returned by the Zobel filter and the error weighting factor will yield useful results.

Improvements in signal to noise ratio may be made if, in sampling the set of previously coded data, the first block to

Original (Low contrast)

(a)

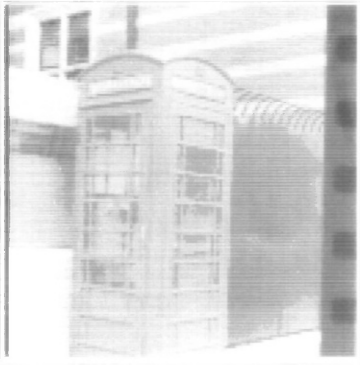
Rate = 1.239 bits/pix.
S/N = 24.99 dB

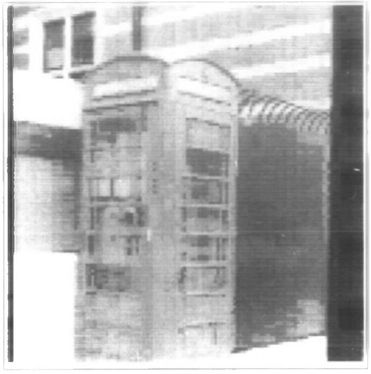
(b)

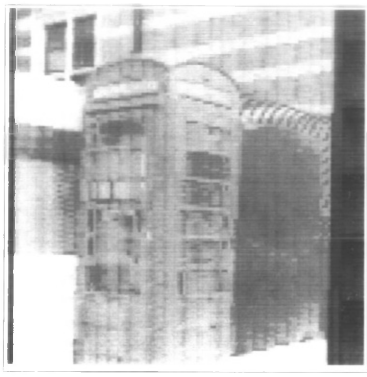
Rate = 0.989 bits/pix.
S/N = 22.88 dB

(c)

Figure 3.18 Image coding via MPPCD scheme with 2-D blocks. Edges are weighted.







satisfy the distortion constraint is not chosen automatically. Instead, the best of the previous blocks is chosen, if it satisfies the distortion constraint. The results of doing this are shown in figures 3.17i and 3.17j, where improved S/N values were obtained.

3.6 Consideration of speech properties.

The task of coding speech may be approached in two ways. The first relies upon trying to approximate the speech waveform as closely as possible. The second relies upon extracting the important features of speech, coding and sending these. In either case it is useful to understand the particular properties of speech, so that we may take advantage of these. For the second approach, it is necessary that the speech generation and perception processes are studied in great detail. In appendix 2 therefore, the rudiments of speech production and hearing are described. In this section, vocoders, speech coders which make use of the properties of speech, are described. The results of experiments concerning the application of the MPPCD scheme in the LPC vocoder, are presented.

3.6.1 The vocoder

A vocoder is a speech coder which relies upon the parameterisation of the short term spectrum of speech. The vocoder was first implemented by Dudly [Dudly-(1940)]. It is supposed that for the comprehension of speech, the character of the phase spectrum was unimportant. The power spectral characteristics of blocks of speech are ascertained, the parameters representing these are then coded for transmission. These parameters are used at the receiver to reconstruct a version of the original speech. The following details the different types of vocoder.

3.6.1.1 The channel vocoder

This employs a bank of bandpass filters for spectrum analysis. Envelope detection is done to evaluate the magnitude of each frequency component. The magnitude values are sent to the receiver periodically. In addition, information indicating the short term signal variance and whether the block in question is voiced or unvoiced, is transmitted.

3.6.1.2 The formant vocoder

This relies upon modelling the vocal tract as an inductor-capacitor (LC) filter network. The component values for this network are evaluated regularly, the pole frequencies of the resulting transfer function are approximated and transmitted. Generally the first 3 or 4 formant frequencies or pole frequencies are sufficient to characterise the phoneme. [Ainsworth-(1976), Chapters 6 and 7] The Q factor or bandwidth associated with each pole is also transmitted.

3.6.1.3 The LPC vocoder,

Spectrum analysis, in this vocoder, is done by modelling the vocal tract by a filter network. The filter coefficients (or alternative parameters) for this filter are extracted periodically. These are coded and transmitted.

For all the above vocoders, the speech is analysed to ascertain whether or not it is voiced. When it is voiced, the speech synthesiser, be it filter bank or just one filter, is excited with a sequence of pulses. The pitch is extracted at the transmitter and is used at the receiver to generate a sequence of pulses with an appropriate period.

The LPC vocoder will be described in more detail because the remainder of this chapter deals with the use of the MPPCD scheme with an LPC vocoder.

For LPC analysis, the vocal tract is modelled as several coupled tubes of different lengths and crossectional areas. The speech perceived at the mouth is modelled as the sound observed at the open end of this set of tubes, when a point source generates some excitation at some position in the set of tubes. When a non-nasal phoneme is spoken the speech generation process may be modeled as simply a sequence of tubes with the excitation end closed. The vocal tract then has an electrical analogue which is a lossy LC ladder network, whose transfer function is all-pole [Fant-(1960)] When the vowel is nasal, the resulting electrical analogue for the vocal tract is a filter whose transfer function is no longer all-pole. Similarly transfer functions for the electrical analogue associated with fricative consonants have zeros in addition to poles [Heinz and Stevens-(1961)]. The LPC vocoder however, models the transfer function of the vocal tract as simply an all-pole filter. This is done for the following reasons.

1) Voiced sounds are well modelled by all pole transfer functions.
2) A large quantity of zeros are required to make any improvement in the sound quality achieved by the all-pole model.
3) Most importantly, the parameters of the all-pole filter model may

be evaluated in a straight-forward manner.

Improvements to the LPC vocoder have been in two directions,
1) Reduction in coding rate by assigning as few bits to the coding of the filter parameters, pitch and gain as possible.
2) Efforts to increase the subjective quality of the coded speech.

We shall first discuss the former. A typical LPC vocoder will analyse speech blocks of duration 10msecs to 30msecs (for a sampling rate of 8kHz, this means blocks of length 80 to 240 samples). A twelfth order filter might be used for the frequency analysis of the block under consideration. Each of the Log-Area-Ratios might be quantised for representation with 8 bits. 8 bits might be used to code the pitch, 7 bits to code the gain and 1 bit the voice/unvoiced decision. Thus 112 bits might be sent per block. The coding rate is between 11.2kbits/sec and 3.73kbits/sec. The following is a list of the reported methods used to reduce the rate.

a) Reduction in the number of filter coefficients: The effects of doing this were reported by Atal and Hanauer-(1971).

b) Coarser quantisation of the Log-Area-Ratios (or some other parameters eg. reflection coefficients or inverse sine of reflection coefficients). A.H. Gray and J.D. Markel-(1976) discuss in great detail, the spectral variation due to the quantisation of the various reflection coefficients. Their work allows a cookbook type design procedure for bit allocation for the LPC parameters.

c) The linear predictive or DPCM coding of the LPC parameters, from block to block was studied by Sumbar-(1975)

d) The linear transformation of the LPC filter parameters, allows the transform coding of these parameters. Sumbar-(1975) and

Fussel-(1980) studied the effects of bit rate reduction by the use of the Karhunen-Loeve transform.

e) Vector quantisation of the filter parameters (or the spectral envelope associated with these parameters). Vector quantisation involves finding a small number of spectral shapes which are deemed representative of the set of spectral patterns generated in speech. Vector quantisation has been studied by Ahmadi-(1980), Linde, Buzo and Gray-(1980), Buzo, A.H. Gray, R.M. Gray and Markel-(1980a) and-(1980b), Gray, Gray, Robodello and Shore-(1981), Abut, Gray and Robodello-(1982) and Wilson-(1983). Briefly, the following is done for vector quantisation. A long sequence of typical speech, refered to as a training sequence is analysed. Blocks are clustered into a relatively small number of spectral patterns. These are stored in identical libraries, at both the transmitter and receiver. In implementing the coder, a block is analysed and the LPC filter parameters extracted. The spectrum is then evaluated from these filter coefficients. The spectrum is compared with the members of the library of filter coefficients or spectral patterns. The library member whose spectrum is closest to the spectrum of the block being considered has its coordinate transmitted. This allows very considerable rate reduction; for a library of 256 members, only 8 bits are used to represent the spectrum of a block instead of the 96 that might be needed in a conventional LPC scheme.

f) The variable rate transmission of the parameters for the coding allows some bit rate reduction. This is contrived by not transmitting the filter parameters for every block of speech. The MPPCD scheme may be used to achieve bit rate reduction in this manner. The idea of effecting some bit rate reduction by not transmitting a set of parameters for each block of speech, was introduced by Magill-(1973). This method of increasing the efficiency of the LPC vocoder has received little attention, some of the few papers on this are by Magill-(1973), Viswanathan, Makhoul, Shwartz and Higgins-(1982). These authors reported schemes where, for each block LPC analysis is undertaken. The spectrum resulting from the evaluated LPC filter is compared with the spectrum of the previous block of speech. If these two spectra, the filter coefficients, the Log-Area-Ratios or some other characterisation parameters are close to within some distortion criterion, the parameters of the present block are not sent. At the receiver, these are approximated by those of the previous block. This is very much like the run length encoding of the block parameters or spectra. Papamichalis and Barnwell-III-(1980 and 1983) have expanded this to include the alternative of sending only some of the parameters per block, in addition to sending of all or none of these. The decision concerning what to send is likened to a branch in a tree and a dynamic programming algorithm was proposed to search for the best path in this tree.

3.6.2 The MPPCD scheme and the LPC vocoder

The MPPCD scheme is applied in this manner. Decide on a memory size, M=128 or 256 say. For every block, LPC analysis is undertaken, 12 filter coefficients are evaluated. The spectrum assocaited with this block is then evaluated at 64 frequency points from 0 to π rads/s. (64 is chosen because this results in a spectral resolution of 62.5Hz. A coarser resolution is inadvisable since it is known experimentally that people are able to identify the effects of a 60Hz variation and a 120Hz variation in the first and second formant positions respectively. ([Ainsworth-(1976)] chapter 6) This spectral pattern is compared with the spectral patterns of some of the previously encoded blocks.

The comparison is conducted in this manner. It is presumed that each of the 128 or 256 memory locations contains a set of filter coefficients. Alternativelty, if memory is easily available, 128 or 256 sets of frequency patterns may also be stored in the memory locations. The contents of the memory locations are associated with previously encoded blocks. We shall refer to these memory locations as a library. If the spectral pattern of a block is sufficiently close to that of a member of this library, the coordinates of this member is transmitted. At the receiver, the parameters of this block are approximated by those of the relevant library member. If no library member is similar to the block under test, to within a preset distortion limit, a member of the library is removed. The spectral pattern for the block under test or its filter parameters are included in the library; these are also transmitted to the receiver. The choice of the library member removed when a new addition is made to this, is a subject for further research. For the results presented here, the earliest library member is removed. The members of the library are numbered according to their "age".

3.6.2.1 What distortion measure?

Several distortion measures for the matching of parameters between different blocks, have been proposed in the literature for speech processing. Most are spectral distortion measures or derived from spectral distance measures. This is because the human ear appears to do some spectral analysis, and thus a frequency deviation measure would seem a good one. Below is a list of the well known spectral measures reported in the literature, their advantages and their disadvantages.

1) Log spectral measures:

$$D = (\sum_{\omega_i} |\ln\frac{\hat{f}(\omega_i)}{f(\omega_i)}|^p)^{\frac{1}{p}} \qquad 3.7$$

These require log computations. The normalised versions of the above are;

$$D = \min_{\lambda}(\sum_{\omega_i} |\ln\frac{\hat{f}(\omega_i)}{\lambda f(\omega_i)}|^p)^{\frac{1}{p}} \qquad 3.8$$

For p=1, ln(λ)=median of sequence ln($\hat{f}(\omega_i)/f(\omega_i)$)=u(ω_i) say. For p=2, ln(λ)= mean of sequence u(ω_i)

The latter is easier to calculate, making it more attractive, despite the fact that the square of the quantity u(ω_i) is computed.

2) Itakura-Saito distortion(I-T) measure

$$D = \min_{\lambda} \sum_{\omega_i} \{\frac{\hat{f}(\omega_i)}{\lambda f(\omega_i)} - \ln|\frac{\hat{f}(\omega_i)}{\lambda f(\omega_i)}| - 1\} \qquad 3.9$$

This measure is non-symmetric.

3) The Cosh measure: This is derived from the Itakura-Saito measure, and is a symmetrical version of that.

$$D = \frac{1}{2} \sum_{i}\left(\frac{\hat{f}(\omega_i)}{f(\omega_i)} - \ln|\frac{\hat{f}(\omega_i)}{f(\omega_i)}| - 1 + \frac{f(\omega_i)}{\hat{f}(\omega_i)} - \ln|\frac{f(\omega_i)}{\hat{f}(\omega_i)}| - 1\right)$$

$$= \frac{1}{2} \sum_{\omega_i}\left(\frac{\hat{f}(\omega_i)}{f(\omega_i)} + \frac{f(\omega_i)}{\hat{f}(\omega_i)} - 2\right) \qquad 3.10$$

It is refered to as the cosh measure because of its character, with respect to the quantity $u(\omega_i)$.

$$
\begin{aligned}
D &= \frac{1}{2}\sum_{\omega_i}\left(\frac{\hat{f}(\omega_i)}{f(\omega_i)} + \frac{f(\omega_i)}{\hat{f}(\omega_i)} - 2\right) \\
&= \frac{1}{2}\sum_{\omega_i}\left(\exp u(\omega_i) + \exp -u(\omega_i) - 2\right) \\
&= \cosh u(\omega_i) - 1 \qquad 3.11
\end{aligned}
$$

Figure 3.19 shows the relative effects of a spectral component's deviation, in ratio, from its actual value, for several distortion measures. It may be seen that the Itakura-Saito measure is non-symmetric, being rather lenient if the approximating spectral component is too small. The Itakura-Saito and the class of log-spectral-ratio measures all require a log operation, which is computationally quite expensive. This being the case there is no obvious advantage of the I-T measure over the Log-spectral-ratio measures. The I-T measure over half of its domain, has a character similar to the "Cosh" measure, which is in actual fact derived from it. This character being very lenient when the spectral ratio is small but very strict when this is large. The "Cosh" measure has the advantage though that it does not involve any log computation. By doing some subjective testing it was decided that the mean square log-spectral-ratio performed marginally better than the I-T measure and was employed in all the tests. (Refer to R.M Gray, Buzo, A.H. Gray and Matsuyama-(1980) for a discussion of the properties of some of the spectral distance measures)

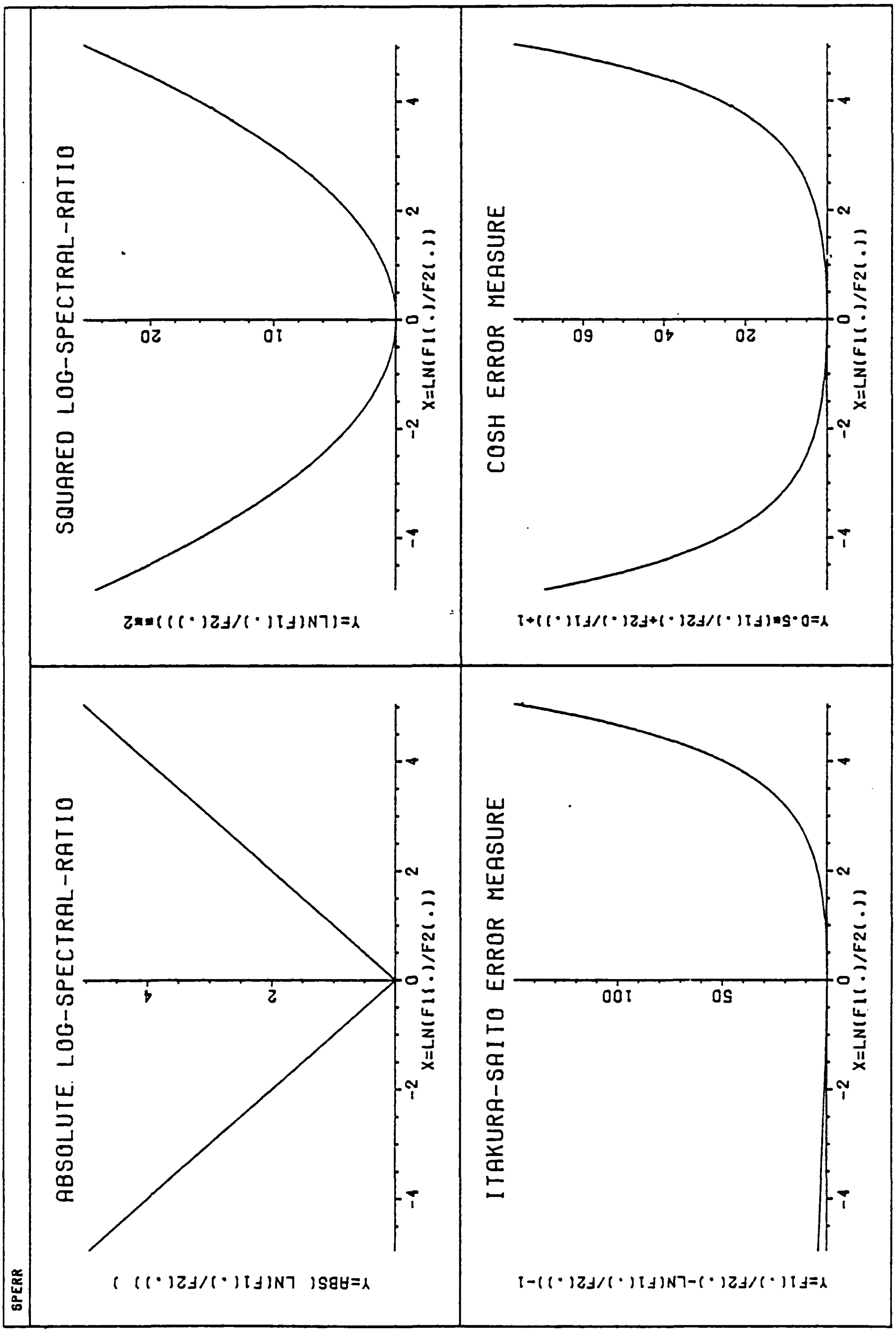


Figure 3.19 Relative effects of spectral deviation for various error measures.

3.6.2.1 Implementation, results and discussion.

The coding approach descibed was implemented, initially with the standard white noise/pulse train excitation for synthesis of the speech waveform at the receiver. The voice/unvoiced decision and pitch extraction were accomplished using the auto-correlation method. (For a review of the pitch extraction schemes available, the reader is advised to refer to Rabiner, Cheng, Rosenberg and McGonegal-(1976))

Due to the size of the library being used, a combination of error measures is employed. Thus an initial error condition has to be satisfied. The computation, in doing this, requires very much less effort than checking the mean square log spectral ratio for each of the library members. The formant positions for each member of the library are computed and stored along with the filter coefficients and the spectral response, each time an addition is made to the library. A library member is considered a viable candidate for approximating the parameters of a block in question, if the formant positions of the block in consideration and those associated with this library member are close. Closeness in this case means the following: Let the n-th formants be at positions p_n and q_n for the two parameters being compared, then the conditions below should hold.

$$|p_n - q_n| \leq 1 \quad \text{or} \quad \frac{|p_n - q_n|}{\min(p_n, q_n)} \leq 0.25 \qquad 3.12$$

When the conditions above are met, the mean square log-spectral-ratio is evaluted.

The learning characteristics of the scheme are indicated in table 3.10, for different error limits. As expected more blocks have their parameters approximated by members of the library as more speech is coded and the library fills up. Table 3.11 shows the rates obtained for various speech files and error limits. Listening tests indicate that an error limit so that about a third to half of the LPC parameters are new, is suitable for transmission without great deterioration in speech quality. Relaxation of the error limit beyond this, results in an impairment of intelligibility, for example some listeners thought, AN APPLE A DAY KEEPS THE BUTCHER AWAY instead of .....THE DOCTOR AWAY, was being uttered.

	number of	number of	ratio	
	new blocks	lib. blocks	new/lib	
1-st 50 blocks	31	19	1.6316	
2-nd 50 blocks	35	15	2.1875	
3-rd 50 blocks	17	33	0.4857	Error limit =50.0
4-th 50 blocks	28	22	1.2727	
5-th 50 blocks	21	29	0.7241	
6-th 50 blocks	20	30	0.6667	
7-th 50 blocks	13	37	0.3513	
8-th 50 blocks	12	38	0.3157	
9-th 50 blocks	22	28	0.7857	
last 25 blocks	9	16	0.5625	
1-st 50 blocks	36	14	2.5714	
2-nd 50 blocks	38	12	3.1667	
3-rd 50 blocks	26	24	1.083	Error limit =30.0
4-th 50 blocks	39	11	3.5454	
5-th 50 blocks	31	19	1.6316	
6-th 50 blocks	31	19	1.6313	
7-th 50 blocks	29	21	1.3809	
8-th 50 blocks	25	25	1.0	
9-th 50 blocks	31	19	1.6316	
last 25 blocks	15	10	1.5	

Table 3.10 Learning characteristics for MPPCD scheme. Error measure is squared log-spectral-ratio

The formants are checked, for 1-st formant 250Hz and upwards,
for 2-nd formant 750Hz and upwards. Block size=192.

3.6.3 The excitation problem

3.6.3.1 The FFT approach

The vocoders as described before, despite thier great compression capabilities, have one fault. This is their tendency to sound electrical. This is attributable to the inadequacy of the excitation signal. The judgement of whether an utterance is voiced or unvoiced is often wrong. Some of the phonemes are best excited by a signal which is periodic in addition to having some noise superimposed on it. Examples are the voiced fricatives {ν as in van; ð as in this; ʒ as in zoo; ʒ as in azure}

The evaluation of the correct pitch when a block being analysed is voiced, is not easy. Another point to note is that, there is rapid pitch variation when an utterance is being made, noted by Pierce and David-(1961). This gives some character to the speech. Although the pitch period is transmitted every 10-30msec (ie the average duration of about 2 pitch periods), the pitch period is evaluated over every 2 or 3 blocks. Thus about 5 pitch periods are averaged and transmitted.

Experiments conducted here have indicated that a contributant to the electrical characteristic of vocoder speech is the loss of phase information at frequencies below 1kHz. The following are the details of these experiments.

1) The FFT of blocks of speech was computed, for each block the phase signal was set to zero over the whole of the frequency band. The result was that the speech sounded electrical (robot

like)

2) The phase signal was set to zero for frequency points below an eigth of the sampling frequency. The other frequency points were left unaltered. The speech still sounded electrical.

3) The phase signal was set to zero for all frequency points greater than an eigth of the sampling frequency. The values at all other frequency points were left unaltered. The speech was now considerably improved. It now sounded slightly electrical.

4) The phase signal was set to zero for all frequency points above a quarter of the sampling frequency, for all frequencies below this, the phase was left untouched. The resulting speech sounded perfect.

The conclusion is that some phase information is important at frequencies below 1kHz.

The first departure from the vocoder excitation approach described thus far was reported by Schroeder and David-(1960). They related the experiments conducted in the development of what they described as a "high fidelity vocoder". In this paper they give a detailed yet simple account of the problems they encountered in trying to design a vocoder for transmitting 10kHz speech over a 3kHz channel. They eventually decided to excite their vocoder with a whitened signal derived from a low pass filtered version of their speech. This was called the "voice excited vocoder". The low pass filtered speech actually contained all the neccesary information for an excitation waveform. It contains the fundamental pitch signal and with some non-linear or possibly linear processing to extend the frequency range of this signal, we obtain the following. A pseudo excitation waveform, in synchronism with the actual excitation when the signal is voiced and a noise like signal when it is unvoiced. The use of this in a normal vocoder was subsequently reported by David, Schroeder, Logan and Prestigiacomo-(1962). Here the low pass filtered speech employed to generate an excitation waveform had a bandwidth of 700Hz. The art of designing voice excited vocoders has flourished since.

The voice excited vocoders achieve less compression than the voice/unvoiced excition based vocoders, with the advantage that the speech sounds more natural and speaker identification is easier. Voice excited LPC vocoders have been designed which operate from 4.8kb/sec to 12kb/sec

An alternative is to excite the LPC filter by a low pass filtered and coded version of the residual signal after applying the LPC inverse filter. The vocoder is then refered to as a "residual excited vocoder".

A residual excited LPC vocoder was implemented for use with the MPPCD scheme. The residual excitation was obtained in the following manner. The residual signal was obtained in the time domain for each block of speech. This signal was Fourier transformed so that we obtain the frequency domain representation of the residual. Blocks of size 192 are used. The values of the first 16 or 32 frequency components are quantised and transmitted. This represents low-pass filteration of the residual to 667Hz or 1.33kHz respectively. At the receiver the N residual (N=16 or 32) components are inverse Fourier transformed. A full band excitation signal is generated by inserting an appropriate number of zeros between the residual samples available.

For the coding of the frequency domain residual, the real and imaginary parts may be modelled as being Gaussian or the magnitude and phase parts may be modelled as being respectively Raleigh and evenly distributed. This may be confirmed by studying figure 3.20. The models allow the use Lloyd-Max quantisers for these. The alternative of quantising the magnitude and phase signals was chosen because it allows the study of the relative importance of these. By listening tests it was decided to accord more bits to the phase signal. For each frequency component, 6 bits were allocated for coding. 4 bits were used to code the phase and 2 bits to code the magnitude.

When 16 frequency components were used for coding, that is a bandwidth of 667Hz for the residual, 96 bits are required to be sent per block to code the excitation. This represents a large increase over the say 16 bits which would need to be sent in the voiced/unvoiced based vocoder. For the case where 32 frequency components of the residual are transmitted, 192 bits are sent per block. The latter case results in a transmission rate of 12kb/sec if we allocate 4kb/sec for transmitting the LPC parameters. The speech obtained in doing this is of very good quality. It is better than that obtained using transform coding for the same rate and better than that obtained using the straigtforward MPPCD scheme of section 3.4 for approximately the same rate.

The former case results in a coding rate of 8kb/sec if we

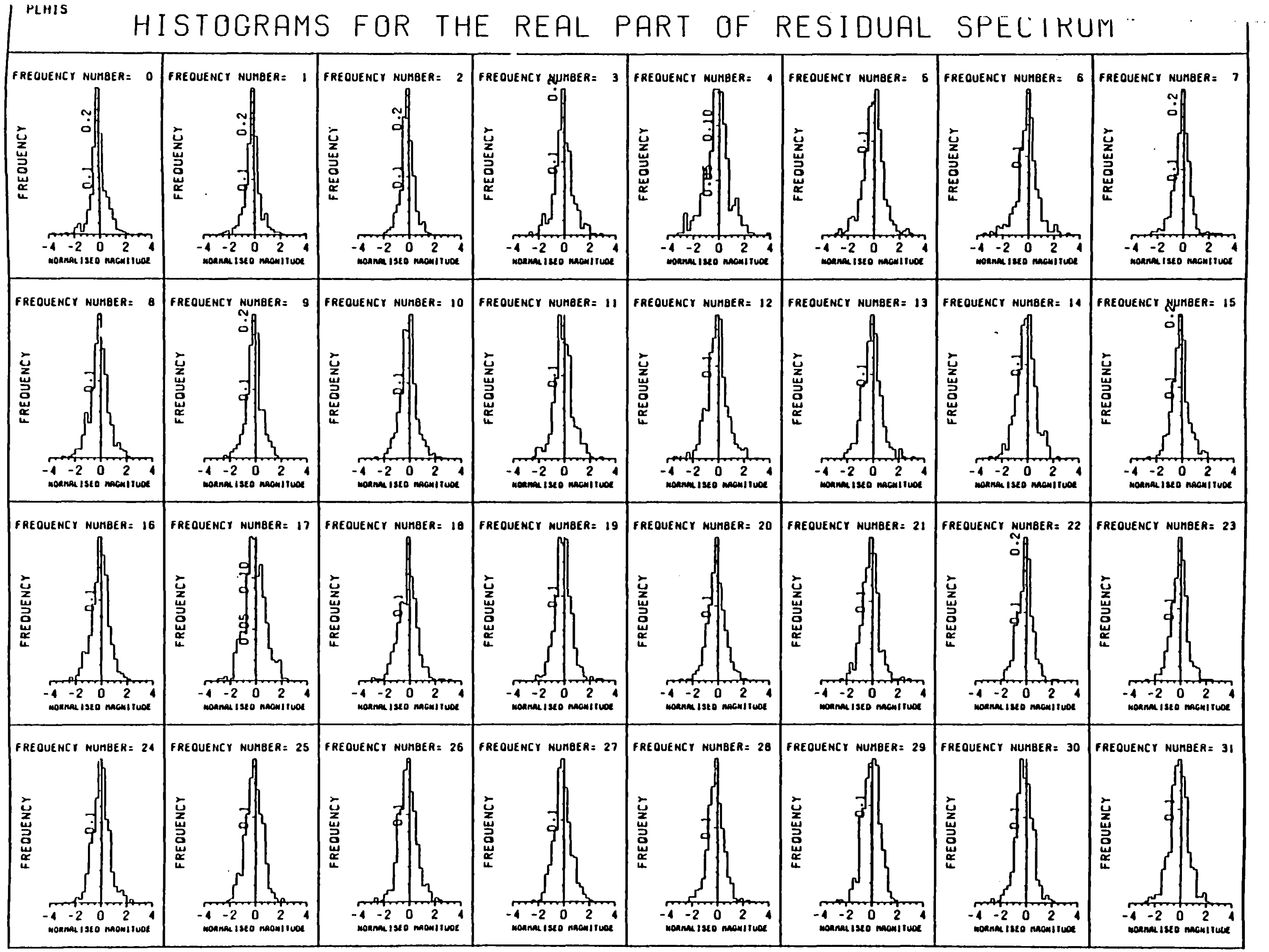


Figure 3.20 Spectral distributions of the residual signal, after linear prediction.

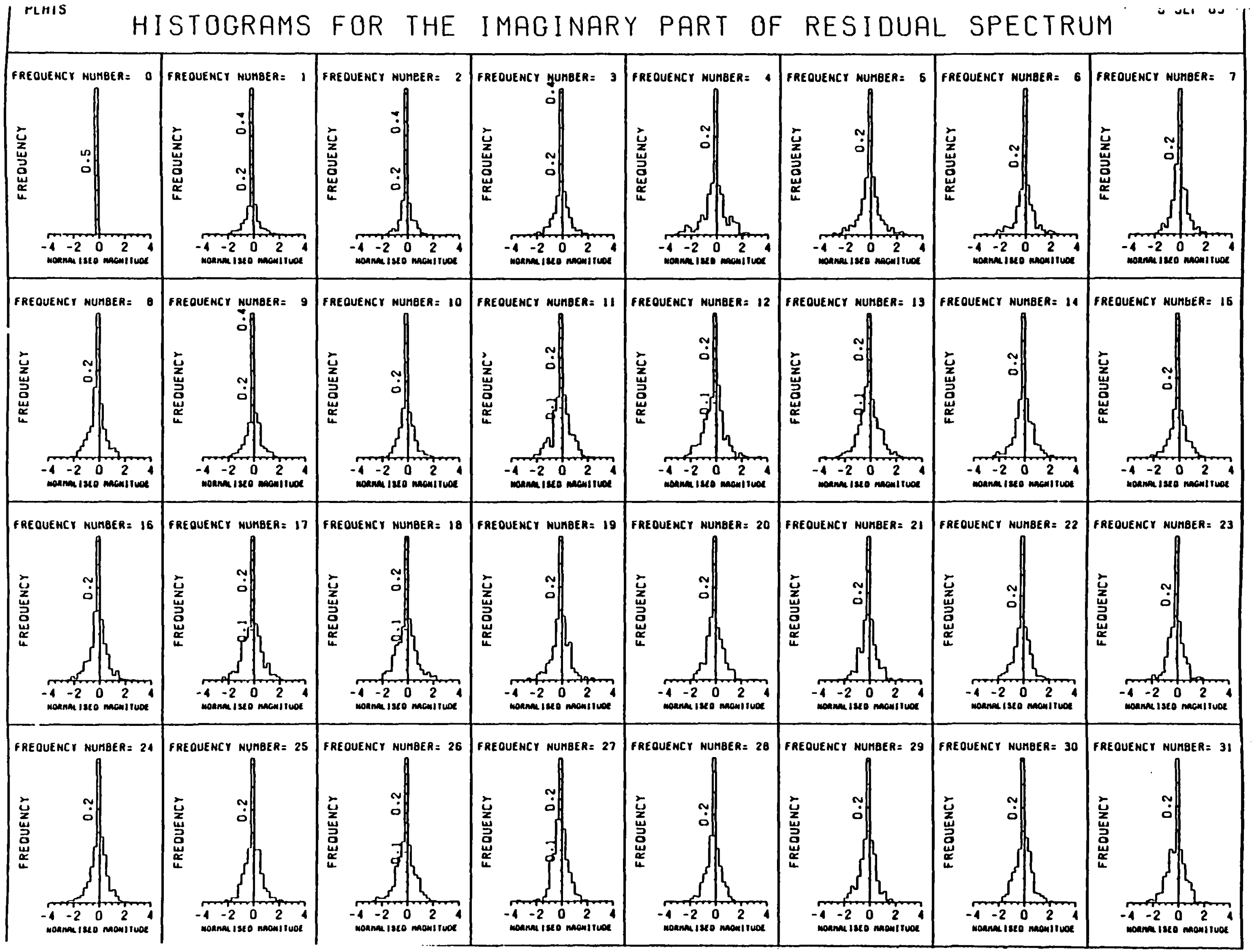


Figure 3.20 (continued)

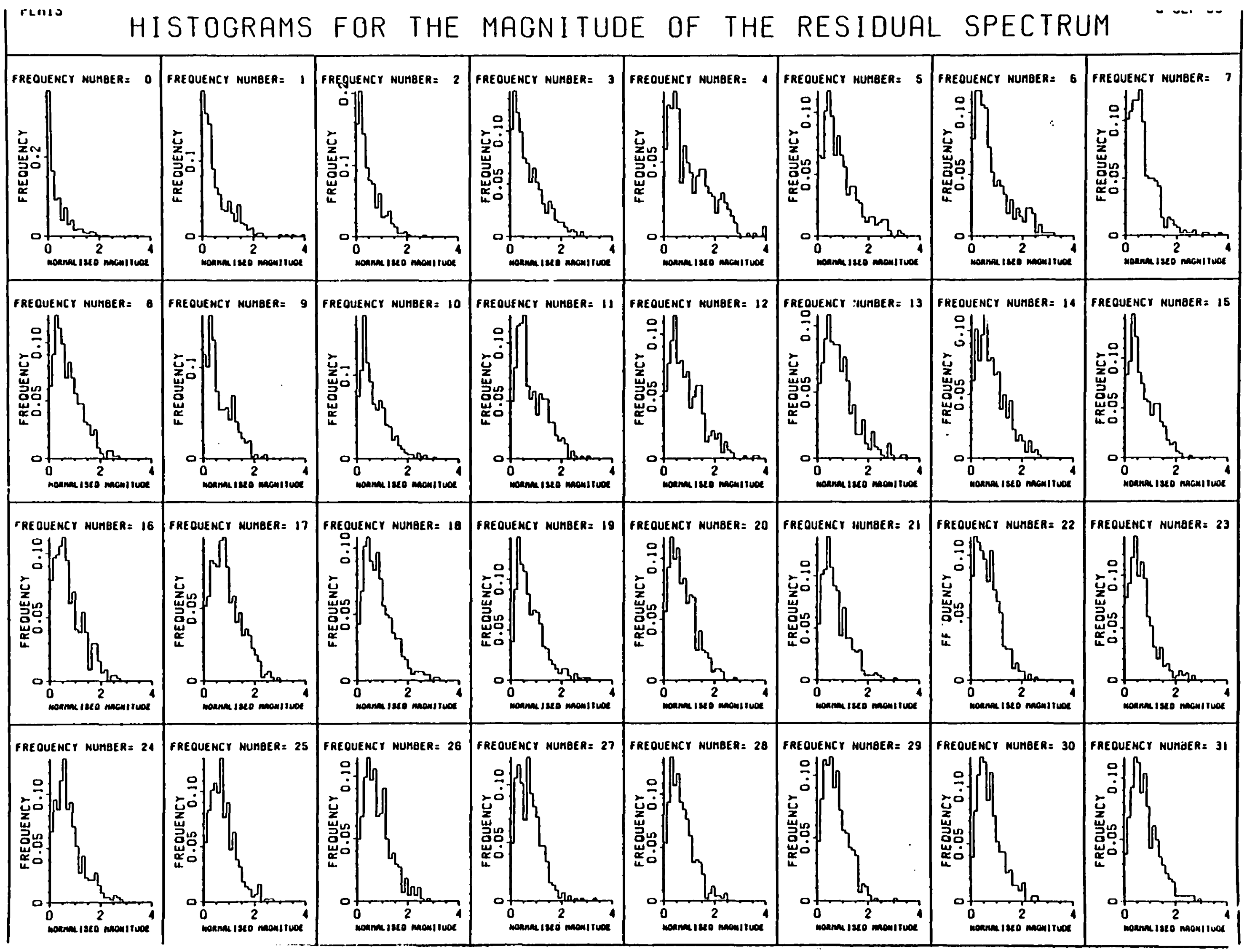


Figure 3.20 (continued)

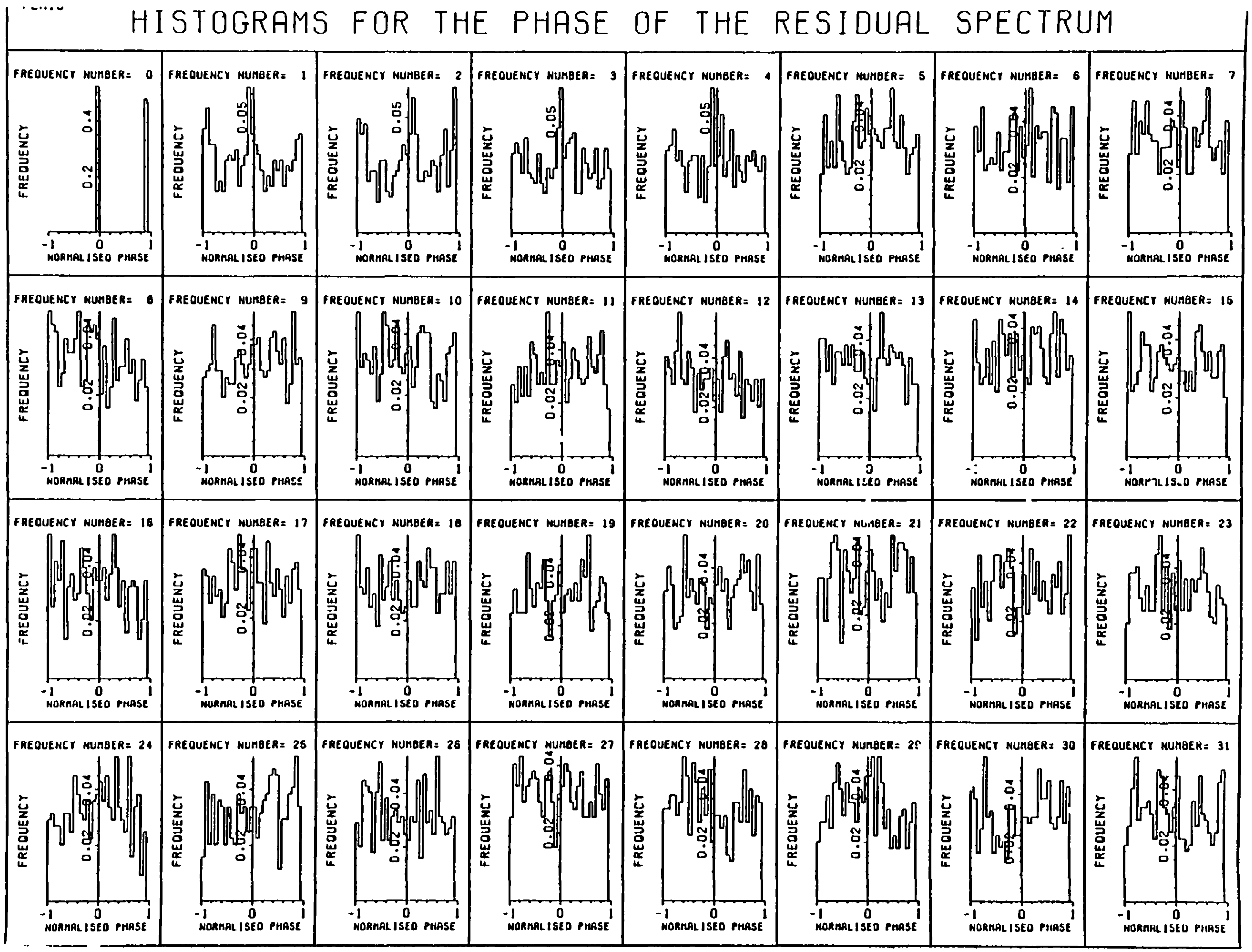


Figure 3.20 (continued)

allocate 4kb/sec for the transmission of the LPC parameters. The speech obtained is of good quality. An electrical character is beginning to encroach and it does not sound as good as the results of the 12kb/sec scheme. It is however of considerably better quality than the simple LPC vocoder with the voiced/unvoiced type excitation, transform coded speech or the MPPCD scheme at the same rate.

Of course we may use the MPPCD scheme as described before, choosing a rate for the excitation signal, 8kb/sec or 4kb/sec, transmit the excitation at this rate and use the LPC parameters to create a library. The occasion of observing a block whose LPC parameters are sufficiently "different" from all the members of this library means the transmission of these parameters and the inclusion of these in the library.

Instead a scheme, which results in transmission at a uniform rate, is opted for. For each block we decide whether the LPC parameters should be transmitted and included in the library. If so these parameters are transmitted with 104 bits/block and the lower rate for the transmission of the excitation information is chosen. 90 bits are transmitted for this. Another m bits are used to indicate that a block of LPC parameters is being sent.

If we find that the transmission of the LPC parameters for the block in question, is unneccessary, we opt for the higher rate for transmitting the residual information. 186 bits/block are used for this. Another 6 bits are transmitted, to indicate which member of the library of LPC parameters to use to approximate those of the present block.

	If filter parameters are from library	If filter parameters are new.
Frequency points coded and number of bits/harmonic	2nd to 32nd harmonics at 6 bits/harmonic total=186 bits	2nd to 16th harmonics at 6 bits/harmonic total=90 bits
Standard deviation of residual harmonics	6 bits	6 bits
Filter parameters	Library coordinate coded with 8 bits	Reflection coeffs quantised & coded with 104 bits. 10 bits for the 1st four 8 bits for the 5th to 12th
Fact that filter parameters are from or not library.	m bits	m bits
TOTAL	200+m bits if m=4 rate=8.5kb/s	200+m bits if m=4 rate=8.5kb/s

Table 3.11 Bit allocation strategy for FFT based residual excited LPC, for case where block size is 192

The advantage of this scheme, is that we have the choice of using a better residual signal for the cases where we distort the LPC model's spectral estimate. This results in an improvement since we go some way to correcting this distortion.

3.6.3.2 <u>The TDRIA scheme</u> [Wilson-(1983) and Atal and Remde-(1982)]

This involves the use of several impulses, of various magnitudes and of non-uniform spacing to excite the LPC filter. This may be envisaged to be an advancement upon the residual excited scheme described above, since another degree of freedom has been included in the coding process. The drawback of this is that some bits have to be employed to define the positions of the residual impulses. For the two alternative schemes which use 96 and 192 bits for the transmission of the residual information, we use 12 and 24 residual impulses respectively, per block. The impulse allocation scheme may be implemented using a multipath search, that is a dynamic programming approach. Let the total block size be N (192) and the total number of residual impulses required be L. Suppose a block with M samples has to have k impulses allocated. Then a possible dynamic programming equation is,

$$C_k(M) = \min_{1 \le z \le M-k+1} \{ C_{k-1}(M-z) - t(z| \text{ previous allocation }) \} \qquad 3.13$$

where $C_k(M)$ is the cost associated with the "optimal" way of allocating k impulses over the first M samples of the block in question. t(z|previous allocations) is the magnitude of the reduction in cost associated with allocating another impulse at position M-z, given previous allocations. Thus at stage k, $C_k(M)$ has to be evaluated for $M \in \{k,...,N\}$ ie for k<<N, approximately N

cost computations. For each value of M, the best z may be one of M-k+1 possible values. Thus the number of operations is of the order N^2k.

Alternatively a single path search may be used, but with several passes. Initially a reasonable allocation of impulses is established. (eg find the best way of allocating one impulse, given this, find the best way of allocating a second impulse, and so on. This is the scheme advocated by Wilson and Atal and Remde.) Subsequently, several passes may be made. In each pass, all impulse positions but one are fixed. The best position for this one impulse is then evaluated. This is similar to the non-hierarchical clustering schemes. The system advocated here however, is dictated by simplicity.

A suboptimal scheme is used to ascertain the positions of the residual impulses. The space of possible impulse positions is partitioned into L non-overlapping regions (L is the number of residual impulses transmitted per block). One residual impulse is permitted to lie within each region. A single pass, single path search is undertaken in order to determine where to place the impulses. This saves on the number of bits required to code the impulse positions, this being the smallest integer not less than $\log_2[N/L]$.

Residual impulse values:

Let y(n) be the signal being coded. Suppose k-1 impulses had already been placed at positions $p_1, \ldots, p_{k-1}$ with magnitudes $v_1, \ldots, v_{k-1}$. Also suppose

$$w(n) = \sum_i a_i w(n-i) \qquad 3.14$$

where w(0)=1, w(n)=0 ∀n<0. w(n) is the impulse response of the LPC filter A. Let $e_{k-1}(n)$ be the resulting error signal, where

$$e_{k-1}(n) = y(n) - \sum_i^{k-1} v_i w(n-p_i) \qquad 3.15$$

Then upon the emplacement of a new residual impulse at p_k, the total mean squared error is

$$\sum_n (e_k(n)^2) = \sum_n (e_{k-1}(n) - v_k w(n-p_k))^2 \qquad 3.16$$

The value of v_k resulting in a minmum value for mean square error is

$$v_k = \frac{\sum_{n=p_k}^{N} e_{k-1}(n) w(n-p_k)}{\sum_{n=p_k}^{N} w(n-p_k)^2} \qquad 3.17$$

For the k-th region, the position where an impulse may be placed such that the error in minimised is computed. The best values of p_k and v_k are noted and the $e_k(n)$ sequence evaluated. This process is continued until the positions of all the impulses are decided. During this process, no quantisation of the impulse amplitudes is undertaken. Upon deciding the positions of all the impulses, the amplitudes are recalculated as follows. Let

$$e_k(n) = y(n) - \sum_{i=1}^{L} v_i w(n-p_i) \qquad 3.18$$

Then the solution of the linear equations resulting from setting $\frac{\partial \sum_n e(n)^2}{\partial v_k}$ to zero for all k gives the v_k values.

$$\frac{\partial \sum_n e(n)^2}{\partial v_k} = -2\sum_n \{y(n) - \sum_{i=1}^{L} v_i w(n-p_i)\} w(n-p_k) \qquad 3.19$$

Setting this to zero gives.

$$\sum_n y(n)w(n-p_k) = \sum_{i=1}^{L} v_i \sum_n w(n-p_k)w(n-p_i) \qquad 3.20$$

We require therefore to solve the matrix equation

$$[\ R\ ][\underline{v}] = [\underline{s}] \qquad 3.21$$

where each member r_{ij} of R is

$$r_{ij} = \sum_n w(n-p_i)w(n-p_j) \qquad 3.22$$

and each member s_i of $\underline{S}$ is

$$s_j = \sum_n y(n)w(n-p_i) \qquad 3.23$$

After the solution of equation 3.20 to evaluate this set v_1 to v_k, these are quantised. Unfortunately R is not a matrix which allows the fast solution of equation 3.20. The above matrix equation may therefore only be solved with On^3 operations.

It ought to be mentioned that the quantisation of the v_i values, after the solution of the matrix equation 3.21 implies that those values obtained are not neccessarily the optimum quantised values. For cases where a small number of levels are used to approximate the v_i values, that is, 8 or less, the following

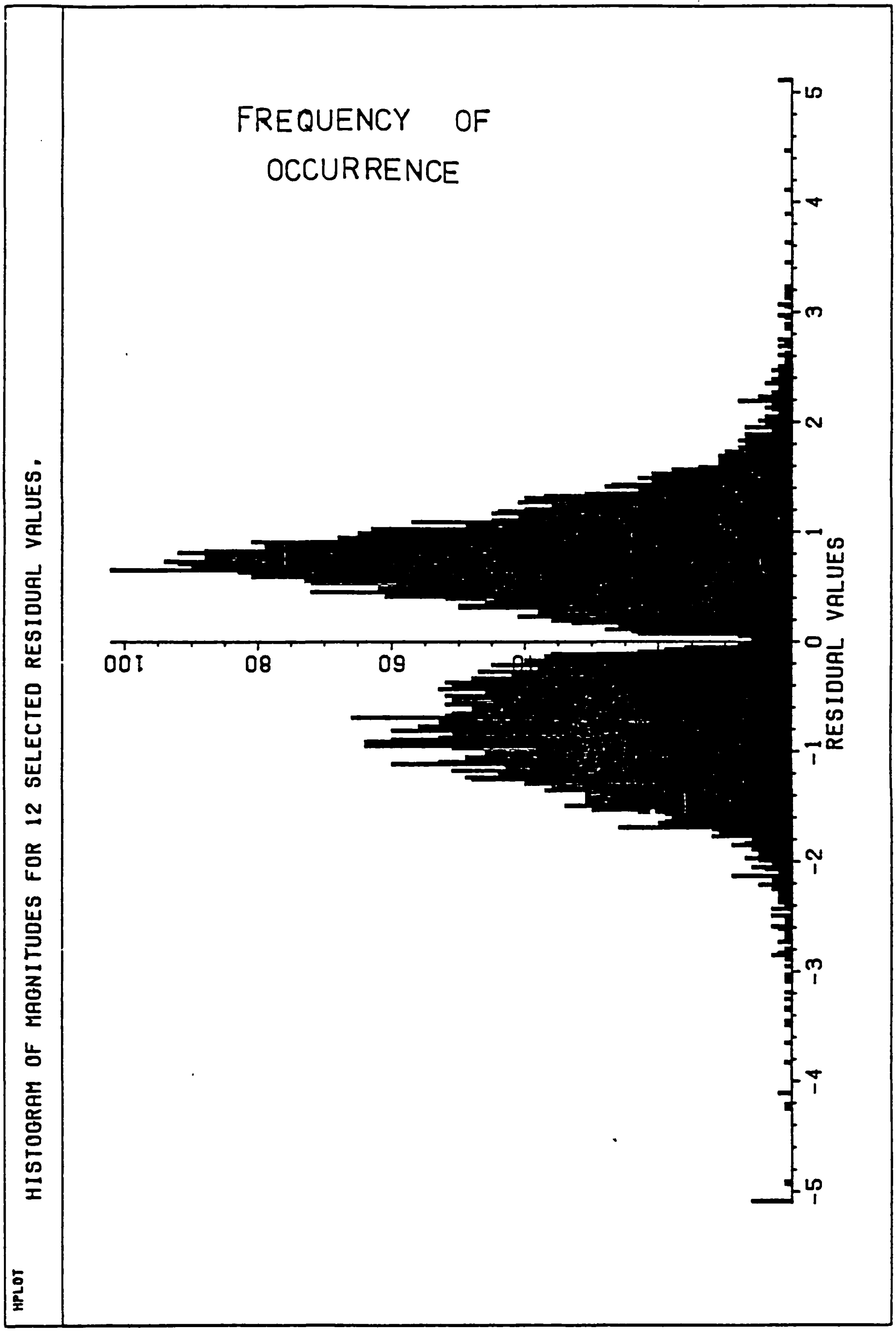


Figure 3.21 Histogram of residual magnitude in TDRIA coding.

alternative method may be more appropriate. A tree or trellis search may be used, in the same manner as in DPCM, with the difference that symbols are transmitted at irregular intervals.

Recently, there has been some interest in the use of the multiple excitation signal for the recomputation of the LPC parameters for speech coding. Essentially, an attempt is made to optimise both the LPC parameters and the pulse positions and amplitudes. This is instead of just evaluating the LPC parameters in the normal manner, leaving these unaltered whilst pulse positions and amplitudes are evaluated [Jain and Hangartner-(1984) and Parker, Alexander and Trussel-(1984)].

Two examples, with block sizes of 96 and 192, the bit allocation scheme is as follows:

BL SIZE=96:	LPC coeffs new	LPC coeffs from lib.
	4 impulses/block	12 impulses/block
	5 bits/impulse for pos	3 bits/impulse for pos
	3 bits/impulse for ampl	4 bits/impulse for ampl
	60 bits/10 LPC paramters	8 bits for library
	92+6+m bits total.	92+6+m bits total.

BL SIZE=192:	LPC coeffs new	LPC coeffs from lib.
	12 impulses/block	24 impulses/block
	4 bits/impulse for pos	3 bits/impulse for pos
	4 bits/impulse for ampl	5 bits/impulse for ampl
	104 bits/12 LPC paramters	8 bits for library
	200+6+m bits total.	200+6+m bits total.

	If filter parameters are from library	If filter parameters are new.
Number of impulses used and bits allocated/block	12 impulses, 3 bits code position and 4 bits code amplitude total=84 bits	4 impulses, 5 bits code position and 3 bits code amplitude total=32 bits
Standard deviation of residual impulses	6 bits	6 bits
Filter parameters	Library coordinate coded with 8 bits	Reflection coeffs quantised & coded with 60 bits.
Fact that filter parameters are from or not library.	m bits	m bits
TOTAL	98+m bits if m=4 rate=8.5kb/s	98+m bits if m=4 rate=8.5kb/s

Table 3.12 Bit allocation strategy for TDRIA based residual excited LPC, for case where block size is 96

The impulse amplitudes are quantised using a two sided Raleigh model. The histogram of figure 3.20 attempts to justify the use of this model. Figure 3.21 shows a typical speech waveform and the allocated residual impulses.

For all the above excitation alternatives, an indication of the standard deviation for the speech block, is coded using logarithmic quantisation and employs 6 bits for transmission. Table 3.12 shows the bit allocation schemes. m bits are used to indicate whether or not the LPC parameters are to be represented by a member of the library. It is suggested that a value greater than 1 is used such that some degree of error correction may be undertaken.

In the concepts described above, block sizes of 192 and 96 have been used. For cases where large coding delays are undesirale and therefore shorter block lengths have to be employed, this coding concept becomes even more attractive. Thus for block sizes of say 96 samples, (maximum delay of 36msecs+transmission delay) it should be expected that adjacent blocks have very similar LPC parameter sets. Hence the transmission of new parameters per block would not occur very often.

3.7 Listening Tests

The coding systems described here, were simulated on a computer. Several speech sentences were used. In each case the speech was sampled at 8kHz and digitised with 12 bits/sample accuracy. Below is a list of the types of coding schemes compared. Included are the results of coding, using Pulse Code Modulation (PCM), these PCM coded sentences set easily recognisable standards against which the methods may be compared.

1) SR8KK, original
2) SR8KK, PCM 4-bit linear
3) SR8KK, PCM 4-bit mu-law
4) SR8KK, coded via MPPCD scheme, error factor=0.1, rate=8.14kb/s, sn=10.64db
5) SR8KK, coded via MPPCD scheme, error factor=0.04, rate=11.34kb/s, sn=14.57db
6) SR8KK, coded via MPPCD scheme, error factor=0.025, rate=13.023kb/s, sn=16.6db
7) SR8KK, Straightforward LPC vocoder with voiced/unvoiced excitation block size=192, rate=4.67kb/s
8) SR8KK, LPC vocoder with MPPCD and voiced/unvoiced excitation error limit=10.0, block size=192, rate=4.67kb/s
9) SR8KK, LPC vocoder with MPPCD and voiced/unvoiced excitation error limit=20.0, block size=192, rate=3.1kb/s
10) SR8KK, LPC vocoder with MPPCD and voiced/unvoiced excitation error limit=30.0, block size=192, rate=2.5kb/s
11) KABITH, LPC vocoder with voiced/unvoiced excitation, block size=192, rate=4.67kb/s

12) KABITH, LPC vocoder with MPPCD and voiced/unvoiced excitation error limit=20.0, block size=192, rate=3.016kb/s

13) KABITH, LPC vocoder with MPPCD and voiced/unvoiced excitation error limit=30.0, block size=192, rate=2.59kb/s

14) TABITH, LPC vocoder and voiced/unvoiced excitation, block size=192, rate=4.67kb/s

15) TABITH, LPC vocoder with MPPCD and voiced/unvoiced excitation error limit=20.0, block size=192, rate=2.8kb/s

16) TABITH, LPC vocoder with MPPCD and voiced/unvoiced excitation error limit=30.0, block size=192, rate=2.3kb/s

17) SR8KK, LPC vocoder with residual excitation(FFT coded) block size=192, rate=6.0kb/s

18) LONG-FILE, LPC vocoder with residual excitation(FFT coded) block size=192, rate=8.0kb/s

19) LONG-FILE, LPC vocoder with residual excitation(FFT coded) block size=192, rate=12.0kb/s

20) LONG-FILE, LPC vocoder with residual excitation(FFT coded) formant positions used. Error limit=40.0, block size=192, rate=8.5kb/s

21) LONG-FILE, LPC vocoder with residual excitation(FFT coded) formant positions used. Error limit=65.0, block size=192, rate=8.5kb/s

22) SR8KK, DCT transform coded speech (Zelinski and Noll adaptation strategy), rate=8.0kb/s, sn=9.166db

23) SR8KK, DCT transform coded speech (Zelinski and Noll adaptation strategy), rate=12.0kb/s, sn=11.65db

24) SR8KK, DCT transform coded speech (Zelinski and Noll adaptation strategy), rate=16.0kb/s, sn=13.84db

25) LONG-FILE, LPC vocoder with residual excitation(TDRIA coded) block size=96, rate=8.0kb/s

26) LONG-FILE, LPC vocoder with residual excitation(TDRIA coded) block size=96, rate=12.0kb/s

27) LONG-FILE, LPC vocoder with residual excitation(TDRIA coded) formant positions used. Error limit=40.0, block size=96, rate=8.5kb/s

28) LONG-FILE, LPC vocoder with residual excitation(TDRIA coded) formant positions used. Error limit=50.0, block size=96, rate=8.5kb/s

29) LONG-FILE, LPC vocoder with residual excitation(TDRIA coded) block size=48, rate=8.0kb/s

30) LONG-FILE, LPC vocoder with residual excitation(TDRIA coded) block size=48, rate=12.0kb/s

31) LONG-FILE, LPC vocoder with residual excitation(TDRIA coded) formant positions used. Error limit=30.0, block size=48, rate=8.5kb/s

32) LONG-FILE, LPC vocoder with residual excitation(TDRIA coded) formant positions used. Error limit=40.0, block size=48, rate=8.5kb/s

33) LONG-FILE, LPC vocoder with residual excitation(TDRIA coded) formant positions used. Error limit=50.0, block size=48, rate=8.5kb/s

In the tests several listeners were asked to indicate preference, by assigning a mark out of 10, for each sentence listened to. The sentences were presented a pair at a time.

The general outcome of the tests is indicated by the list of

File no	Scores	File no	Scores	Comment
1	1,1,1,1,1,1,1,1	17	0,0,0,0,0,0,0,0	
1	1,1,1,1,1,1,1,1	2	0,0,0,0,0,0,0,0	
2	0,0,0,0,0,0,0,0	3	1,1,1,1,1,1,1,1	
17	1,1,1,1,1,.5,0,0	22	0,0,0,0,0,.5,1,1	
17	0,0,0,0,0,0,0,0	3	1,1,1,1,1,1,1,1	
18	0,0,.5,.5,0	19	1,1,.5,.5,1	
18	0,0,1,1,0	20	1,1,0,0,1	Close result
18	1,1,0,0,1	21	0,0,1,1,0	Close result
29	0,0,0,0,0	18	1,1,1,1,1	Definite pref
18	1,.5,0,1,1	30	0,.5,1,0,0	
29	0,0,0,0,0	31	1,1,1,1,1	
29	0,0,0,.5,.5	30	1,1,1.5,.5	
29	0,.5,0,0,1	32	1,.5,1,1,0	Close results
29	0,.5,0,0,1	33	1,.5,1,1,0	Close result
4	0,1,1,1,0	22	1,0,0,0,1	
23	1,0,0,1,1	5	0,1,1,0,0	
6	0,0,0,0,0	24	1,1,1,1,1	

Table 3.14 Preference chart for the speech coding schemes compared. 0 or 1 indicate a positive preference, 1 indicates actual preference. 0.5 indicates no preference.

table 3.14 which gives the numbers of listeners who prefered which sentences of each of the pairs presented.
A complete check of subjective quality was not undertaken for all possible pairs of coded sentence s listed. This is because of the pointlessness of doing this for most pairs. For example, test sentences 7 to 16 were not compared subjectively with the results of other methods. This is because the voiced/unvoiced excitation scheme used resulted in artificial sounding speech which is always not preferable. These sentences are included in the list, to indicate the coding rates achieved for a not considerable reduction in quality compared to straightforward vocoder schemes (sentences 7, 11 and 14).

Results are given for pairs of sentences so that with these, the relative subjective quality of the methods presented in this chapter, may be established.

The following is a summary of the results obtained.
Tests 4 and 5 show the preference of 4 bit mu-law PCM over residual excited- LPC at 6kb/s and preference of the latter over 8kb/s DCT coding.
The 7th and 8th tests are very important and indicate that there is very little difference between the residual excited schemes with and without the updated memory of LPC parameters, when the coding block size is large.
Preferences are more defined for small block sizes (tests 11, 13 and 14). Here the method with an updated library is prefered. In all cases, as the error limit is increased, beyond a point, allowing greater spectral distortion, the listeners dislike the result.
Tests 15, 16 and 17, show that the basic MPPCD result is less less preferable to DCT coding at high bit rates, but is preferable at low bit rates.

3.8 Conclusions and discussion

In this chapter a methodology for data compression has been presented and investigated. For each application, we have endevoured to compare this with alternative schemes. In most applications the scheme has worked reasonably well. In some cases however, coding methods particularly suited to the data to be coded have performed better.

The scheme has the following disadvantages.

a) The basic scheme has been shown to be incapable of achieving very efficient compression compared with other methods particularly well suited to the data to be compressed. For example, it achieves worse compression ratios than Huffman coding, for independent letter sources of known statistics, as expected. It achieves worse compression for a given mean square error than Discrete Cosine Transform coding for sources which are well modelled as 1-st order auto-regressive. It is difficult to extend the method to two dimensional data because of the neccessity of some 'time' axis. The two dimensional nature of image element correlation is thus more difficult to take advantage of.

b) The basic MPPCD scheme results in a variable transmission rate. Whilst this is unavoidable in zero distortion coding for any efficient data compression scheme anyway, it is inconvenient in the case of coding with a fidelity criterion. Thus, in implementation for transmission over a fixed rate channel, one requires the use of a buffer and feedback in the following manner.

The status of an output buffer is continually monitored. When this buffer is close to being overfilled, the distortion limit is relaxed so that the coder output rate is reduced. When the buffer is close to being emptied, the distortion criterion is tightened such that the coding rate is increased.

c) The MPPCD scheme, though simple in realisation, requires a lot of computational effort at the encoder. A lot of comparisons of data blocks are required, this may be very time consuming.

The MPPCD scheme has the following advantage over most schemes. a) The system is very flexible. The user can choose at will, any distortion measure. The flexibility of the scheme is demonstated by its ability to code both speech and image data and also be applicable for zero distortion coding. Thus the system may be configured to code anything with minimal alteration of the receiver.

b) The system is a viable alternative to transform and adaptive multipath coding, in a situation where considerable computation is tolerable at the encoder but intolerable at the receiver. This situation may occur when information is being broadcast to several receivers, where because there is only one broadcast point, considerable capital may be expended in equipping this with powerful processors. Transform coding, for example in contrast requires an almost as much computational capability at the receiver as at the transmitter, in order to implement the inverse transformation. Another example of the type of situation being refered to, is the case of compression for data storage. Here the data is stored once but may be retrieved several times. It is acceptable to employ considerable computation in the job of storage, whereas it is undesirable to require considerable processing power in retrieval if this is to be done several times.

In this chapter, results have been presented for the application of the MPPCD scheme to efficient LPC speech coding. Two broad approaches have been investigated. The first results in a variable transmission rate and relies upon the use of a voiced/unvoiced model of speech generation for excitation. In that section it is shown that it is possible to achieve greater compression for little loss in subjective quality, by learning a library of LPC parameters.

The second, a more satisfactory approach, involves the use of a more sophisticated excitation scheme. Two excitation methods were investigated. Encouraging results were obtained for speech coding at around 8.5kb/s, which unlike residual excited LPC in conjunction with vector quantisation requires no extensive prior processing.

It is significant to say that the approach to data compression where use is made of previously coded data, is a new and encouraging field for data compression. There are a few salutory comments to be made though. Most studies in data compression, presume that there are no transmission errors. Now although by good channel coding, the probability of error may be made vanishingly small, these errors still occur. For block coding where no use is made of previously coded data, the effects of channel errors in the duration of a block are confined solely to this block of data. This is not so for a coding method like the MPPCD scheme, where errors in the transmission of. a block will have an effect on all subsequent blocks of data.

The only way in which this may be countered is to define superblocks, such that the following is done. At the beginning of each new superblock, coding is started anew. No information from previously coded superblocks are employed. In this way errors are confined to the individual superblocks. Provided these superblocks are sufficiently large, the resulting compression inefficiency should be negligible.

CHAPTER 4 ADAPTIVE DATA CODING WITH MEMORY, A THEORETICAL DISCUSSION

4.1 Introduction

In this chapter a theoretical discussion of the performance of the coding scheme described in the previous chapter is given. We shall refer to this as, the Matching of Patterns in Previously Coded Data or "MPPCD" scheme. It is shown that in the limit as the block lengths tend to infinity, the MPPCD scheme becomes efficient, for sources with large redundancy.

We begin the chapter with a brief discussion of the concept of the information associated with a source. The information content or the Shannon entropy of a source is then linked to the minimum rate at which this source may be coded for transmission. Following this, the MPPCD method is described and in mathematical notation, the rate at which it will code a source is given. At this juncture sources considered will have a discrete outcome set and coding will be noiseless. In the limit as the block sizes considered tend to infinity, it is shown that the coding rate is close to the Shannon entropy of the source, for signals of large redundancy. In showing this we employ the Shannon-McMillan-Brieman asymptotic equipartition theorem "AEP". This is a very interesting and important theorem associated with the probability distributions of long sequences from ergodic sources.

A discussion of the coding performance of the MPPCD scheme is undertaken for the situation when we allow distortion (coding with a fidelity criterion). Treatment of this case is considerably more difficult compared with the zero distortion situation. Consideration of this case is necessary however, because in general the sources we deal with are of continuous amplitude, where coding is only feasible with distortion or else the data rates that are permitted do not allow zero distortion coding. Several assumptions are made so that the analysis is tractable. In the course of this discussion, rate-distortion theory is introduced. Rate-distortion theory allows the generalisation of information theory and more specifically source coding, to encompass the class of sources with a countable or uncountably infinite outcome space. Rate-distortion theory deals with the problem of having to transmit data from a source whose rate of information generation is greater than the capacity of the channel over which the data is required to be transmitted. In addition, a theorem is developed that is similar to the Shannon-McMillan-Brieman AEP theorem, but relates to sources with a continuous in addition to discrete outcome space. With the assumptions made, it is shown that as the block lengths become very large, the coding scheme becomes very efficient for situations of large distortion.

In this chapter we offer complete proofs for most theorems considered, even though some tedious proofs of known results may be found elsewhere. Two very important theorems used herein, the Ergodic theorem and the theorem concerning the convergence of conditional expectation are proved in the appendices.

4.2 Information

In the transmission of information we are concerned with the rate at which information is generated by a statistical source. Knowledge of this tells us how much "effort" needs to be devoted to the business of communicating the outcomes from this source. Thus the need arises for a quantitative measure of "information". Intuitively, information may be related to uncertainty. That is, it may be said that the more uncertain we are about the outcome of an experiment, the more knowledge or information we gain after the event of observing this outcome. A measure of information, in that case, should be related to the statistics of a random source so that the more uncertain we are about, or the more random the source's outcomes, the higher the information value we attach to the source.

To this end three axioms which a measure of information should satisfy, were proposed by communications workers of the 1940's. Consider a statistical source defined by its probability mass function p_i $\forall i$, a measure of information $I(p_1, p_2, \ldots)$ should satisfy the following:

1) $I(p_1, p_2, \ldots)$ should be continuous in p_i for all i

2) $I(p_1, p_2, \ldots)$ should be maximum when all possible events are equally likely. The information, when all events are equally likely should be a monotonically increasing function of the cardinality of the set of possible events.

3) Additivity; the total information obtained from several independent sources should be a weighted sum of the individual information associated with each source plus additional information indicating the uncertainty as to

which source is being observed at each instant.

Shannon's second theorem says that a function $I(p_1, p_2, \ldots)$ that satisfies those axioms is

$$I(p_1, p_2, \ldots) = H(p_1, p_2, \ldots)$$
$$= -\sum_i p_i \log_b p_i \qquad 4.1$$

This is refered to as the Shannon entropy function. The base of the log is generally set to 2. This information measure was first introduced by Shannon in his classic paper of 1948 [Shannon-(1948a]. Good textbooks which go through the foundations of information theory and the philosophy that led to the Shannon entropy function are very many, examples of which are; [Brillouin-(1956), Reza-(1961) and Cherry-(1978)].

4.3 The MPPCD scheme

Consider a source which is required to be coded and which has an outcome set Ω with a finite number of members. A block of symbols of length M from this source is defined on the product set

$$\Omega^M = \Omega \times \Omega \times \ldots \times \Omega = \bigtimes_{i=1}^{M} \Omega$$

From this set of possible outcomes may be generated a Borel-Σ field $\mathfrak{F}$ and on this field a probability measure P is defined. The source whose outcomes are sequences of length M is called $(\Omega^M, \mathfrak{F}, P)$. Let the set Ω have cardinality C say. Ω^M has cardinality C^M. Choose a number N so that $\log_2 N$ is small compared with $\log_2 C^M$. Decide on a sequence of lengths $L_1, L_2, \ldots, L_N$ so that $M = L_1 < L_2 < L_3 < \ldots < L_N$.

In the simplest form, the coding procedure is as follows:

Consider a sequence of source symbols of length L_N. Perform C^M experiments on the set of previously coded outcomes to see if these symbols had occured before, note that these are known to both the transmitter and receiver. If this L_N length sequence had occured before, send the coordinate of the instant in the past when it was transmitted. This will be represented with $\log_2 C$ bits; in addition $\log_2 N$ bits are sent to indicate the length of the data block being coded. Then go to the next block of length L_N and try again. If the L_N sequence is not found after C^M experiments on the set of previously coded symbols, consider coding a sequence of length L_{N-1}, see if this may be observed in C^M experiments on the set of previously coded symbols. If so send the coordinate of the point in the previously coded sequence where this was observed. This is done with $\log_2 C^M$ bits and an additional $\log_2 N$ symbols. Failure to find an L_J sequence in C^M experiments on the previously encoded sequence results in an attempt with a smaller length L_{J-1} of data. The coding procedure as outlined above is continued until the event of a failure to code a block of length L_2. The actual data of length $L_1 = M$ is then transmitted, with $\log_2 C^M$ bits, with an additional $\log_2 N$ bits to indicate the length of the block encoded. After this one goes on to the next block of length L_N and continues in the same manner.

4.4 Performance bounds for noiseless coding

The coding rate of the scheme is:

$$\frac{(\log_2 N + \log_2[C^M])}{\text{average length}} \qquad 4.2$$

By average length the following is meant; we will encode varying lengths of data with C^M experiments associated with previous outcomes, the average length $\bar{L}$ is the average length of an encoded block. To ascertain bounds on the rate, we find bounds on the average length. As usual with most coding methods, it is impossible to ascertain the theoretical performance under practical conditions. We do the next best thing and content ourselves with performance as some parameter is pushed to some extreme. In this case, we ascertain what happens as an "elementary" block size tends towards infinity. What is done is to increase the cardinality of the outcome set by considering a block of size "k" source symbols as the elementary source symbol. The following example shows what is meant.

i) Let $M=L_1=1$, $L_2=2,\dots\dots\dots,L_N=N$. In coding, consider firstly whether one can encode N symbols with C experiments on the set of previous outcomes. Upon failure, try to code $N-1$ symbols on C experiments on the set of previous outcomes, and so on. Upon successive failures to code, block sizes are reduced until we have a block of size one, then this symbol is sent using $\log_2 C$ symbols.

ii) Let $k=3$. Once again let $M=L_1=1$, $L_2=2$, $\dots,L_N=N$. In coding, consider firstly whether a block of N elementary symbols or 3N symbols may be observed during C^3 experiments on the set of previous outcomes. Upon failure, try to code $3N-3$ symbols or $N-1$ elementary symbols in C^3 experiments on the set of previous outcomes, and so on. When one gets to six symbols and is unable to code this, the first three symbols are the sent, using $\log_2 C^3$ symbols.

What is done is to find the asymptotic performance of the scheme as $k \longrightarrow \infty$. It will be demonstrated that as $k \longrightarrow \infty$, the average length $\bar{L}$ is the largest integer such that the inequality 4.3 holds.

$$\bar{L} < \frac{\log_2 C}{H_\infty} \qquad 4.3$$

C is the cardinality of the source symbol outcome set and H_∞ is the Shannon per symbol entropy of the source in the limit as the block size tends to infinity. Suppose Y^{L_N} is to be coded. The probability that Y^{L_N} is encodable using $R = C^k$ independent observations from the previous outcomes is:

$$\{1-(1-p(Y^{L_N}))^R\}$$

The average length is written as follows.

$$\begin{aligned}\bar{L} = {} & \text{Expectation over all } \; Y^{L_N} \; \text{sequences} [\![[\{1-(1-p(Y^{L_N}))^R\}] \times L_N \\ & +[\{1-(1-p(Y^{L_{N-1}}))^R\} - \{1-(1-p(Y^{L_N}))^R\}] \times L_{N-1} \\ & +[\{1-(1-p(Y^{L_{N-2}}))^R\} - \{1-(1-p(Y^{L_{N-1}}))^R\}] \times L_{N-2} \\ & + \; . \; . \; . \; . \; . \; . \\ & +[\{1-(1-p(Y^{L_2}))^R\} - \{1-(1-p(Y^{L_3}))^R\}] \times L_2 \\ & +[\{ \quad (1-p(Y^{L_2}))^R\}] \times L_1]\!] \end{aligned} \qquad 4.4$$

Where L_J is the number of "k length" elementary symbols being encoded. It will be assumed from now on that $L_J = J$. The above equation is obtained by this reasoning: The sequence of length L_J is coded if it is possible to find this sequence in R experiments but impossible to find a longer sequence in R experiments. Thus the probability of coding an L_J length sequence, Y^{L_J}, is the probability of observing the outcome Y^{L_J} in R experiments <u>minus</u> the probability of observing its subset outcome $Y^{L_{J+1}}$ in R experiments. The average length may be written as:

$$\begin{aligned}\bar{L} =& E\{[1-(1-p(Y^N))^R](N-(N-1)) \\ &+ [1-(1-p(Y^{N-1}))^R]((N-1)-(N-2)) \\ &+ \ldots + [1-(1-p(Y^2))^R](2-1)+1\} \\ =& E\{[1-(1-p(Y^N))^R] + [1-(1-p(Y^{N-1}))^R] \\ &+ .. + [1-(1-p(Y^2))^R] + 1\}\end{aligned} \tag{4.5}$$

It will be shown that all the terms in the equation 4.5 tend to zero or unity in the limit as $k \longrightarrow \infty$, thereby giving some bounds on the length $\bar{L}$. Consider the function $F = 1-(1-p(Y^J))^R$. For $R=C^k$ we have

$$F = 1-(1-p(Y^J))^{(C^k)} \tag{4.6}$$

where

$$Y^J = \{x_1, \ldots, x_k \;;\; x_{k+1}, \ldots, x_{2k} \;;\; \cdot\;\cdot\;\cdot \;; x_{1+(J-1)k}, \ldots, x_{Jk}\} \tag{4.7}$$

Each of the x_i is an original source symbol.

$$\begin{aligned}F =& 1-\exp[\ln((1-p(Y^J))^{(C^k)})] \\ =& 1-\exp[C^k \ln(1-p(Y^J))]\end{aligned} \tag{4.8}$$

Let $u(k) = \frac{1}{C^k}$ and $v(k) = \ln(1-p(Y^J))$.

This gives

$$F = 1-\exp\{\frac{v(k)}{u(k)}\} \tag{4.9}$$

It will be seen that as k → ∞, $v(k) \longrightarrow 0$ and $u(k) \longrightarrow 0$. L'Hopital's rule gives:

$$\lim_{k\to\infty} F = 1 - \exp\Big(\big(\frac{dv(k)}{dk}\big|_{k=\infty}\big)/\big(\frac{du(k)}{dk}\big|_{k=\infty}\big)\Big)$$

$$= 1 - \exp\Big(+C^k \frac{dp(Y^J)}{dk}/(1-p(Y^J))\ln C\Big)\Big|_{k=\infty} \qquad 4.10$$

Now

$$p(Y^J) = p\{x_1,\ldots,x_k\,;\,x_{k+1},\ldots,x_{2k}\,;\,\ldots\,;\,x_{1+(J-1)k},\ldots,x_{Jk}\}$$

may be shown, by the Shannon-McMillan-Brieman AEP theorem, to tend towards the value 2^{-JkH_∞} as k → ∞, for a set S_1 of Y^J space and zero for all $Y^J \in \bar{S}_1$. Thus $\forall Y^J \in S_1$

$$\lim_{k\to\infty} F = 1 - \exp\{C^k \frac{d}{dk} 2^{-JkH_\infty}/(1-p(Y^J))\ln C\}\big|_{k\to\infty}$$

and $\forall\ Y^J \in \overline{S}_1$

$$\lim_{k\to\infty} F \approx 1 - exp\Big(C^k \frac{d(0)}{dk}/(1-p(Y^J))\ln C\Big)\Big|_{k=\infty}.$$

But
$$\frac{d}{dk}(2^{-JkH_\infty}) = 2^{-JkH_\infty}\ln(2^{-JH_\infty})$$

Thus
$$\lim_{k\to\infty} F = 1 - \exp\Big(-C^k 2^{-JkH_\infty}(J H_\infty \frac{\ln 2}{\ln C})\Big) \qquad 4.11$$

The exponent goes to −∞ as k → ∞ for $2^{-JH_\infty}C > 1$ and goes to zero as $k \rightarrow \infty$ for $2^{-JH_\infty}C < 1$. Thus for

$$J < \frac{\log_2 C}{H_\infty}, \quad F \rightarrow 1 \qquad 4.12$$

$$J > \frac{\log_2 C}{H_\infty}, \quad F \rightarrow 0 \qquad 4.13$$

The expectation

$$E(\lim_{k \rightarrow \infty} F) = 1 \text{ or } 0 \text{ for}$$

$$J < \frac{\log_2 C}{H_\infty} \text{ and } J > \frac{\log_2 C}{H_\infty} \text{ respectively} \qquad 4.14$$

Thus

$$\lim_{k \rightarrow \infty} \overline{L} = J$$

where J is the largest integer smaller than $\frac{\log_2 C}{H_\infty}$ or

$$\overline{L} > \frac{\log_2 C}{H_\infty} - 1 \qquad 4.15$$

Now it should be pointed out $\overline{L}$ is the average number of k length sequences per block that is coded. Thus the actual average length is given by

$$L_{\text{av}} = k\overline{L} > k(\frac{\log_2 C}{H_\infty} - 1) \qquad 4.16$$

The coding rate is;

$$R = \frac{\log_2 N + \log_2 C^k}{L_{Av}} < \frac{\log_2 N + \log_2 (C^k)}{k(\frac{\log_2 C}{H_\infty} - 1)}$$

and for $k \rightarrow \infty$ we have.

$$R < \frac{H_\infty}{1 - \frac{H_\infty}{\log_2 C}} \qquad 4.17$$

For a source with large redundancy, that is $H_\infty << \log_2 C$, the coding scheme is efficient. We can make the performance tighter by not restricting lengths to be integer multiples of k. The disadvantage of this is that we have to devote more bits to the coding of the block sizes as the number N of possible block sizes increase.

4.4.1 The Shannon-McMillan-Brieman Asymptotic Equipartition theorem

This is a fundamental theorem associated with the probability distributions of long sequences from an ergodic source. The AEP is a direct consequence of the ergodic theorem. The theorem says that the probability of occurrence of a block of symbols $\mathbf{X}^N$ of length N, as N tends to infinity, behaves as follows: Every N length sequence $\mathbf{X}^N$ is a point in the N dimensional product set $\overset{N}{\underset{i=1}{X}} \Omega$. Then the product set may be partitioned into two disjoint subsets S_1 and $\bar{S}_1$ as N tends to infinity. All $\mathbf{X}^N$ sequences that belong to S_1 occur with almost constant probability 2^{-NH_∞} and all $\mathbf{X}^N$ sequences that belong to $\bar{S}_1$ occur with almost zero probability. The physical interpretation is that for long sequences we observe that some particular options almost never occur; the others which occur, do so with almost constant probability. This theorem is rather useful and is often used as a justification for the use of block source codes of fixed length and rate. Before going on to the theorem proper, we should state the ergodic theorem, the foundation upon which the AEP is built. Consider a source with a countable outcome space Ω , a Borel-Σ field contructed from this space, and a probability measure $\mu(.)$, defined on this field. Consider a bounded μ-measurable function $g(\omega)$ defined on the space Ω , with mean defined as follows;

$$\hat{g} = \int_{\forall \omega \in \Omega} g(\omega) d\mu(\omega) \qquad 4.18$$

Then if the source (Ω, $\mathfrak{F}$,μ) is ergodic and T^i is a coordinate shift of i positions, $T^i(\omega_j) = \omega_{i+j}$;

$$\lim_{N \to \infty} \frac{1}{N} \sum_{i=0}^{N-1} g(T^i(\omega_0)) = \hat{g} \qquad 4.19$$

That is, the time average of the function $g(\omega)$ tends towards the mean $\hat{g}$ as the block length is increased.

Now we shall give a proof for the AEP theorem. Suppose the sequence of functions $g_k(x_i)$ are defines thus;

$$g_k(x_i) = -\log_2 p(x_i | x_{i-1}, x_{i-2}, \ldots, x_{i-k})$$

$$\text{and} \quad g_0(x_i) = -\log_2 p(x_i)$$

$$d\mu(\omega) = p(\omega) d(\omega) \quad \text{where this exists} \qquad 4.20$$

The Kolmogorov-Sanai[Billingsley-(1966)] theorem says that

$$H_\infty = \lim_{n \to \infty} H(x_0 | x_{-1}, \ldots, x_{-n}) \qquad 4.21$$

where

$$H(x_0 | x_{-1}, \ldots, x_{-n}) = -\int \log_2 p(x_0 | x_{-1}, \ldots, x_{-n}) d\mu(x_0, x_{-1}, \ldots, x_{-n}) \qquad 4.22$$

and H_∞ is by defintion the Shannon per symbol entropy associated with a source. Where

$$H_\infty = \lim_{n\to\infty}[-\frac{1}{n+1}\int \log_2 p(x_0, x_{-1}, \ldots, x_{-n})\mathrm{d}\mu(x_0, x_{-1}, \ldots, x_{-n})] \qquad 4.23$$

The theorem is then as follows

If $$f_N(X^N) = \frac{1}{N}\sum_{l=0}^{N-1} g_l(T^l(x_0)) = -\frac{1}{N}\log_2 p(x_{N-1}, x_{N-2}, \ldots, x_0) \qquad 4.24$$

Then $f_N(X^N) \rightarrow H_\infty$

Now we may write

$$\begin{aligned}\int_{\forall X^\infty} |f_N(X^N) - H_\infty|\,\mathrm{d}\mu(X^N) &\le \int_{\forall X^\infty} |\frac{1}{N}\sum_{l=0}^{N-1} g_l(T^l(x_0)) - \frac{1}{N}\sum_{l=0}^{N-1} g_\infty(T^l(x_0))|\,\mathrm{d}\mu(X^\infty) \\ &+ \int_{\forall X^\infty} |\frac{1}{N}\sum_{l=0}^{N-1} g_\infty(T^l(x_0)) - H_\infty|\,\mathrm{d}\mu(X^\infty) \\ &\le \frac{1}{N}\sum_{l=0}^{N-1}\Big(\int_{\forall X^\infty} |g_l(T^l(x_0)) - g_\infty(T^l(x_0))|\,\mathrm{d}\mu(X^\infty)\Big) \\ &+ \int_{\forall X^\infty} |\frac{1}{N}\sum_{l=0}^{N-1} g_\infty(T^l(x_0)) - H_\infty|\,\mathrm{d}\mu(X^\infty) \qquad 4.25\end{aligned}$$

The fact that we have an invariant function $g_\infty(.)$ in the second integral of the right hand side allows the ergodic theorem to be used here, thus it may be observed that this term tends to zero. This follows from the fact that the mean of the function $g_\infty(.)$ is the conditional entropy defined in equation 4.22 and is equal to H_∞.

The first term is described in McMillan's paper as the Cesaro mean, this is the mean in time of a generally decreasing sequence whose limit is zero. The limit of the sequence

$$\int_{\forall X^{\infty}} | g_l(T^l(x_0)) - g_{\infty}(T^l(x_0)) | \, \mathrm{d}\mu(X^{\infty}) \qquad 4.26$$

being zero follows from the fact of the convergence of conditional probabilities. The theorem on the convergence of conditional probabilities is proved in an appendix. Therefore the first term on the right hand side also converges to zero, proving the theorem.

The following corollary of the above statement is actually what is known as the AEP theorem.

Corollary. Given any $\delta>0$ there exists an integer N such that sequences of length greater than N fall into two classes, S_1 and $\bar{S}_1$. Class S_1 has a total probability mass greater than $1-\delta$ and class $\bar{S}_1$ a total probability mass less than δ, that is

$$\int_{Y^M \in S_1 \;\; M>N} \mathrm{d}\mu(Y^M) \geq 1-\delta \quad \text{and} \quad \int_{Y^M \in \bar{S}_1, \;\; M>N} \mathrm{d}\mu(Y^M) \leq \delta \qquad 4.27$$

Every sequence Y^M that belongs to class S_1 has almost the same probability of occurrence, this falls between the limits defined below.

$$2^{-M(H_{\infty}+\delta)} \leq p(Y^M) \leq 2^{-M(H_{\infty}-\delta)} \qquad 4.28$$

Proof. By the Chebychev inequality

$$\mathrm{Prob}\{|-\frac{1}{M}\log_2 p(Y^M)-H_\infty|>\delta\}$$

$$\leq \frac{1}{\delta}\mathrm{E}\{|-\frac{1}{M}\log_2 P(Y^M)-H_\infty|\} \qquad 4.29$$

By the statements preceeding this corollary it had been shown that

$$f_M(X^M) = -\frac{1}{M}\log_2 p(Y^M) \rightarrow H_\infty \qquad 4.30$$

Thus for any δ^2, $\exists$N such that $\forall$M > N

$$\int_{\forall Y^\infty} |-\frac{1}{M}\log_2 p(Y^M)-H_\infty| d\mu(Y^\infty) = \mathrm{E}\{|-\frac{1}{M}\log_2 p(Y^M)-H_\infty|\} \leq \delta^2 \qquad 4.31$$

Therefore

$$\mathrm{Prob}\{|-\frac{1}{M}\log_2 p(Y^M)-H_\infty|>\delta\} \leq \delta$$

But that set $\bar{S}_1$ is the set such

$$|-\frac{1}{M}\log_2 p(Y^M)-H_\infty|>\delta$$

This has total probability mass less than δ. Thus S_1 has probability mass greater than 1-δ and $\forall Y^M \in S_1$

$$-\delta \leq -\frac{1}{M}\log_2 p(Y^M)-H_\infty \leq +\delta \qquad 4.32$$

This concludes the proof.

This theorem was first noted by Shannon [Shannon-(1948a) theorem 3] where it was offered with a rather sketchy proof. The first thorough proof was given by McMillan [McMillan-(1953)], whose proof is followed here. The expansion of this to cover joint sources is described very thoroughly by Dobrushin [Dobrushin-(1963)]. For a treatment of the theorem in the context of ergodic theory see Billingsley [Billingsley-(1965)] pp 129-136.

4.5 Performance bounds for coding with a fidelity criterion.

The coding scheme for the situation where distortion is allowed is almost the same as that described in section 4.3. The difference here is that in searching the set of previous or previously coded outcomes Λ_R , we allow ourselves to code a block Y^{L_N} , if we find a $\hat{Y}^{L_N} \in \Lambda_R$, such that $d(Y^{L_N}, \hat{Y}^{L_N}) \leq d^*$; we declare coding a success and approximate Y^{L_N} by $\hat{Y}^{L_N}$. d^* is a distortion limit set beforehand. Let

$$\underset{R}{\text{Prob}}(\exists\ \hat{Y}^{L_N} : d(Y^{L_N}, \hat{Y}^{L_N}) \leq d^*_{L_N})$$

be the probability of finding, for a given sequence Y^{L_N} , an approximation within distortion d^* , in Λ_R. The probability of coding a particular sequence Y^{L_M} of length L_M, $M < N$ is the probability of observing Y^{L_M}, but not a longer sequence $Y^{L_{M+i}}$ within distortion d^*, in Λ_R. In briefer notation, this is

$$\underset{R}{\text{Prob}}[\{\exists\ \hat{Y}^{L_M} : d(Y^{L_M}, \hat{Y}^{L_M}) \leq d^*_{L_M}\} \bigcap \{\nexists\ \hat{Y}^{L_J} : d(Y^{L_J}, \hat{Y}^{L_J}) \leq d^*_{L_J},\ \forall J > M\}]$$

As before we attempt to bound the coding rate by finding bounds on the average length. This is given below.

$$\begin{aligned}
\overline{L} = \underset{\forall Y^{L_N}}{\text{E}} \Big(& \underset{R}{\text{Prob}}[\exists\ \hat{Y}^{L_N} : d(Y^{L_N}, \hat{Y}^{L_N}) \leq d^*_{L_N}].L_N \\
& + \underset{R}{\text{Prob}}[\{\exists\ \hat{Y}^{L_{N-1}} : d(Y^{L_{N-1}}, \hat{Y}^{L_{N-1}}) \leq d^*_{L_{N-1}}\} \bigcap \\
& \quad \{\nexists\ \hat{Y}^{L_N} : d(Y^{L_N}, \hat{Y}^{L_N}) \leq d^*_{L_N}\}].L_{N-1} \\
& + \underset{R}{\text{Prob}}[\{\exists\ \hat{Y}^{L_{N-2}} : d(Y^{L_{N-2}}, \hat{Y}^{L_{N-2}}) \leq d^*_{L_{N-2}}\} \bigcap \\
& \quad \{\nexists\ \hat{Y}^{L_I} : d(Y^{L_I}, \hat{Y}^{L_I}) \leq d^*_{L_I},\ \text{some } I > N-2\}].L_{N-2} \\
& + \ldots\ldots\ldots\ldots \\
& + \underset{R}{\text{Prob}}[\{\exists\ \hat{Y}^{L_2} : d(Y^{L_2}, \hat{Y}^{L_2}) \leq d^*_{L_2}\} \bigcap \\
& \quad \{\nexists\ \hat{Y}^{L_I} : d(Y^{L_I}, \hat{Y}^{L_I}) \leq d^*_{L_I},\ \text{some } I > 2\}].L_2 \\
& + \underset{R}{\text{Prob}}[\{\nexists\ \hat{Y}^{L_I} : d(Y^{L_I}, \hat{Y}^{L_I}) \leq d^*_{L_I},\ \text{some } I > 1\}].L_1 \Big)
\end{aligned} \qquad 4.33$$

The evaluation of these probabilities is difficult, if not impossible, therefore an assumption has to be made in order that we may proceed. Now

$$\begin{aligned}
&\underset{R}{\mathrm{Prob}}\Big(\{\exists\, \hat{Y}^{L_J} : d(Y^{L_J},\hat{Y}^{L_J}) \le d^{*}_{L_J}\} \bigcap \{\nexists\, \hat{Y}^{L_I} : d(Y^{L_I},\hat{Y}^{L_I}) \le d^{*}_{L_I},\ \text{some}\ I > J\}\Big)\\
&=\underset{R}{\mathrm{Prob}}\Big(\ \exists\, \hat{Y}^{L_J} : d(Y^{L_J},\hat{Y}^{L_J}) \le d^{*}_{L_J}\Big) -\\
&\quad\underset{R}{\mathrm{Prob}}\Big(\{\exists\, \hat{Y}^{L_J} : d(Y^{L_J},\hat{Y}^{L_J}) \le d^{*}_{L_J}\} \bigcap \{\exists\, \hat{Y}^{L_I} : d(Y^{L_I},\hat{Y}^{L_I}) \le d^{*}_{L_I},\ \text{some}\ I > J\}\Big)
\end{aligned} \qquad 4.34$$

Assumption. The quantity to the right hand side of the minus sign above, is to be simplified by the following assumption: If a block of length Y^{L_I} may be found, within some distortion d^*, in Λ_R , then a block Y^{L_J} of smaller length may be found to within distortion d^*, in Λ_R. In mathematical notation we replace

$$\underset{R}{\mathrm{Prob}}\Big(\{\exists\, \hat{Y}^{L_J} : d(Y^{L_J},\hat{Y}^{L_J}) \le d^{*}_{L_J}\} \bigcap \{\exists\, \hat{Y}^{L_I} : d(Y^{L_I},\hat{Y}^{L_I}) \le d^{*}_{L_I},\ \text{some}\ I > J\}\Big)$$

by

$$\underset{R}{\mathrm{Prob}}\Big(\exists\, \hat{Y}^{L_{J+1}} : d(Y^{L_{J+1}},\hat{Y}^{L_{J+1}}) \le d^{*}_{L_{J+1}}\Big)$$

We may therefore write the average length as

$$\begin{aligned}
\bar{L} = \underset{\forall Y^{L_N}}{\mathrm{E}}\Big(&\underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^{L_N} : d(Y^{L_N},\hat{Y}^{L_N}) \le d^{*}_{L_N}].(L_N - L_{N-1})\\
&+\underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^{L_{N-1}} : d(Y^{L_{N-1}},\hat{Y}^{L_{N-1}}) \le d^{*}_{L_{N-1}}].(L_{N-1} - L_{N-2})\\
&+\ldots\ldots\ldots\ldots\\
&+\underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^{L_2} : d(Y^{L_2},\hat{Y}^{L_2}) \le d^{*}_{L_2}].(L_2 - L_1) + L_1\Big)
\end{aligned} \qquad 4.35$$

We shall consider the particular case when $L_1=1$, $L_2=2,\ldots,L_N=N$. Then the average length may be rewritten as

$$\begin{aligned}\bar{L} = \mathrm{E}\Big(& \underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^N : d(Y^N,\hat{Y}^N) \le d^*_N] \\ & +\underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^{N-1} : d(Y^{N-1},\hat{Y}^{N-1}) \le d^*_{N-1}] \\ & + \dots\dots\dots \\ & +\underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^2 : d(Y^2,\hat{Y}^2) \le d^*_2] + 1\Big)\end{aligned} \qquad 4.36$$

Now we shall concentrate on the evaluation of the terms

$$\underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^I : d(Y^I,\hat{Y}^I) \le d^*_I]$$

It may be shown that as an "elementary" block size tends to infinity, for a given distortion value d*, some of these probability values tend to one and others to zero. As was done in section 4.4 where noiseless coding was considered, we define an "elementary" source symbol as a sequence of k original source symbols. Y^I therefore refers to kI original source symbols or I elementary source symbols. Now

$$\mathrm{Prob}[\exists\, \hat{Y}^I : d(Y^I,\hat{Y}^I) \le d^*_I] = 1 - \left(1 - \begin{array}{l}\text{Probability of observing a se-}\\ \text{quence within a region } d^*_I \text{ of } Y^I\\ \text{in one experiment conducted in the}\\ \text{space of previously coded symbols.}\end{array}\right)^R \qquad 4.37$$

Let the probability of observing a sequence within a region d^*_I of Y^I in any one experiment be $p(Y^I,d^*)$. Let $d^*_I=d^*$ for all I. We are concerned therefore with evaluating $1-[1-p(Y^I,d^*)]^R$. We may write

$$1-[1-p(Y^I,d^*)]^R = 1-\exp\{R\ln[1-p(Y^I,d^*)]\} \qquad 4.38$$

We shall now tie R to k in the following manner. Let $C^k = R$. Thus in implementation of the coding scheme, if one is unable to find a block of length greater than the "elementary" source symbol, ie. k original source symbols, then just one elementary source symbol is coded. This is done by quantising it to the nearest of C^k levels.

The number of quantisation levels is chosen so that, presuming a maximum ignorance or flat distribution between the amplitude extrema, the quantisation introduces less distortion than d*. Thus

$$1-\exp\{R\ln[1-p(Y^I,d^*)]\} = 1-\exp\{C^k\ln[1-p(Y^I,d^*)]\} \qquad 4.39$$

As was done in section 4.4 we look at what happens as k ⟶ ∞.

$$1-\exp\{C^k\ln[1-p(Y^I,d^*)]\}|_{k\to\infty} = 1-\exp\{C^k\frac{\frac{d}{dk}p(Y^I,d^*)}{(1-p(Y^I,d^*))\ln C}\}|_{k\to\infty}$$

4.40

by L'Hopital's rule. At this point we need to make use of the theorem of section 4.5.1. This says that the probability of observing any symbol, within the horizon of Y^N defined by d(X^N,Y^N) $\leq$ d*, is almost a constant for some Y^N, this constant being $2^{-NR(d^*)}$ and almost zero elsewhere, as N goes to infinity. R(d*) is the rate-distortion (r-d) function of the source X^N sequences, calculated as defined in section 4.5.1 part i. In keeping with custom the r-d function of section 4.5.1 is defined using napierian logarithms. The r-d value given here refers to that which would have been obtained if the log is to base 2. This is done here because it allows our results to be given in bits (instead of nats). Thus

$$\frac{d}{dk}p(Y^I,d^*) = \frac{d}{dk}2^{-IkR(d^*)} = -2^{-IkR(d^*)}.I.R(d^*)\ln 2 \qquad 4.41$$

$$1-\exp\{C^k\ln[1-p(Y^I,d^*)]\}|_{k\to\infty} = 1-\exp\{-C^k2^{-IkR(d^*)}\frac{IR(d^*)\ln 2}{(1-p(Y^I,d^*))\ln C}\}|_{k\to\infty}$$

4.42

$$\to 0 \quad \forall I > \frac{\log_2 C}{R(d^*)} \quad \text{and} \to 1 \quad \forall I < \frac{\log_2 C}{R(d^*)} \qquad 4.43$$

The average length is thus

$$\underset{\substack{\text{all uncoded}\\ Y^I \text{ sequences}}}{\mathrm{E}} \{1 + \underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^2 : d(Y^2, \hat{Y}^2) \le d_2^*] + \underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^3 : d(Y^3, \hat{Y}^3) \le d_3^*] + \ldots \mathrm{etc}\}$$

We know that for every J, pr($\exists Y^J$:d(Y^J,Y^J) $\leq$ d*) is very close to either unity or zero, depending on the value of J and the actual Y^J. There are some Y^J values of almost zero probability of being observed in Λ_R, to within distortion d*, even though others of the same length are observable within distortion d* . The average length may thus be written as

$$\underset{\substack{\text{all uncoded}\\ Y^I \text{ sequences}}}{\mathrm{E}} \{1 + \underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^2 : d(Y^2, \hat{Y}^2) \le d_2^*] + \cdots + \underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^I : d(Y^I, \hat{Y}^I) \le d_I^*] \}$$

Where I is the largest integer smaller than $\log_2$C/R(d*). Let the region (Y^I,ϵ) be the set of all I length sequences within distortion ϵ of Y^I. Let p_u(Y^J,ϵ) be the probability of observing, in one experiment, an uncoded sequence within distortion ϵ of Y^J. Now, let us concentrate on the expectation of observing a particular length.

$$\text{Suppose } A_J = \underset{\substack{\text{all uncoded}\\ Y^J \text{ sequences}}}{\mathrm{E}} \{\underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^J : d(Y^J, \hat{Y}^J) \le d_J^*]\}$$

$$= \lim_{\epsilon \to 0} \sum_{\substack{\text{all disjoint}\\ (Y^J,\epsilon) \text{ regions}}} \underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^J : d(Y^J, \hat{Y}^J) \le d_J^*] P_u(Y^J, \epsilon)$$

$$= \lim_{\epsilon \to 0} \sum_{\substack{\text{all members of a}\\ \text{set } S_1 \text{ of disjoint}\\ (Y^J,\, \epsilon) \text{ regions.}}} \underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^J : d(Y^J, \hat{Y}^J) \le d_J^*] P_u(Y^J, \epsilon)$$

$$+ \lim_{\epsilon \to 0} \sum_{\substack{\text{all members of the}\\ \text{complementary set } \bar{S}_1\\ \text{of disjoint } (Y^J,\epsilon)\\ \text{regions.}}} \underset{R}{\mathrm{Prob}}[\exists\, \hat{Y}^J : d(Y^J, \hat{Y}^J) \le d_J^*] P_u(Y^J, \epsilon) \qquad 4.44$$

The set of (Y^J,ϵ) regions is divided into two subsets for this reason: It may be envisaged that for ~~a~~ some particular (Y^J,ϵ) region, the probability of finding in Λ_R, a $\hat{Y}^J$ such that $d(Y^J,\hat{Y}^J) \leq d^*$ for every $Y^J \in (Y^J,\epsilon)$ is almost zero. It should be recalled that in the space of the coded signal, there are d^* spheres, with almost constant probability and other d^* spheres of almost zero probability mass. These two classes are represented by S_1 and $\bar{S}_1$. Then

$$A_J \approx \sum_{\substack{\text{all disjoint}\\ (Y^I,\epsilon)\text{ regions}}} P_u(Y^I,\epsilon) + \delta \qquad 4.45$$

δ is an arbitrarily small quantity representing the total contribution from the space $\bar{S}_1$. The above equation follows from the fact that for all (Y^I,ϵ) regions which form the centres of observable d^* spheres in the space of the coded signal, $pr(\exists\hat{Y}^I : d(Y^I,\hat{Y}^I) \leq d) \simeq 1$.

The next point to be established is the total mass of the set S_1. This may be observed to be arbitrarily close to unity, since any (Y^I,ϵ) region that occurs with any frequency, is coded within distortion d^* and included in the space of coded symbols. Therefore any (Y^I,ϵ) region of non-zero probability mass (measured according to the probability distribution function defined on the uncoded signal space), will be observed within distortion d^*, in the set of previously coded symbols with finite probability. Therefore $A_J \simeq 1$. It may be concluded that

$$\bar{L} = 1 + \sum_{j=2}^{I} A_J \approx\geq \frac{\log_2 C}{R(d^*)} - 1 \qquad 4.46$$

The coding rate is

$$\frac{\log_2 R + \log_2 N}{k\hat{L}} \approx\leq \frac{R(d^*)\log_2 N}{k(\log_2 C - R(d^*))} + \frac{R(d^*)}{1-\frac{R(d^*)}{\log_2 C}} \qquad 4.47$$

As k becomes very large, the first term on the right hand side goes to zero, giving

$$\text{Rate} \approx \leq \frac{R(d^*)}{1 - \frac{R(d^*)}{\log_2 C}} \qquad 4.48$$

It should be noted that the R(d*) value refered to here, is not that for the original source but that for the signal of previously coded symbols.

The following is a non-rigorous discussion of the relationship between the r-d value of the coded and original signals. The r-d values for the original and coded signals are refered to as $R_u(d^*)$ and $R_c(d^*)$ respectively. The coding scheme described may be looked at in the following manner: Consider N separate spaces, each containing C^k points strategically positioned, the spaces of smallest and largest dimensions being respectively k and kN dimensional. The C^k points are the members of the set Λ_R of previously encoded symbols. In implementing the coding scheme, one observes blocks of length kL, $1 \lesssim L \lesssim N$. These blocks are samples from a space of dimension kL. We try to encode a block with one of the C^k points. Always trying the largest dimension first. The foregoing has indicated that for long sequences, blocks of one particular length are almost always encoded. In other words coding is done in almost exclusively one particular dimension. This is the kI-th dimension where

$$I > \frac{\log_2 C}{R(d^*)} - 1 \qquad 4.49$$

It is only on very few occasions that the length of a coded block is less than kI. We may consider the occasion of coding a shorter block as replacing a member of the space of C^k points with a new point. Since this is done a very small proportion of the time, the C^k points may be considered to span the space of kI length blocks that occur and kI is the largest dimension space which will be spanned by C^k symbols.

Next we look at the meaning of the rate-distortion value at distortion d*, for a source. Each of the N dimensional regions of distortion radius d* (d* spheres), have almost a constant probability mass of $2^{-NR(d^*)}$ or almost zero probability mass as $N \rightarrow \infty$. In other words, it is possible to span the N dimensional space of all occurrences to within a distortion d^* by packing this space with spheres, each of distortion radius arbitrarily close to d* and of probability arbitrarily close to $2^{-NR(d^*)}$. Indeed $2^{NR(d^*)}$ spheres are sufficient. An alternative way of looking at the r-d function is this: Suppose one has $2^{NR(d^*)}$ spheres and is allowed to choose a dimension so that these d^* spheres will span the space of all possible outcomes, then N is the maximum dimension that one could pick. This is because if one could pick the dimension N+J, and span this space, then the coding rate in bits per source symbol would be $\frac{N}{N+J}$ R(d*), which would mean that R(d*) is not the infimum per symbol rate. This contradicts the definition of the r-d function.

It is known that the space of previously coded sequences of block length kI, has a rate, for zero distortion of a very small quantity greater than $\log_2 C^k$. This is because almost all sequences in the set of previously coded symbols of length kI are obtained originally from other previously encoded symbols. These may be transmitted with $\log_2 C^k$ symbols. The small number ϵ represents the additional rate associated with new additions to the set of C^k points of previously coded data, so that the following statement holds:

$$kIR_C(d=0) = \log_2(C^k) + \epsilon \qquad 4.50$$

Also

$$kIR_C(d=d^*) = kIR_C(d^*) \le R_C(d=0) = \log_2(C^k) + \epsilon \qquad 4.51$$

It is known that C^k points almost totally span the space of all kI length blocks that may be observed in the uncoded signal space. The fact that a longer block length may not be found that enables the space of blocks of this length to be spanned by C^k points allows us to presume that

$$2^{kIR_u(d^*)} \approx C^k \quad \text{and} \quad kIR_u(d^*) \approx \log_2 C^k \qquad 4.52$$

Thus using 4.51,

$$\begin{aligned} kIR_C(d^*) &\le \log_2(C^k) + \epsilon \\ &\approx \le kIR_u(d^*) + \epsilon \\ R_C(d^*) &\approx \le R_u(d^*) + \epsilon_1 \end{aligned} \qquad 4.53$$

ϵ_1 arbitrarily small. This allows us to say that the coding rate is approximately as given below:

$$\text{rate} \approx \le \frac{R_u(d^*)}{1 - \frac{R_u(d^*)}{\log_2 C}} \qquad 4.54$$

This concludes the discussion.

The next section gives the details of the theorem for the convergence of the probability mass of d^* spheres.

4.5.1 A partition theorem for sequences observed with finite precision

Theorem. Consider a source with an outcome space Ω, a Borel-Σ field $\mathfrak{F}$ constructed from this space and a probability measure q(.) defined on this field. The product space $\overset{N}{\underset{i=1}{X}}\Omega$ is partitioned into two disjoint regions S and $\bar{S}$ as N tends towards infinity. The probability of observing any sequence X^N, emitted from this source, that is close to some Y^N, within distortion d*, tends towards the constant exp(-NR(d*)) for all Y^N belonging to S and zero for all Y^N belonging to $\bar{S}$ as N tends to infinity. The distortion class considered, consists of the absolute and square error, single letter distortion measures. That is the distortion $d(X^N,Y^N)$ between the two sequences X^N and Y^N is $\frac{1}{N}\sum|x_i-y_i|$ or $\frac{1}{N}\sum(x_i-y_i)^2$. The theorem also holds for any other single letter distortion measures with a difference distortion measure $d(|x-y|)$ such that the Fourier transform of the function exp{-d(.)} exists everywhere and is strictly non-zero for all frequencies. R(d*) is the rate-distortion value of the source at the distortion d*.

Proof. The proof of this theorem is rather long and involves several intermediate theorems. The proof is similar to that employed for the Shannon-McMillan-Brieman Asymptotic Equipartition theorem. Shannon [Shannon 1959] proved a similar theorem as lemma 1 in the referenced paper. In that paper however, he considers only the single letter r-d function. Here, the limiting r-d function is considered and it is shown that the joint density functions that solve the r-d optimisation problem, converge for a class of distortion functions.

The proof given here relies greatly upon the ergodic theorem and the convergence theorem for conditional probability. The proof will be given in the following order.

i) The definition of the rate-distortion function

ii) A sketch of the proof

iii) A proof of the convergence of the probability functions which give the rate-distortion function for a given d*.

i) The rate-distortion function R(d*) for a given distortion value d*, is the minimum rate at which a source may be coded so that the distortion is less than d*. In mathematical notation,

$$R(d^{*}) = \inf_{p(y|x)} \int q(x) \int p(y|x) \ln[\frac{p(y|x)}{\int q(u) p(y|u) \mathrm{d}u}] \mathrm{d}y \, \mathrm{d}x \qquad 4.56$$

such that

$$d^{*} \geq \int q(x) \int p(y|x) d(x,y) \mathrm{d}y \, \mathrm{d}x \qquad 4.57$$

and

$$1 = \int p(y|x) \mathrm{d}y \quad \forall x \qquad 4.58$$

x is an outcome of the source random variable and y is an outcome of the approximation or reproduction random variable. q(x) is the source probability density function and p(y|x) is the conditional density function governing the approximation of the outcome x of the source by the value y. The x and y may be vectors in which case the integral signs represent multiple integrals. The solution of this optimisation problem involves rewriting the objective function to include the constraints, with a Lagrange multiplier ρ for the constraint of equation 4.57 and a Lagrange multiplier function f(x) for the multiple constraints of equation 4.58. Let

$$R(\mathrm{p},\rho,f) = \inf_{p(y|x),\rho,f(x)} \{ \int \int p(y,x) [\ln \frac{p(y|x)}{w(y)} + \rho d(x,y) + \ln \frac{f(x)}{q(x)}] dy\, dx \} \tag{4.59}$$

be the new objective function, where w(y) is defined as below

$$w(y) = \int p(y|x) q(x) dx \tag{4.60}$$

On differentiating with respect to p(y|x) and setting the result to zero, we obtain the following equations.

$$0 = [\ln \frac{p(y|x)q(x)}{f(x)w(y)} + \rho d(x,y)] \quad \text{and thus}$$

$$1 = \int f(x) \exp\{-\rho d(x,y)\} dx \tag{4.61}$$

$$\frac{q(x)}{f(x)} = \int w(y) \exp\{-\rho d(x,y)\} dy \tag{4.62}$$

After solving equations 4.61 and 4.62, the p(y|x) which yeilds the minimum is defined as.

$$p(y|x) = \frac{f(x)w(y)}{q(x)} \exp\{-\rho d(x,y)\} \tag{4.63}$$

These results were first obtained by Shannon [Shannon-(1948b) section 27]. For a more complete discussion see the following; [Gallager-(1968)] p457 or [Berger-(1971)] pp29-32 and pp88-90

ii) A sketch of the proof. It will be shown that as N tends to infinity,

$\frac{1}{N}\ln[\frac{p_N(X^N|Y^N)}{q_N(X^N)}]$ tends to $R_\infty(d^*)$.

Where $R_\infty(d^*)$ is the rate-distortion function value at distortion d^* when block sizes considered tend towards infinity. $p_N(X^N|Y^N)$ is the conditional density function associated with minimising the rate for a given distortion, when the block size considered is N.

It will also be shown that the distortion $d(X^N,Y^N)$ between any pair of sequences X^N and Y^N tends towards d^* as the block size N goes to infinity Both these proofs presume that the joint source (X^N,Y^N) is ergodic.

We shall proceed by assuming that the above two statements are correct. This will be demonstrated later. With these assumptions we develop the proof that the space of the X^N random variable, for a given Y^N divides into two regions as defined before. Now for a given Y^N,

$$\text{Prob}\{X^N : |\frac{1}{N}\ln[\frac{p_N(X^N|Y^N)}{q_N(X^N)}] - R_\infty(d^*)| > \epsilon\}$$

$$\leq \frac{1}{\epsilon}\text{E}\{|\frac{1}{N}\ln[\frac{p_N(X^N|Y^N)}{q_N(X^N)}] - R_\infty(d^*)|\} \qquad 4.64$$

by the Chebyshev inequality. But by the fact that

$\frac{1}{N}\ln[\frac{p_N(X^N|Y^N)}{q_N(X^N)}]$ may be made as close to $R_\infty(d^*)$ as wanted by taking N large enough, $\text{E}\{|\frac{1}{N}\ln[\frac{p_N(X^N|Y^N)}{q_N(X^N)}] - R_\infty(d^*)|\}$

may be made smaller than δ^2. The following statement may be claimed: For a given Y^N, there exists an N such that

$$\text{Prob}\{X^N : |\frac{1}{N}\ln[\frac{p_N(X^N|Y^N)}{q_N(X^N)}] - R_\infty(d^*)| > \delta\} \leq \delta \qquad 4.65$$

There are therefore for a given Y^N and N large enough, two disjoint regions S_1 and $\bar{S}_1$ in the space spanned by X^N where this is true:

$$\int_{S_1(Y^N)} p_N(X^N|Y^N)\mathrm{d}X^N > 1-\delta$$

$$\int_{\bar{S}_1(Y^N)} p_N(X^N|Y^N)\mathrm{d}X^N \leq \delta \qquad 4.66$$

In the region S_1

$$\left|\frac{1}{N}\ln[\frac{p_N(X^N|Y^N)}{q_N(X^N)}] - R_\infty(d^*)\right| \leq \delta \qquad 4.67$$

and in the region $\bar{S}_1(Y^N)$

$$\left|\frac{1}{N}\ln[\frac{p_N(X^N|Y^N)}{q_N(X^N)}] - R_\infty(d^*)\right| > \delta \qquad 4.68$$

In the region $S_1(Y^N)$ we have

$$q_N(X^N)\exp\{N(R_\infty(d^*)-\delta)\} \leq p_N(X^N|Y^N) \leq q_N(X^N)\exp\{N(R_\infty(d^*)+\delta)\} \qquad 4.69$$

Now integration of $p(X^N|Y^N)$ over the region $S_1(Y^N)$ gives $1-\delta$. Thus

$$\int_{S_1(Y^N)} q_N(X^N)\exp\{N(R_\infty(d^*)-\delta)\}\mathrm{d}X^N \leq 1-\delta \leq \int_{S_1(Y^N)} q_N(X^N)\exp\{N(R_\infty(d^*)+\delta)\}\mathrm{d}X^N$$

and

$$(1-\delta)\exp\{-N(R_\infty(d^*)+\delta)\} \leq \int_{S_1(Y^N)} q_N(X^N)\mathrm{d}X^N \leq (1-\delta)\exp\{-N(R_\infty(d^*)-\delta)\} \qquad 4.70$$

What has been shown so far is that for any $\delta > 0$, we may find an N large enough so that the next few statements are true:

Statement 1. A given Y^N is to be approximated by X^N. Governing the set of possible X^N that may be used to approximate Y^N is the conditional density function $p(X^N|Y^N)$.

Statement 2. The space of all X^N used to approximate this Y^N may be partitioned into two regions $S_1(Y^N)$ and $\bar{S}_1(Y^N)$. The total mass of the conditional density function in the region $S_1(Y^N)$ is greater than $1-\delta$.

Statement 3. The probability of observing an outcome from the region $S_1(Y^N)$ is $\int_{S_1(Y^N)} q_N(X^N)\mathrm{d}X^N$

which is almost constant for all Y^N and equal to $\exp\{N(R_\infty(d^*) \pm \delta)\}$.

Next we have to show that the region $S_1(Y^N)$ corresponds to the region where distortion $d(X^N,Y^N) \leq d + \epsilon$, where ϵ is an arbitrarily small value greater than zero.

Given a Y^N, we have;

$$\text{Prob}\{X^N : |d(X^N,Y^N)-d^*| > \delta\} \leq \frac{1}{\delta}\int_{-\infty}^{\infty} p_N(X^N|Y^N)|d(X^N,Y^N)-d^*|\mathrm{d}X^N \tag{4.71}$$

Now by the ergodic theorem, it is known that $d(X^N,Y^N) \longrightarrow d^*$ and

$$\int_{-\infty}^{\infty} p_N(X^N|Y^N)|d(X^N,Y^N)-d^*|\mathrm{d}X^N$$

may be made less than δ^2 by the choice of an N adequately large. Thus $\exists$N such that

$$\text{Prob}\{X^N : |d(X^N,Y^N)-d^*| > \delta\} \leq \delta \tag{4.72}$$

This implies that for a given Y^N, there exists in the space of X^N values a region $S_2(Y^N)$ of probability mass greater than $1-\delta$, that is

$$\int_{S_2(Y^N)} p_N(X^N|Y^N)\mathrm{d}X^N \geq 1-\delta \tag{4.73}$$

where within this region,

$$|d(X^N,Y^N)-d^*| \leq \delta \qquad 4.74$$

Thus importantly for us within this region $d(X^N,Y^N) \leq d^*+\delta$.

Combining this and the set of previous statements, we have the fact that for any $\delta > 0$ and $\epsilon > 0$, we can find an N large enough so that for any Y^N, we can find regions of conditional probability mass

$$\int_{S_1(Y^N)} p_N(X^N|Y^N)\mathrm{d}X^N, \quad \int_{S_2(Y^N)} p_N(X^N|Y^N)\mathrm{d}X^N \qquad 4.75$$

respectively greater than $1-\epsilon$ and $1-\delta$. For region $S_1(Y^N)$ $\int_{S_1} q_N(X^N)\mathrm{d}X^N$ is almost constant and $\forall Y$ almost equal to $\exp(-NR(d^*))$. For region S_2, $d(X^N,Y^N) \leq d^*+\delta$. From these it may be said that region $S_1 \cap S_2$ is of mass almost 1. This is because

$$\int_{S_1 \cap S_2} p_N(X^N|Y^N)\mathrm{d}X^N = \int_{S_1} p_N(X^N|Y^N)\mathrm{d}X^N - \int_{S_1 \cap \bar{S}_2} p_N(X^N|Y^N)\mathrm{d}X^N \qquad 4.76$$

Now

$$\int_{S_1 \cap \bar{S}_2} p_N(X^N|Y^N)\mathrm{d}X^N \leq \int_{\bar{S}_2} p_N(X^N|Y^N)\mathrm{d}X^N \leq \delta \qquad 4.77$$

Thus

$$\int_{S_1 \cap S_2} p_N(X^N|Y^N)\mathrm{d}X^N \geq \int_{S_1} p_N(X^N|Y^N)\mathrm{d}X^N - \delta \geq 1-\epsilon-\delta \qquad 4.78$$

This concludes our proof.

To recapitulate therefore the theorem may be summarised thus:

For a given Y^N, the X^N space may be partitioned into two regions S and $\bar{S}$, $S=S_1(Y^N)\cap S_2(Y^N)$, which has almost unity probability mass, that is, almost X^N which are used to approximate Y^N fall into this region. These regions S for any Y^N have the dual property that all X^N in a region S differs from Y^N by at most $d^*+\delta$, and any regions S may be have a member observed with almost constant probability, $\exp(-NR(d^*))$.

iii) Now we proceed to the formal proofs of the convergence of

$$\ln[\frac{p_N(X^N|Y^N)}{q_N(X^N)}] \quad \text{to} \quad R(d^*)$$

First we shall give some definitions. Let

$$R_N(d^*) = \inf_{p'_N(Y^N|X^N)}[\frac{1}{N}\int_{\infty}^{\infty} q(X^N)(\int_{\infty}^{\infty} p'_N(Y^N|X^N)\ln[\frac{p'_N(Y^N|X^N)}{w'_N(Y^N)}]\mathrm{d}Y^N)\mathrm{d}X^N]$$

$$w'_N(Y^N) = \int_{\infty}^{\infty} p'_N(Y^N|X^N)q(X^N)\mathrm{d}X^N \qquad 4.79$$

$q(X^N)$ is the probability density function associated with the X^N random variable. The minimisation is subject to the following conditions:

$$\int q(X^N)(\int p'_N(Y^N|X^N)d(X^N,Y^N)\mathrm{d}Y^N)\mathrm{d}X^N \leq d^*$$

$$\int p'_N(Y^N,X^N)\mathrm{d}Y^N = 1, \qquad \forall X^N \qquad 4.80$$

It will be assumed that the conditional density function $p'_N(Y^N|X^N)$ that solves the minimisation problem is $p_N(Y^N|X^N)$. Let $R(d^*) = \lim_{N\to\infty} R_N(d^*)$. Next we define a function

$$g_{Nl}(x_k,y_k) = \ln[\frac{p_N(x_k,y_k\,|\,x_{k-1},\ldots,\,x_{k-l}\,;\,y_{k-1},\ldots,y_{k-l})}{w_N(y_k\,|\,y_{k-1}\ldots,y_{k-l})q(x_k\,|\,x_{k-1},\ldots,x_{k-l})}] \qquad 4.81$$

This function will be abbreviated as

$$g_{Nl}(x_k,y_k) = \ln[\frac{p_N(x_k,y_k|X^l_{k-1},Y^l_{k-1})}{w_N(y_k|Y^l_{k-1})q(x_k|X^l_{k-1})}] \qquad 4.82$$

The superscripts ℓ indicate the lengths of the blocks X^l_{k-1} and Y^l_{k-1} on which the random variables x_k and y_k are conditioned. The coordinates of the first random variables of the blocks X^l_{k-1} and Y^l_{k-1} are indicated by their subscripts. It may be noted that,

$$\ln[\frac{p_N(Y^N,X^N)}{w_N(Y^N)q(X^N)}] = g_{N,N-1}(x_N,y_N) + g_{N,N-2}(x_{N-1},y_{N-1}) + \ldots. + g_{N,0}(x_1,y_1)$$

$$= \sum_{i=0}^{N-1} g_{Ni}(T^i(x_1,y_1)) \qquad 4.83$$

where $T^i(.)$ is a time shift operation and

$$R_N(d^*) = E\{\frac{1}{N}\sum_{i=0}^{N-1} g_{Ni}(T^i(x_i,y_i))\} \qquad 4.84$$

Thus

$$R(d^*) = \lim_{N\to\infty} R_N(d^*) \qquad 4.85$$

and

$$\lim_{N\to\infty}\{\frac{1}{N}\sum_{i=0}^{N-1} g_{Ni}(T^i(x_i,y_i))\} \quad \rightarrow \quad R(d^*)$$

What we shall show is that as N grows larger and larger, the apparently time averaged quantity $g_{Ni}(x_1,y_1)$ tends towards its actual average

In which case we may write that

$$\frac{1}{N}\ln[\frac{p_N(X^N,Y^N)}{w_N(Y^N)q(X^N)}] \rightarrow R(d^*)$$

Thus the quantity inside the log tends towards a constant as N ⟶ ∞. Formally we shall show that

$$|R(d^*) - \frac{1}{N}\sum_{i=0}^{N-1} g_{Ni}(T^i(x_1,y_1))| \rightarrow 0$$

almost everywhere or

$$\lim_{N\rightarrow\infty}\int|R(d^*) - \frac{1}{N}\sum_{i=0}^{N-1} g_{Ni}(T^i(x_1,y_1))|\mathrm{d}P(X^\infty,Y^\infty) = 0 \qquad 4.86$$

We rely greatly on the ergodic theorem, thus the joint process (X^N,Y^N) must be ergodic.

We know that

$$\begin{aligned}\int|R(d^*) - \frac{1}{N}\sum_{i=0}^{N-1} g_{Ni}(T^i(x_1,y_1))|\mathrm{d}P &\leq \int|R(d^*) - \frac{1}{N}\sum_{i=0}^{N-1} g_{\infty\infty}(T^i(x_1,y_1))|\mathrm{d}P \\ &+ \int|\frac{1}{N}\sum_{i=0}^{N-1} g_{\infty\infty}(T^i(x_1,y_1)) - \frac{1}{N}\sum_{i=0}^{N-1} g_{Ni}(T^i(x_1,y_1))|\mathrm{d}P \\ &\leq \int|R(d^*) - \frac{1}{N}\sum_{i=0}^{N-1} g_{\infty\infty}(T^i(x_1,y_1))|\mathrm{d}P \\ &+ \frac{1}{N}\sum_{i=0}^{N-1}\int|g_{\infty\infty}(T^i(x_1,y_1)) - g_{Ni}(T^i(x_1,y_1))|\mathrm{d}P\end{aligned} \qquad 4.87$$

$g_{\infty\infty}(T^i(x_1,y_1))$ is an invariant function with N. The probability density function associated with this is $p_\infty(X^\infty,Y^\infty)$, corresponding to the case that solves the minimisation problem for an infinitely long sequence. This defines the first subscipt of the function

$g_{\infty\infty}(x_1,y_1)$. The second subscipt means that the conditioning on the random variables $T^i(x_1,y_1)$ is the infinite X^∞ and Y^∞ sequences that happened prior to $T^i(x_1,y_1)$. Unlike $g_{Ni}(.,.)$, the function $g_{\infty\infty}(.,.)$ is N invariant and i invariant. Therefore there follows immediately from the ergodic theorem the fact that

$$\int |E\{g_{\infty\infty}(x_1,y_1)\} - \frac{1}{N}\sum_{i=0}^{N-1} g_{\infty\infty}(T^i(x_1,y_1))| dP \rightarrow 0$$

provided the joint sequences (X^∞,Y^∞) are ergodic. Concerning the term

$$E\{g_{\infty\infty}(x_1,y_1)\}$$

all that is required is that we show that

$$R(d^*) = E\{g_{\infty\infty}(x_1,y_1)\} \qquad 4.88$$

After this we concentrate on the second term of the right hand side of equation 4.87 and show that this goes to zero.

Lemma. Let

$$g_{NN-1}(x_1,y_1) = \ln[\frac{p_N(x_0,y_0 \mid x_{-1},\ldots,x_{-N+1}; y_{-1},\ldots,y_{-N+1})}{w_N(y_0 \mid y_{-1},\ldots,y_{-N+1})q(x_0 \mid x_{-1},\ldots,x_{-N+1})}] \qquad 4.89$$

then $\lim_{N\rightarrow\infty} E\{g_{NN-1}(x_1,y_1)\} = R(d^*)$.

Proof. The proof is simillar to Fano's proof of the convergence of conditional entropy [Fano-(1961)] pp86-88. We will first show that $E(g_{NN-1}(x_1,y_1)) \leq R_N(d^*)$. We then show that for any $\delta > 0$, however small we may find an N large enough so that $E(g_{NN-1}(x_1,y_1)) \geq R_N(d^*) - \delta$. Making δ tend to zero and hence N tend toward infinity, we conclude that $\lim_{N\rightarrow\infty} E\{g_{NN-1}(x_1,y_1)\} = R(d^*)$

Now

$$N R_N(d^*) = \mathrm{E}\{\ln\Big(\frac{p_N(x_N, y_N | X_{N-1}^{N-1}, Y_{N-1}^{N-1})}{w_N(y_N | Y_{N-1}^{N-1}) q(x_N | X_{N-1}^{N})}\Big)\} + \mathrm{E}\{\ln\Big(\frac{p_N(Y_{N-1}^{N-1}, X_{N-1}^{N-1})}{w_N(Y_{N-1}^{N-1}) q(X_{N-1}^{N-1})}\Big)\}$$

4.90

Next we note that the joint density function $p_N(Y^N | X^N)$ satisfies the distortion constraint. That is

$$\int q(X_N^N) \int p_N(Y_N^N | X_N^N)[\frac{1}{N} \sum_{i=1}^{N} d(x_i, y_i)] \mathrm{d}X^N \, \mathrm{d}Y^N \leq d^*$$

$$\text{or } \frac{1}{N} \int \int p_N(x_i, y_i) d(x_i, y_i) \mathrm{d}x_i \, \mathrm{d}y_i \leq d^*$$ 4.91

By measure invariance with time shifts

$$\frac{1}{N-1} \sum_{i=1}^{N-1} \int \int p_N(x_i, y_i) d(x_i, y_i) \mathrm{d}x_i \mathrm{d}y_i \leq d^*$$ 4.92

hence the marginal density function $p_N(X^{N-1} | Y^{N-1})$ also satisfies the distortion constraint. It is known that of all the joint density functions of length N-1, the one which gives the smallest rate, in addition to satisfying the distortion constraint is $p_{N-1}(X^{N-1}, Y^{N-1})$. Thus

$$\mathrm{E}\{\ln\Big(\frac{p_{N-1}(Y^{N-1}, X^{N-1})}{w_{N-1}(Y^{N-1}) q(X^{N-1})}\Big)\} = (N-1) R_{N-1}(d^*) \leq \mathrm{E}\{\ln\Big(\frac{p_{N-1}(Y^{N-1}, X^{N-1})}{w_N(Y^{N-1}) q(X^{N-1})}\Big)\}$$

4.93

Then

$$N R(d^*) = \mathrm{E}\{g_{N,N-1}(x_1, y_1)\} + \mathrm{E}\{\ln \frac{p_N(Y^{N-1}, X^{N-1})}{w_N(Y^{N-1}) q(X^{N-1})}\}$$

$$\geq \mathrm{E}\{g_{N,N-1}(x_1, y_1)\} + (N-1) R_{N-1}(d^*)$$ 4.94

also we know that

$$R_{N-1}(d^*) \geq R_N(d^*)$$ 4.95

and hence

$$N R_N(d^*) \geq \mathrm{E}\{g_{N,N-1}(x_1,y_1)\} + (N-1)R_N(d^*) \qquad 4.96$$

Therefore

$$\mathrm{E}\{g_{N,N-1}(x_1,y_1)\} \leq R_N(d^*) \qquad 4.97$$

Now we show that there exists an N such that

$$\mathrm{E}\{\ln\left(\frac{p_N(x_N,y_N|X^{N-1}_{N-1},Y^{N-1}_{N-1})}{w_N(y_N|Y^{N-1}_{N-1})q(x_N|X^{N}_{N-1})}\right)\} \geq R_N(d^*)-\epsilon \quad \text{for any } \epsilon > 0 \qquad 4.98$$

This will be done by considering the numerator and denominators separately. Consider the numerator

$$\mathrm{E}(\ln p_N(x_N,y_N|X^{N-1}_{N-1},Y^{N-1}_{N-1})) = -h_N(y_N,x_N\,|X^{N-1},Y^{N-1}) \qquad 4.99$$

Now $h_N(y_k\ ,x_k\,|Y^{k-1},X^{k-1})$ forms a non-increasing sequence with k $\forall$ k<N (see Berger [Berger-(1971)] problem 4.1 page 140). Thus

$$\begin{aligned} h_N(X^N,Y^N) &= \mathrm{E}(-\ln p_N(X^N,Y^N)) \\ &= \mathrm{E}(-\ln p_N(x_N,y_N|X^{N-1}_{N-1},Y^{N-1}_{N-1})\,) \\ &+ \mathrm{E}(-\ln p_N(x_{N-1},y_{N-1}|X^{N-2}_{N-2}\ ,Y^{N-2}_{N-2}\ )\,) \\ &+ \ldots\ldots. \\ &+ \mathrm{E}(-\ln p_N(x_2,y_2|x_1,y_1)\,) + \mathrm{E}(-\ln p_N(x_1,y_1) \end{aligned} \qquad 4.100$$

But since

$$\mathrm{E}(-\ln p_N(x_N,y_N|X^{N-1}_{N-1},Y^{N-1}_{N-1})) \leq \mathrm{E}(-\ln p_N(x_k,y_k|X^{k-1}_{k-1},Y^{k-1}_{k-1})) \qquad \forall\, k<N$$

$$h_N(X^N,Y^N) = \mathrm{E}(-\ln p_N(X^N,Y^N)) \geq N h_N(x_N,y_N\,|X^{N-1},Y^{N-1}) \qquad 4.101$$

Thus

$$-\mathrm{E}(\ln p_N(x_N,y_N|X^{N-1}_{N-1},Y^{N-1}_{N-1})) \leq \frac{1}{N}h_N(X^N,Y^N)$$

$$\mathrm{E}\{\ln[p_N(x_N,y_N|X^{N-1}_{N-1},Y^{N-1}_{N-1})]\} \geq \frac{1}{N}\,\mathrm{E}\{\ln p_N(X^N,Y^N)\} \qquad 4.102$$

The next thing to prove is that the numerator function will obey this; for any $\epsilon > 0$ an M may be found such that for all N > M

$$\mathrm{E}\{\ln[\frac{1}{w_N(y_N|Y_{N-1}^{N-1})q(x_N|X_{N-1}^{N-1})}] \geq \frac{h_N(Y^N)+h_N(X^N)}{N} - \epsilon \qquad 4.103$$

Where

$$h_N(Y^N) = -\mathrm{E}\{\ln w_N(Y^N)\} = -\sum_{k=1}^{N} \mathrm{E}\{\ln w_N(y_k|Y_{k-1}^{k-1})\} \qquad 4.104$$

$$h_N(Y^N) = -\mathrm{E}\{\ln q_N(X^N)\} = -\sum_{k=1}^{N} \mathrm{E}\{\ln q(x_k|X_{k-1}^{k-1})\} \qquad 4.105$$

For a given N, $-\mathrm{E}\{\ln[w_N(y_k|Y^{k-1})]\}$ is a non-increasing function with k, $\forall k < N$. Thus the following inequality is true.

$$h_N(Y^N) \leq -\mathrm{E}\{\ln w_N(y_j|Y_{j-1}^{j-1})\}(N-j+1) + (j-1)\mathrm{E}\{\ln\frac{1}{w_N(y_1)}\}, \quad j < N \qquad 4.106$$

Similarly

$$h_N(X^N) \leq -\mathrm{E}\{\ln q(x_k|X_{k-1}^{k-1})\}(N-k+1) + (k-1)\mathrm{E}\{\ln\frac{1}{q(x_1)}\}, \quad k < N \qquad 4.107$$

Hence

$$-\mathrm{E}\{\ln(q(x_k|X_{k-1}^{k-1})w_N(y_k|Y_{k-1}^{k-1})\} \geq \{\frac{h_N(Y^N)+h_N(X^N)}{N}\frac{N}{N-k+1}$$

$$-\frac{k-1}{N-k+1}\mathrm{E}\{\ln\frac{1}{q(x_1)w_N(y_1)}\} \qquad 4.108$$

Our next statement relies on the convergence of conditional entropy. Consider a ratio M to k, let $\alpha = \frac{M}{k}$ say. For any finite α however large, an M may be found that makes k large enough so that for any $\delta > 0$ we have

$$-\mathrm{E}\{\ln[w_M(y_M|Y^{M-1}_{M-1})q(x_M|X^{M-1}_{M-1})]\} \geq -\mathrm{E}\{\ln[w_M(y_{\frac{M}{\alpha}}|Y^{\frac{M}{\alpha}-1}_{\frac{M}{\alpha}-1})]\} - \mathrm{E}\{\ln[q(x_{\frac{M}{\alpha}}|X^{\frac{M}{\alpha}-1}_{\frac{M}{\alpha}-1})]\} - \delta \qquad 4.109$$

What we are saying is that due to the convergence of conditional entropy, for M sufficiently large, the value $k=\frac{M}{\alpha}$ will be large enough such that

$$-\mathrm{E}\{\ln[w_M(y_{\frac{M}{\alpha}}|Y^{\frac{M}{\alpha}-1}_{\frac{M}{\alpha}-1}).q(x_{\frac{M}{\alpha}}|X^{\frac{M}{\alpha}-1}_{\frac{M}{\alpha}-1})]\}$$

is very nearly equal to

$$-\mathrm{E}\{\ln[w_M(y_M|Y^{M-1}_{M-1})q(x_M|X^{M-1}_{M-1})]\}$$

Going back to 4.109 then we have,

$$-\mathrm{E}\{\ln[w_M(y_M|Y^{M-1}_{M-1})q(x_M|X^{M-1}_{M-1})]\} \geq -\mathrm{E}\{\ln[w_M(y_{\frac{M}{\alpha}}|Y^{\frac{M}{\alpha}-1}_{\frac{M}{\alpha}-1})]\} - \mathrm{E}\{\ln[q(x_{\frac{M}{\alpha}}|X^{\frac{M}{\alpha}-1}_{\frac{M}{\alpha}-1})]\} - \delta$$

4.110

Using 4.108

$$-\mathrm{E}\{\ln[w_M(y_M|Y^{M-1}_{M-1})q(x_M|X^{M-1}_{M-1})\} \geq [\frac{h_M(X^M)+h_M(Y^M)}{M}][\frac{M}{M-\frac{M}{\alpha}+1}]$$

$$-\frac{(\frac{M}{\alpha}-1)}{(M-\frac{M}{\alpha}+1)}\mathrm{E}\{\ln[\frac{1}{q(x_1)w_M(y_1)}]\}-\delta$$

$$\geq \frac{h_M(Y^M)+h(X^M)}{M} - \frac{(1-\frac{\alpha}{M})}{(\alpha\frac{M+1}{M}-1)}.\,\mathrm{E}\{\ln\frac{1}{q(x_1)w_N(y_1)}\}$$

$$-\delta$$

4.111

Allowing α to go as large as we want and M to go to infinity gives

$$\mathrm{E}\{\ln[\frac{1}{w_M(y_M|Y^{M-1}_{M-1})q(x_M|X^{M-1}_{M-1})}]\} \geq \frac{h_M(X^M)+h_M(Y^M)}{M} - \epsilon - \delta \qquad 4.112$$

for any ϵ , $\delta > 0$. Considering this in addition to the statement arising from 4.102 that is

$$\mathrm{E}\{\ln[p_M(x_M,y_M|X^{M-1}_{M-1},Y^{M-1}_{M-1})]\} \geq \frac{1}{M}\mathrm{E}\{\ln[p_M(Y^M,X^M)]\} \qquad 4.113$$

gives

$$\mathrm{E}\{g_{M,M}(x_1,y_1)\} = \mathrm{E}\{\ln[\frac{p_M(x_M,y_M|X^{M-1}_{M-1},Y^{M-1}_{M-1})}{w_M(y_M|Y^{M-1}_{M-1})q(x_M|X^{M-1}_{M-1})}]$$

$$\geq \frac{1}{M}\mathrm{E}\{\ln[\frac{p_M(X^M,Y^M)}{w_M(Y^M)q(X^M)}]\} - \delta_1$$

$$= R_M(d^*) - \delta_1 \qquad 4.114$$

for some M, for any $\delta_1 > 0$.

This concludes the proof.

Thus refering back to inequality 4.87, by the ergodic theorem, the first integral of the right hand side tends towards zero as $N \rightarrow \infty$. Next we consider the second term, the summed integrals of equation 4.87.

In this section it is shown that most of the members of the k-varying sequence of functions $g_{kk-1}(,.,), g_{kk-2}(,.,),....$ tend towards the invariant function $g_{\infty\infty}(,.,)$. This, in conjunction with the fact that the expectation of $g_{\infty\infty}(,.,)$ is $R(d^*)$ allows us to prove the theorem. In fact all we require is that for any $\epsilon > 0$, an N may be found such that for all $k > N$, the following equation is true.

$$\int \left| \frac{1}{k} \sum_{i=1}^{k} \{ g_{ki}(T^i(x_1,y_1)) - g_{\infty\infty}(T^i(x_1,y_1)) \} \right| dP \le \epsilon \qquad 4.115$$

As usual P is a probability measure defined on the field $\mathfrak{F}$ constructed on the product outcome space for (X^∞, Y^∞). Firstly a sketch of the proof will be given and details filled in later. A very important theorem concerning the convergence of conditional probabilities is of great importance to this proof. This theorem says that the sequence of probability distribution functions $p(y|A_1)$, $p(y|A_1,A_2)$, $p(y\ |A_1,A_2,A_3)$ etc. converge. The convergence is in this sense.

$$\int |p(y|A_1,A_2,\ldots,A_N) - p(y|A_1,A_2,\ldots,A_\infty)| dP \rightarrow 0$$

P is a probability measure over the infinite sequence of outcomes $A_1, A_2, \ldots, A_\infty$. This tells us that for large enough $g_{ki}(T^i(x,y))$ will be close to $g_{kk}(T^i(x,y))$ for most of the i so that

$$\frac{1}{k}\sum_{i=1}^{k} g_{ki}(T^i(x_1,y_1)) \approx \frac{1}{k}\sum_{i=1}^{k} g_{kk}(T^i(x_1,y_1)) \qquad 4.116$$

It is required to be shown that the sequence $g_{kk}(T^i(x,y))$ converges with k, for every X^k, Y^k pair of random variables. This is different from what was shown in the previous section where the expectation of $g_{kk}(,.,)$ as $k \to \infty$ was shown to converge to the rate-distortion value for a given distortion. In this case, it is important that for almost each point X^k, Y^k of non-zero measure $g_{kk}(T^i(x_1,y_1))$ converges.

Now

$$g_{kk}(\,.\,) - g_{k+1,k+1}(\,.\,) = \ln[\frac{p_k(X_0^k, Y_0^k)}{w_k(Y_0^k)q(X_0^k)}] - \ln[\frac{p_k(X_{-1}^{k-1}, Y_{-1}^{k-1})}{w_k(Y_{-1}^{k-1})q(X_{-1}^{k-1})}]$$

$$-\left(\ln\frac{p_{k+1}(X_0^{k+1}, Y_0^{k+1})}{w_{k+1}(Y_0^{k+1})q(X_0^{k+1})}] - \ln[\frac{p_{k+1}(X_{-1}^{k}, Y_{-1}^{k})}{w_{k+1}(Y_{-1}^{k})q(X_{-1}^{k})}]\right) \qquad 4.117$$

We know that

$$\frac{p_k(X_0^k, Y_0^k)}{w_k(Y_0^k)q(X_0^k)} = \frac{f_k(X_0^k)}{q(X_0^k)}\exp\{-\rho_k d(X_0^k, Y_0^k)\} \quad \text{and} \qquad 4.118$$

$$\frac{p_{k+1}(X_0^{k+1}, Y_0^{k+1})}{w_{k+1}(Y_0^{k+1})q(X_0^{k+1})} = \frac{f_{k+1}(X_0^{k+1})}{q(X_0^{k+1})}\exp\{-\rho_{k+1} d(X_0^{k+1}, Y_0^{k+1}) \qquad 4.119$$

(from 4.63)

Now

$$p_k(X_{-1}^{k-1}, Y_{-1}^{k-1}) = \int\int p_k(X_0^k, Y_0^k)\mathrm{d}x_0\,\mathrm{d}y_0$$

$$= w_k(Y_{-1}^{k-1})\exp\{-(\frac{k-1}{k})\rho_k d(X_{-1}^{k-1}, Y_{-1}^{k-1})\}\,.$$

$$\int\int w_k(y_0 \mid Y_{-1}^{k-1})f_k(X_0^k)\exp\{-\frac{\rho_k}{k}d(x_0,y_0)\}\mathrm{d}x_0\,\mathrm{d}y_0 \qquad 4.120$$

where

$$d(X_0^k, Y_0^k) = \frac{1}{k}\sum_{i=0}^{k-1} d(x_i, y_i) \qquad 4.121$$

Therefore

$$\frac{p_k(X_{-1}^{k-1}, Y_{-1}^{k-1})}{w_k(Y_{-1}^{k-1})} = \zeta_k(X_{-1}^{k-1}, Y_{-1}^{k-1})\exp\{-\frac{k-1}{k}\rho_k d(X_{-1}^{k-1}, Y_{-1}^{k-1})\} \qquad 4.122$$

where

$$\zeta_k(X_{-1}^{k-1}, Y_{-1}^{k-1}) = \int\int w_k(y_0 \,|\, Y_{-1}^{k-1}) f_k(X_0^k)\exp\{-\frac{\rho_k}{k} d(x_0, y_0)\}\,\mathrm{d}x_0\,\mathrm{d}y_0 \qquad 4.123$$

also

$$\frac{p_{k+1}(X_{-1}^{k}, Y_{-1}^{k})}{w_{k+1}(Y_{-1}^{k})} = \zeta_{k+1}(X_{-1}^{k}, Y_{-1}^{k})\exp\{-\frac{k}{k+1}\rho_{K+1} d(X_{-1}^{k}, Y_{-1}^{k})\} \qquad 4.124$$

where

$$\zeta_{k+1}(X_{-1}^{k}, Y_{-1}^{k}) = \int\int w_{k+1}(y_0 \,|\, Y_{-1}^{k}) f_{k+1}(X_0^{k+1})\exp\{-\frac{\rho_k}{k+1} d(x_0, y_0)\}\,\mathrm{d}x_0\,\mathrm{d}y_0 \qquad 4.125$$

Thus

$$\begin{aligned} g_{kk}(\,.\,) - g_{k+1,k+1}(\,.\,) = {} & \{\ln\frac{f_k(X_0^k)}{q(X_0^k)} - \ln\frac{\zeta_k(X_{-1}^{k-1}, Y_{-1}^{k-1})}{q(X_{-1}^{k-1})}\} \\ & - \{\ln\frac{f_{k+1}(X_0^{k+1})}{q(X_0^{k+1})} - \ln\frac{\zeta_{k+1}(X_{-1}^{k-1}, Y_{-1}^{k-1})}{q(X_{-1}^{k})}\} \\ & + \frac{(k-1)}{k}\rho_k d(X_{-1}^{k-1}, Y_{-1}^{k-1}) - \rho_k d(X_0^k, Y_0^k) + \rho_{k+1} d(X_0^{k+1}, Y_0^{k+1}) \\ & - \frac{k}{k+1}\rho_{k+1} d(X_{-1}^{k}, Y_{-1}^{k}) \end{aligned} \qquad 4.126$$

Now we can concentrate on establishing the convergence of the term

$$\ln\frac{f_k(X_0^k)}{f_{k+1}(X_0^{k+1})} - \ln\frac{\zeta_k(X_{-1}^{k-1}, Y_{-1}^{k-1})}{\zeta_{k+1}(X_{-1}^{k}, Y_{-1}^{k})}$$

At this point we have to forego the luxury of generality. This is because the general distortion measure does not allow us to describe the character of the functions $f_J(X_0^J)$ or $f_{J+1}(X_{-1}^{J+1})$. We shall therefore restrain ourselves to the absolute difference and square difference distortion measures defined as follows.:

$$d(X^N,Y^N) = \frac{1}{N}\sum_{n=0}^{N} d(|x_n - y_n|),\ \text{where}\ d(z) = z^2\ \text{or}\ |z| \qquad 4.127$$

These distortion measures are chosen because they define kernels $\exp(-\rho_N d(X^N,Y^N))$ which are convolutional. They allow the solution of the integral equations

$$\int f_N(X^N)\exp\{-\rho_N d(X^N,Y^N)\}\,\mathrm{d}X^N = 1 \qquad 4.128$$

$$\int f_{N+1}(X^{N+1})\exp\{-\rho_{N+1} d(X^{N+1},Y^{N+1})\}\,\mathrm{d}X^{N+1} = 1 \qquad 4.129$$

(see 4.61)

in a straigthforward manner.

Theorem. The eigenfunctions of the kernel

$$K(X^N,Y^N) = \exp\{-\rho_N d(X^N,Y^N)\} \qquad 4.130$$

where

$$d(X^N,Y^N) = \frac{1}{N}\sum_{N=1}^{N-1} d(|x_n - y_n|), \qquad d(x) = x^2\ \text{or}\ |x| \qquad 4.131$$

are $\exp\{-j\cdot\sum_{i=1}^{N}\omega_i x_i\}$

The eigenfunctions form a continuous spectrum, are real and non-zero.

Proof. Let

$$\begin{aligned} A_N(\omega^N,Y^N) &= \int_{-\infty}^{\infty} \exp(-j\sum_{i=1}^{N}\omega_i, x_i)K(X^N,Y^N)\mathrm{d}X^N \\ &= \int_{-\infty}^{\infty} \exp(-j\sum_{i=1}^{N}\omega_i, x_i)\,\mathcal{K}(\frac{1}{N}\sum_{i=1}^{N} d'(|x_i - y_i|))\,\mathrm{d}X^N \end{aligned} \qquad 4.132$$

Then

$$A_N(\omega^N, Y^N) = \int_{-\infty}^{\infty} \exp(-j \sum_{i=1}^{N} \omega_i(x_i + y_i))\, \mathcal{K}(\frac{1}{N} \sum_{i=1}^{N} d'(|x_i|)))\, dX^N \qquad 4.133$$

Also let $$\lambda(\omega^N) = \int_{-\infty}^{\infty} \exp(-j \sum_{i=1}^{N} \omega_i x_i)\, \mathcal{K}(\frac{1}{N} \sum_{i=1}^{N} d'(|x_i|))\, dX^N \qquad 4.134$$

be the Fourier transform of the function $K\{ \frac{1}{N} \sum_{i=1}^{N} d'(x_i)\}$.

Then

$$A(\omega^N, Y^N) = \exp(-j \sum_{i=1}^{N} \omega_i, y_i).\lambda(\omega^N) \qquad 4.135$$

By the symmetry of the kernel we may write

$$\lambda(\omega^N) = 2\int_{0}^{\infty} \cos(\sum_{i=1}^{N} \omega_i x_i)\, \mathcal{K}(\frac{1}{N} \sum_{i=1}^{N} d'(|x_i|))\, dX^N \qquad 4.136$$

thus showing that $\lambda(\omega^N)$ is real. We write the homogeneous Fredholm integral of the second kind as,

$$\int_{-\infty}^{\infty} \exp(-j \sum_{i=1}^{N} \omega_i x_i) K(X^N, Y^N)\, dX^N = \lambda(\omega^N).\exp(-j \sum_{i=1}^{N} \omega_i y_i) \qquad 4.137$$

Thus proving that the eigenvalues are real and form a continuous spectrum and the eigenfunctions are the exponential functions $\exp\{-j \cdot \sum_{i=1}^{N} \omega_i\, x_i\}$. That the eigenvalues $\lambda(\omega^N)$ are strictly non-zero is shown as follows. It is known that

$$\lambda(\omega^N) = \int_{-\infty}^{\infty} \exp(-j \sum_{i=1}^{N} \omega_i x_i)\, \mathcal{K}(\sum d'(|x_i|))\, dX^N$$

$$= \prod_{i=1}^{N} \int_{-\infty}^{\infty} \exp(-j \sum_{i=1}^{N} \omega_i x_i)\, \mathcal{K}(\frac{1}{N} d'(|x_i|))\, dx_i \qquad 4.138$$

For the square distortion measure, $K\{ \frac{1}{N} d'(x_i)\} = \exp\{- \frac{\rho_N}{N} x_i^2 \}$

It is known that the Fourier transform of $\exp\{-ax^2\}$ is $\sqrt{\frac{\pi}{a}} \exp\{- \frac{\omega^2}{4a}\}$. Thus $\lambda(\omega^N) = \prod_{i=1}^{N} \sqrt{\frac{\pi N}{\rho_N}} \exp\{- \frac{N\omega_i^2}{4\rho_N}\}$ This function is always strictly positive.

For the absolute difference distortion measure

$$\mathcal{K}(\frac{1}{N}d'(|x_i|)) = \exp\{-\frac{\rho_N}{N}|x_i|\} \qquad 4.139$$

We know that the Fourier transform of the function exp{-a|x|} is $\dfrac{2a}{a^2+\omega^2}$. Thus $\lambda(\omega^N) = \prod_{i=1}^{N} \dfrac{2N\rho_N}{(\rho_N + N^2\omega_i^2)}$

This is also strictly positive for all ω^N. This concludes the proof.

Now $$1 = \int f(X^N)\exp\{-\rho_N d(X^N, Y^N)\}\mathrm{d}X^N \qquad 4.140$$

Thus $$0 = \int \{f(X^N) - \frac{1}{\int\exp\{-\rho_N d(U^N, Y^N)\}\,\mathrm{d}U^N}\}\exp\{-\rho_N d(X^N, Y^N)\}\,\mathrm{d}X^N$$

$$\int \{f(X^N) - \frac{1}{\int\exp\{-\frac{\rho_n}{N}\sum_{i=1}^{N} d'(|u_i|)\}\,\mathrm{d}U^N}\}\exp\{-\rho_N d(X^N, Y^N)\}\,\mathrm{d}X^N \qquad 4.141$$

But if all the eigenvalues of the kernel are non-zero, the above can only be true if the function being transformed by the kernel is zero. Therefore for the two distortion measures we can say that the only function $f_N(X^N)$ that solves the equation

$$\int_{-\infty}^{\infty} f_N(X^N)\exp\{-\rho_N d(X^N, Y^N)\}\,\mathrm{d}X^N = 1 \qquad 4.142$$

is the constant

$$1/\int_{-\infty}^{\infty} \exp\{-\frac{\rho}{N}\sum_{i=1}^{N} d'(|x_i|)\}\,\mathrm{d}X^N$$

For the square difference distortion measure,

$$\int_{-\infty}^{\infty} \exp\{-\frac{\rho_N}{N}d'(|X^N|)\}\,\mathrm{d}X^N = \prod_{i=1}^{N}\int_{-\infty}^{\infty}\exp\{-\frac{\rho_N}{N}x_i^2\}\,\mathrm{d}x_i$$

$$= \left(\sqrt{\frac{\pi N}{\rho_N}}\right)^N \qquad 4.143$$

Thus

$$f_N(X^N) = \Big(\sqrt{\frac{\rho_N}{\pi N}}\Big)^N \qquad 4.144$$

Similarly

$$f_{N+1}(X^{N+1}) = \Big(\sqrt{\frac{\rho_{N+1}}{\pi N+1}}\Big)^{N+1} \qquad 4.145$$

Now we are in a position to investigate the character of the functions $\zeta_N(.,.)$ and $\zeta_{N+1}(.,.)$

$$\zeta_{N+1}(X^N_{-1}, Y^N_{-1}) = \int\int w_{N+1}(y_0 \mid Y^N_{-1}) f_{N+1}(X_0^{N+1}) \exp\{-\frac{\rho_{N+1}}{N+1}(x_0, y_0)^2\} \, \mathrm{d}x_0 \, \mathrm{d}y_0$$

$$= \Big(\sqrt{\frac{\rho_{N+1}}{\pi(N+1)}}\Big)^N \qquad 4.146$$

Also

$$\zeta_N(X^{N-1}_{-1}, Y^{N-1}_{-1}) = \Big(\sqrt{\frac{\rho_N}{\pi N}}\Big)^{N-1} \qquad 4.147$$

Therefore

$$|g_{NN} - g_{N+1,N+1}| = |\{\ln[\frac{p_N(Y_0^N, X_0^N)}{q(X_0^N) w_N(Y_0^N)}] - \ln[\frac{p_N(Y^{N-1}_{-1}, X^{N-1}_{-1})}{q(X^{N-1}_{-1}) w_N(Y^{N-1}_{-1})}]\}$$

$$-\{\ln[\frac{p_{N+1}(Y_0^{N+1}, X_0^{N+1})}{q(X_0^{N+1}) w_{N+1}(Y_0^{N+1})}] - \ln[\frac{p_{N+1}(Y^N_{-1}, X^N_{-1})}{q(X^N_{-1}) w_{N+1}(Y^N_{-1})}]\}|$$

$$\leq |\{\ln[f_N(X^N)] - \ln[\zeta_N(X^{N-1}_{-1}, Y^{N-1}_{-1})]\} - \{\ln[f_{N+1}(X^{N+1})] - \ln[\zeta_{N+1}(X^N_{-1}, Y^N_{-1})]\}|$$

$$+ |\ln[q(x_0 \mid X^N_{-1})] - \ln[q(x_0 \mid X^{N-1}_{-1})]| + |\frac{\rho_N}{N} - \frac{\rho_{N+1}}{N+1}| d(x_0, y_0)$$

$$= \frac{1}{2}|\ln[\frac{(N+1)\rho_N}{N\rho_{N+1}}]| + |\ln[q(x_0 \mid X^N_{-1})] - \ln[q(x_0 \mid X^{N-1}_{-1})]| + |\frac{\rho_N}{N} - \frac{\rho_{N+1}}{N+1}| d(x_0, y_0)$$

4.148

Thus

$$\|g_{NN} - g_{N+1,N+1}\|_1 = \int |g_{NN} - g_{N+1,N+1}| \mathrm{d}P$$
$$\leq \frac{1}{2}|\ln\frac{(N+1)\rho_N}{N\rho_{N+1}}| + \|\ln[q(x_0|X^N_{-1})] - \ln[q(x_0|X^{N-1}_{-1})]\|_1 + |\frac{\rho_N}{N} - \frac{\rho_{N+1}}{N+1}|d^*$$

4.149

The third term may instantly be recognised as going to zero as N goes to infinity. By the convergence of conditional probability the second term may also be observed to tend to zero. Next we need to show that the first term tends to zero. This is done by showing that ρ_N and ρ_{N+1} tend towards each other as N becomes bigger thereby sending $\frac{(N+1)\rho_N}{N\rho_{N+1}}$ to one and the first term to zero. Before the theorem for the convergence of the sequence $\rho_1, \rho_2, \ldots.$ is given, we shall look at the case where an absolute difference distortion measure is used.

For the absolute difference distortion measure,

$$f_N(X^N_0) = \left(\frac{\rho_N}{2N}\right)^N, \quad f_{N+1}(X^{N+1}_0) = \left(\frac{\rho_{N+1}}{2(N+1)}\right)^{N+1} \qquad 4.150$$

and

$$\zeta_N(X^{N-1}_{-1}, Y^{N-1}_{-1}) = \left(\frac{\rho_N}{2N}\right)^{N-1} \text{ and } \zeta_{N+1}(X^N_{-1}, Y^N_{-1}) = \left(\frac{\rho_{N+1}}{2(N+1)}\right)^N$$

4.151

Then

$$\|g_{NN} - g_{N+1,N+1}\|_1 \leq |\ln\frac{(N+1)\rho_N}{N\rho_{N+1}}| + \|\ln[q(x_0|X^N_{-1})] - \ln[q(x_0|X^{N-1}_{-1})]\|_1 + |\frac{\rho_N}{N} - \frac{\rho_{N+1}}{N+1}|d^*$$

4.152

Lemma. The sequence of numbers $\rho_1, \rho_2, \rho_3, \ldots.$ for a given distortion converges.

Proof. The value $-\rho_i$ is the gradient of the rate-distortion function obtained by considering i length sequences, at the distortion value d^*. Obviously the sequence of r-d functions obtained by considering successively larger blocks, converges. To prove this lemma therefore, we have to prove two lemmas the first is that the $-\rho_i$ are the gradients of the r-d functions at the distortion value d^*; the second is that the convergence of the rate distortion functions imply the convergence of their gradient functions almost everywhere.

Lemma. Suppose that the joint density function $p_N^*(Y^N,X^N)$ that enables one to achieve the minimum rate for a given distortion d^*, is given by

$$p_N(Y_N,X_N) = \frac{w_N(Y_N)f_N(X^N)}{q(X^N)}\exp\{-\rho_N d(X^N,Y^N)\} \qquad 4.153$$

Then the number $-\rho_N$ is the gradient of the r-d function at this distortion.

Proof. (From [Gallager 1968] p457 section 9.4; [Berger 1971] theorem 2.5.1 p33)

We need the fact that the rate distortion function is a convex non-increasing function and almost everywhere differentiable. The argument then goes as follows. We know that at distortion d^*, the minimum rate is defined as $R_N(d^*)$ where

$$R_N(d^*) + \rho_N d^* = \inf_{p_N'(Y^N|X^N)\ \&\ \lambda:\tilde{d} \le d^*} \{\frac{1}{N} I(X^N,Y^N) + \lambda\tilde{d}\} \qquad 4.154$$

λ is a Lagrange multiplier to take care of the distortion constraint. $\bar{d}$ and I are the distortion and the mutual information respectively, obtained when the conditional density function is $p_N'(Y^N|X^N)$. We repeat that $R_N(d^*)$ is the infimum rate that may be obtained for all distortion values less than d^*. For any $p_N'(Y^N|X^N)$ a pair of values I and $\bar{d}$ for mutual information and distortion are obtained, where by definition, $\forall \bar{d} < d^*$,

$$R_N(d^*) + \rho_N d^* \le \frac{1}{N} I(X^N, Y^N) + \rho_N \bar{d} \qquad 4.155$$

We can draw a line with slope $-\rho_N$ between the points $(d^*, R_N(d^*))$ and $(0, \{R_N(d^*) + \rho_N d^*\})$ as shown in figure 4.1. This represents for all values $\bar{d} < d^*$, a lower bound on the r-d function. This is because for any value of distortion, say $\bar{d}'$, that is less than d^* and has an associated infimum rate $R_N(\bar{d}')$, we have by equation 4.154

$$R_N(d^*) + \rho_N d^* \le R_N(\bar{d}') + \rho_N \bar{d}' \qquad 4.156$$

Therefore $R_N(d^*)$ always lies above that line. The next thing is to ascertain what happens for distortion values greater than d^*. Here the objective of the optimisation problem is changed. We try to find the minimum distortion that may be attained for a rate less than $R_N(d^*)$. Consequently the Lagrange multiplier operates upon $R_N(d^*)$.

$$r.R_N(d^*) + \Delta = \inf_{p_N'(Y^N|X^N)\ \&\ \lambda : \frac{1}{N} I \le R_N(d^*)} \left(\lambda \{ \frac{1}{N} I(X^N, Y^N) \} + \bar{d} \right) \qquad 4.157$$

By definition, the minimisation problem done this way will lead to the same value of distortion, that is $\Delta = d^*$. Hence

$$r.R_N(d^*) + d^* = \inf_{(.)} \left(\lambda \{ \frac{1}{N} I(X^N, Y^N) \} + \bar{d} \right) \qquad 4.158$$

But this may be rewritten as

$$R_N(d^*) + \frac{1}{r}d^* = \inf_{(.)}\left(\frac{1}{N}I(X^N,Y^N) + \frac{1}{\lambda}\bar{d}\right) \qquad 4.159$$

The minimisation of the right hand side will give

$$R_N(d^*) + \frac{1}{r}d^* \leq R_N(\bar{d}'') + \frac{1}{r}\bar{d}'' \qquad 4.160$$

for all values of rate and distortion such that $R_N(\bar{d}'') = R_N(d^*)$. The inclusion of the value $\bar{d}''=d^*$ gives the equality and hence $r=1/\rho_N$. It may therefore be said that for all values of rate $R_N(\bar{d}'')$ less than $R_N(d^*)$, the rate distortion function is greater than the line of slope $-\rho_N$ going from $(d^*,R(d^*))$ to $(\{d^*+\frac{1}{\rho_N}R(d^*)\},0)$ as shown in figure 4.2. Therefore the rate distortion function is greater than the line of slope $-\rho_N$ which touches it a $(d^*,R(d^*))$. By the convexity of the r-d function (see [Berger 1971] theorem 2.4.1, p27; [Gallager 1968]; [Shannon 1959]) this line should be a tangent and hence the gradient of the r-d function at distortion d^* is $-\rho_N$.

Lemma. By the convexity and monotonicity of the sequence of rate distortion functions, the sequence of r-d functions obtained by considering successively larger block sizes, have gradient functions that converge almost everywhere.

Proof. To prove this we need the following two facts:

1) The sequence of r-d functions converge almost everywhere, that is for all d^*: $0 < d^* < d_{max}$. For any subregion, $x < d^* < x+\delta$, and for any $\epsilon > 0$, an N may be found such that for all $n > N$, $|R_N(d^*)-R_{N+1}(d^*)| \leq \epsilon$

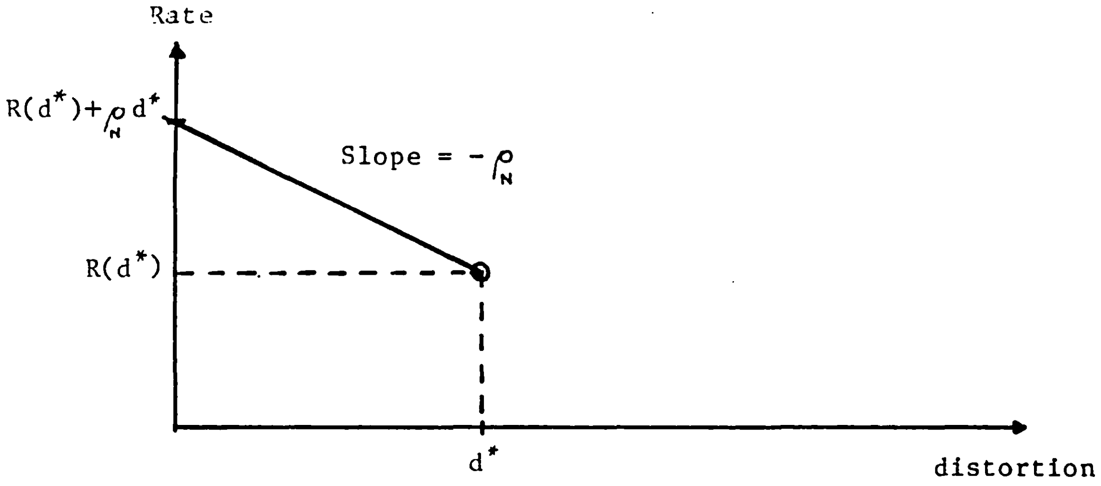


Figure 4.1

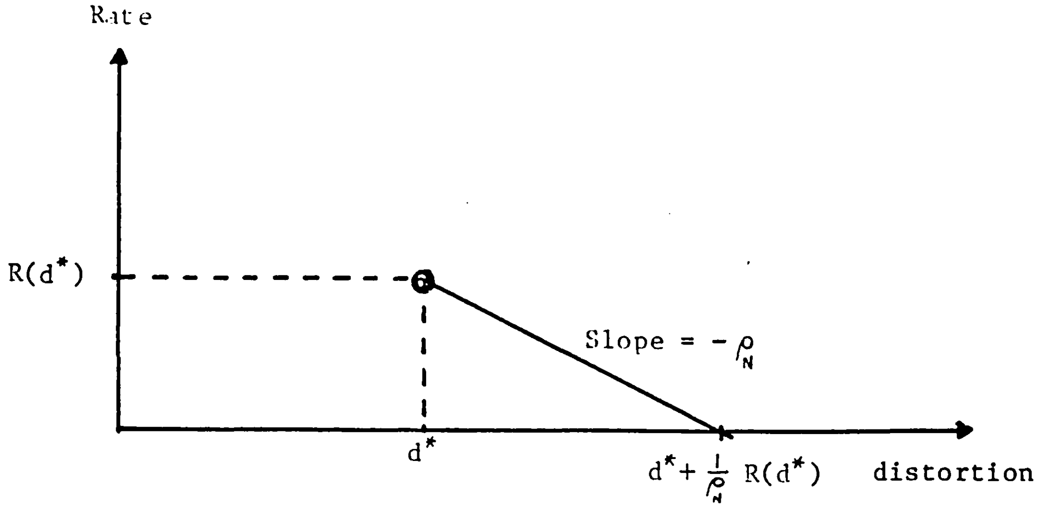


Figure 4.2

2) The r-d function must have a continuous slope, for all values of distortion except $d^* = 0$ or d_{max} .

The two conditions are proved to be true in [Gallager-1968] pp 491 and 492 and p463 and [Berger 1971] p463. Armed with these two we proceed as follows: Consider an x, ϵ, δ and n so that $|R_N(d^*) - R_{N+1}(d^*)| \leq \epsilon$. Over this region an upper bound on the absolute value for the difference between the gradient functions $R'_N(d\ )$ and $R'_{N+1}(d\ )$ will be established

Since the gradient functions are continu ous, for any n the numbers

$$R''_N(d^*) = \frac{R'_N(d^*) - R'_N(d^* + \zeta)}{\zeta}, \qquad R''_{N+1}(d^*) = \frac{R'_{N+1}(d^*) - R'_{N+1}(d^* + \zeta)}{\zeta}$$

exist and are bounded for any $\zeta > 0$. Now Let

$$z'(d^*) = R'_N(d^*) - R'_{N+1}(d^*) \qquad 4.161$$

Then since the function $z(d^*) = R_N(d^*) - R_{N+1}(d^*)$ is bounded by $\pm\epsilon$, within the region in question,

$$z(d^*) = \int_x^{d^*} z'(v)\mathrm{d}v + \epsilon_0 \qquad 4.162$$

is bounded by $\pm\epsilon$. Let the maximum $|R''_N(d^*) - R''_{N+1}(d^*)|$ be z''_m (over all ζ and d^*) maximum instantaneous value of z'_m may be observed to be $\sqrt{2\epsilon z''_m}$. Now since z''_m is finite, ϵ may be made as small as possible by increasing N thus making z' as small as one wants. This concludes the proof.

4.6 Conclusion and Discussion

In this chapter a theoretical discussion of the performance of the MPPCD scheme was presented. As usual, analysis of this scheme as it might be practically implemented is not feasible. The operation of the scheme as some particular parameter, in this case the block size is pushed towards infinity, was studied. This is instructive as far as understanding the capabilities of the scheme are concerned.

Analysis was relatively straigthforward for the noiseless coding situation. It was shown that for sources with large redundancy (the entropy is much less than $\log_2 C$, C being the source alphabet size), the coding rate for this scheme approaches the Shannon entropy value. For sources with little redundancy, at the expense of trying to code with very many possible block sizes, the scheme could be made to perform at close to the Shannon entropy value of the source.

Analysis was undertaken for the case of coding with distortion. This was considerably less straigthforward, compared with the noiseless coding case. A few assumptions were made concerning the source. These, in addition to the development of a theorem concerning the probability mass functions of long sequences from ergodic sources observed with finite precision, allowed analysis to continue. At that point the discussion had to be conducted along more heuristic lines, it was then argued that the coding scheme performs in a similar manner to its performance in noiseless coding. For sources with a rate distortion value, at distortion d , which is much less than $\log_2 C$ (C is the number of levels in a uniform quantisation scheme that achieves a distortion of less than d), the system is efficient. For sources where this is otherwise, performance may be made tighter by using more alternative block sizes.

In the analysis of the situation where coding is with respect to a distortion measure, a theorem was developed which said the following. For ergodic sources, the probability mass associated with the N dimensional regions of distortion less than d is almost of constant value $2^{-NR(d^*)}$ for some regions and almost zero for all other regions. The theorem was first noted by Shannon. The proof offered for this however, involved only the use of the joint density function which solved the variational problem for the single dimensional rate distortion function. The proof offered here is more general, although applicable to single letter square and absolute distortion measures only.

The following is a discussion of the connection between this scheme and universal coding. The minimum rate at which a source may be encoded is determined by the statistics of this source. The design of a coding scheme that is efficient for a particular source relies on the knowledge of the statistics of the source being coded. For a significant proportion of the sources whose compression is considered, the statistics are not known. Some sources, behave as if they are composite, that is, from one relatively long period to another, the source statistics may be observed to change. For these two types of source, it is important to use a coding scheme that works reasonably efficiently for all sources whose statistics belong to a certain superclass. A family of coding schemes which work well for sources of unknown statistics or varying statistics is the family of universal codes. The essentials of universal coding are described in a comprehensive paper by Davisson [Davisson 1973]. Most of the present universal coding algorithms rely upon a certain degree of statistical analysis of a particular block to be encoded. The code symbols sent to the receiver are; 1) a sequence indicating the statistics of the block in question and 2) a sequence representing the code symbols associated with the "optimal" coding of the source, bearing in mind the information about the statistics. The aim of the MPPCD scheme was to effect the coding of sources whose statistics were unknown, in a reasonably efficient manner. This scheme differs from universal coding methods in the following two ways:

1) No direct assessment of the source statistics for a block is done.
2) In the MPPCD scheme the coding of a block involves the use of previously coded blocks of data. Universal coding methods however, take non-overlapping blocks, establish the statistics and encode these blocks accordingly. No account is taken of other blocks in the past or future, for the coding of a particlar block.

The MPPCD scheme may be considered as viable for the coding of sources with slowly varying statistics and similar to universal coding by reasoning thus:

The scheme codes blocks on a basis set of C members which are previously encoded blocks. The coding of a block therefore depends on previous blocks. If we consider superblocks consisting of the block to be coded and the previously coded C blocks, then the following may be said: Part of a superblock is coded at a time. The coding of this segment of a superblock is independent of other superblocks. The superblocks are shifted in such a manner as to overlap with their previous superblocks. If the period between the instants when the statistics vary is much longer than the size of a superblock, then the system adapts well to variations in source statistics.

It is in this way that the MPPCD scheme resembles a universal coding scheme and justifies its investigation.

CHAPTER 5 THE ENCODING OF SCALARS

5 The encoding of scalars

5.1 Introduction

Scalar encoding is an alternative to block coding which can offer improvements in these ways; the coding schemes may be made less complex and coding may be implemented with little time delay. The former is of special importance in image compression where the data generation rate is so large that complex compression schemes are impracticable in real time. It should be pointed out that these advantages may be only be obtained at the expense of compression capability. In the cases where scalar schemes are used to obtain compression rates comparable to those attainable using block coding, comparable complexity result and just as much delay is suffered. Thus scalar schemes are only really advantageous in cases where large compression is not required.

The encoding of scalars involves the allocation of channel digits to represent each individual source symbol generated. For most schemes, the value of each individual source symbol may be retrieved, with some distortion, without waiting for a whole block of data. The exception is the class of schemes where a multipath search is conducted. In most scalar coding schemes, the decoding process entails simply the evaluation of the appropriate approximation symbol given the channel symbol received and the previously received channel symbols.

In this chapter some examples of scalar encoding are given. Following this a brief description of the theory used for the design of "optimal" scalar encoders as reported in the literature will be given. This chapter serves as a preamble to work reported in chapter 6 on scalar encoding.

5.2 Pulse Code Modulation (PCM)

This is the simplest and most basic of all the digital encoding techniques. It was first reported and patented in 1939 by A.H. Reeves. Early descriptions of practical PCM schemes are given by Goodall-(1947) and-(1951) and a good general desciption is given by Oliver, Pierce and Shannon-(1948). A time continuous signal is sampled so that the sampling frequency is greater than twice the highest frequency component in the input signal. The lowest rate that a signal may be sampled at is termed the Shannon-Nyquist rate [Shannon-(1948)]. The resulting samples are then coded for transmission. A number of quantisation levels is chosen. The choice involves making a compromise between excessive noise and transmission rate. To each of the "N" values that a source symbol may take after quantisation (each member of the resulting source alphabet), $\log_2 N$ bits are assigned. N is chosen, in general, to be a power of 2. A PCM system with a uniform transmission rate attempts no redundancy reduction, by the allocation of a variable number of bits to a source alphabet member. The simplest PCM schemes "linearly" quantise the sample space of the source symbols into N levels. The source symbol, when observed to have a value within a given quantisation region, causes a certain sequence of bits to be transmitted. At the receiver, this source symbol is approximated by the centroid or mean of the appropriate quantisation region.

Various improvements to the basic PCM scheme have been reported. The most common is non-linear quantisation. A-Law and μ-Law quantisation schemes are the accepted standards for speech transmission. These employ a fine quantisation grid for low values of the source signal and a coarser grid for high values of the source symbol. The A and μ-law characteristics are shown in figure 5.1. A more systematic approach to non-linear quantisation is offered by the methods of Lloyd-(1982) and Max-(1961). For a source with a known probability distribution, these methods try to achieve minimum distortion granted a certain number N of quantisation levels.

An optimum quantisation scheme as far as the mean square error is concerned, should have the following properties.

1) Granted a set of partitions $x_1, x_2, \ldots x_{N-1}$ an optimum set of centroids $m_1, m_2, \ldots, m_N$ should satisfy the following.

$$m_i = \frac{\int_{x_{i-1}}^{x_i} up(u)\mathrm{d}u}{\int_{x_{i-1}}^{x_i} p(u)\mathrm{d}u} \qquad 5.1$$

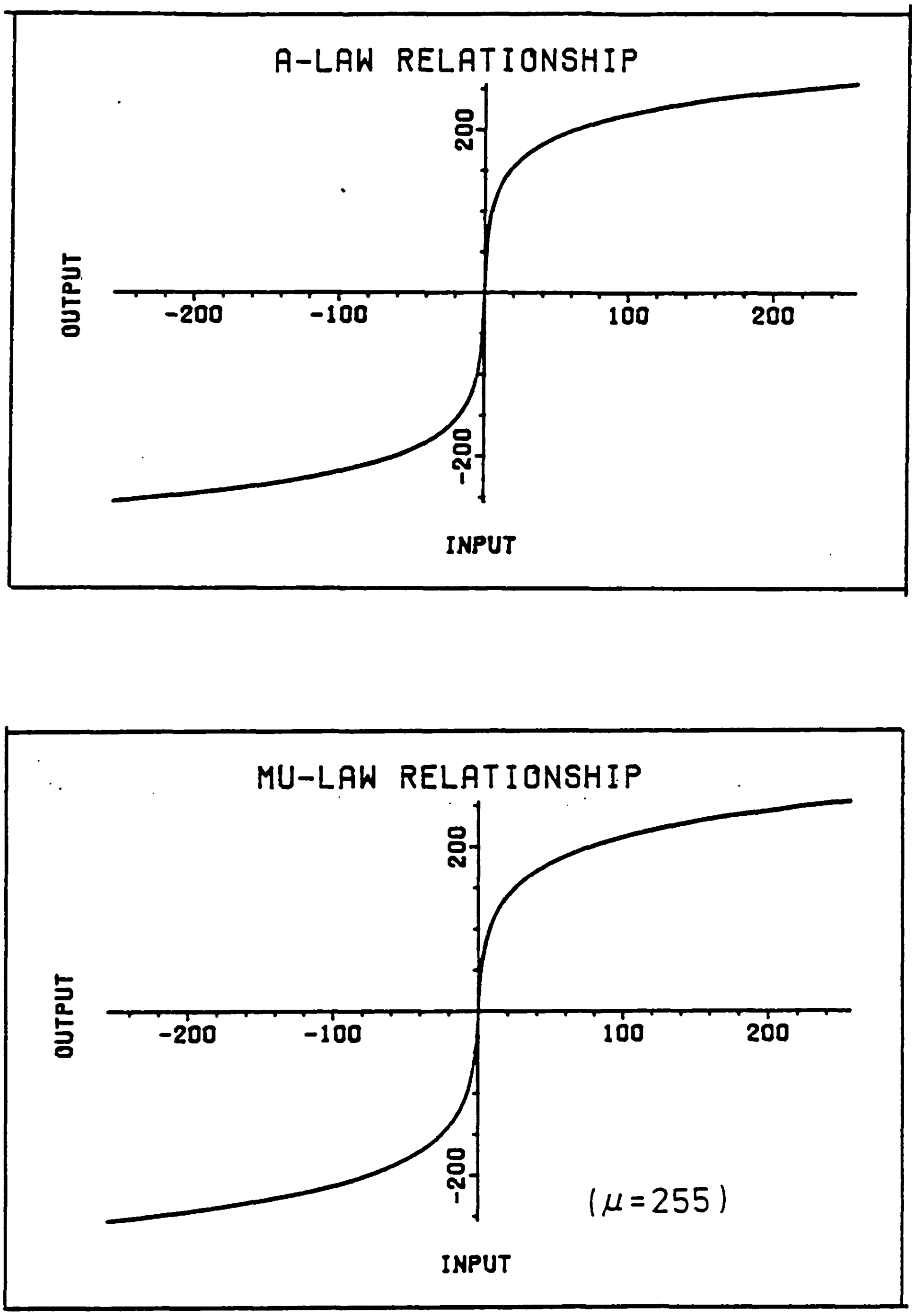
p(.) is the probability density function for the source.

2) Granted a set of centroids $m_1, m_2, \ldots, m_N$ an optimum set of partitions $x_1, x_2, \ldots, x_{N-1}$, should satisfy the following.

$$x_i = \frac{m_i + m_{i+1}}{2} \qquad 5.2$$

The schemes by Lloyd and Max try to define an optimum quantisation scheme by the successive invocation of equations 5.1 and 5.2.

For applications where a variable transmission rate is allowed, more efficient ways of bit allocation may be used. For each channel symbol possible after quantisation, a different number



A-LAW RELATIONSHIP

MU-LAW RELATIONSHIP

Figure 5.1 A-law and μ-law characteristics.

of bits may be assigned for transmission. The bit assignment rule is decided according to the relative frequency of occurrence of each channel symbol. In general, the larger the probability of occurrence of a symbol, the fewer the number of bits assigned to this symbol. The optimum bit assignment scheme was discovered by Huffman in 1952.

To date, most of the digital communication links employ uniform rate PCM with some non-linear quantisation scheme.

5.3 Delta modulation (DM)

Delta modulation is an advancement on PCM which attempts to use inter-symbol dependence to obtain some data compression. In DM the effective sampling rate is very much larger than the Shannon-Nyquist lower limit. For example, 40kHz is used to obtain coding of a reasonable quality for 4kHz bandwidth speech. Figure 5.2 shows a delta modulation transmitter and receiver pair. A brief explanation of how this works is as follows:

The source waveform is clocked in at the rate w say, which is much greater than the Nyquist lower bound. At the instant n say, let the clocked source symbol be $x(n)$ and suppose the previously generated symbol has been decoded as $\tilde{x}(n-1)$. The delta modulator then sends a channel symbol, "1" or "0", to indicate the polarity of the error or difference between $x(n)$ and $a\tilde{x}(n-1)$ (a is refered to as the integrator multiplier). If a "1" is sent the error is presumed to be $+e$ and if "0" is sent the error is presumed to be $-e$. This quantised error value, e or $-e$, is added to $a\tilde{x}(n-1)$ and used to approximate $x(n)$. This value, $\tilde{x}(n)$ is used to help encode $x(n+1)$

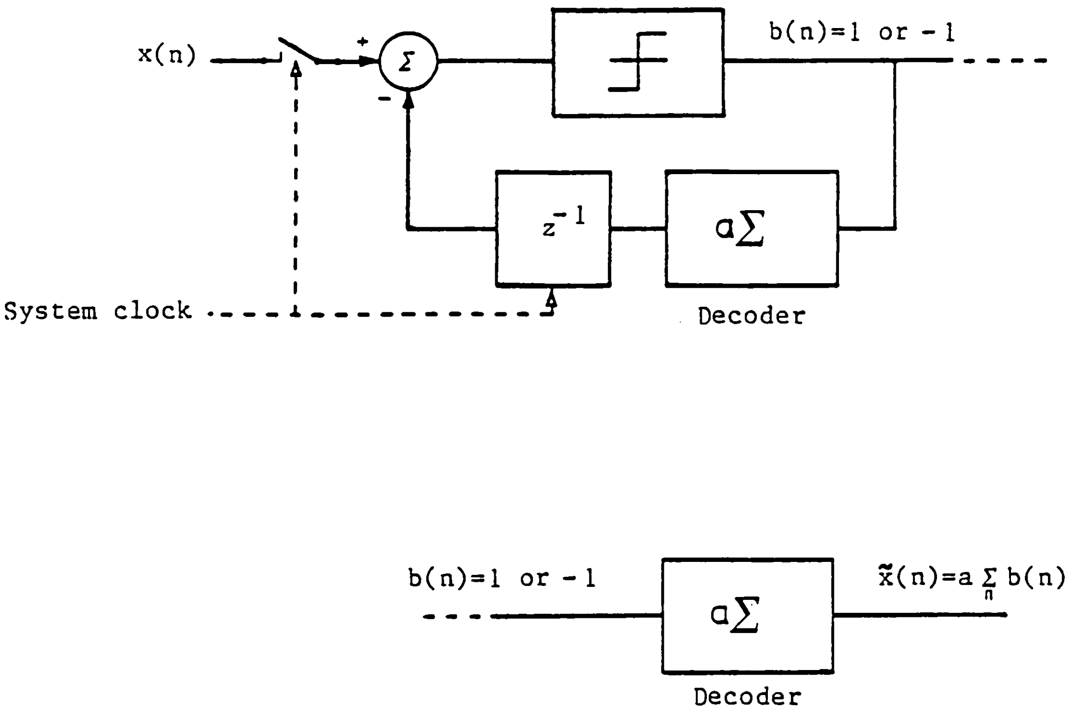


Figure 5.2 A delta modulator transmitter and receiver pair.

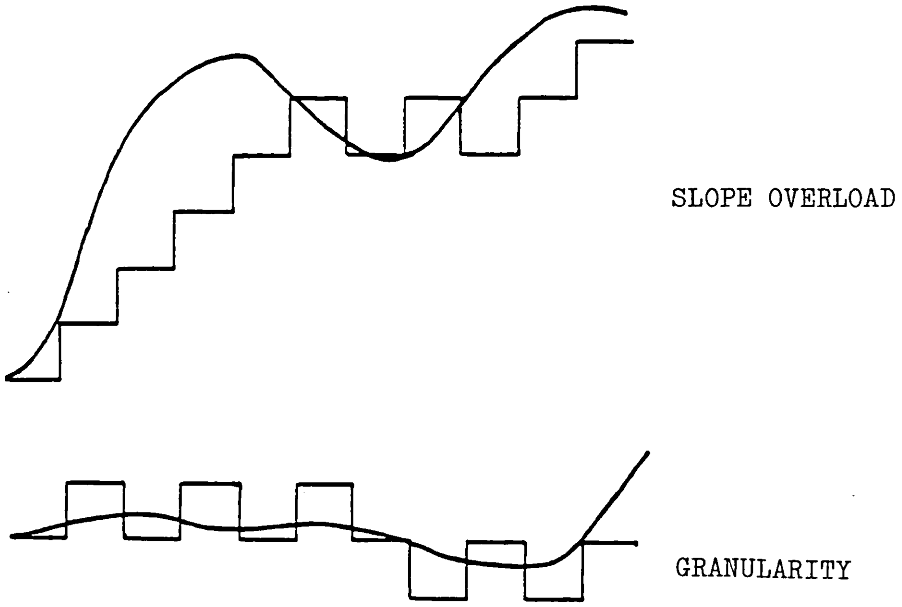


Figure 5.3 Phenomena of slope overload and granularity.

and so on. The delta modulator effects a feedback process, the fact that it uses previous estimates of a signal gives it its advantage over PCM.

An alternative way of interpreting the functioning of the delta modulator is this. Presume a first order auto-regressive model for the source. The model is defined by the following equations,

$$\begin{aligned} x(n) &= a\tilde{x}(n-1) + e(n) \\ \tilde{x}(n) &= a\tilde{x}(n-1) \pm e_q(n) \end{aligned} \qquad 5.3$$

The error signal e(n) is then quantised to one of two levels $\pm e_q(n)$.

Several adaptive methods for delta modulation have been reported. These generally work by changing the quantisation levels for the error signal e(n), depending on the short term magnitude of this. These fall into two classes, instantaneous adaptation schemes and syllabic adaptation schemes. The latter are descibed by Tomozowa and Kaneko-(1968) and Bolin and Brown-(1968). In general these detect periods when the signal magnitude is too large for the step size, by monitoring the sequence of ones and zeros generated by the encoder. The mean number of ones and zeros is used to increase or reduce the step size. The instantaneous schemes work by changing the step size based on a decision using a few of the ones and zeros released from the encoder. The most common method is that reported by Jayant-(1970), Cumminsky, Jayant and Flanagan-(1973) and Goodman and Gersho-(1974).

The use of a second integrator in the feedback loop, acts as a means of ensuring that there is little slope overload. The effect of a sequence of ones at the encoder output(indicating slope overload), is to force a ramp at the input to the second integrator, whose output then rises in the manner of a 2-nd order function to match the input. This has the disadvantage that it is liable to overshoot.

The third variation on the theme of delta modulation is that concerning the use of a variable sampling rate. Work in this direction has been reported by Vanlandingham and Bogdanski-(1980) and Un and Cho-(1982). These adapt the delta modulation sampling rate according to the degree of the local activity of the input waveform. For example, in the paper of Vanlandingham and Bogdanski, an estimate is made of the local second differential. The larger this is, the smaller the sampling period used. This is the philosophy of run-length coding which is considered in the next section. Steel-(1975) gives a very thorough presentation of the various DM systems available. A comparison of delta modulation systems, is given in the paper by Un and Lee-(1980).

5.4 Run length coding

This is a differential coding scheme, where a non-uniform transmission rate is obtained. It has been applied mostly to picture coding and in particular to facsimile pictures.

Briefly, a typical run length coding scheme does the following: An error criterion is set apriori, an estimate $\hat{x}(n)$ of a source symbol $x(n)$ is made based upon the values of the decoded approximations to the previous source symbols. If the difference between $x(n)$ and $\hat{x}(n)$ is above the error threshold, $x(n)$ is sent as it is or the error signal $x(n)-\hat{x}(n)$ is sent. If otherwise, nothing is transmitted. At the receiver, the symbol $x(n)$ is approximated by $\hat{x}(n)$. When a symbol is transmitted, the period between this and the latest of the previous symbols transmitted, is also transmitted. Generally, the estimate of a symbol $x(n)$ is simply the estimate for the previous symbol, $x(n-1)$. Thus the decoded output of a run length coder consists of straight line approximations to the source waveform. Run length coding is very similar to delta modulation with "a", the integrator gain set to unity. The difference is that the line segments used in delta modulation are always one inter-symbol period long. Run length coding is particularly suitable for the coding of data where long sequences of data are of approximately the same value.

A block diagram showing a simple run length coder is given in figure 5.4. The following references indicate the types of run length coder reported in the literature. Gonzalez and Wintz-(1977) section 6.3.3, Pratt-(1978) section 22.3, Gouriet-(1957), Cherry, Kubba, Pearson and Barton-(1963) and Robinson and Cherry-(1967).

5.5 Tree and Trellis coding

The basic idea underlying both these methods are explained in the foregoing. Consider a sequence of random variables, each with a sample set $\Omega=\{\omega_1,\omega_2,\ldots,\omega_N\}$, so that a typical sequence of outcomes is $x(1),x(2),\ldots,x(n),\ldots$ where each $x(n)$ belongs to Ω. Tree and Trellis coding techniques try to find an optimum set of

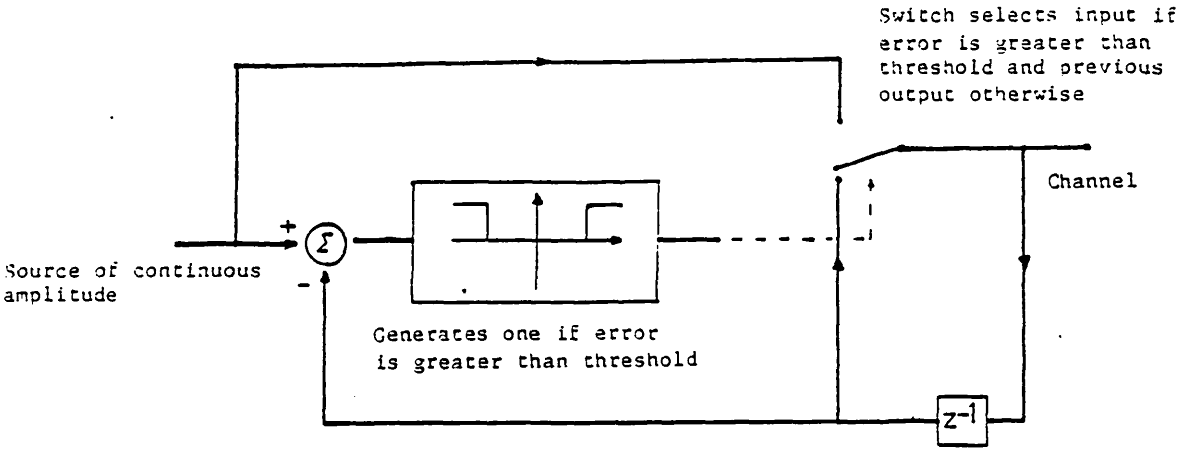

Switch selects input if error is greater than threshold and previous output otherwise
Channel
+
-
Σ
Source of continuous amplitude
Generates one if error is greater than threshold
z^{-1}

Figure 5.4a A block diagram of a Run-length encoder.

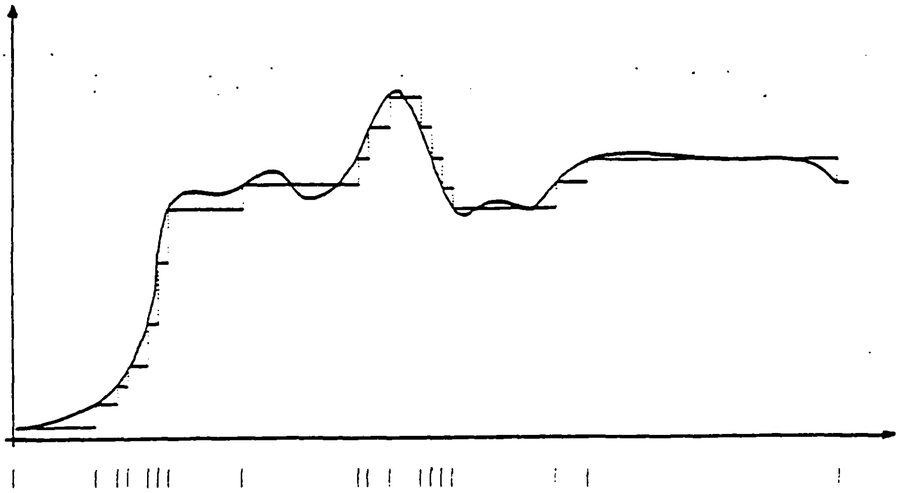


Figure 5.4b Example of the input waveform and its approximation , obtained with a simple run-length-encoder. The short vertical lines against the horizontal axis show the transmission instants.

channel symbols b(1),b(2),...,b(n) to represent the sequence of source outcomes x(1),x(2),...,x(n). The representation is such that a set of approximating $\tilde{x}(1),\tilde{x}(2),\ldots,\tilde{x}(n)$ may be decoded from b(1),b(2),...,b(n) so that the distortion $d(x(1),x(2),\ldots ; \tilde{x}(1),\tilde{x}(2),\ldots)$ is minimised. Tree and Trellis coding schemes differ from block coding schemes in that the former schemes choose the set b(1),b(2),...,b(n) and hence $\tilde{x}(1),\tilde{x}(2),\ldots,\tilde{x}(n)$ sequentially.

5.5.1 Tree coding

A study of figure 5.5 and the following explanation shows in detail how a typical tree coding scheme works. The whole of the sample space for the sequence of bits which may be used to transmit approximations to the outcomes {a(-n),...,a(-1),a(0),a(1),...,a(n)} may be represented by an infinite sized tree. The actual sequence of bits {b(-n),...,b(-1),b(0),b(1),...,b(n)} employed to code a source sequence may be likened to a particular path in the tree. A finite length of data {x(1),...,x(n)}, has a sub-tree associated with the sample space of bits which may be used to represent these. Tree coding is the business of assigning approximation sequences to paths in a tree and given these, to find good paths to traverse when coding actual source sequences.

Figure 5.6 shows a sub-tree where the darker branches describe a path representing the sequence b(1),...,b(n). In the following sections two examples of tree encoders are given.

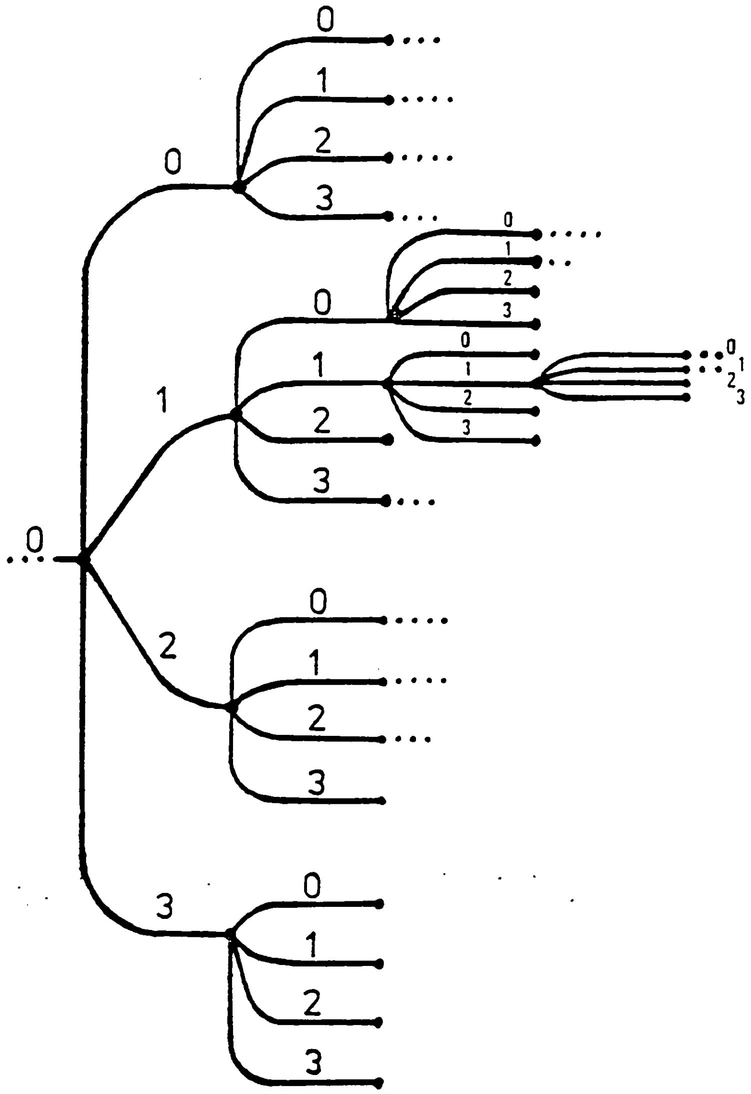


Figure 5.5 A tree, symbolising the options that the channel symbols may take, in the process of scalar coding.

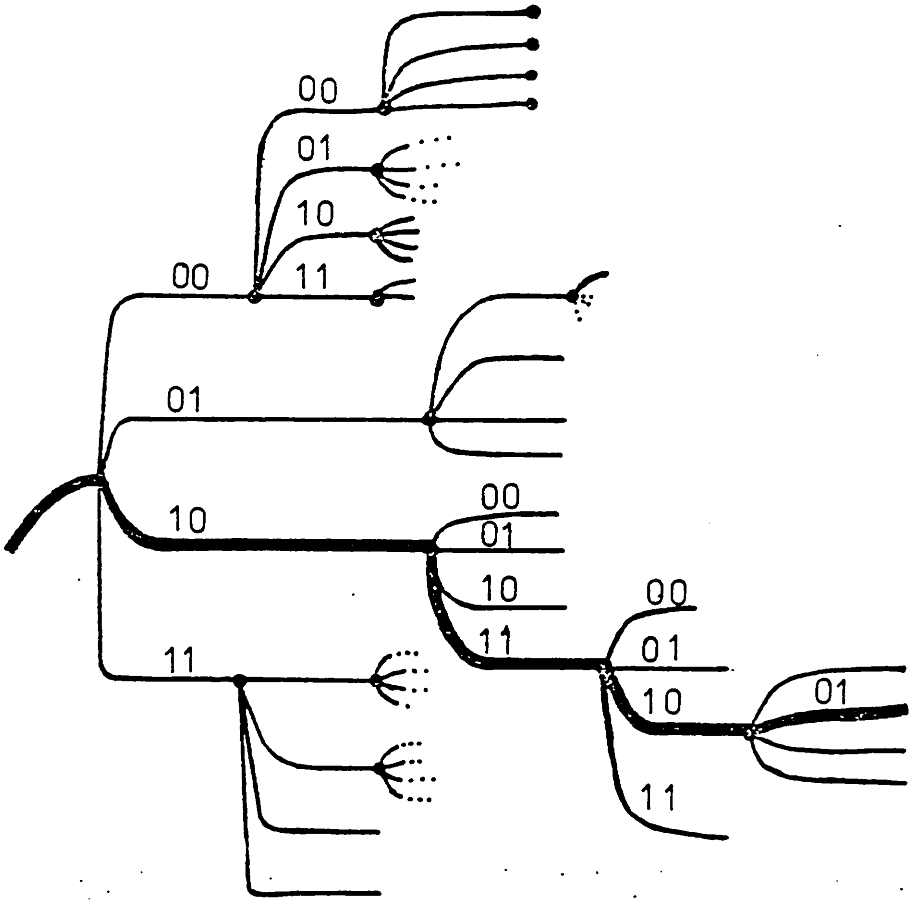


Figure 5.6 A subtree, with an example path and path map.

5.5.1.1 Example 1. Differential Pulse Code Modulation (DPCM)

Suppose a source generates a sequence of symbols x(n),.... In implementing a differential pulse code modulator, an estimate $\hat{x}(n)$ of the outcome x(n) is made, employing a linear combination of the previous approximations.

$$\hat{x}(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) \qquad 5.4$$

The values of $a_1, a_2, \ldots, a_p$ which lead to estimates $\hat{x}(n)$ of least deviation from the correct values, are computed.

In DPCM, an outcome x(n) is coded and represented by a quantised version of the difference between itself and its estimate. Refering back to figure 5.6, it may be seen that DPCM effects tree encoding in the following way. At the previous instant "i-1" the DPCM coder may be envisaged to have been at some node in the tree. The symbol x(n-1) has been approximated by $\tilde{x}(n-1)$. The coder is said to be at stage i-1. Next the coder decides which of the alternative branches in the tree it may take, given that the symbol x(n) has just been observed. The destination node represents the symbol to be used to approximate the observed source symbol. Associated with each branch that may be chosen are some channel symbols or binary digits. In this particular case, the alternative destination nodes represent the values

$$\tilde{x}(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) + q_1(e(n))$$

$$\text{or} \quad \sum_{i=1}^{P} a_i \tilde{x}(n-i) + q_2(e(n))$$

$$\text{or} \quad \ldots\ldots\ldots\ldots \qquad 5.5$$

$$\text{or} \sum_{i=1}^{P} a_i\tilde{x}(n-i) + q_L(e(n))$$

The quantities $q_1(e(n)), q_2(e(n)), \ldots, q_4(e(n))$ are alternative approximations to the estimation error. To each of these possible approximations is assigned a sequence of bits.

Adaptive differential PCM (ADPCM) and linear predictive coding (LPC) are advancements on DPCM. ADPCM is DPCM where an adaptive quantisation scheme is used to encode the error associated with the linear prediction of a sample or where the coefficients for linear prediction are adapted regularly. Linear predictive coding is a term used in speech coding for ADPCM where the prediction coefficients are adapted regularly. More emphasis is placed on the coding the prediction coefficients than on the linear estimation error.

The following is some of the literature on DPCM and ADPCM. This is nowhere near a comprehensive list, but these ought to give a good impression of the work done in the area and more significantly the combined references of these papers should indicate where to look for more information. Harrison-(1952), Elias-(1955), Jayant-(1974), Flanagan et al-(1979) may be consulted for general work on DPCM. Methods which rely on the particular properties of speech and images are:

1) For speech coding, pitch synchronous prediction, where estimation is made using previous outcomes which are a pitch period in the past [Atal-(1982)].

2) For images Candy and Bosworth-(1972) and Maragos, Schafer and Mersereau-(1984) do 2-dimensional spatial prediction. Limb and

Rubenstein-(1978), Netravali-(1977) and Zschunke-(1977) make use of the detection of edges and plane areas for coding.

5.5.1.2 Example 2. General tree coding

DPCM as described earlier is a particular case of tree coding. A general tree coding scheme may vary from DPCM or ADPCM in the following ways.

1) The schemes described so far assign "reproduction" symbols to the branches of the code tree by means of a linear predictive mechanism. A reproduction symbol is the approximation symbol obtained when a particular link in a tree is chosen at any stage in the coding process. In the use of a linear prediction model, the reproduction symbol for each node in the tree is determined by these two quantities: A linearly predicted quantity obtained by using the reproduction symbols associated with the nodes traversed in going to the node in question and an error signal associated with the branch which joins the node in question with the previous node visited. This need not be so, the values assigned to each node of the tree may be determined by another process. The process of determining which reproduction values to associate with going to a particular node via a particular sequence of nodes is called "colouring".

2) In DPCM a multipath search is not conducted to ascertain the best path to use in coding a block of data. In multipath search schemes, a decision is not made concerning which path to employ at each sample instant. A number of possible paths are considered as candidates and a choice is made only after a whole block of data has been considered.

A particular tree coding scheme will be explained in the following section. The results of using this scheme were reported by Anderson and Bodie in 1977. A flow diagram for this scheme is presented in figure 5.7. The colouring of the tree is based upon linear prediction. When the i-th source symbol is under consideration, the coder is said to be at the i-th stage in the tree. Initially, a number of possible paths to be considered is decided, for example let this be 6. At each stage of the tree 6 paths to this stage are considered. Associated with each path, is a bit stream and a sequence of reproduction symbols. For each of the 6 possible paths, the following is done. The nodes traversed till the {i-1}-th stage (the reproduction sequence defined by these nodes) are used to estimate the i-th outcome. The estimation is effected by linear prediction. From each of the 6 nodes at the {i-1}-th stage terminating each of the 6 paths under consideration, emanate "m" possible branches. m is refered to as the generation exponent. Each of the m possible branches has a sequence of bits assigned.

There are therefore 6m possible values which may be assumed by the approximation to the i-th outcome. These outcomes are the estimates associated with each of the 6 alternative paths to nodes at the {i-1}-th stage and associated with each, m possible values for the error. Of these 6m possible reproduction symbols at the i-th stage, 6 are chosen. The coder then advances to the {i+1}-th stage and proceeds as explained before.

After the consideration of a block of say N source symbols, one of the 6 alternative paths is selected as the best. The bit

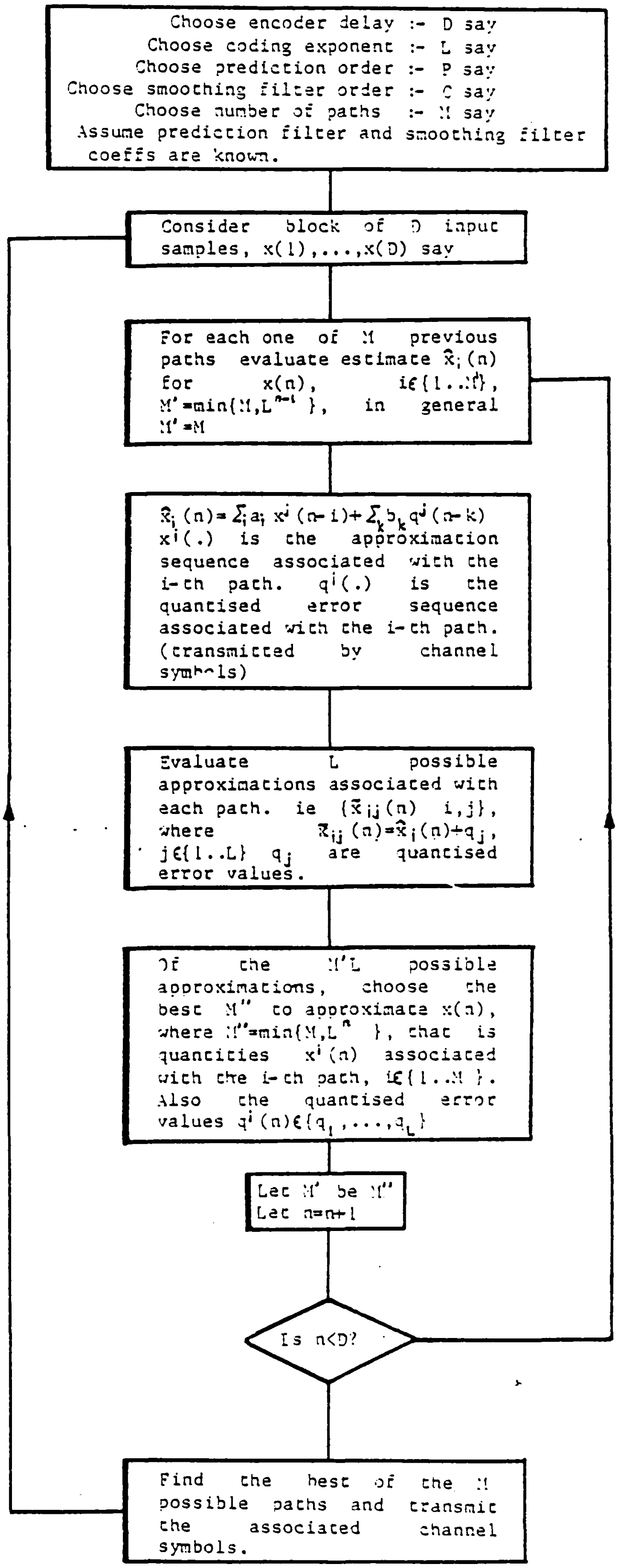

Choose encoder delay :- D say
Choose coding exponent :- L say
Choose prediction order :- P say
Choose smoothing filter order :- Q say
Choose number of paths :- M say
Assume prediction filter and smoothing filter coeffs are known.
Consider block of D input samples, x(1),...,x(D) say
For each one of M previous paths evaluate estimate $\hat{x}_i(n)$ for x(n), $i\in\{1..M'\}$, $M'=\min\{M,L^{n-1}\}$, in general $M'=M$
$\hat{x}_i(n)=\sum_i a_i x^i(n-i)+\sum_k b_k q^i(n-k)$
$x^i(.)$ is the approximation sequence associated with the i-th path. $q^i(.)$ is the quantised error sequence associated with the i-th path. (transmitted by channel symbols)
Evaluate L possible approximations associated with each path. ie $\{\tilde{x}_{ij}(n)\ i,j\}$, where $\tilde{x}_{ij}(n)=\hat{x}_i(n)+q_j$, $j\in\{1..L\}$ q_j are quantised error values.
Of the M'L possible approximations, choose the best M'' to approximate x(n), where $M''=\min\{M,L^n\}$, that is quantities $x^i(n)$ associated with the i-th path, $i\in\{1..M\}$. Also the quantised error values $q^i(n)\in\{q_1,...,q_L\}$
Let M' be M''
Let n=n+1
Is n<D?
Find the best of the M possible paths and transmit the associated channel symbols.

Figure 5.7 Flow diagram of the tree coding scheme by Anderson and Bodie

sequence defining this path is released for transmission. The process restarts, using initially only one node, that terminating the best path just chosen.

In addition to the procedure described above, Anderson and Bodie chose to smooth the quantised error sequence by sending this sequence through a short, 2 or 3 length, FIR filter. This filter was chosen to have zero transmission at the input sampling rate. The operation of this filter was as follows. Suppose the sequence x(1),x(2),... is the undistorted input sequence and $\hat{x}(1),\hat{x}(2),\ldots$ is the sequence of linear estimates obtained using coded previous outcomes. The error sequence is e(n),e(n),..., where these are defined thus;

$$\hat{x}(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) \qquad \text{and} \qquad x(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) + e(n) \qquad 5.6$$

Each e(n) is approximated by a quantised version, q(e(n)). In normal DPCM or linear predictive coding, the receiver will use the quantities q(e(n)),q(e(n+1)),... in conjuction with the linear prediction model parameters to estimate the input signal. In this case, a linear combination of the previous quantised error values is added to the linear estimate obtained using the reproduction sequence defined by the coded previous outcomes. This is the estimate for the i-th input symbol. Of the 6m paths defined by the 6 estimates and the m alternative quantised error values the best six are retained. These 6 are used to define the approximations for the next stage. The process model is thus an auto-regressive moving average system defined by the following relationship.

$$x(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) + \sum_{i=1}^{Q} b_i q(e(n-i)) + q(e(n)) + \zeta(n) \qquad 5.7$$

and

$$\tilde{x}(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) + \sum_{i=1}^{Q} b_i q(e(n-i)) + q(e(n)) \qquad 5.8$$

where e(n) are the error values

In the paper of Anderson and Bodie, the quantities $a_1,\ldots,a_P$ are estimated by using an auto-regressive model for the source. The quantities $b_1,\ldots,b_Q$ are chosen in an ad-hoc manner. In fact, we have been unable, by experiment to observe an improvement in the signal to noise ratio resulting from the use of the quantised error signal.

Wilson and Hussain-(1977) reported an adaptive scheme based on the method of Anderson and Bodie. In this scheme, a set of estimation parameters were evaluated at regular intervals and transmitted with a side channel to the receiver.

5.5.2 Trellis coding

Trellis coding is essentially the same as tree coding. The difference is the graph structure used to represent the possible bit assignment schemes for coding a sequence of data.

A trellis is a structure which unlike the tree structure has branches which remerge. A tree with branches labelled such that sections of this are repeated over and over again, may be represented by a less redundant structure, a trellis. Figures 5.8a, 5.8b and 5.8c show a tree and two possible trellis representations. It is to be observed that the trellis is adequate to indicate all the possible paths associated with the uniform rate coding of a

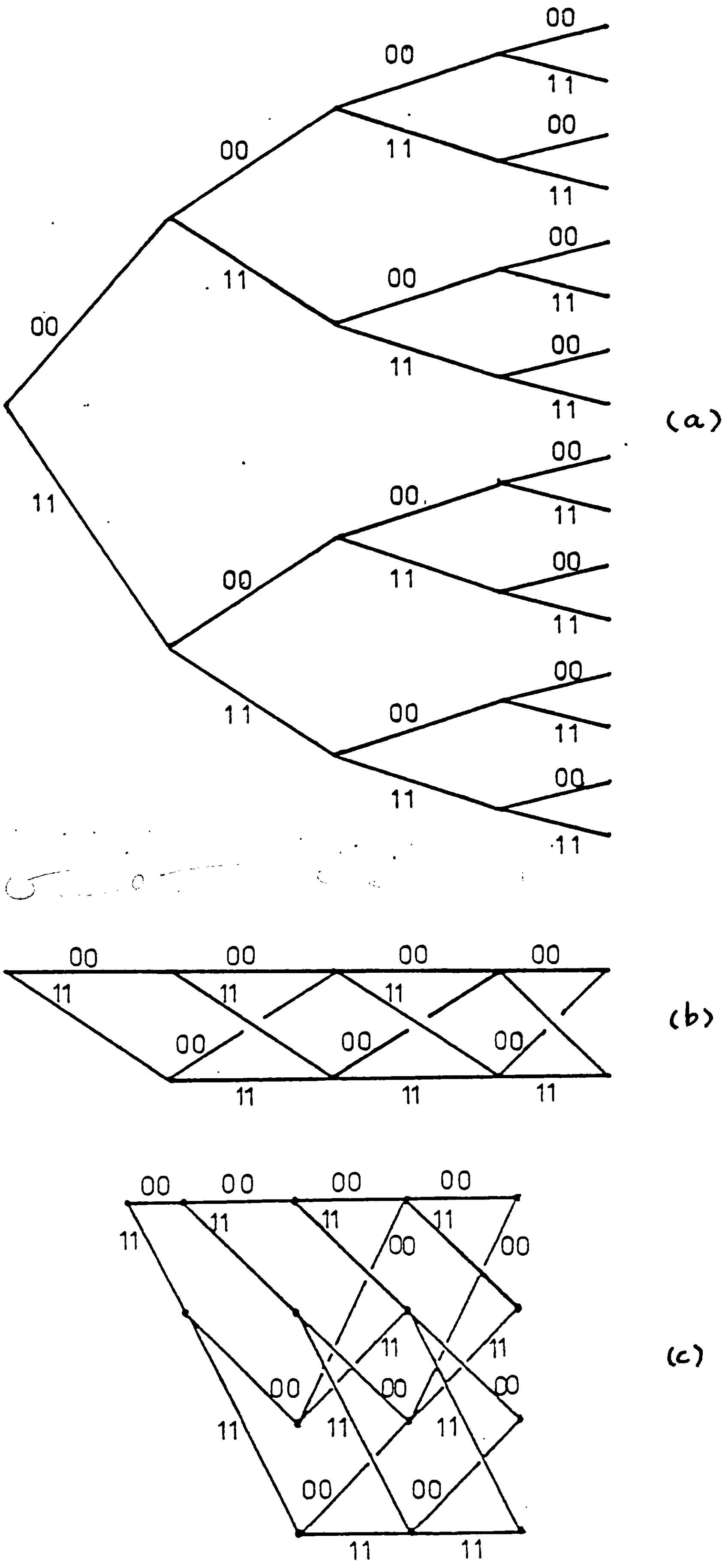


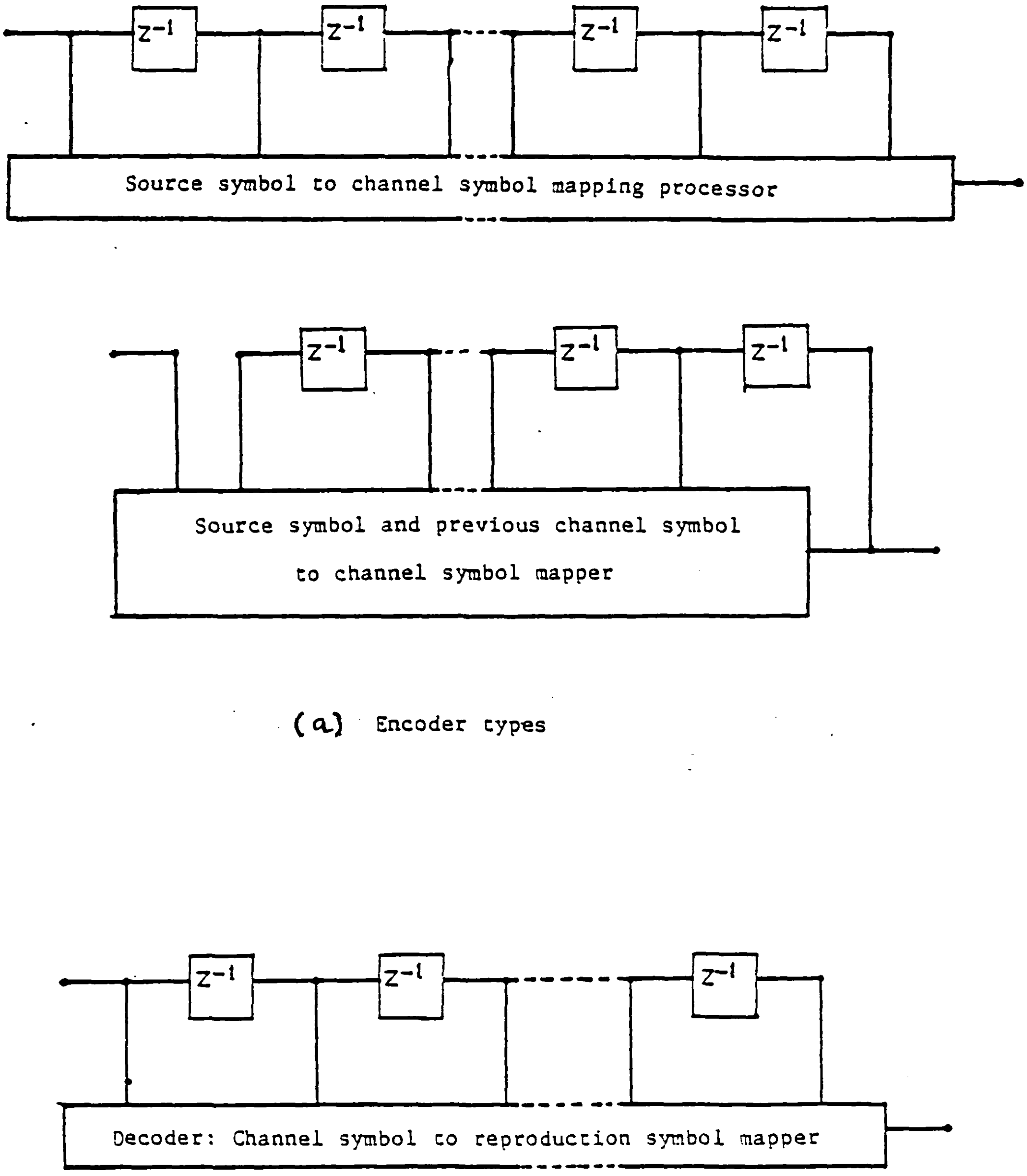
Figure 5.8 A tree and alternative trellis representations.

source. (By uniform it is meant that a fixed number of bits is assigned for coding each source symbol). Tree and trellis coding are fundamentally the same. Both methods try to find a good path through a graph representing the possible ways of assigning bits for coding sequences. The only difference is that the rather compact structure of the trellis indicates one particular scheme for finding a path through this graph. The fact that vertices which would be distinct in a tree are merged in a trellis, directs one to do a path search using the following principle. If there are m paths which lead to a given node, the choice of the best path to this node is made once and for all. For subsequent processing no consideration is given to how this particular vertex was reached. This is the underlying principle for a well known algorithm for decoding convolutional codes, the Viterbi algorithm. (see Viterbi and Omura-(1979) section 7.4, Forney-(1973)).

5.5.2.1 Algorithms for trellis coding

Firstly a few definitions will be given. A trellis encoder is shown in figure 5.9a and a decoder in figure 5.9b. The encoder is a convolutional coder. That shown in figure 5.9a has a constraint length of K. K is the number of input symbols which are employed to generate the channel sequence {...,b(n),...} using some function $f_K(.)$ such that $b(n)=f_K(x(n),x(n-1),...,x(n-K+1))$. The x(n) are the source symbols.

The essentials of a trellis coding scheme are summarised by these statements



z^{-1}
Source symbol to channel symbol mapping processor
Source symbol and previous channel symbol to channel symbol mapper
Decoder: Channel symbol to reproduction symbol mapper

(a) Encoder types

(b) Decoder

Figure 5.9 A convolutional (possibly trellis) coder and decoder.

1) A codebook or decoding scheme is designed, either apriori or adaptively.

2) Granted this decoding scheme, a good encoding scheme, like the Viterbi algorithm, is chosen. This is used to assign bits to the input symbols.

3) At the receiver, the channel bit stream are used to generate an approximation sequence to the source symbols.

The field associated with the design of good multipath search schemes, has been exceedingly well researched. The challenge in the field of both tree and trellis coding is the design of good decoding schemes (alternatively refered to as colouring schemes). To see why the term "decoding scheme" is applied here, consider the following. A decoding scheme is that which decides which "reproduction" symbols should be associated with a sequence of channel bits. The job of generating a set of reproduction symbols for a sequence of channel bits is performed at the decoder and thus this assignment scheme is termed a decoding scheme. Really therefore, it is not difficult to assign a set of bits to represent a sequence of input symbols, given a scheme for going from these bits to reproduction symbols. It is however difficult to choose a good scheme for deciding a set of reproduction symbols given a sequence of bits. Most methods reported in the literature use a decoding or colouring scheme based upon linear prediction. A few of the methods which specifically use a multi-path search, as apart from DPCM, ADPCM and LPC are the schemes reported by Anderson and Bodie-(1975), Linde and Gray-(1978), Jayant and Christensen-(1978), Wilson and Hussain-(1979), Fehn and Noll-(1980), Matsuyama-(1981), Fehn and Noll-(1982) and and Modestino and Bhaskaran-(1981) for image coding.

The literature provide theoretical results concerning the feasiblilty of designing tree and trellis coders which achieve compression close to the rate-distortion bounds for various types of sources. (Jelinek-(1969) and Viterbi and Omura-(1974) for memoryless sources. R. M. Gray-(1977) for ergodic sources). They provide as yet very few methods for the design of tree and trellis coders.

5.5.2.2 Trellis compression

A scheme in which the design of a trellis decoder is attempted in an optimal manner, has recently been reported by Stewart, Gray and Linde-(1982). The following is an explanation of the scheme. A code-book is arbitrarily chosen, this is such that for every feasible path map of length k, (k is the constraint length of the decoder) a reproduction symbol is assigned. Refer to this code-book as C^k. It has members $\{b(1),...,b(k)=u_1 ; y_1\}$, $\{b(1),...,b(k)=u_2 ; y_2\},.....,\{b(1),...,b(k)=u_M ; y_M\}$

Each b(n) is a channel symbol or a sequence of bits and each y_j is the reproduction symbol associated with the channel symbol u_j.

A training sequence of source symbols $\{x(1),x(2),....\}$ is fed to the encoder. This encoder implements the Viterbi algorithm to find the best bit map given the trellis representation and the codebook C^k. With the output of the encoder, a decoder with a codebook C^k is driven. A set of approximations to the training sequence $\{x(1),x(2),...\}$ is obtained.

The following information is therefore available.

1) A set of k length input symbols from the training sequence and corresponding to each k length sequence the associated k length sequence of reproduction symbols.

2) Also available is the sequence of bits transmitted to obtain a k length sequence of reproduction symbols.

This information is used to alter the codebook C^k in this manner. For the codebook value $y_i \in C^k$ an update, defined as follows is employed.

$$y_i = \frac{\sum_{\forall\, n:\hat{x}(n)=y_i} x(n)}{\sum_{\forall\, n:\hat{x}(n)=y_i} 1} \qquad 5.9$$

This is done for all samples in the training sequence. A new codebook C^k is thus designed. The new codebook is then used to code the training sequence and using the method described above, an update of the new codebook obtained. This is done over and over again till there is little noticable difference in the codebook as a result of a repeat of this. This codebook is then considered to be "optimal".

To provide greater insight an example is given. Consider a trellis encoder with a constraint length k=3, which may transmit a zero or one at each input sample instant. The trellis diagram of an encoder is given in figures 5.10. Table 5.1a gives the beginning of a training sequence and table 5.1b gives an initial codebook employed to design a decoder for this trellis coding system. Following the trellis diagrams of figure 5.11 to 5.16 and the associated tables 5.2 to 5.7 should show the precise working of the

10 14 9 16 13 1 16 10

Table 5.1a Beginning of training sequence.

Channel symbols			Reproduction symbols
b(n-2)	b(n-1)	b(n)	y(n)
0	0	0	1
0	0	1	4
0	1	0	7
0	1	1	9
1	0	0	10
1	0	1	12
1	1	0	14
1	1	1	18

Table 5.1b Initial codebook used to illustrate the Stewart-Linde-Gray method for establishing codebook values for convolutional coding.

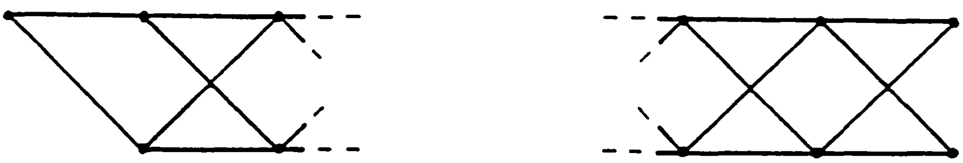
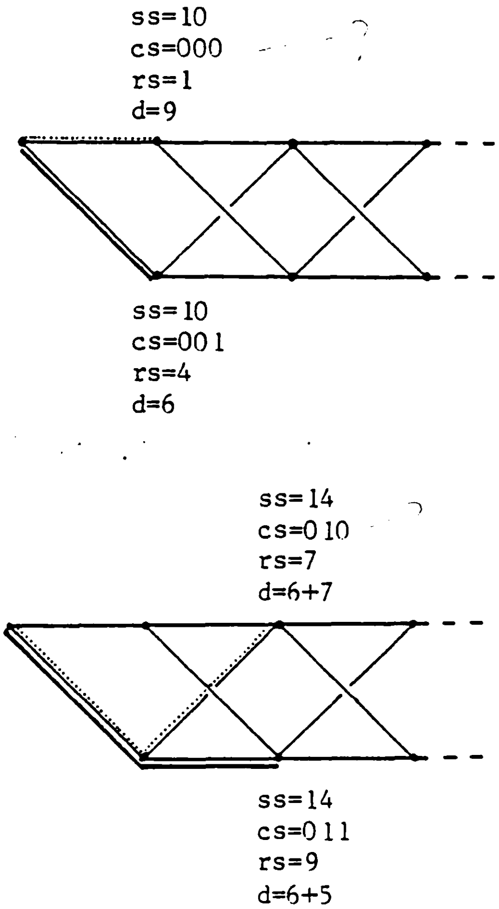


Figure 5.10 Example of trellis contruction used to illustrate Stewart-Linde-Gray method for establishing codebook values.



ss = source symbol
cs = channel symbol
rs = reproduction symbol
d = distortion or distance
the above associated with the latest node of the path in consideration.

The full and dotted lines indicate the alternative paths in consideration.

Figure 5.11a-g Example of trellis coding procedure.

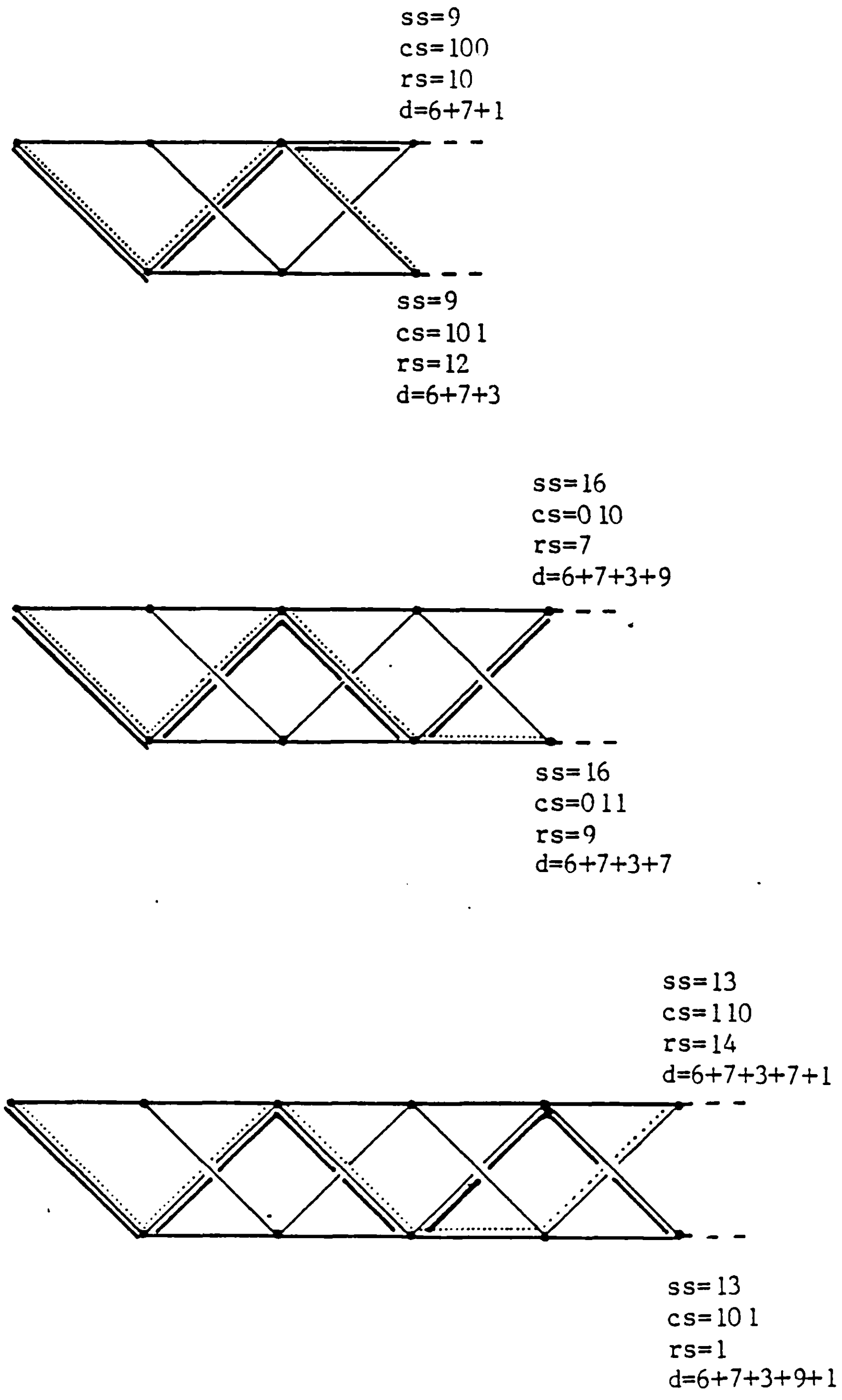


Figure 5.11a-g Example of trellis coding procedure.

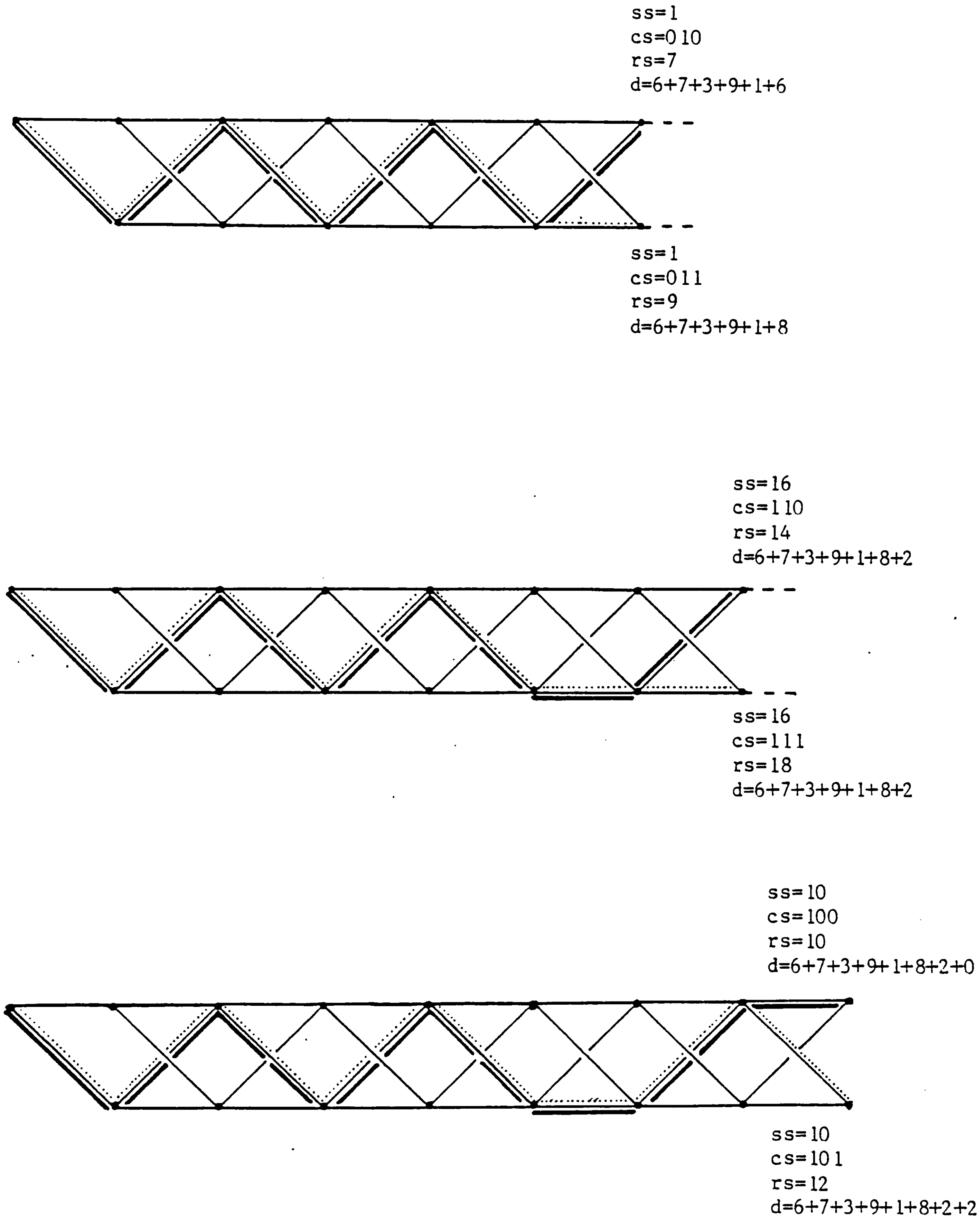


Figure 5.11a-g Example of trellis coding procedure.

system. The dark lines and the dotted lines show the two possible paths to each stage.

5.6 Conclusion

No new results were presented in this chapter. Data compression methods refered to as scalar encoding schemes have been discussed. It had been presumed that the reader is well aquainted with the more popular schemes; PCM, DM, DPCM and run length coding. Therefore the desciption of these has been brief. The not so well known methods, tree and trellis coding, especially the latter have received more attention in this chapter. It is hoped that the contents of this chapter will enable the reader to appreciate more fully, what is presented in the next chapter.

Source sequence	=	10	14	9	16	13	1	16	10
Reproduction sequence	=	1	7	12	7	12	9	14	10
Channel sequence	=	1	0	1	0	1	1	0	0

Table 5.2a Source, reproduction and channel sequences

Channel symbols			Reproduction symbols		
b(n-2)	b(n-1)	b(n)	y(n)		
0	0	0	1	=	1
0	0	1	(4+10)/2	=	7
0	1	0	(7+14+16)/3	=	12.33
0	1	1	(9+1)/2	=	5
1	0	0	(10+10)/2	=	10
1	0	1	(12+9+13)/3	=	11.33
1	1	0	(14+16)/2	=	15
1	1	1	18	=	18

Table 5.2b New codebook after a single pass of the input sequence.

CHAPTER 6 ADAPTIVE DATA COMPRESSION WITH MEMORY, THE SCALAR CODING APPROACH

6.1 Introduction

In this chapter some results of investigation into methods of scalar encoding are presented. As discussed in the previous chapter, scalar encoding techniques, offer some potential advantages over block coding; mostly in the direction of reduced complexity. In addition, scalar coding techniques are often used when coding delays are intolerable.

Most research in the subject of scalar encoding has resulted in suggestions for the improvement and analysis of multipath search coding (MSC) schemes. For the efficient operation of tree and trellis coding schemes (examples of MSC) a good 'colouring' or 'decoding' scheme is required. To design a good colouring scheme requires a knowledge of the statistics of the source to be coded. More often than not, the precise source statistics are unknown apriori.

An adaptive scheme is then called for. The local statistics for the signal being coded are ascertained and using these, varying 'colouring' or 'decoding' schemes are used for compression.

In this chapter some results on adaptive tree and trellis coding are presented. Some of the results presented are from schemes which work in a similar manner to the MPPCD methods first described in chapter 3. MPPCD stands for "the Matching of Patterns in Previously Coded Data". Coding is done on a block by block basis; for each block to be coded, the statistics of a previously coded source with similar statistics to the present block, are used to design a 'colouring' scheme. The system uses a very small quantity of extra bandwidth to code the local statistics, as compared to other adaptive tree and trellis coding schemes.

The chapter is organised as follows: The results of tree coding by the methods of Anderson and Bodie-(1975) are presented and compared with those obtained by using, firstly an adaptive quantiser and secondly, an adaptive 'colouring' scheme. The latter is similar to the experiments of Wilson and Hussain-(1977). These results are then compared with those obtained by the approach presented here.

Adaptive trellis coding is proposed for the case when the 'colouring' or 'decoding' scheme is based on a codebook as reported by Steward, Linde and Gray-(1982). Due to the magnitude of the transmission rate ordinarily neccessary to specify the colouring strategy, adaptive methods had not been previously reported in the literature.

In the same spirit as the rest of this work, a 'colouring' scheme which is specified using the statistics of previously coded blocks of data, is presented

6.2 Tree coding

The binary digits which code the outcomes of a source, may be supposed to represent paths in a tree. Figure 6.1 shows a tree, a path through this tree and the binary digits representing this path. Tree coding involves two jobs.

The first is finding a means of 'colouring' a tree. That is, choosing 'reproduction' sequences or symbols to associate with paths or branches in the tree.

The second task is that of finding, given a 'colouring' rule, a strategy for choosing a path in a tree such that the resulting reproduction sequence matches the input sequence reasonably well.

An example of a colouring scheme is that defined by linear predictive analysis and of a path search strategy is the M-algorithm [Jelinek and Anderson-(1971)].

6.3 The colouring problem

Suppose a search scheme has been established for doing tree or trellis coding. The job to be tackled is that of appropriately colouring the tree or trellis. There are two reported approaches to the problem.

The first and the most commonly adopted approach is founded on linear prediction. Reproduction symbols used to colour the tree or trellis are derived using a linear combination of previously coded symbols, in conjunction with quantised versions of the error between an actually observed symbol and its estimate.

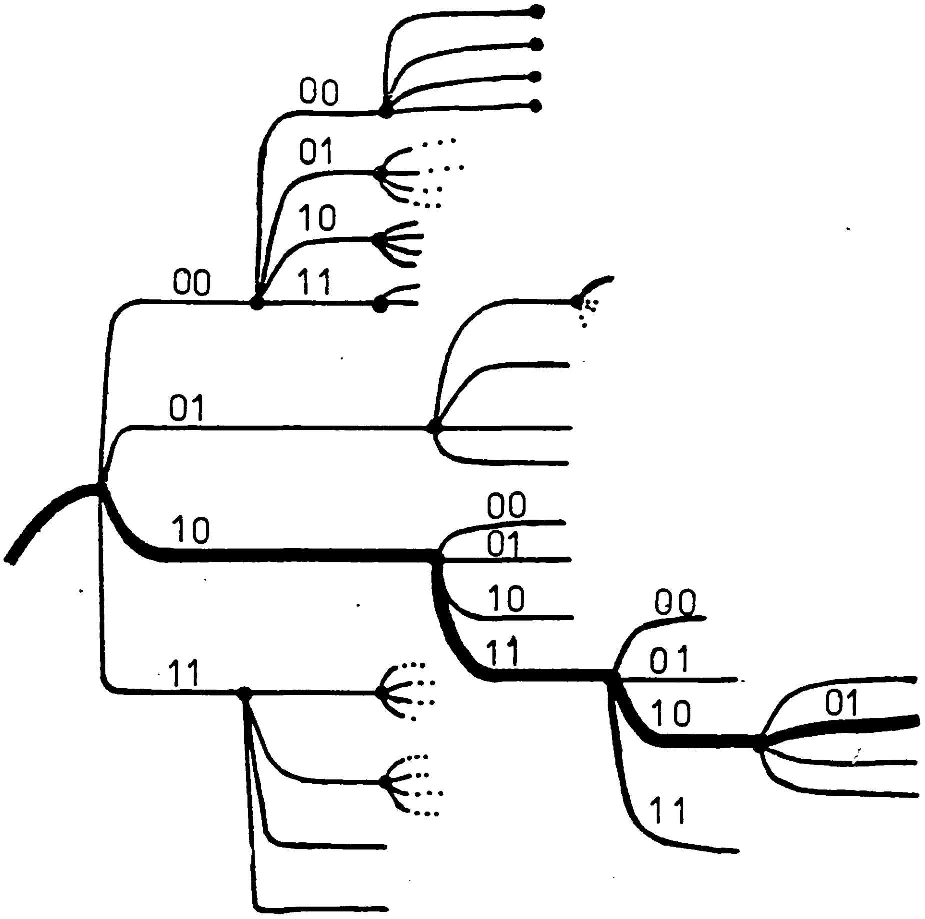


Figure 6.1 A tree, showing a possible path and the bit sequence which code a block of data.

Other methods, rarely reported upon, are based on 'vector quantisation' [Stewart, Linde and Gray-(1982)]

In this section we discuss some problems associated with multipath coding based on linear prediction and the attempts to solve these problems. The equations governing linear predictive coding are;

$$x(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) + \epsilon(n) \qquad 6.1$$

$$= \sum_{i=1}^{P} a_i \tilde{x}(n-i) + q(n) + \zeta(n) \qquad 6.2$$

$$\text{and} \quad \tilde{x}(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) + q(n) \qquad 6.3$$

where x(n) is the n-th input symbol. $\tilde{x}(n)$ is the n-th reproduction symbol. At the receiver, x(n) will be approximated by $\tilde{x}(n)$. The $\{a_i\}$ are the linear predictive filter coefficients. These will be refered to as the 'linear prediction parameters'. $\epsilon(n)$ is the difference between the linearly predicted value for x(n), that is

$$\hat{x}(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) \qquad 6.4$$

and x(n). q(n) is a quantised version of $\epsilon(n)$. In linear predictive based multipath coding, several values of q(n) are considered at each stage of coding. (In single path coding only one value of q(n) is chosen) $\zeta(n)$ is the actual coding error for symbol x(n). In summary, the coding methodology is defined by the following equation;

$$x(n) = \{\sum_{i=1}^{P} a_i \tilde{x}(n-i) + q(n)\} + \zeta(n) \qquad 6.5$$

The problem in optimum linear predictive coding is to evaluate the coefficients $\{a_i\}$ so as to minimise the average mean-square-error; $E(\zeta(n)^2)$. The following approximate model is used to determine the

$\{a_i\}$ in practice.

$$\tilde{x}(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) + q(n) \qquad 6.6$$

where the a_i are evaluated in order to minimise $E(q(n)^2)$ and there is no restriction on the values q(n) may take.

If $\tilde{x}(n)$ is sufficiently close to x(n), the model of equation 6.6 suffices. It is a simple matter to find a_i to minimise the mean-square value for q(n). With the supposition that the $\tilde{x}(n)$ and x(n) sequences had similar statistics, this set of a_i values should result in a small $E(\epsilon(n)^2)$ and hence $E(\zeta(n)^2)$. In this section a less imprecise model for linear predictive coding is presented. The model described by equation 6.1 is used.

$$x(n) = \sum_{i=1}^{P} a_i \tilde{x}(n-i) + \epsilon(n) \qquad 6.7$$

To find the values $\{a_i\}$ which give the least mean square $\epsilon(n)$ sequence, we differentiate $E(\epsilon(n)^2)$ with respect to each a_i. This yeilds a Wiener-Hopf matrix equation

$$[Y][\underline{a}] = [\underline{z}] \qquad 6.8$$

where Y is the auto-correlation matrix for the $\tilde{x}(n)$ sequence. $\underline{a}$ is the vector of filter coefficients and $\underline{z}$ is the cross-correlation between the x(n) and $\tilde{x}(n)$ sequences.

$$\underline{z} = \{E(x(n)\tilde{x}(n-1)), E(x(n)\tilde{x}(n-2)), ..., E(x(n)\tilde{x}(n-P))\}^T \qquad 6.9$$

The following assumptions are made; the error sequence $\zeta(n)=x(n)-\tilde{x}(n)$ is uncorrelated with $\tilde{x}(n-k)$, for all k and $E(\zeta(n)\zeta(n-j))=0$, for all $j\neq 0$. This results in a matrix equation identical to the "Normal equations" except for a positive factor λ which contributes to the diagonal. That is,

$$\begin{pmatrix} r_0+\lambda & r_1 & \cdot & \cdot & r_{p-1} \\ r_1 & r_0+\lambda & r_1 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ r_{p-1} & r_{p-2} & \cdot & \cdot & r_0+\lambda \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ a_p \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \\ \cdot \\ \cdot \\ \cdot \\ r_p \end{pmatrix} \qquad 6.10$$

where λ is the variance of the error signal $\zeta(n)$.

The problem then, is that of estimating the variance of the error signal. This is not straightforward because a set of a_i values are required for estimating the variance of the coding error $\zeta(n)$ and yet, the value of λ is required in order to evaluate the set $\{a_i\}$. Of course, the evaluation of this, may be attempted recursively, it is however thought that this would involve a lot of computation, and for real time application, is impractical.

The next approach is to make a direct estimate of the coding error. This may be done by estimating the rate-distortion function of the source. Alternatively, the filter coefficients, assuming zero error, may be used to make an estimate of the variance of the error signal $\zeta(n)$. The second scheme, being simpler was followed. It is quite straightforward, to estimate the coding error variance, given a set of coefficients b_i, (b_i are the coefficients represented by the model of equation 6.6)

$$\lambda = c\prod_{i=1}^{P}(1-k_i^2) \qquad 6.11$$

where the k_i are the reflection coefficients generated as intermediate products in the filter coefficient computation process.(appendix 1) c is a constant dependent upon the statistical model used to describe the $\epsilon(n)$ sequence and the number of quantisation levels used to represent q(n)

$$c \approx \frac{E(\zeta(n)^2)}{E(\epsilon(n)^2)} \qquad 6.12$$

For the work reported here, the signal is coded in blocks and the prediction coefficients adapted per block. A set of coefficients are calculated using the Burg method[Burg-1968], initially on the original data. The block of data to be coded, then has noise of the appropriate variance λ added to it. Each new reflection coefficient calculated, in the Burg algorithm, when calculated is quantised for transmission.

6.3.1 Results and discussion

In implementing the above scheme, the estimation error signal is modelled as being of either a Gaussian or Laplacian distribution. Depending on which is used, one of a set of values u(1),...,u(L) obtained using a Lloyd-Max quantiser are used to approximate the error signal. When the Gaussian model is used for the prediction error signal,

L=4 implies	u(1)=-1.51M	and L=2 implies	u(1)=-0.798M
	u(2)=-0.4528M		u(2)=0.798M
	u(3)=0.4528M		
	u(4)=1.51M		
	c = 0.1175		c = 0.3634

When the Laplacian model is used for the prediction error signal,

L=4 implies	u(1)=-1.81M	and L=2 implies	u(1)=-0.707M
	u(2)=-0.39M		u(2)=0.707M
	u(3)=0.39M		
	u(4)=1.81M		
	c = 0.1765		c = 0.5

M is an estimate of the standard deviation of the prediction error signal. The results are shown in table 6.1. Very satisfactory signal to noise ratio values are obtained, in comparison to say transform coding, especially where the step sizes are adapted regularly. As expected, the signal to noise ratio achievement improves with increasing the number of paths.

The results of adaptive tree coding of images are shown in figures 6.2 and 6.3. It may be observed that on some occasions the system is unable to cope with the very rapid variation in amplitude which occurs at feature edges. The computed step size is inadequate and a whole line sometimes, is badly coded. Were the step size larger, coding error would be poor in plane areas. This is a problem not observed in speech coding, where an average "good" step size enables reasonable coding of the whole of a block. What is therefore required is an adaptive step size computation, this appears to be very important for image coding. A step size adaptation algorithm as reported by Jayant-(1970), Cumminsky, Jayant and Flanagan-(1973) and Goodman and Gersho-(1974) was used. As before, the quantised error signal is one of L values if the transmission rate is $\log_2 L$ bits/symbol, where these values are chosen with a assumption that the error is of a Gaussian or Laplacian distribution. As mentioned before a multiplying constant M_n , is applied to each of a set of numbers determined by the model for the predicted error, to generate the set of possible quantised error values. M_n is a function of the estimated prediction error standard deviation. This multiplier M_n is now allowed to vary at each sample instant. The step size variation logic is shown in the equation below.

```
PATHS=  1   EXPONENT=  2   BUFFER LENGTH= 128   S/N=    8.52888
PATHS=  4   EXPONENT=  2   BUFFER LENGTH= 128   S/N=   11.28734
PATHS=  8   EXPONENT=  2   BUFFER LENGTH= 128   S/N=   11.55060
PATHS=  1   EXPONENT=  4   BUFFER LENGTH= 128   S/N=   13.59127
PATHS=  4   EXPONENT=  4   BUFFER LENGTH= 128   S/N=   16.48544
PATHS=  8   EXPONENT=  4   BUFFER LENGTH= 128   S/N=   16.77807
PATHS=  1   EXPONENT=  8   BUFFER LENGTH= 128   S/N=   17.62343
PATHS=  4   EXPONENT=  8   BUFFER LENGTH= 128   S/N=   19.27146
PATHS=  8   EXPONENT=  8   BUFFER LENGTH= 128   S/N=   19.67548
```

Table 6.1b Results of multipath tree coding of speech. A 4th order predictor is used and the coefficients are kept fixed. Syllabic companding is used for coding the step size. For each block of 256, a new variance estimate is made of the prediction error and use to evaluate new quantisation levels. A Gaussian model is used for the prediction error signal.

```
PATHS=  1   EXPONENT=  2   BUFFER LENGTH= 128   S/N=    7.99171
PATHS=  4   EXPONENT=  2   BUFFER LENGTH= 128   S/N=   10.18226
PATHS=  8   EXPONENT=  2   BUFFER LENGTH= 128   S/N=   10.62836
PATHS=  1   EXPONENT=  4   BUFFER LENGTH= 128   S/N=   14.67488
PATHS=  4   EXPONENT=  4   BUFFER LENGTH= 128   S/N=   17.20935
PATHS=  8   EXPONENT=  4   BUFFER LENGTH= 128   S/N=   17.41278
```

Table 6.1c Results for multipath tree coding of speech, conditions are the same as those of table 6.1b except that Laplacian model is used for the prediction error signal and quantisation levels chosen accordingly.

```
PATHS=  1    EXPONENT=  4    BUFFER LENGTH=  32    S/N=    10.31530
PATHS=  4    EXPONENT=  4    BUFFER LENGTH=  32    S/N=    12.28051
PATHS=  8    EXPONENT=  4    BUFFER LENGTH=  32    S/N=    12.79777
PATHS=  4    EXPONENT=  4    BUFFER LENGTH=  64    S/N=    12.27789
PATHS=  4    EXPONENT=  4    BUFFER LENGTH= 128    S/N=    12.35737
PATHS=  1    EXPONENT=  2    BUFFER LENGTH= 128    S/N=     5.88364
PATHS=  4    EXPONENT=  2    BUFFER LENGTH= 128    S/N=     7.81316
PATHS=  8    EXPONENT=  2    BUFFER LENGTH= 128    S/N=     8.06916
PATHS=  1    EXPONENT=  8    BUFFER LENGTH= 128    S/N=    12.52345
PATHS=  4    EXPONENT=  8    BUFFER LENGTH= 128    S/N=    14.08971
PATHS=  8    EXPONENT=  8    BUFFER LENGTH= 128    S/N=    14.56803
PATHS=  1    EXPONENT=  4    BUFFER LENGTH= 128    S/N=    10.31580
PATHS=  4    EXPONENT=  4    BUFFER LENGTH= 128    S/N=    12.35737
PATHS=  8    EXPONENT=  4    BUFFER LENGTH= 128    S/N=    12.75397
```

Table 6.1a Results of non-adaptive multipath tree coding, Gaussian model is used for quantising residual. A 4th order predictor is used.

PATHS=	1	EXPONENT=	2	BUFFER LENGTH=	128	S/N=	8.44675
PATHS=	4	EXPONENT=	2	BUFFER LENGTH=	128	S/N=	11.91222
PATHS=	8	EXPONENT=	2	BUFFER LENGTH=	128	S/N=	12.28032
PATHS=	1	EXPONENT=	4	BUFFER LENGTH=	128	S/N=	15.22299
PATHS=	4	EXPONENT=	4	BUFFER LENGTH=	128	S/N=	19.22484
PATHS=	8	EXPONENT=	4	BUFFER LENGTH=	128	S/N=	19.93847
PATHS=	1	EXPONENT=	8	BUFFER LENGTH=	128	S/N=	21.41381
PATHS=	4	EXPONENT=	8	BUFFER LENGTH=	128	S/N=	24.71616
PATHS=	8	EXPONENT=	8	BUFFER LENGTH=	128	S/N=	25.46754

Table 6.1d Results of adaptive multipath tree coding of speech. A 4th order predictor is employed. The 1st two reflection coefficients quantised and coded with 8 bits/coefficients and the 3rd and 4th with 4 bits/coefficient. The prediction coefficients and prediction error estimates are updated every 256 sample periods. A Gaussian model is employed for the prediction error signal for exponent=2, otherwise a Laplacian model is used. Adaptation information rate=30bits/block

PATHS=	1	EXPONENT=	2	BUFFER LENGTH=	128	S/N=	7.92061
PATHS=	4	EXPONENT=	2	BUFFER LENGTH=	128	S/N=	10.77414
PATHS=	8	EXPONENT=	2	BUFFER LENGTH=	128	S/N=	11.18465
PATHS=	1	EXPONENT=	4	BUFFER LENGTH=	128	S/N=	13.59804
PATHS=	4	EXPONENT=	4	BUFFER LENGTH=	128	S/N=	17.47357
PATHS=	8	EXPONENT=	4	BUFFER LENGTH=	128	S/N=	18.15260
PATHS=	1	EXPONENT=	8	BUFFER LENGTH=	128	S/N=	20.23861
PATHS=	4	EXPONENT=	8	BUFFER LENGTH=	128	S/N=	23.14097
PATHS=	8	EXPONENT=	8	BUFFER LENGTH=	128	S/N=	23.86367

Table 6.1e Results of speech coding using adaptive multipath tree coding. The coding curcumstances are the same as those for table 6.1d except that the prediction coefficients and prediction error variance values are updated every 512 sample periods.

Figure 6.2a AFTAB original.

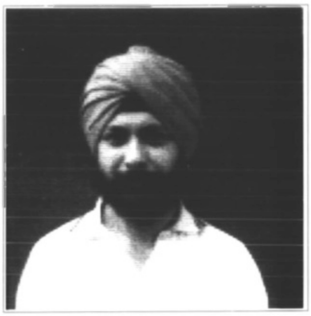
Figure 6.2b Step size adaptive tree coding of image AFTAB. Fixed 4-th order predictor used. Number of paths=1, Block size=128, S/N=20.69dB.

Figure 6.2c Step size adaptive tree coding of image AFTAB. Fixed 4-th order predictor used. Number of paths=4, Block size=128, S/N=22.71dB.

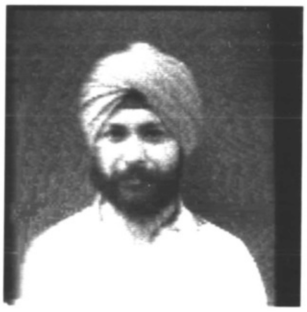
Figure 6.2d Adaptive tree coding of image AFTAB. Adaptive 4-th order predictor used. Number of paths=1, Block size=128, S/N=20.99dB.

Figure 6.2e Adaptive tree coding of image AFTAB. Adaptive 4-th order predictor used. Number of paths=4, Block size=128, S/N=22.65dB.

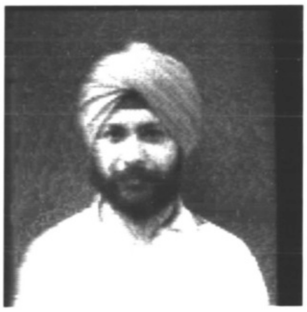
(a)



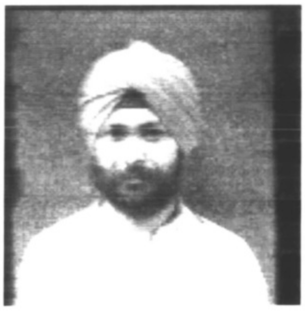
(b)



(c)



(d)



(e)

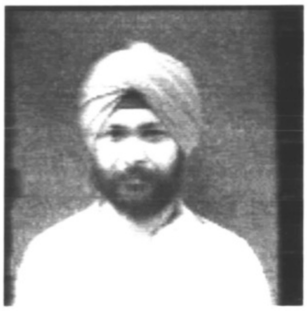


Figure 6.3a TELEBOX original.

Figure 6.3b Step size adaptive tree coding of image TELEBOX. Fixed 4-th order predictor used. Number of paths=1, Block size=128, S/N=17.71dB.

Figure 6.3c Step size adaptive tree coding of image TELEBOX. Fixed 4-th order predictor used. Number of paths=4, Block size=128, S/N=19.64dB.

Figure 6.3d Adaptive tree coding of image TELEBOX. Adaptive 4-th order predictor used. Number of paths=1, Block size=128, S/N=16.30dB.

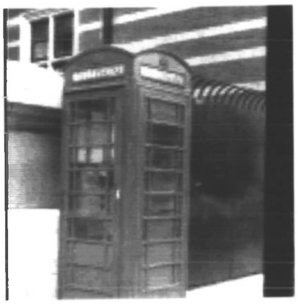
Figure 6.3e Adaptive tree coding of image TELEBOX. Adaptive 4-th order predictor used. Number of paths=4, Block size=128, S/N=17.81dB.

Figure 6.3f GEORGE original.

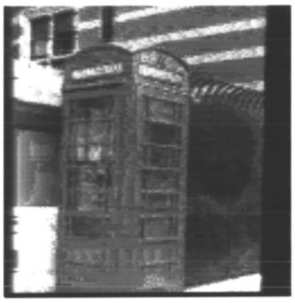
Figure 6.3g Adaptive tree coding of image GEORGE. Adaptive 4-th order predictor used. Number of paths=1, Block size=128, S/N=13.51dB.

Figure 6.3h Adaptive tree coding of image GEORGE. Adaptive 4-th order predictor used. Number of paths=4, Block size=128, S/N=13.68dB.

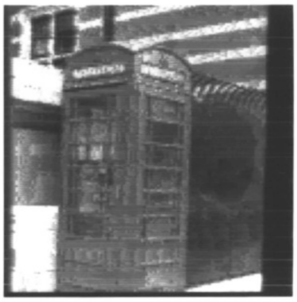
(a)



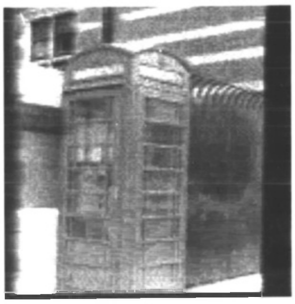
(b)



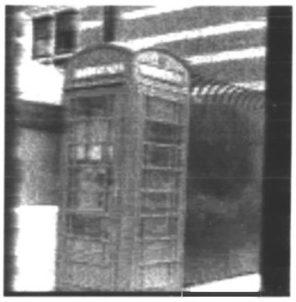
(c)



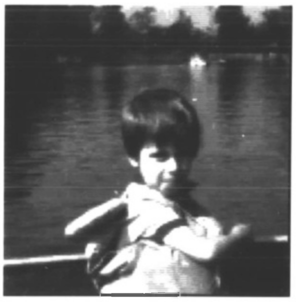
(d)



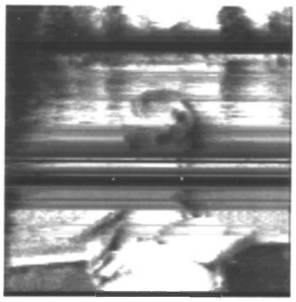
(e)



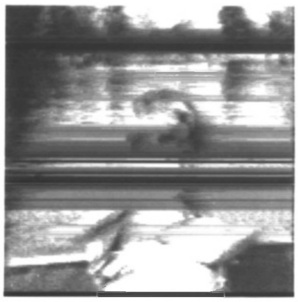
(f)



(g)



(h)



$$\begin{aligned} M_n &= \alpha M_{n-1} \quad \text{if } (q(n-1), q(n-2)) \in \{(u(1), u(1)), \quad (u(L), u(L))\} \\ &= \frac{1}{\alpha} M_{n-1} \quad \text{otherwise} \\ \alpha &= 1.4 \end{aligned} \qquad 6.13$$

In addition to the above scheme, at the beginning of each coded block, a new estimate of an average M is computed; this is denoted as M_1 and used to initiate the step size adaptation scheme. This is because it was observed that the above adaptation scheme was liable to become unstable if left unattended. A further variation on the above adaptation scheme was employed. This is particularly suited to image coding, where a large step size is only required in the region of an edge. This step size adaptation algorithm is shown below.

$$\begin{aligned} M_n &= \alpha M_{n-1} \quad \text{if } (q(n-1), q(n-2)) \in \{(u(1), u(1)), \quad (u(L), u(L))\} \\ &= M_1 \quad \text{otherwise} \\ \alpha &= 1.4 \end{aligned} \qquad 6.14$$

This results in an increase in step size at an edge when the step sizes being employed are too small, presumably when an edge is observed. When the step size is too large, presumably when one is no longer in a busy region, the step size is immediately returned to the average step size estimated for that block, instead of the gentle reduction implied by the above scheme. The results are shown in figure 6.4.

6.4 Adaptive tree coding by parameter matching

In this section, the results of an adaptive multipath tree coding scheme are presented. The scheme relies upon a library of filter coefficients evaluated from the previously coded blocks of

Figure 6.4a AFTAB original.

Figure 6.4b Adaptive tree coding of image AFTAB Adaptive 4-th order predictor used. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=1, Block size=128, S/N=21.82dB.

Figure 6.5b Adaptive tree coding of image AFTAB Adaptive 4-th order predictor used. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=4, Block size=128, S/N=23.57dB.

Figure 6.4d TELEBOX original.

Figure 6.4e Adaptive tree coding of image TELEBOX Adaptive 4-th order predictor used. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=1, Block size=128, S/N=17.25dB.

Figure 6.4f Adaptive tree coding of image TELEBOX Adaptive 4-th order predictor used. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=4, Block size=128, S/N=18.86dB.

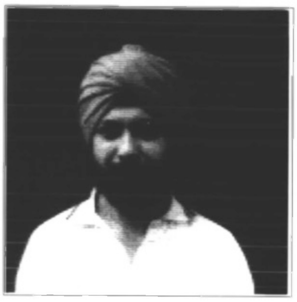
Figure 6.4g GEORGE original.

Figure 6.4h Adaptive tree coding of image GEORGE Adaptive 4-th order predictor used. Instantaneous step size

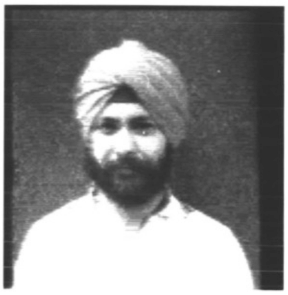
adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=1, Block size=128, S/N=19.93dB.

Figure 6.4i Adaptive tree coding of image GEORGE Adaptive 4-th order predictor used. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=4, Block size=128, S/N=21.94dB.

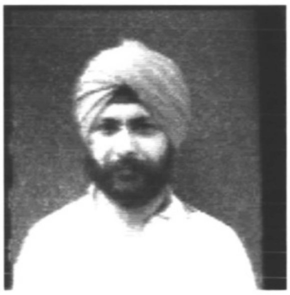
a



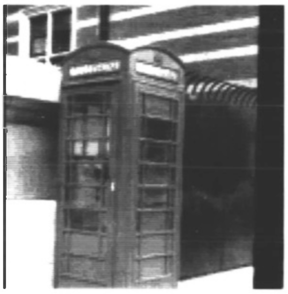
b



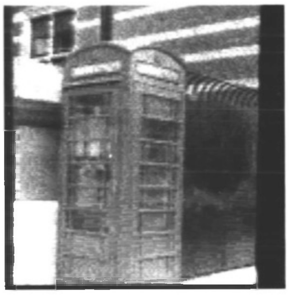
c



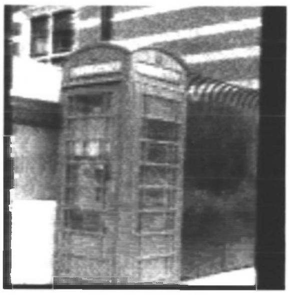
d



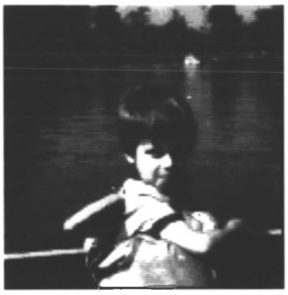
e



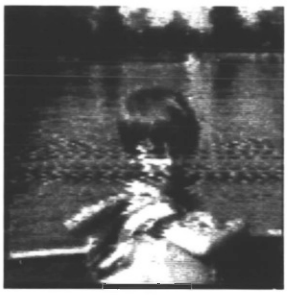
f



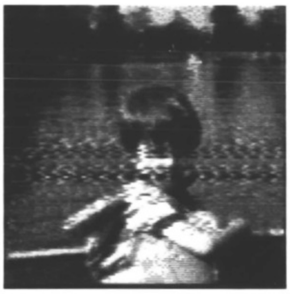
g



h



i



data. Firstly, a description of the coding system:

For a block of length N samples, P reflection coefficients are extracted using the Burg-Maximum Entropy method. This set of reflection coefficients are compared with the members of a library of reflection coefficients. In the computer simulation of the scheme, 64 sets of reflection coefficients are employed. The coding scheme is based upon the M-path search as described by Anderson and Bodie-(1975) and Wilson and Hussain-(1977) and later by Matsuyama and Gray-(1980).

The points requiring discussion are; the way in which the library is formed and the way in which a library member is chosen as the basis of a linear predictive system.

For a library of K parameter sets, K-2 sets are the LPC parameters associated with previously coded blocks. One parameter set consists of P zeros (reflection or prediction coefficients) and another, a set which represents the long term statistics of the source reasonably well. Each time a block, represented by a vector X is coded, so that an approximation vector $\tilde{X}$ is obtained, the LPC parameters are extracted for the block $\tilde{X}$. Note that the $\tilde{X}$ sequence is known at both the receiver and transmitter. This set of LPC parameters are included in the duplicate libraries maintained at both the transmitter and receiver. Before coding a block, the LPC parameters of this block are evaluated. Refer to the filter coefficients thus obtained as $A=\{a_1,\ldots,a_P\}$. Now of the K (64 say) library LPC parameter sets, we require to find that which allows the least mean square error coding of the block under consideration. This is undertaken by comparing the coefficients A with those B say, of each member of the library. The next section shows how the coding error is estimated.

6.4.1 The variance of the error signal in linear estimation based tree coding.

In this section we shall consider the options posed in the approximation of the variance of the coding error signal.

Let $$e_0(n) = x(n) - \sum_{i=1}^{P} b_i x(n-i) \qquad e_1(n) = x(n) - \sum_{i=1}^{P} b_i \tilde{x}(n-i) \qquad 6.15$$

$$\text{where} \quad \tilde{x}(n) = \sum_{i=1}^{P} b_i \tilde{x}(n-i) + q(n) \qquad 6.16$$

q(n) is some quantised signal which may take only one of L values. We attempt to identify the magnitude of the error signal $\zeta(n)$ defined as follows.

$$\zeta(n) = x(n) - \tilde{x}(n) \qquad 6.17$$

Now by the coding mechanism, q(n) is some function of $e_1(n)$. For single path coding, this function Q say, defined below, is deterministic.

$$q(n) = Q(e_1(n)) \qquad 6.18$$

For multipath coding, Q is a stochastic function.

$$\begin{aligned} \text{now} \quad \zeta(n) &= e_1(n) - q(n) \quad \text{from 6.15, 6.16 and 6.17} \\ &= Q_1(e_1(n)) \end{aligned} \qquad 6.19$$

(Q_1 shown in fig 6.5)

Where Q_1 represents the possibly stochastic function 1- Q. Thus $E(\zeta(n)^2)$ is dependent upon $E(e_1(n)^2)$ and the variance magnification or reduction effect of Q_1.

$$\text{Let} \quad K = E\left\{\frac{E(\zeta(n)^2)}{E(e_1(n)^2)}\right\} \qquad 6.20$$

Then we have two tasks to tackle. The first is to ascertain some constant K, a function of

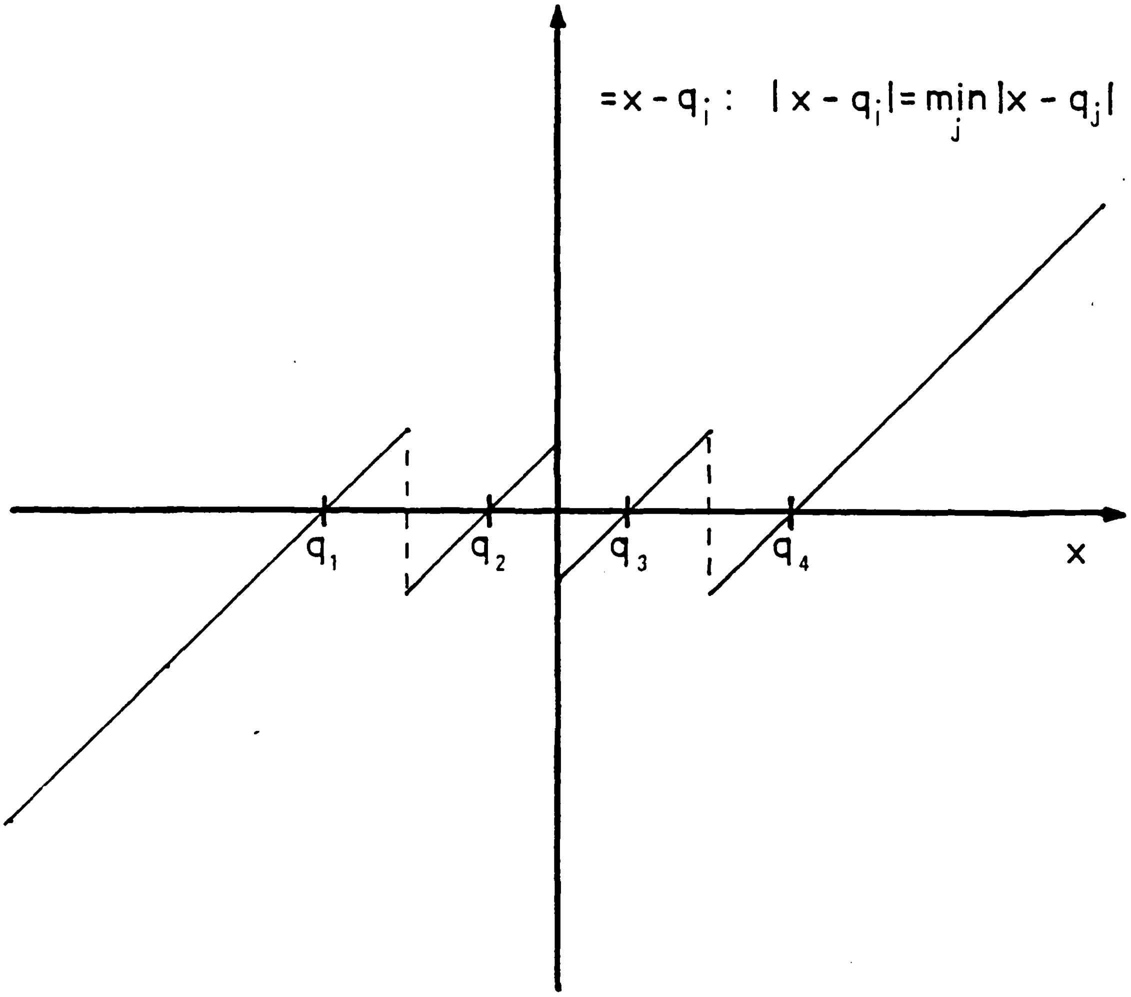


Figure 6.5 Plot of function Q_1 relating $e_1(n)$ to $\zeta(n)$ that is, the quantisation function.

1) the source statistical model,

2) the number of quantisation levels of the error signal and

3) the number of paths considered in the tree coding algorithm.

The second is the estimation of $E(e_1(n)^2)$. We shall consider this first.

Suppose
$$\begin{aligned} e_1(n)^2 &= \{x(n) - \sum_{i=1}^{P} b_i\tilde{x}(n-i)\}^2 \\ &= \{x(n) - \sum_{i=1}^{P} b_i(x(n-i) - \zeta(n-i))\}^2 \\ &= \{x(n) - \sum_{i=1}^{P} b_i x(n-i) + \sum_{i=1}^{P} b_i\zeta(n-i)\}^2 \end{aligned}$$

then
$$E(e_1(n)^2) = E(e_0(n)^2) + 2\sum_{i=1}^{P} b_i E(e_0(n)\zeta(n-i)) + \sum_{i=1}^{P}\sum_{j=1}^{P} b_i b_j E(\zeta(n-i)\zeta(n-j)) \qquad 6.21$$

By the use of the fact that $e_0(n) = e_1(n) - \sum_{i=1}^{P} b_i\zeta(n-i)$ we write an alternative form of $E(e_1(n)^2)$

$$E(e_1(n)^2) = E(e_0(n)^2) + 2\sum_{i=1}^{P} b_i E(e_1(n)\zeta(n-i)) - \sum_{i=1}^{P}\sum_{j=1}^{P} b_i b_j E(\zeta(n-i)\zeta(n-j)) \qquad 6.22$$

Let us write A, the ratio between $E(e_1(n)^2)$ and $E(e_0(n)^2)$. That is

$$A = \frac{E(e_1(n)^2)}{E(e_0(n)^2)} \qquad 6.23$$

Two methods of approximating A were tried, this because of the difficulty of evaluating the quantities

$$2\sum_{i=1}^{P} b_i E(e_0(n)\zeta(n-i)) + \sum_{i=1}^{P}\sum_{j=1}^{P} b_i b_j E(\zeta(n-i)\zeta(n-j))$$

or
$$2\sum_{i=1}^{P} b_i E(e_1(n)\zeta(n-i)) - \sum_{i=1}^{P}\sum_{j=1}^{P} b_i b_j E(\zeta(n-i)\zeta(n-j))$$

The first relied on the expectation that this ratio would change slowly, as one went from block to block. The value of the previous block's A is used in a present block.

The second relied upon the following assumption

$$2\sum_{i=1}^{P} b_i E\left(e_0(n)\zeta(n-i)\right) + \sum_{i=1}^{P}\sum_{j=1}^{P} b_i b_j E\left(\zeta(n-i)\zeta(n-j)\right) \approx k_1^2 E\left(\zeta(n)^2\right) \qquad 6.24$$

where k_1 is the first reflection coefficient associated with the sequence $b_1, \ldots, b_P$ of prediction filter coefficients. This assumption is dictated by a combination of what might be reasonably approximated, k_1 and $E(\zeta(n)^2)$ and by the assumption that the error signals are uncorrelated in the following ways.

$$E\left(e_0(n)\zeta(n-i)\right) \approx 0 \qquad \forall i \quad \forall i \neq 0$$

$$E\left(\zeta(n-i)\zeta(n-j)\right) \approx 0 \qquad \forall i \neq j \qquad 6.25$$

and the effect on A of the use of a model of order greater than 1, may be neglected. Thus

$$E\left(e_1(n)^2\right) \approx \underline{B}^T R \underline{B} + k_1^2 E\left(\zeta(n)^2\right) \qquad 6.26$$

R has members $r_{ij} = E(x(n-i)x(n-j))$ and $B = \{1, -b_1, \ldots, -b_P\}$

In the alternative assumption, where the ratio $A = \frac{E(e_1(n)^2)}{E(e_0(n)^2)}$ resulting from the previous block is used for the present block, we obtain the approximation

$$E\left(e_1(n)^2\right) \approx E\left(e_0(n)^2\right)\frac{E\left(e_1(n)^2\right)}{E\left(e_0(n)^2\right)}\Big|_{\text{from previous block}} \qquad 6.27$$

The next task is to estimate $E(\zeta(n)^2)$, given $E(e_1(n)^2)$. By the assumption that the signal $e_1(n)$ is of a Gaussian or Laplacian distribution and that the signal is simply quantised such that the error signal $\zeta(n)$ is the resulting quantisation error, we estimate $E(\zeta(n)^2)$. For a Gaussian source, the quantisation error variances are v=0.3634 and 0.1175 for 1 and 2bits/symbol coding. For the Laplacian source, the quantisation errors are v=0.5 and 0.1765 respectively for 1 and 2bits/symbol coding. Thus for the assumption of equation 6.27

$$E(\zeta(n)^2) \approx E(e_0(n)^2)\frac{E(\zeta(n)^2)}{E(e_0(n)^2)}\Big|_{\text{from previous block}} \qquad 6.28$$

and for the assumption of equation 6.26

$$E(\zeta(n)^2) \approx v[\underline{B}^T R \underline{B} + k_1^2 E(\zeta(n)^2)] = \frac{v\underline{B}^T R \underline{B}}{1 - vk_1^2} \qquad 6.29$$

6.4.2 Results and discussion

The scheme described above was implemented with various library sizes and used to code both speech and image data. Adaptation with various block sizes was investigated. For each block the following side information was sent. 6 bits to code the step size information and 6 bits to code the library coordinate in use. The results are given in table 6.2 and figure 6.6.

Coding with a block size of 96 in the above scheme results in the same bit rate as the adaptive tree coding scheme of section 6.3 with a block size of 256. (For the method of section 6.3, 24 bits are transmitted per block to represent the prediction filter coefficients) Comparing the results of the two schemes, we observe that the adaptive predictor of section 6.4 gives superior results when single path coding is being undertaken. On other occasions the simpler scheme of section 6.3 performed better. This is probably because the values of v employed are too large for multipath coding.

When the estimation scheme of equation 6.28 is employed, rather dis apointing results were obtained.

```
PATHS=  1    EXPONENT=  2    BUFFER LENGTH= 128     S/N=         9.78233
PATHS=  4    EXPONENT=  2    BUFFER LENGTH= 128     S/N=        11.16974
PATHS=  8    EXPONENT=  2    BUFFER LENGTH= 128     S/N=        10.96774
PATHS=  1    EXPONENT=  4    BUFFER LENGTH= 128     S/N=        16.67945
PATHS=  4    EXPONENT=  4    BUFFER LENGTH= 128     S/N=        18.28475
PATHS=  8    EXPONENT=  4    BUFFER LENGTH= 128     S/N=        18.33772
PATHS=  1    EXPONENT=  8    BUFFER LENGTH= 128     S/N=        19.18630
PATHS=  4    EXPONENT=  8    BUFFER LENGTH= 128     S/N=        21.33475
PATHS=  8    EXPONENT=  8    BUFFER LENGTH= 128     S/N=        21.87574
```

Table 6.2a Results of adaptive multipath tree coding of speech. A 8th order predictor is employed. The prediction parameters are extracted from the previously coded symbols. 64 sets of parameters extracted from the previously coded block are stored and sampled to find the best approximate set for a target block. 6 bits to indicate prediction coeffs and 6 bits for prediction error estimates are transmitted every 128 sample periods. A Gaussian model is employed for the prediction error signal for exponent=2, otherwise a Laplacian model is used. Adaptation information coded with 12bits/block.

```
PATHS=  1    EXPONENT=  2    BUFFER LENGTH=  96     S/N=          .00368
PATHS=  4    EXPONENT=  2    BUFFER LENGTH=  96     S/N=        12.45977
PATHS=  8    EXPONENT=  2    BUFFER LENGTH=  96     S/N=        13.08969
PATHS=  1    EXPONENT=  4    BUFFER LENGTH=  96     S/N=        16.93618
PATHS=  4    EXPONENT=  4    BUFFER LENGTH=  96     S/N=        19.04717
PATHS=  8    EXPONENT=  4    BUFFER LENGTH=  96     S/N=        19.92685
```

Table 6.2b Results of speech coding using adaptive multipath tree coding. The scheme is identical to that which generated the results of table 6.2a except that the filter coefficient and prediction error variance information is transmitted every 96 sample instants. Adaptation information coded with with 12bits/block

PATHS=	1	EXPONENT=	4	BUFFER LENGTH=	96	S/N=	14.92796
PATHS=	4	EXPONENT=	4	BUFFER LENGTH=	96	S/N=	16.00607
PATHS=	8	EXPONENT=	4	BUFFER LENGTH=	96	S/N=	16.67022
PATHS=	1	EXPONENT=	2	BUFFER LENGTH=	96	S/N=	-.24332
PATHS=	4	EXPONENT=	2	BUFFER LENGTH=	96	S/N=	5.71741
PATHS=	8	EXPONENT=	2	BUFFER LENGTH=	96	S/N=	6.10713

Table 6.2c Results of adaptive multipath tree coding of speech. A 8th order predictor is employed. The prediction parameters are extracted from the previously coded symbols. 64 sets of parameters extracted from the previously coded block are stored and sampled to find the best approximate set for a target block. 6 bits to indicate prediction coeffs and 6 bits for prediction error estimates are transmitted every 96 sample periods. A Gaussian model is employed for the prediction error signal for exponent=2, otherwise a Laplacian model is used. Adaptation information coded with 12bits/block. In estimating the variance of the prediction error signal, the actual prediction error for the previously coded block, is employed.

PATHS=	1	EXPONENT=	4	BUFFER LENGTH=	128	S/N=	15.77090
PATHS=	4	EXPONENT=	4	BUFFER LENGTH=	128	S/N=	18.39192
PATHS=	8	EXPONENT=	4	BUFFER LENGTH=	128	S/N=	18.60713
PATHS=	1	EXPONENT=	2	BUFFER LENGTH=	128	S/N=	8.21394
PATHS=	4	EXPONENT=	2	BUFFER LENGTH=	128	S/N=	11.97855
PATHS=	8	EXPONENT=	2	BUFFER LENGTH=	128	S/N=	12.46636

Table 6.2d Results of adaptive multipath tree coding of speech. A variable order predictor is employed. The prediction parameters are extracted from the previously coded symbols. 64 sets of parameters extracted from the previously coded block are stored and sampled to find the best approximate set for a target block. In addition, for each set of prediction or reflection coefficients, the best order to use for prediction is ascertained. For the example above, orders of 1 to 4 are allowed. The order is coded with 2 bits. 6 bits to indicate prediction coeffs and 6 bits for prediction error estimates are transmitted every 128 sample periods.

A Gaussian model is employed for the prediction error signal for exponent=2, otherwise a Laplacian model is used. Adaptation information coded with 14bits/block.

Figure 6.6a AFTAB original

Figure 6.6b Adaptive tree coding of image AFTAB Adaptive 4-th order predictor used. Coefficients derived from previously coded data. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=1, Block size=064, S/N=22.18dB.

Figure 6.6c Adaptive tree coding of image AFTAB Adaptive 4-th order predictor used. Coefficients derived from previously coded data. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=4, Block size=064, S/N=22.17dB.

Figure 6.6d Adaptive tree coding of image AFTAB Adaptive variable order predictor used. Coefficients derived from previously coded data. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=1, Block size=128, S/N=22.52dB.

Figure 6.6e Adaptive tree coding of image AFTAB Adaptive variable order predictor used. Coefficients derived from previously coded data. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=4, Block size=128, S/N=24.41dB.

Figure 6.6f TELEBOX original.

Figure 6.6g Adaptive tree coding of image TELEBOX. Adaptive variable order predictor used. Coefficients derived from previously coded data. Instantaneous set size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=1, Block size=128, S/N=19.473dB.

Figure 6.6h Adaptive tree coding of image TELEBOX Adaptive variable order predictor used. Coefficients derived from previously coded data. Instantaneous step size adaptation used, step size increases by factor 1.4, on slope overload, and drops instantaneously to default otherwise. Number of paths=1, Block size=128, S/N=17.836dB.

a

b

c

d

e

f

g

h

For the case where a variable predictor order is employed, signal to noise ratios very similar values to fixed order system were obtained, with a little improvement when a single path search is employed.

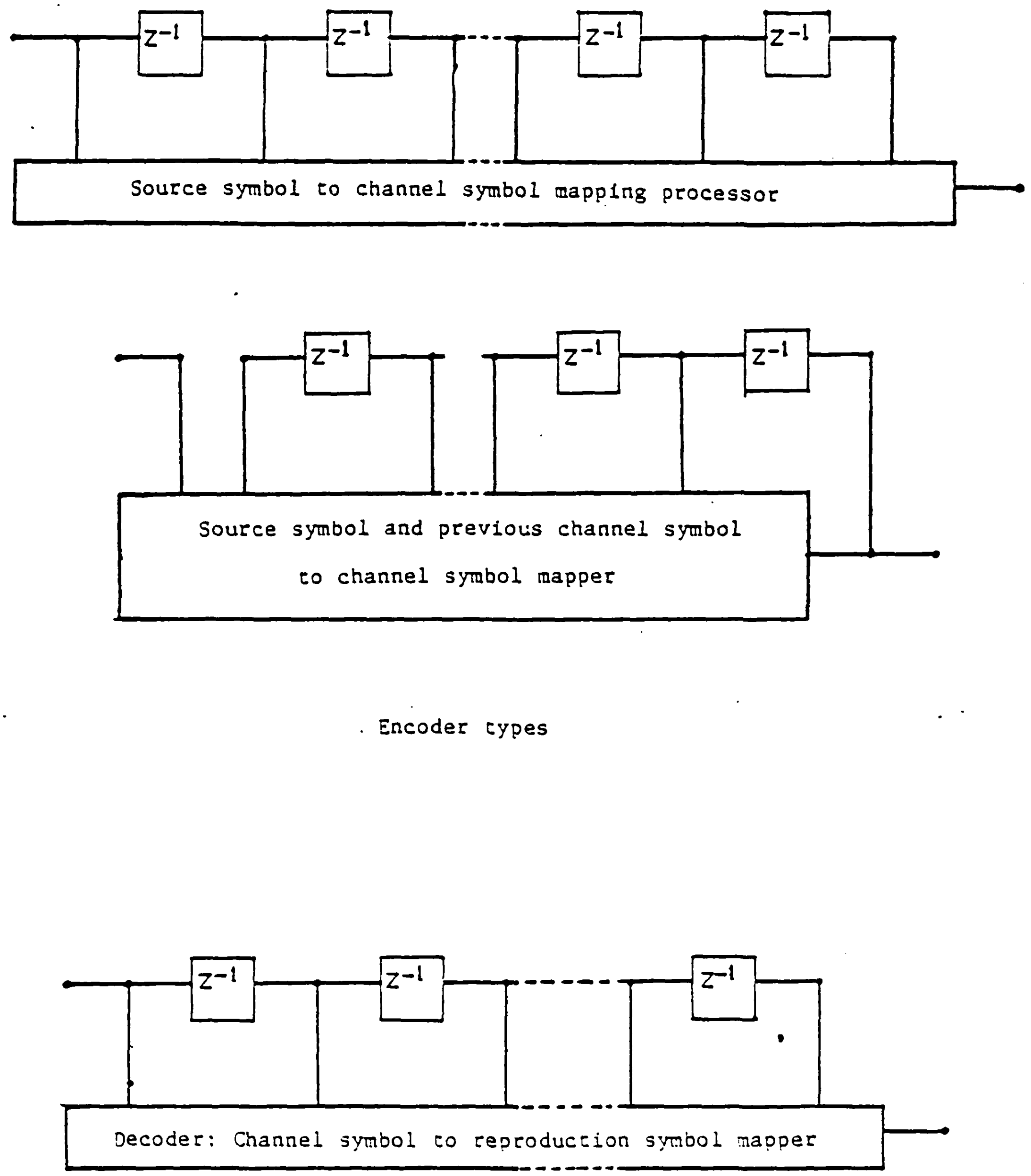
6.5 Trellis coding with a codebook

Trellis coding is an alternative to tree coding for multipath search coding. Most schemes for trellis coding have been very similar to those for tree coding and have been based on linear prediction. In fact, the M-path tree coding and trellis coding schemes are very similar, the only differences being how many paths are selected at each sample instant and the method by which these paths are selected. (The trellis structure, in general, is more restrictive in the paths which may be selected)

As has been stressed before, the focus of attention in this chapter is subject of 'colouring schemes' used in multipath search coding and not the way the trees or trellises are searched. In this direction, a relatively recent and interesting method for colouring will be discussed, and in this section, a method proposed for overcoming a disadvantage of this scheme.

This colouring scheme was proposed by R. M. Gray and results presented by Stewart, Gray and Linde-(1982). The basic algorithm was described in chapter 5, but for the purposes of a reminder, it will be described again very briefly.

Multipath search coding as described by the above authors requires a codebook described in the following manner. Suppose the coding rate to be used is 'm' channel symbols per source symbol. Also suppose the channel symbol space is of membership size C. The coding exponent is $L=C^m$. Then if a convolutional code with constraint length K is used, a library of C^K members, results. A block diagram of the convolutional coder is shown in figure 6.7. Simply, a sequence of previous channel symbols indicates a section


Z-1
Source symbol to channel symbol mapping processor
Source symbol and previous channel symbol to channel symbol mapper
Decoder: Channel symbol to reproduction symbol mapper

Encoder types

Decoder

Figure 6.7 Block diagram of convolutional coder.

of a codebook to search. In this portion of the codebook, there are only L possible values of reproduction symbol $\tilde{y}$ which may be used to approximate y. Corresponding to each is a sequence of channel symbols which may be transmitted. The "best" reproduction sequence is selected and the corresponding channel sequence transmitted. At the next source sample instant, the set of channels symbols which had been transmitted will be employed to define which section of the codebook will be searched for a reproduction symbol. A positive feature of this coding scheme is that of low complexity in implementation. There are no multiplications, unlike linear prediction based multipath coding. The only operations are lookups and comparisons. The drawback of the system is the difficulty of defining a good codebook. This codebook is dependent upon source statistics and even when these are known, there is no obvious one pass scheme for determining the codebook values.

A practical scheme for codebook definition has been suggested by Linde, Stewart and Gray-(1982), which is based upon the quantisation axioms of Lloyd-(1982) and Max-(1962). This is a recursive, not a 'one pass' algorithm, with its attendant problems of stability and convergence. In the following paragraphs, the quantisation/codebook definition algorithm as suggested by the above authors is described and the problems encountered in the application of this scheme discussed. After this, a proposed adaptive version of the codebook based trellis coder is described.

6.5.1 Codebook definition

Suppose a codebook for a certain application were defined as shown in table 6.3. The constraint length k is 2, the channel

b(n-1)	b(n)	c b(n)
		b(n-1)
1	1	-1.30
2	1	.30
1	2	-.30
2	2	1.30

Table 6.3 Example of the codebook entries for a convolutional coder.

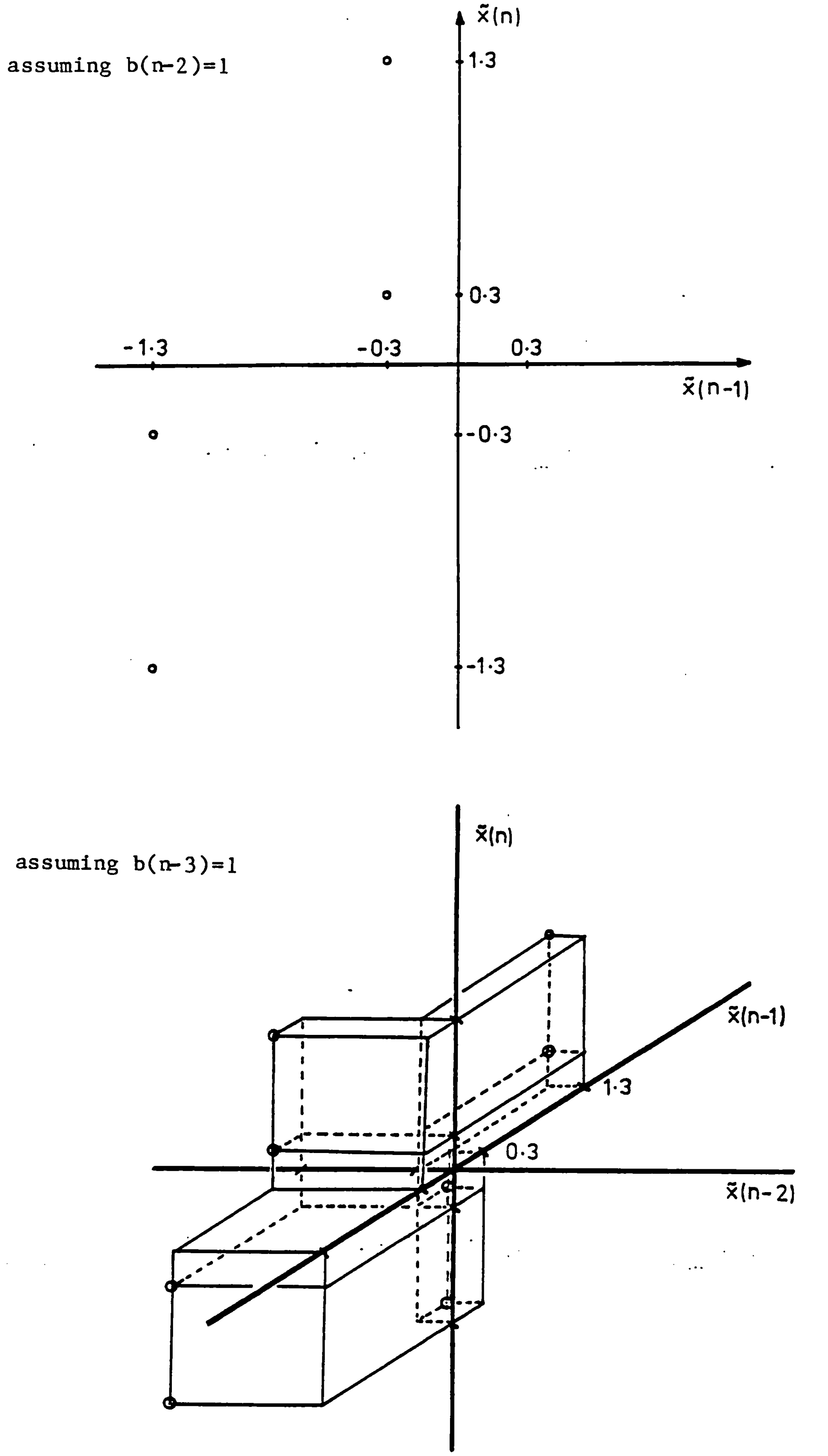


Figure 6.8 Positions of 2 and 3-dimensional centroids for the example codebook of table 6.3.

symbol space has a cardinality of 2 and the number of channel symbols transmitted per source symbol is one. Then the 2-dimensional space for the source symbol pair {x(n),x(n-1)} would have possible reproduction symbol pairs situated as shown in figure 6.8. By considering the space formed by more source symbols, and the associated possible reproduction symbols, the diagram of figure 6.8 may be extended to higher dimensions. We may then observe that the coding scheme is similar to block quantisation, where we want to find for the large dimensional space, the set of reproduction vectors which will allow the coding of the source symbols with "small" error. We may thus set about choosing this set of reproduction vectors by the same techniques as are used in block or vector quantisation. The "colouring" task may therefore be tackled using the concepts of Max-Lloyd quantisers. Thus we present the two statements for quantisation; but first a few definitions

1) A centroid; granted a certain distortion measure $d(x,\bar{x})$ such that for a region S,

$$\int_S \mathrm{d}P(x)d(x,m) = \min_{\forall y \in S} \int_S \mathrm{d}P(x)d(x,y) \qquad 6.30$$

we have a centroid m, for that region.

A partition is the boundary between two disjoint regions which touch.

A Max-Lloyd quantiser is designed by successively finding the best partitions and centroids for a test signal. This procedure has been used to define codebooks for speech and image data. For each block of data, the signal was normalised with respect to its mean and variance. These were quantised and coded separately. The normalised data is then coded by the method described.

The results are shown in table 6.4 for various coding rates, constraint lengths and block sizes.

A pair of codebooks were designed for image coding, using a data base of four images. Eight passes were employed, after which the codebook members had converged. The resulting codebooks shown in table 6.5 were employed to code three of the images used in the database; AFTAB, TELEBOX and GEORGE, for a 2 bit/pixel rate. The results are shown in figure 6.9. As may be observed, there is significant improvement as the constraint length is increased. It must be concluded that for such a simple coding scheme, good results are obtained.

6.5.2 Problems

There are two main difficulties associated with the scheme as it stands. The first is that the codebook construction procedure as described here has dubious convergence properties. Although a discussion of the possible reasons for nonconvergence is undertaken, we have felt unable to suggest a better solution. The second is that precise source statistics or very large quantities of typical source data are required for the design of this sort of convolutional coder.

6.6 An adaptive codebook based coder

It would seem obvious that an adaptive version of the codebook based coder would be suggested. The design of an adaptive coder is frought with problems however, the greatest of these being that of transmitting a new codebook or description of a new codebook. In

Figure 6.9a AFTAB original

Figure 6.9b Trellis coding of image AFTAB with codebook.
Constraint length=2, rate=2bits/pixel, S/N=23.42db

Figure 6.9c Trellis coding of image AFTAB with codebook.
Constraint length=4, rate=2bits/pixel, S/N=25.30db

Figure 6.9d TELEBOX original

Figure 6.9e Trellis coding of image TELEBOX with codebook.
Constraint length=2, rate=2bits/pixel, S/N=22.92db

Figure 6.9f Trellis coding of image TELEBOX with codebook.
Constraint length=4, rate=2bits/pixel, S/N=23.99db
Figure 6.9g GEORGE original.

Figure 6.9g GEORGE original

Figure 6.9h Trellis coding of image GEORGE with codebook.
Constraint length=2, rate=2bits/pixel, S/N=26.46db

Figure 6.9i Trellis coding of image GEORGE with codebook.
Constraint length=4, rate=2bits/pixel, S/N=28.48db

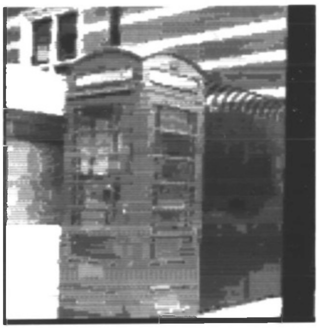
a

b

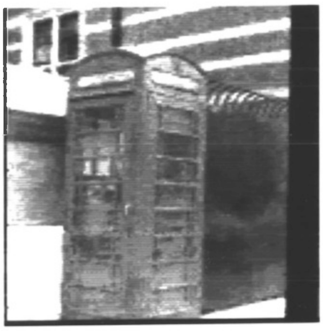
c

d

e



f



g

h

i

```
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N=  9.31
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 13.43
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 13.82
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 14.35
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 13.74
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 14.50
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.49
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.78
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.77
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.54
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.47
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.44
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.25
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.40
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.40
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.47
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.37
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.37
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.54
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.42
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.54
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.57
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.42
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.42
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.61
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.51
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.41
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.56
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.55
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.48
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.66
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.50
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.46
```

Table 6.4a Results showing the learning characteristics of convolutional scheme as described by Stewart-Linde-Gray, for a data file of 512 speech samples. Each of the S/N values above is the result of one pass over the data.

```
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N=  9.56
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 14.48
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 14.70
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 13.76
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.03
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.55
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.26
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.36
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 14.92
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.90
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.94
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.40
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.11
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.17
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.84
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.40
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.22
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.98
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.41
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.22
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.98
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.41
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.22
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.98
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.41
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.22
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.98
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.41
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.22
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.98
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.41
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 16.22
EXPONENT=4  CONSTRAINT LENGTH= 3  BLOCK SZ= 64  S/N= 15.98
```

Table 6.4b Results showing the learning characteristics of convolutional scheme as described by Stewart-Linde-Gray, for a data file of 256 speech samples. Each of the S/N values above is the result of one pass over the data.

```
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 11.23
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 11.23
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 13.50
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 14.12
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 14.48
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 14.80
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.01
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.10
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.19
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.22
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.30
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.35
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.39
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.41
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.45
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.45
EXPONENT=4   CONSTRAINT LENGTH= 3   BLOCK SZ=256   S/N= 15.45
```

Table 6.4c Results showing the learning characteristics of convolutional scheme as described by Stewart-Linde-Gray, for a data file of 14400 speech samples. Each of the S/N values above is the result of one pass over the data.

b(n-1)	b(n)	c b(n) b(n-1)
1	1	-2.30
2	1	-1:31
3	1	-.62
4	1	.24
1	2	-1.33
2	2	-.57
3	2	-.07
4	2	.47
1	3	-2.49
2	3	-.09
3	3	.71
4	3	1.19
1	4	.21
2	4	.45
3	4	1.14
4	4	1.73

Table 6.5a Codebook used for image coding with exponent=4 and constraint length=2. The codebook was learnt using a database of 4 images, with 8 passes of the total data base. The data was coded a block at a time with a block size of 128. Each block was normalised with respect to mean and variance before coding. (image sizes were 128 by 128)

b(n-3),	b(n-2),	b(n-1),	b(n)	$c_{B(\cdot)}(n)$	b(n-3),	b(n-2),	b(n-1),	b(n)	$c_{B(\cdot)}(n)$
1	1	1	1	-2.52	2	1	1	1	-2.48
3	1	1	1	-2.39	4	1	1	1	-2.52
1	2	1	1	-2.02	2	2	1	1	-2.07
3	2	1	1	-2.21	4	2	1	1	-2.44
1	3	1	1	-2.08	2	3	1	1	-2.57
3	3	1	1	-2.17	4	3	1	1	-2.22
1	4	1	1	-2.44	2	4	1	1	-2.60
3	4	1	1	-2.11	4	4	1	1	-1.43
1	1	2	1	-2.04	2	1	2	1	-1.80
3	1	2	1	-1.47	4	1	2	1	-1.50
1	2	2	1	-1.13	2	2	2	1	-1.02
3	2	2	1	-1.35	4	2	2	1	-1.28
1	3	2	1	-1.03	2	3	2	1	-1.31
3	3	2	1	-1.62	4	3	2	1	-2.01
1	4	2	1	-1.68	2	4	2	1	-1.73
3	4	2	1	-1.33	4	4	2	1	-1.88
1	1	3	1	-1.12	2	1	3	1	-1.33
3	1	3	1	-.45	4	1	3	1	-.45
1	2	3	1	-.23	2	2	3	1	-.48
3	2	3	1	-.30	4	2	3	1	-.29
1	3	3	1	-1.29	2	3	3	1	.06
3	3	3	1	-.51	4	3	3	1	-.18
1	4	3	1	-1.12	2	4	3	1	-.44
3	4	3	1	-.07	4	4	3	1	.00
1	1	4	1	-.99	2	1	4	1	-.56
3	1	4	1	.12	4	1	4	1	.35
1	2	4	1	-.33	2	2	4	1	.05
3	2	4	1	.37	4	2	4	1	.80
1	3	4	1	.33	2	3	4	1	.68
3	3	4	1	.47	4	3	4	1	1.02
1	4	4	1	1.04	2	4	4	1	1.35
3	4	4	1	1.55	4	4	4	1	1.68
1	1	1	2	-2.43	2	1	4	2	-1.98
3	1	1	2	-2.93	4	1	4	2	-2.75
1	2	1	2	-1.79	1	2	4	2	-1.24
3	2	1	2	-2.11	4	2	4	2	-1.86
1	3	1	2	-1.33	2	3	4	2	-1.03
3	3	1	2	-3.03	4	3	4	2	-1.33
1	4	1	2	-.85	2	4	4	2	-.92
3	4	1	2	-.86	4	4	4	2	-.83
1	1	2	2	-1.20	2	1	2	2	-1.12
3	1	2	2	-1.12	4	1	2	2	-1.04
1	2	2	2	-.93	2	2	2	2	-.76
3	2	2	2	-.65	4	2	2	2	-.69
1	3	2	2	-.49	2	3	2	2	-.21
3	3	2	2	-.13	4	3	2	2	-.35
1	4	2	2	-.10	2	4	2	2	-.18
3	4	2	2	-.13	4	4	2	2	-.37
1	1	3	2	.18	2	1	3	2	-.38
3	1	3	2	-.35	4	1	3	2	-.18

1	2	3	2	-.18	2	2	3	2	-.17
3	2	3	2	-.02	4	2	3	2	.20
1	3	3	2	.14	2	3	3	2	.28
3	3	3	2	.59	4	3	3	2	.65
1	4	3	2	.39	2	4	3	2	.71
3	4	3	2	.90	4	4	3	2	.80
1	1	4	2	.26	2	1	4	2	.26
3	1	4	2	.32	4	1	4	2	.41
1	2	4	2	.48	2	2	4	2	.42
3	2	4	2	.50	4	2	4	2	1.24
1	3	4	2	1.01	2	3	4	2	.99
3	3	4	2	1.03	4	3	4	2	1.37
1	4	4	2	1.30	2	4	4	2	1.35
3	4	4	2	1.51	4	4	4	2	2.01
1	1	1	3	-2.52	2	1	1	3	-2.21
3	1	1	3	-2.26	4	1	1	3	-.54
1	2	1	3	-1.44	2	2	1	3	-1.33
3	2	1	3	-1.41	4	2	1	3	-.31
1	3	1	3	-.45	2	3	1	3	-.40
3	3	1	3	-.61	4	3	1	3	-.36
1	4	1	3	-.25	2	4	1	3	-.20
3	4	1	3	-.37	4	4	1	3	-.32
1	1	2	3	-.48	2	1	2	3	-.76
3	1	2	3	-.60	4	1	2	3	-.89
1	2	2	3	-.72	2	2	2	3	-.66
3	2	2	3	-.21	4	2	2	3	-.15
1	3	2	3	-.22	2	3	2	3	-.03
3	3	2	3	.09	4	3	2	3	.29
1	4	2	3	.34	2	4	2	3	.49
3	4	2	3	.44	4	4	2	3	.50
1	1	3	3	1.21	2	1	3	3	.34
3	1	3	3	.07	4	1	3	3	.23
1	2	3	3	.26	2	2	3	3	.13
3	2	3	3	.21	4	2	3	3	.54
1	3	3	3	.65	2	3	3	3	.61
3	3	3	3	.77	4	3	3	3	.91
1	4	3	3	.89	2	4	3	3	.95
3	4	3	3	1.25	4	4	3	3	1.37
1	1	4	3	.72	2	1	4	3	.59
3	1	4	3	.84	4	1	4	3	1.08
1	2	4	3	1.16	2	2	4	3	.86
3	2	4	3	.94	4	2	4	3	1.09
1	3	4	3	1.53	2	3	4	3	1.34
3	3	4	3	1.33	4	3	4	3	1.66
1	4	4	3	1.86	2	4	4	3	1.62
3	4	4	3	1.82	4	4	4	3	2.11
1	1	1	4	.53	2	1	1	4	.56
3	1	1	4	.59	4	1	1	4	.56
1	2	1	4	.65	2	2	1	4	.10
3	2	1	4	.57	4	2	1	4	.78
1	3	1	4	-.13	2	3	1	4	.82
3	3	1	4	.22	4	3	1	4	.87
1	4	1	4	.36	2	4	1	4	.33
3	4	1	4	.32	4	4	1	4	.71
1	1	2	4	.33	2	1	2	4	-.21

3	1	2	4	1.09	4	1	2	4	1.12
1	2	2	4	-.25	2	2	2	4	-.28
3	2	2	4	-.07	4	2	2	4	1.19
1	3	2	4	1.16	2	3	2	4	.22
3	3	2	4	.54	4	3	2	4	1.31
1	4	2	4	1.08	2	4	2	4	1.23
3	4	2	4	1.16	4	4	2	4	1.36
1	1	3	4	1.51	2	1	3	4	1.52
3	1	3	4	.26	4	1	3	4	.49
1	2	3	4	.60	2	2	3	4	.61
3	2	3	4	.70	4	2	3	4	.75
1	3	3	4	.75	2	3	3	4	.74
3	3	3	4	.91	4	3	3	4	.98
1	4	3	4	1.45	2	4	3	4	1.57
3	4	3	4	1.63	4	4	3	4	1.77
1	1	4	4	2.03	2	1	4	4	2.06
3	1	4	4	1.60	4	1	4	4	1.56
1	2	4	4	1.48	2	2	4	4	1.25
3	2	4	4	1.40	4	2	4	4	1.52
1	3	4	4	1.60	2	3	4	4	1.78
3	3	4	4	1.80	4	3	4	4	1.80
1	4	4	4	2.27	2	4	4	4	2.31
3	4	4	4	2.28	4	4	4	4	2.37

Table 6.5b Codebook used for image coding with exponent=4 and constraint length=4. The codebook was learnt using a database of 4 images, with 8 passes of the total data base. The data was coded a block at a time with a block size of 128. Each block was normalised with respect to mean and variance before coding. (image sizes were 128 by 128)

general, for efficient compression, the constraint length is large, resulting in a large codebook. To transmit a new codebook directly for each block period, would be prohibitively expensive in channel symbols.

What is required therefore, is an inexpensive method of transmitting an approximate codebook. It is proposed that this may be done by sending the coordinates of previously coded blocks of data. The members of a block of previously coded data would be employed to construct a codebook, which may be used to code subsequent blocks of data.

The coding procedure is as follows: Start with a codebook, designed in an ad-hoc manner. (Probably with some prior knowledge of the source to be coded.) This is used to code the first block of data. The coded data is used to construct a second codebook. For the second block to be coded, the better of these two codebooks is employed. This second block of coded data is then used to design a third codebook. In coding the third block of data, the better of the three codebooks available is employed. This process is continued; at each block period, the coordinate associated with a codebook is transmitted, along with the mean and standard-deviation of the block in question. A limit on the number of codebooks which may be maintained at any one time, is set, thus defining the number of bits transmitted per block period to indicate which codebook is to be used.

Straightforward though this scheme might be, it has two attendent problems. These are:

1) The Lloyd method for the construction of a codebook is inapplicable as it stands. This is because the previous block, from

which a codebook is to be derived had already been coded. (In a sense the members of this block have been quantised.) The method results in a sequence of codebooks which converge to the codebook used to code that previous block of data.

2) The problem of deciding which of the available codebooks to use. The first problem will be discussed first.

The codebook design task is tackled in the following manner. Instead of using the actual sequence of previously coded data to define the codebook, this is used to derive the parameters of a model for the source. These parameters are then employed to define the codebook centroids. Provided the model parameters are fairly insensitive to noise, the effect is to design a codebook which is similar to that which would have been obtained if there were no coding distortion. The following is a mathematical description of the mechanism of a codebook based convolutional coder. This description is required in order to pose the questions whose answering are required in the design of an approximate codebook, using previously coded data.

Suppose the coding exponent is L and the constraint length is k. Then there are L^k sequences of symbols, which define the set of L reproduction symbols which may be used to approximate the n-th source outcome $x(n)$. Let $b(n-k+1),\dots,b(n-1)$ be the sequence of channel symbols which have just been transmitted. Refer to this as $B(n-k+1,,n-1)$. For convenience we say $B(.,,.)$ may take some $(k-1)$-tuple value i, say. Let $p(x(n)|B(n-k+1,,n-1)=i)$ be the probability density function for the source symbol $x(n)$, given that the sequence of $k-1$ previous channel symbols $B(n-k+1,,n-1)$ is i. Then $x(n)$ may take one of values $[c_i(1),c_i(2),\dots,c_i(L)]$. Due to the fact that coding is done by a multipath search, it may not be assumed that non-overlapping regions of the x(n) space map into the $c_i(.)$ values. This would have happened if simply the closest $c_i(.)$ were chosen to approximate x(n), given that B(n-k+1,,n-1)=i. (This would happen with a single path search) Suppose we define the function $w_{im}(z)$ such that $w_{im}(z)p(z|B(.,,.)=i)$ is the probability that the n-th source symbol x(n)=z, given that the n-th reproduction symbol, x(n) is $c_i(m)$ and B(n-k+1,,n-1)=i. Properties worth noting are;

1) $$\sum_m w_{im}(z) = 1 \qquad \forall z, \forall i \qquad 6.31$$

2) $$D = \sum_{\forall i} p(\underline{B}(n-k+1,,n-1) = i) \sum_m \int_{\pm\infty} (x(n) - c_i(m))^2 w_{im}(x(n)) p(x(n)|\underline{B}(n-k+1,,n-1) = i) \; \mathrm{d}x(n) \qquad 6.32$$

where D is the total mean square coding error.

3) In single path coding, $w_{im}(x(n))=1 \qquad x(n) \in S_i(m), \quad x(n)=c_i(m)$

$=0$ elsewhere

The $S_i(m)$ are non-overlapping regions and $\bigcup_m S_i(m)=(-\infty,\infty)$

4) $$D_{sp} = \sum_{\forall i} p(\underline{B}(n-k+1,,n-1) = i) \sum_m \int_{S_i(m)} (x(n) - c_i(m))^2 p(x(n)|\underline{B}(n-k+1,,n-1) = i) \; \mathrm{d}x(n) \qquad 6.33$$

D_{sp} is the total mean square coding error with single path coding.

A good codebook is one which has a set of values $\{c_i(.)\}$ such that with the best choice of $w_{im}(x(n))$, (defined by the path search) D is minimised. At the receipt of a set of channel symbols associated with the coding of a block of data, the first stage towards improving the codebook, is to define a new and better set of centroids $\{c_i(m)\}$. The new set of centroids are

$$c_i(m) = \int_{\pm\infty} x(n) w_{im}(x(n)) \;\; p(x(n)|\underline{B}(n-k+1,,n-1) = i) \mathrm{d}x(n) \qquad 6.34$$

The problems tackled are the estimation of the probability functions p(x(n)|B(n-k+1,,n-1)=i) and w_{im}(x(n)). p(x(n)|B(n-k+1,,n-1)=i) is modelled as being of a Gaussian distribution and parameterised by its mean and variance. The following are a list of assumptions used in the estimation of the c_i(m) from the observed, coded data.

Assumption 1 Let {c_i^p(m)} be a sequence of codebook entries which have been used to code a block of data. Let $c^q{}_i$(m) be new codebook entries, to be computed with information obtained from the previously coded block. If we abbreviate B(n-k+1,,n-1)=i to B=i,

$$E\,(x(n)|\tilde{x}(n) \in \bigcup_{\mathrm{m}} c_i^p(m)) \approx E\,(\tilde{x}(n)|\tilde{x}(n) \in \bigcup_{\mathrm{m}} c_i^p(m)) = \hat{\mu}_{\underline{B}=i} \quad \text{say} \qquad 6.35$$

and

$$E\,(x(n)^2|\tilde{x}(n) \in \bigcup_{\mathrm{m}} c_i^p(m)) \approx E\,(\tilde{x}(n)^2|\,\tilde{x}(n) \in \bigcup_{\mathrm{m}} c_i^p(m)) = \hat{\sigma}^2_{\underline{B}=i} \quad \text{say} \qquad 6.36$$

Where the left-hand-sides are the mean and variance of the distribution p(x(n)|B=i). Given the assumption of a certain distribution class, in this case Gaussian, for p(x(n)|B(.)=i), we may estimate $c^q{}_i$(m) values in the following way.

$$c_i^q(m) = \frac{\int_{\pm\infty} x(n) w_{im}(x(n)). \frac{1}{\hat{\sigma}\sqrt{2\pi}} exp\{-\frac{1}{2}\frac{(x(n)-\hat{\mu})^2}{\hat{\sigma}^2}\}\,\mathrm{d}x(n)}{\int_{\pm\infty} w_{im}(x) p(x|\underline{B} = i)\mathrm{d}x} \qquad 6.37$$

Assumption 2 The next task would be to estimate the function w_{im}(x(n)), we assume that a single path search had been used to code the blocks received. Then

$$c_i^q(m) \approx \frac{\int_{S_i(m)} x(n) \frac{1}{\hat{\sigma}\sqrt{2\pi}} exp\{-\frac{1}{2}\frac{(x(n)-\hat{\mu})^2}{\hat{\sigma}^2}\}\,\mathrm{d}x(n)}{\int_{S_i(m)} p(x|\underline{B} = i)\mathrm{d}x} \qquad 6.38$$

The regions S_i(m) are estimated by insuring that the following holds.

$$\int_{S_i(m)} p(x|\underline{B} = i)\mathrm{d}x = \int_{\pm\infty} w_{im}(x) p(x|\underline{B} = i)\mathrm{d}x \qquad 6.39$$

6.6.1 Implementation details, results and discussion

Preliminary results have been obtained for the scheme described above, which shows that the use of an adaptive codebook is worthwhile. The following is a desciption of some implementation details.

To estimate a new set of centroids $c^q_i(m)$, the following integrals are required over various intervals

$$\int_{S_i(m)} xp(x|\underline{B} = i)\mathrm{d}x \qquad\qquad \int_{S_i(m)} p(x|\underline{B} = i)\mathrm{d}x$$

It would be prohibitively expensive computationaly to evaluate these during the coding process. Thus the quantities below are computed beforehand for several values of a(j)

$$\int_{-\infty}^{a(j)} xp(x|\underline{B} = i)\mathrm{d}x \text{ and } \int_{-\infty}^{a(j)} p(x|\underline{B} = i)\mathrm{d}x$$

a(j) at intermediate values are determined by straight line interpolation.

A default or initial codebook is derived in the following manner. The initial codebook is based upon the Lloyd-Max quantiser controid values. Suppose the centroid values for a code scheme with exponent L, are $y_1,\dots,y_L$. Then when b(n)=j, the codebook B(n-k+1,...,n-1)=i and b(n)=j, are chosen to be purtabations on the value y_j for all i. This is so for all values of j.

The purturbations are chosen in the following way: The reproduction symbol for the channel sequence b(n-k+1),...,b(n) is

$$y_{b(n)} + 2\frac{\sum_{m=1}^{k-1}(b(n-m)-1)L^{k-m+1} - \frac{L^{k-1}}{2} + \frac{1}{2}}{L^{k-1}}$$

where each b(n) may take a value in the set $\{1,\ldots,L\}$. These are equally spaced values centred at $y_{b(n)}$.

Table 6.6 shows the S/N results obtained using the initial codebook described above to code several blocks of speech and the S/N values obtained using the improved codebook computed with the previously coded block. In addition the S/N value obtained after coding the same block with an improved codebook derived using the method of Stewart-Linde-Gray, with one pass, are given.

The speech file SR8KK is coded with various values of codebook size, constraint length, exponent and adaptation block size. The results are shown in table 6.7, for the case when the default codebook is used initially and is the last member of the library of codebooks. These results are better than those obtained when the default codebook is used throughout. Table 6.8 shows the results obtained using an initial codebook which has been evaluated using the standard Stewart-Linde-Gray method on the speech file in question. In this case, since the default codebook is quite good any way, only a slight improvement is achieved by the use of an adaptive codebook.

6.6 Conclusion

In this chapter we present results for speech and image coding using multipath search techniques. The contribution of this chapter has been the demonstration that adaptive multipath search coding may be implemented with adaptation information is transmitted via already coded data.

In the class of coding schemes based upon linear prediction,

S/N-(1)	S/N-(2)	S/N-(3)
12.1507	13.7391	16.6202
10.8299	11.7936	14.5944
11.2953	13.0324	16.0910
10.9024	12.3282	16.4186
11.5785	12.9307	14.4925
11.1117	13.9562	14.6887
11.1104	12.1001	15.4123
11.5235	13.4975	16.2536
11.3584	13.5620	15.0780
11.0058	13.6331	14.6951
11.7421	13.0570	15.4616
11.3826	12.8817	15.2579
11.6164	13.6870	15.3361
11.8695	13.2776	15.0688
11.0217	12.8615	13.7541
11.3755	14.2512	15.2309
10.8699	12.2685	14.4546
10.9358	13.0643	13.6038
11.0238	11.9094	14.6060
6.1729	6.6601	9.1275
11.2452	12.8387	15.7679
10.7783	13.5197	14.2311
11.6797	13.6647	14.9442
11.8418	14.3443	15.9491
11.4500	13.5660	14.7815

Table 6.6 Results of speech coding tests for adaptive trellis coding. The coding scheme is a convolutional coder.
Exponent = 4
Constraint length = 3
Coding delay = 128
Adaptation period = 128

Initial codebook is that described in section 6.6.1
1-st S/N value is actual S/N obtained using the 'initial' (default) codebook.
2-nd S/N value is that obtained using the improved codebook derived employing the already coded data (distorted data)
3-rd S/N value is that obtained if improved codebook is obtained from already coded data (undistorted data).

Exponent	Constraint length	Block size	Lib. size	S/N
4	3	128	32	14.10
4	4	128	32	13.89
4	3	256	32	14.16
4	4	256	32	13.67
4	3	256	16	14.06
4	4	256	16	13.63
2	4	256	32	8.90
2	8	256	32	7.85

Table 6.8 S/N ratio values obtained using adaptive trellis coding with initial codebook which is not learnt using speech data.

4	3	256	32	15.61
2	4	256	32	10.32

Table 6.9 S/N ratio values obtained using adaptive trellis coding with initial codebook which is learnt using speech data.

adaptation using previously coded data is compared with that described by Wilson and Hussain (1977). The latter proving slightly superior, when more than one path is employed.

For speech coding it was observed that "instantaneous" step size adaptation did not yield any improvements in signal to noise ratio. For image coding however, "instantaneous" step size adaptation proved to be of vital importance, a simple adaptation scheme, an alternative to that of Jayant, Cumminsky et al., was introduced and proved to be more suitable.

Multipath search coding using a codebook, and not based upon linear prediction, was described and results presented for coding image and speech data using a non-adaptive version of this. An adaptive version of this is introduced and shown to work well, with adaptation providing a significant signal to noise ratio gains over the use of the data independent initial codebook.

CHAPTER 7 CONCLUSIONS AND SUGGESTIONS FOR FURTHER RESEARCH

The concluding chapter is written in two broad sections. The first section is a summary of the contents of this thesis. It is however different from chapter one, in that here, a more detailed discussion of the results is presented.

In chapter three an approach to data compression was presented. As had been mentioned several times before, what was required was an adaptive coding strategy. The approach described in this thesis makes use of already coded data in its adaptation algorithm. This approach, although not entirely new, has been the focus of little research and in fact its application had been rather limited. In chapter three, several variations on the basic scheme are presented. In addition the results of the application of the general idea to several types of source are presented. It was shown that the approach yielded results which were, although inferior to some well known compression schemes particularly suited to some sources, worked well for a very wide range of data source, requiring very little prior knowledge of the source to be coded. It is also versatile in the fidelity measure which may be incorporated into the coding scheme.

Results were presented for the cases where some source properties were used to aid coding. For the case where coding of the speech waveform was undertaken, the approach suggested here yielded favourable signal to noise ratio values, compared with Adaptive DCT coding. It was noted though that due to fact that this is a waveform coding scheme, it gave subjectively inferior results to other more speech particular coding schemes. Thus, the idea of using previously coded data was used to improve the performance of other well known speech coding schemes. These were the

'Voiced-unvoiced' excited LPC vocoder and the 'residual' excited LPC vocoder. It was shown that a significant reduction in the coding rate for a small reduction in subjective quality may be obtained by transmitting LPC information at a variable rate. It was also shown that an improvement may be made, unfortunately not very large, by the exchange of excitation information bandwidth for LPC coefficient parameter transmission bandwidth, when appropriate.

Chapter four tackled the problem of providing some theoretical 'bone' to the approach suggested in chapter three. Coding with the basic 'MPPCD' scheme is considered, which allows the rate which may be achieved to be written in terms of the probability of observing a sequence of symbols. In the case of coding with zero error, this may be precisely done, with no approximations. The idea of an elementary block size was introduced. This involved coding with super letters, consisting of say n of the original source symbols; n is the elementary block size. As n, the elementary block size, is allowed to tend to infinity, by the Shannon-McMillan-Brieman AEP theorem, the probability of observing a sequence of data, may be written in terms of the Shannon entropy for the source. Using the above information the theoretical capabilities of the basic MPPCD scheme for zero error coding were defined.

An interesting corollary obtained in this chapter, concerns the probability of observing very long sequences of symbols, from an ergodic source, within a certain number of outcomes of a source. It turns out that within a given number of outcomes of a source, L say, there is a critical length, k say, where for any m length sequence, there is almost zero probability of observing this in L outcomes if m>k, and almost unity probability of observing this in L outcomes if m<k.

A similar argument as used for the zero distortion case was used to obtain approximate results on the theoretical performance of the basic `MPPCD' scheme for coding with a fidelity criterion. With the assumption of ergodicity, a theorem for the probability of observing a long sequence of data, with finite precision is presented. The use of theis in a manner similar to the use of the Shannon-McMillan-Brieman AEP theorem and some non-theoretical discussion, allows the derivation of a similar result for coding with a fidelity criterion, as for coding with zero error.

Chapter four is an interesting chapter, for the extreme of thoery or lack of practical bearing in the results presented. It was felt nevertheless, important in that it gives a deeper understanding of the mechanism of the MPPCD scheme. For the author, working for this chapter was particularly enjoyable since it allowed a considerable broadening of the scope of the thesis and allowed a deeper understanding of the character of ergodic sources.

Chapter six dealt with adaptive `Multipath Search Coding' as against block coding of data. A system for adaptive tree coding, based upon linear prediction was proposed. This relied upon coding the prediction parameters by approximating these by the parameters for previously coded data. Several results were presented for the performance of this scheme and it was shown that on some occasions, that is when a single path was used, this scheme performed better than the normal adaptive tree coding scheme.

Results were presented for speech and image coding by a multipath search scheme, where linear prediction is not employed. A codebook is used instead to colour the structure, in this case a trellis, which is searched to provide reproduction symbols coding a source. This rather simple scheme was shown to work well for both speech and image data. An adaptive version of this scheme was proposed and results presented which showed that this gave a better performance compared with using the default codebook. A drawback of the adaptive scheme however was that the attractiveness, the simplicity, of the original scheme was somewhat lost.

Suggestions for further research

Ch3 1) A cause of inefficiency in the basic MPPCD scheme is the fact that a significant number of bits are wasted in coding the block sizes being employed. A method of reducing the transmission rate for this, is suggested below. It is hoped that further research along these lines might prove fruitful.

Recall that in the examples of section 3.3.1, where sources of alphabet size 4 were coded, block sizes 1,2,3&4 were considered. 2 bits were employed to code the block size and at each 'transmission block period' 6 bits were transmitted. (elementary block size=1) In striving to code the block sizes more efficiently, we shall consider coding in sequences of 'transmission block periods'. For example consider the block of 64 'transmission block periods'. Separate the information indicating the block sizes, from that indicating the coordinate in the previously coded data where a similar block may be found. The former forms another sequence of alphabet size 4, which may be coded separately. If the source is in a locally stationary mode, it is expected that the sequence of block size information will be highly redundant and thus lend itself favourably to further compression. By further application of the above scheme more and more redundancy might be removed, with the disadvantage of very large coding delay. Decoder complexity is also very slightly increased. It ought to be stressed that the effects of channel errors are likely to be more serious.

Ch3 2) In chapter 3, most of the proposed schemes required adaptive libraries. Especially important in section 3.6.2 where the

MPPCD scheme is employed to improve the LPC vocoder performance, there is the problem of which library member to remove when a new addition is made to the library. In all the schemes implemented, the earliest library member was removed. It is suggested that an investigation into how the library may best loose an old member be conducted. An alternative to consider is the following. A tally is kept at both the transmitter and receiver of how many times each member of the library had been employed to code another block of data. The library member least used in this way is then removed upon the addition of a new member. A tie breaking rule, for example the age of a library member, may be employed in the case of equality in the number of times a member has been used.

Ch4 1) A very useful result to obtain with further research, would be bounds on the coding rate for the 'MPPCD' scheme, obtained without resorting to the Asymptotic-Equipartition theorem and without having to suppose block sizes to be infinite. This would give an indication of how poor or good, the scheme is when practical sized blocks are employed. This is still under the assumption that the source is ergodic.
An approach might be to evaluate the quantities

$\textit{Expectation over all } Y^{L_N} \textit{ sequences } [\{1-(1-p(Y^{L_{N-1}}))^{R}\} - \{1-(1-p(Y^{L_N}))^{R}\}]$ for each N

These being the weighting applied to the various elementary source symbol block lengths, in order to find the average length. The calculation of the above, for various values of N, would indicate how fast weighting converged to a delta function at some N, as the value 'k', the number of original source symbols which make up the elementary source symbol, is increased.

Ch4 2) Finding a more satisfactory proof for the theorem of section 4.5.1, without ergodic assumptions for the joint distribution between the source and the reproduction symbols, would be a further contribution to this research.

Ch6 1) It is suggested that further research be undertaken to improve the methods used to obtain prediction filter coefficients for linear prediction based multipath search coding.
In section 6.4.1, attempts were made to estimate the coding error in linear prediction based multipath search coding, for any set of prediction filter coefficients $B = \{b_1, \ldots, b_p\}$. The assumption was made that the ratio of prediction error to the coding error, in variance, is the following. Since Max-Lloyd quantiser values were used to generate the quantised versions of the prediction error signal (to add to the predicted signal value to make the estimate), the ratio of the prediction to the coding error was taken to be the Max-Lloyd quantiser, quantisation error value. This is of course dependent upon the model for the signal being quantised. In the cases considered, the prediction error signals were modelled as being Gaussian or Laplacian in distribution. The use of the Max-Lloyd quantiser quantisation error value is perfectly valid for the case where a single path search is undertaken. This is because, the quantised prediction error values are chosen as one would choose the centroids in scalar coding, using centroids determined by the Max-Lloyd algorithms. For a multipath search however, the closest quantised prediction error value is not neccessarily chosen to approximate the quantisation error. The relation between, the prediction error and coding error variances are thus not those obtained assuming simple quantisation. It is therefore suggested that further research be untertaken to find the relationships between; the coding rate, the number of paths, the prediction filter

order and the ratio of prediction error variance to coding error variance for multpath search coding.

Ch6 2) As discussed in section 6.3, the optimal prediction parameters for linear prediction based multi-path search coding, are dependent upon the coding error. An adaptive filter is suggested for the purpose of ascertaining the coefficients of the optimum prediction filter, as an alternative to directly estimating the coding error. The following is a description of the possible operation of the suggested system.

A block of data to be coded, is fed continually to an adaptive filter system as shown in figure 7.1. The coding process is simulated precisely in this system so as to generate the exact coding error sequence. This sequence is then fed back to alter the prediction filter parameters.

Ch6 3) Choosing the best of a set of codebooks for multipath search coding: In this section a measure of deviation between the best codebook for a source and a test codebook, is proposed for Gaussian sources. Given a certain codebook for trellis coding, the parameters of the Gaussian source for which this codebook is best suited may be estimated by feeding the decoder for this codebook with a long sequence of independently distributed channel symbols. This gives the typical sequence of reproduction symbols, $\{\tilde{x}(n)\}$ say, associated with the source which would have been well coded with this codebook. The statistics for such a source are estimated from the sequence $\{\tilde{x}(n)\}$. Suppose the i-th codebook in the library of codebooks has a typical reproduction sequence with statistics set $\{S_i\}$, derived from its $\{\tilde{x}(n)\}$ sequence. Then if the block to be coded has a statistics set $\{S_{true}\}$, then the best codebook of the

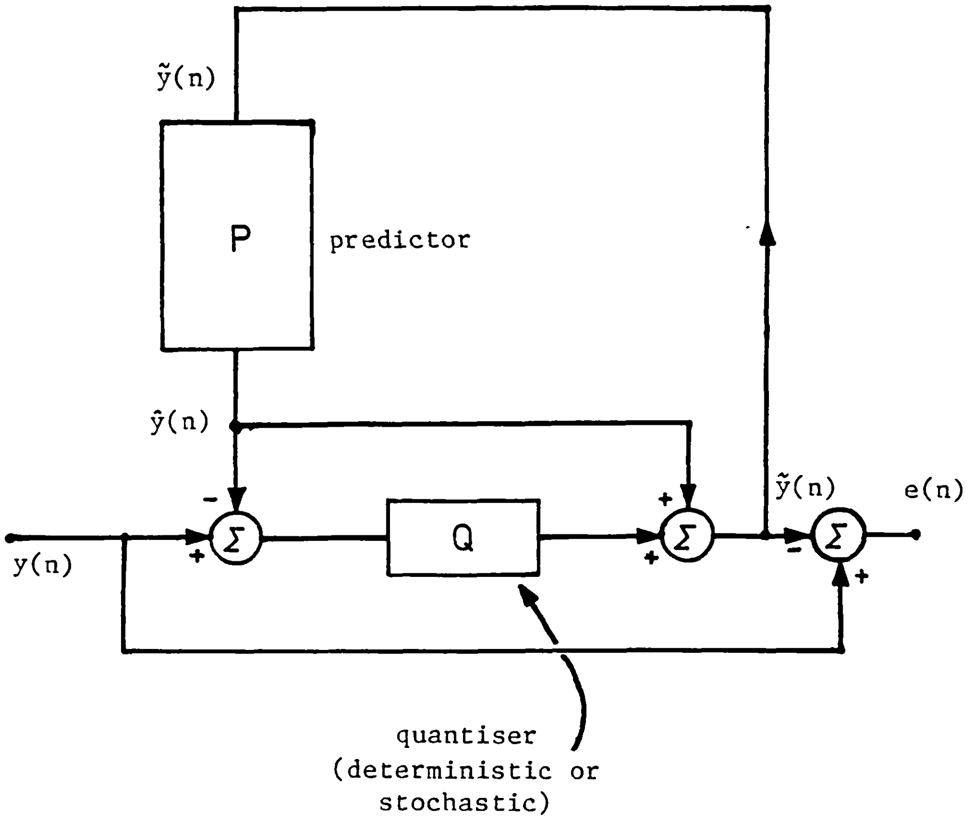


Figure 7.1 Block diagram for linear prediction based multipath coder, where to establish the prediction parameters, adaptation of these is undertaken, using several passes over a given block of data. Feedback for adaptation is taken from the error signal e(n).

library is that whose statistics $\{S_i\}$ differ least from $\{S_{true}\}$. For a Gaussian source, the statistics in question are the auto-correlation function values. A rate-distortion theory based measure of deviation is employed. We shall first evaluate for a given rate the distortion bound obtained as a result of the use of inappropriate statistics, for a one dimensional Gaussian random variable. The spectrum of the Gaussian source with non-independent letters is used to evaluate the weighted average distortion in the same manner as the rate-distortion function is evaluated for the Gaussian non-independent source. From appendix 6, it is shown that the resulting rate and distortion values resulting from modelling a Gaussian source of parameters (μ,σ^2) by an inappropriate model of parameters $(\hat\mu\ ,\hat\sigma^2)$ are the following

$$D(d^*) \le d^* - \hat\sigma^2\left(\frac{d^*}{\hat\sigma^2}\right)^2(1-\frac{\sigma^2}{\hat\sigma^2}) + \left(\frac{d^*}{\hat\sigma^2}\right)^2(\mu-\hat\mu)^2$$

$$\text{and}\quad R(d^*) \le -\frac{1}{2}\left((1-\frac{d^*}{\hat\sigma^2})\{(1-\frac{\sigma^2}{\hat\sigma^2})-(\mu-\hat\mu)^2\}+\ln\frac{d^*}{\hat\sigma^2}\right)$$

It may be observed that when $\sigma^2=\hat\sigma^2$ and $\mu=\hat\mu$, the above simplify to the well known Gaussian rate-distortion function.

$$D = d^*$$
$$R = -\frac{1}{2}\ln\frac{d^*}{\sigma^2}$$

Note that $\hat\sigma^2 \geq \mathbf{d}^*$ and $\sigma^2 \geq \mathbf{d}^*$. For the case where $\hat\mu=\mu=0$, this being of particular interest to us,

$$D(d^*) \le d^* - \hat\sigma^2\left(\frac{d^*}{\hat\sigma^2}\right)^2(1-\frac{\sigma^2}{\hat\sigma^2})$$

$$\text{and}\quad R(d^*) \le -\frac{1}{2}\{(1-\frac{d^*}{\hat\sigma^2})(1-\frac{\sigma^2}{\hat\sigma^2}) + \ln\frac{d^*}{\hat\sigma^2}\}$$

Suppose the source has non-independent outcomes so that the rate distortion bound may only be approached by coding separately, the sequence associated with each spectral component resulting from the

KLT of large blocks of the data. (see Gallagher-(1968), p482) Then the coding scheme, assuming statistics ($\sigma^2(\omega)$), the variance for each frequency or eigenvalue (associated with each eigenfunction), will allocate $\hat{R}(\omega)$ bits for each harmonic ω, such that

$$\int_{\pm\frac{f_s}{2}} \hat{R}(\omega)\mathrm{d}\omega = R_{\text{total}}$$

Suppose we have a finite number of frequencies, so that

$$R_{\text{total}} = \sum_{\forall \omega_i} \hat{R}'(\omega_i)$$

and $\hat{R}'(\omega_i)$ has been computed, given the power spectral density $\sigma^2(\omega_i)$, determined from the library codebook of interest. The parameter value d^* which results in the appropriate rate $\hat{R}'(\omega_i)$ is evaluated. This is substituted into equation 6.4, giving the distortion $D(\omega_i,d^*)$. The summation of this quantity over all the frequencies gives an estimation of the total coding distortion and is a measure of the deviation between two Gaussian sources.

To evaluate d^* such that $R(d^*)$ as defined in eq. 6.5 is $\hat{R}'(\omega)$, the following iteration is used. A point to note beforehand, is that the function $R(d^*)$ is monotonically decreasing with d^*/σ^2, for all $d^*/\sigma^2 \leq 1$, moreover it is convex. A Newton algorithm is consequently guaranteed to solve the equation

$$R'(\omega) + \frac{1}{2}\{(1-x)c + \ln x\} = 0$$

$$\text{where} \quad x = \frac{d^*}{\hat{\sigma}^2} \quad \text{and} \quad c = 1 - \frac{\sigma^2}{\hat{\sigma}^2}$$

The following is the iteration algorithm,

$$x_{n+1} = \frac{x_n[1 - c - 2R - \ln x_n]}{1 - cx_n}$$

An initial value for x should be as small as possible. (To reduce the computational requirement, the log function may be replaced by a piecewise approximation. For x taking values from 1/16384 to 1, in steps of 1/16384. This requires a 2^{14} sized lookup table.)

APPENDICES

APPENDIX 1

A1.1 The Levinson-Durbin and the Burg Maximum entropy methods for system parameter estimation

The system identification problem is that of finding the parameters of a given model such that the parameters best suit a certain stochastic source. An often used model is the AUTO-REGRESSIVE (AR) model. An AR source is characterised as follows. Suppose a source generates outcomes x(n) at some instant n then

$$x(n) = \sum_{i=1}^{P} a_i x(n-i) + e(n) \qquad \text{APE 1.1}$$

The model parameters $\{a_i\}$ are evaluated such that the variance of the error sequence {e(n)} is minimised.

$$\text{Let} \quad E(e(n)^2) = E(\{x(n) - \sum_{i=1}^{P} a_i x(n-i)\}^2)$$

$$= E(x(n)^2) - 2\sum_{i=1}^{P} a_i E(x(n)x(n-i)) + \sum_{i=1}^{P}\sum_{j=1}^{P} a_i a_j E(x(n-i)x(n-j)) \qquad \text{APE 1.2}$$

The coefficients $\{a_i\}$ are evaluated by differentiating the quantities of APE1.2 with respect to each a_k and setting the result to zero.

$$\frac{\partial E(e(n)^2)}{\partial a_k} = -2E(x(n)x(n-k)) + 2\sum_{i=1}^{P} a_i E(x(n-i)x(n-k)) \qquad \text{APE 1.3}$$

Refering to E((n-i)(n-k)) as r(|i-k|) one obtains

$$\begin{pmatrix} r(0) & \dots & r(p-1) \\ \cdot & & \\ \cdot & & \\ \cdot & & \\ r(p-1) & \dots & r(0) \end{pmatrix} \begin{pmatrix} a_1 \\ \cdot \\ \cdot \\ \cdot \\ a_p \end{pmatrix} = \begin{pmatrix} r(1) \\ \cdot \\ \cdot \\ \cdot \\ r(p) \end{pmatrix} \qquad \text{APE 1.4}$$

The solution of the above gives the $\{a_k\}$ values. Solution of this normally takes $O(P^3)$operations. The Levinson-Durbin method employs $O(P^2)$ operations and is as follows. Let the set of numbers $\{a_{k1}, \ldots, a_{kk}\}$ solve the equation APE1.4 for the k-th order case. Then

$$\begin{bmatrix} r(0) & \ldots & r(k-1) \\ \cdot & & \\ \cdot & & \\ \cdot & & \\ r(k-1) & \ldots & r(0) \end{bmatrix} \begin{bmatrix} a_{k,1} \\ \cdot \\ \cdot \\ \cdot \\ a_{kk} \end{bmatrix} = \begin{bmatrix} r(1) \\ \cdot \\ \cdot \\ \cdot \\ r(k) \end{bmatrix} \qquad \text{APE 1.5}$$

Thus neglecting the bottom row,

$$\begin{bmatrix} r(0) & \ldots & r(k-2) \\ \cdot & & \\ \cdot & & \\ r(k-2) & \ldots & r(0) \end{bmatrix} \begin{bmatrix} a_{k,1} \\ \cdot \\ \cdot \\ a_{k,k-1} \end{bmatrix} = \begin{bmatrix} r(1) \\ \cdot \\ \cdot \\ r(k-1) \end{bmatrix} - a_{kk} \begin{bmatrix} r(k-1) \\ \cdot \\ \cdot \\ r(1) \end{bmatrix} \qquad \text{APE 1.6}$$

Thus by multiplying both sides by the inverse of the above matrix,

$$\begin{bmatrix} a_{k,1} \\ \cdot \\ \cdot \\ \cdot \\ a_{k,k-1} \end{bmatrix} = \begin{bmatrix} a_{k-1,1} \\ \cdot \\ \cdot \\ \cdot \\ a_{k-1,k-1} \end{bmatrix} - a_{kk} \begin{bmatrix} a_{k-1,k-1} \\ \cdot \\ \cdot \\ \cdot \\ a_{k-1,1} \end{bmatrix} \qquad \text{APE 1.7}$$

a_{kk} is called the k-th reflection coefficient and may refered to here as R_k.

These reflection coefficients turn out to be important parameters for signal processing. APE1.4 may be expanded as follows.

$$\begin{bmatrix} r(0) & \ldots & r(k) \\ \cdot & & \\ \cdot & & \\ \cdot & & \\ r(k) & \ldots & r(0) \end{bmatrix} \begin{bmatrix} -1 \\ a_{k,1} \\ \cdot \\ \cdot \\ a_{kk} \end{bmatrix} = \begin{bmatrix} -E_k \\ \cdot \\ 0 \\ \cdot \\ \cdot \end{bmatrix} \qquad \text{APE 1.8}$$

A top row way be created such that APE1.8 is obtained, where E_k is the variance of the prediction error signal e(n) for a k-th order model. Now from APE1.7 it may ascertained that

$$\begin{bmatrix} -1 \\ a_{k,1} \\ \cdot \\ \cdot \\ \cdot \\ a_{k,k-1} \\ a_{kk} \end{bmatrix} = \begin{bmatrix} -1 \\ a_{k-1,1} \\ \cdot \\ \cdot \\ \cdot \\ a_{k-1,k-1} \\ 0 \end{bmatrix} - a_{kk} \begin{bmatrix} 0 \\ a_{k-1,k-1} \\ \cdot \\ \cdot \\ \cdot \\ a_{k-1,1} \\ -1 \end{bmatrix} \qquad \text{APE 1.9}$$

Also from APE1.8 it may be observed that by the multiplication of each side of APE1.9 by the k-th dimensional autocorrelation matrix, the following is obtained.

$$\begin{bmatrix} -E_k \\ . \\ 0 \\ . \\ . \end{bmatrix} = \begin{bmatrix} -E_{k-1} \\ . \\ 0 \\ . \\ -B_{k-1} \end{bmatrix} - a_{kk} \begin{bmatrix} -B_{k-1} \\ . \\ 0 \\ . \\ -E_{k-1} \end{bmatrix} \qquad \text{APE 1.10}$$

where

$$B_k = r(k+1) - \sum_{i=1}^{k} a_{ki} r(k+1-i) \qquad \text{APE 1.11}$$

Thus

$$a_{kk} = R_k = \frac{B_{k-1}}{E_{k-1}} \qquad \text{APE 1.12}$$

$$\text{and} \qquad E_k = (1 - a_{kk}^2) E_{k-1} \qquad \text{APE 1.13}$$

This concludes the Levinson-Durbin algorithm.

A1.2 The Burg Maximum Entropy method

This employs the relationship of equation APE1.7 for evaluating the k-th order prediction coefficients from those of the (k-1)-th order coeficients. Unlike the Levinson-Durbin method however, no prior knowledge of the auto-correlation function is required. Processing is undertaken directly upon the outcomes of the source to parameterised.

Suppose
$$f_k(n) = x(n) - \sum_{i=1}^{k} a_{ki} x(n-i) \qquad \text{APE 1.14}$$

$$b_k(n) = x(n-k) - \sum_{i=1}^{k} a_{ki} x(n-k+i) \qquad \text{APE 1.15}$$

where $f_k(n)$ and $b_k(n)$ are respectively the forward and backward prediction error for the k-th order model at instant n. Now using APE1.7, it may be shown that

$$\begin{aligned} f_k(n) &= f_{k-1}(n) - a_{kk} b_{k-1}(n-1) \\ b_k(n) &= b_{k-1}(n-1) - a_{kk} f_{k-1}(n) \end{aligned} \qquad \text{APE 1.16}$$

In the Burg method, a_{kk} is evaluated so as to minimise $\sum_{n=-\infty}^{\infty}(f_k^2(n) + b_k^2(n))$. The data outside the block of concern is assumed to be zero.

Now
$$\frac{\partial}{\partial a_{kk}} \sum_{-\infty}^{\infty} (f_k(n)^2 + b_k(n)^2) = -\sum_{\pm\infty} \Big(\{f_{k-1}(n) - a_{kk} b_{k-1}(n-1)\}\, b_{k-1}(n-1)$$
$$+ \{b_{k-1}(n-1) - a_{kk} f_{k-1}(n)\} f_{k-1}(n) \Big)$$

APE 1.17

Setting this to zero gives.

$$\sum_{\pm\infty} 2 f_{k-1}(n) b_{k-1}(n-1) - a_{kk} \sum_{\pm\infty} \{ b_{k-1}(n-1)^2 + f_{k-1}(n)^2 \} = 0 \qquad \text{APE 1.18}$$

Thus

$$a_{kk} = \frac{2 \sum_{\pm\infty} f_{k-1}(n) b_{k-1}(n)}{\sum_{\pm\infty} \{ b_{k-1}(n-1)^2 + f_{k-1}(n)^2 \}} \qquad \text{APE 1.19}$$

Hence for each order, $a_{kk} = R_k$ is evaluated by passing over the data with the prediction filter, forward and backward, to obtain $f_{k-1}(n)$ and $b_{k-1}(n)$. These are then used to evaluate a_{kk}. Use of this in equation APE1.7 allows the algorithm to proceed.

APPENDIX 2

A brief description of the Human speech generation and hearing systems

The speech generator may be considered to be a very sophisticated wind instrument. The wind source is air from the lungs. The ingenuity of the speech generator is demonstrated by the variety of sounds which may be generated. The large variety of sounds achievable by the vocal generator, is attributable to the different ways in which the steady air flow from the lungs are converted into audible vibration. The conversion into audible vibration is done by the introduction of a constriction in the path of the steady flow of air from the lungs. Each sound which may be generated in the production of speech is called a phoneme. The phonemes are classified according to how the audible vibration is generated.

Voiced sounds. For the generation of these, the air flow from the lungs are converted into audible vibration in the vocal chords. The position of this is shown in figure APD2.1. The vocal chords are mascular tissue attached to the inside of the larynx. The larynx is a cartilageneous box open at the top and the bottom; the larynx is situated at the top of the trachea. The region between the vocal chords is called the glottis. In breathing the vocal chords are kept open allowing the free passage of air. During the generation of a voiced phoneme, the vocal chords vibrate. It is supposed that the vibration is not under direct nerve action, however the tension associated with the attachment of the chords to the larynx and the effective mass of these is under direct mascular control. This allows a large variation in the frequency at which the vocal chords may vibrate. For men the pitch or frequency of

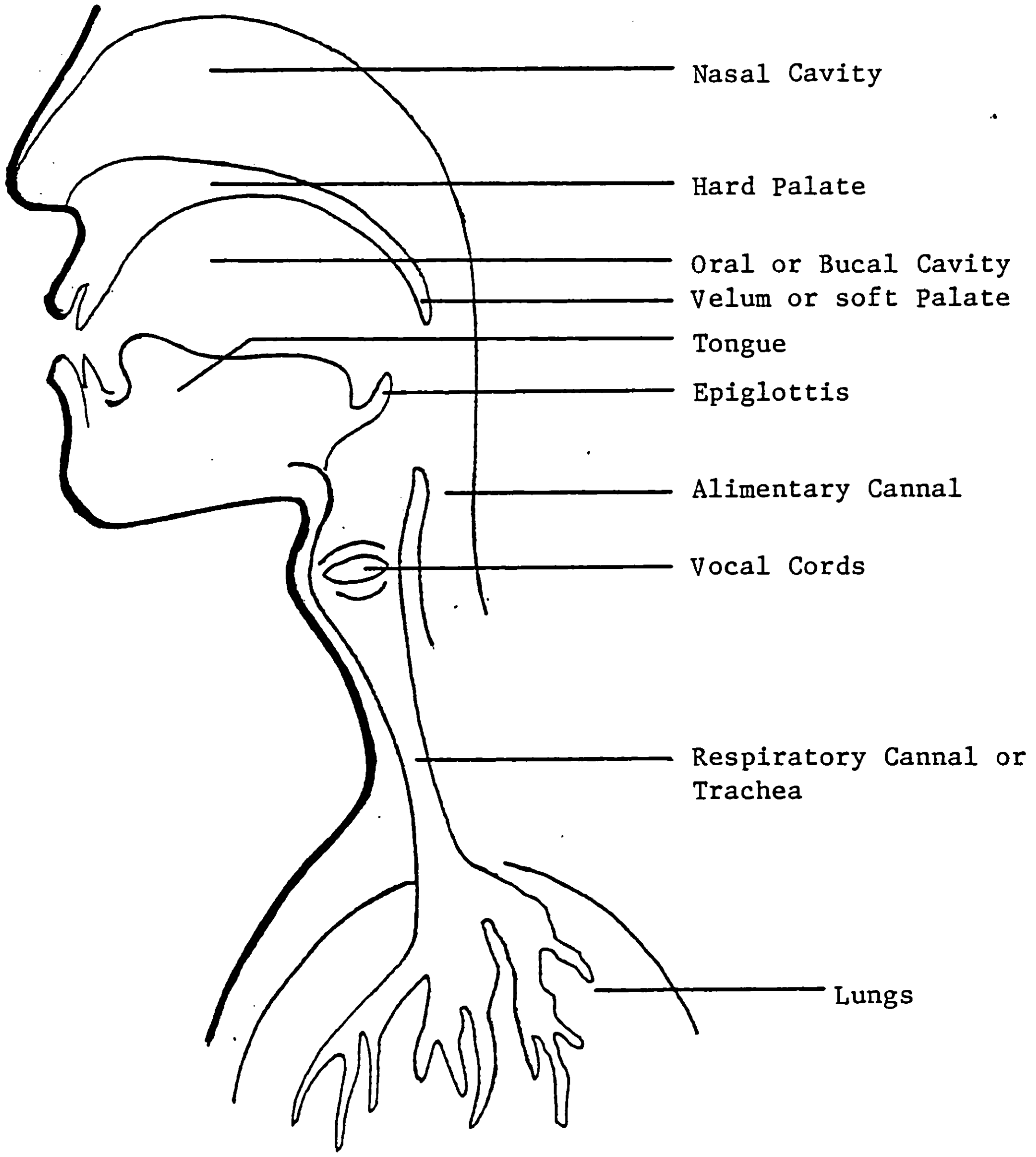


Figure APD2.1 Block diagram of the human speech generation apparatus.

vibration during speech is around 150Hz and for women around 250Hz. For little children the pitch may be as high as 400 Hz. (For women and children, the vocal chords tend to be shorter and thinner) The operation of the vocal chords is excellently described in a non-technical manner by Pierce and David-(1958) in chapter 4 of their book.

Unvoiced fricatives. The excitation for these is generated as follows. The vocal chords are kept open, and a constriction is formed at some point in the vocal tract. This causes turbulence and a noise like sound is made. For example to produce the "f" sound, a constriction is created at the lips, to produce the sound "θ", a constriction is created at the roof of the mouth, just behind the top front teeth, by the tongue.

Plosives. The vibration for these is generated by completely closing the vocal tract at some point for a short while. This allows air pressure to build up behind the point of closure. Upon the sudden release of this pressure, a plosive sound is made.

Combinations of the above three form the excitation for most of the phonemes. Other interesting sounds made are for example the vibration of the tongue in the manner of the vocal chords but at the rate of about 30Hz in the production of the rolled "r". The generation of the almost pure tone by whistling when producing an "s". An interesting point to note is that the vocal chords are not used when whispering. Whispering relies on the faint white noise generated when one allow the free passage of air from the lungs.

The second contributant to the sound generated in speech, is the colouring due to the vocal tract. Sections of the vocal tract may be modelled as concatenated tubes of different topologies, that is different lengths, cross-sectional areas etc. Depending on the shape and sizes of these tubes, resonances are induced which colour the primary auditory vibration generated by combinations of the methods mentioned above. Thus dependent upon how the bucal cavity is shaped, the nasal cavity is shaped and where the epiglottis and velum are positioned, in addition to the types of primary excitation used, a large ensemble of sounds may be produced. Table APT2.1 shows the phonemes of the British English language and how these are classified.

The hearing apparatus is an even more complicated device. Figure APD2.2a shows, in rough detail, the anatomy of the ear. The pinna, the ear drum and the oscicles combine to form a filter, a mechanical impedance matching device and a companding device. These serve to present sound in an appropriate manner for the inner ear. The inner ear contains the cochlea. This is the transducer for converting mechanical information to nervous information, as well as doing some preprocessing so that information is in an appropriate state for the brain to analyse. The cochlea is of great interest because it might allow us to say which information is lost in going from the mechanical signal to the signal which goes to the brain. We may thus neglect this information in coding.

A simplified drawing of a longitudinal section of an uncoiled cochlea is shown in figure APD2.2b. The cross-section is shown in figure APD2.2c. The important features of the cochlea are the basilar membrane and the organ of corti. The basilar membrane tapers as one goes from the basal end to the apical end of the cochlea. It vibrates in response to excitation from the oval

Phoneme	Class	Example
/b/	Voiced plosives	bat
/d/		dog
/g/		get
/p/	Voiceless plosives	pig
/t/		tell
/k/		kick
/m/	Nasals	man
/n/		null
/ŋ/		sing
/w/	Glides	well
/r/		ran
/l/		let
/j/		you
/h/	Voiceless fricatives	hat
/f/		fix
/θ/		thick
/s/		sat
/ʃ/		ship
/v/	Voiced fricatives	van
/ð/		this
/z/		zoo
/ʒ/		azure
/dʒ/	Affricates	joke
/tʃ/		chew
/i/	Front vowels	seat
/ɪ/		bit
/ɛ/		head
/æ/		hat
/ɑ/	Back vowels	cart
/ɒ/		rod
/ɔ/		cord
/ʊ/		would
/u/		rude
/ɜ/	Middle vowels	dirt
/ʌ/		hut
/ə/		the

Table APT2.1 Phonemes of British English.

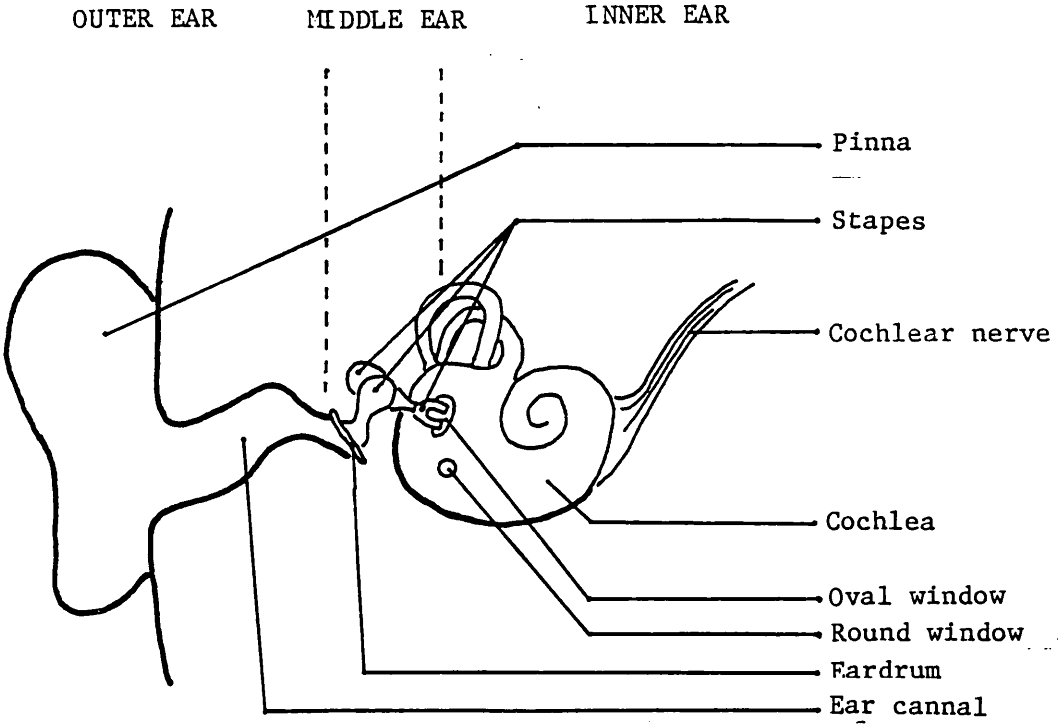


Figure APD2.2a Block diagram of human hearing apparatus.

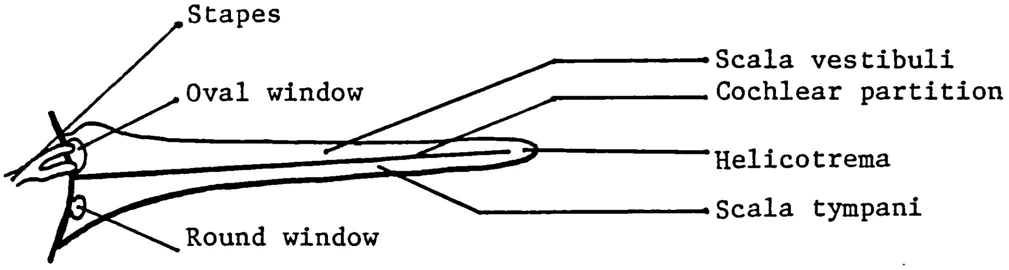


Figure APD2.2b Block diagram of unfolded cochlea.

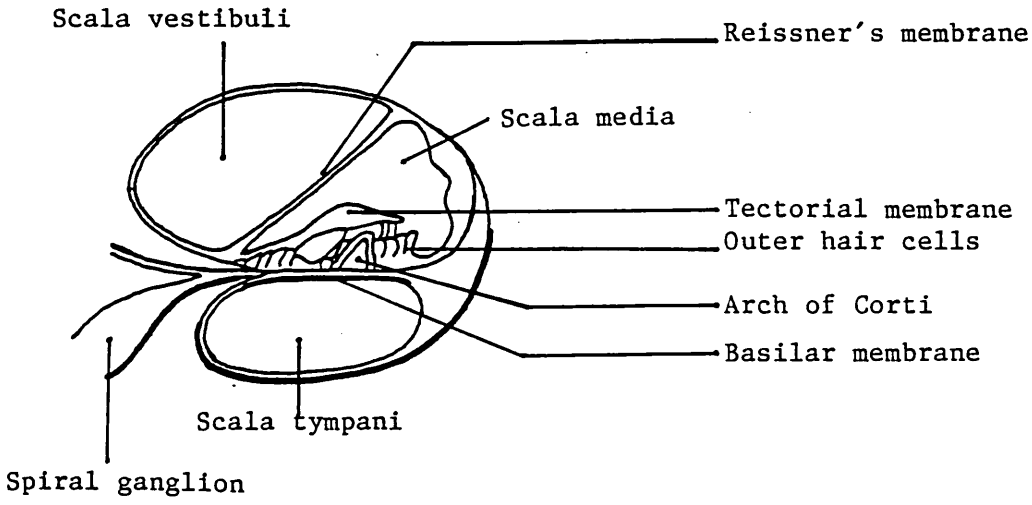


Figure APD2.2c Block diagram of cross-section of cochlea.

window, so that depending upon the frequency of the oscillation, the position on the membrane where the maximum amplitude of vibration occurs, varies. The basilar membrane therefore, performs some spectral analysis. The signal observed at each point on the membrane is a band-pass filtered version of the signal supplied at the oval window with a roughly constant Q along the length of the membrane. A rough drawing of the characterisics is given in figure APD2.3. The oscillation at any particular point of the basilar membrane causes the hairs of the sensory cells of the organ of corti to bend. This causes nerve impulses to be sent by the nerve cells to which the particular hairs are attached. The frequency of the firing of the nerve cells is dependent upon the quantity of bending of the associated hairs. Limitations on the frequency at which nerve cells may fire (approximately 4kHz as reported by Rose et al.-(1968)) give an indication of the ability of the ear to distinguish phase at high frequencies. There is some dispute about the band pass characteristics at different points on the basilar membrane. There is also some suggestion that the hairs of the organ of corti aid the frequency discrimination abilities of the hearing process. Investigation of sensitivity to phase has shown that the human ear is sensitive to phase at low frequencies, probably up to 1 or 2kHz. At higher frequencies, the ear is reportedly insensitive to phase. Recently it has been reported that sensitivity to phase at higher frequencies have been observed. The perception of phase occurred though only when the signal is noiselike and not of a simple harmonic structure. Thus these experiments were conducted using signals with a Gaussian spectrum with a centre frequency is the frequency of interest. It is not certain though, if this result was due to the perception of phase differences in lower frequency

THE AUDITORY SYSTEM

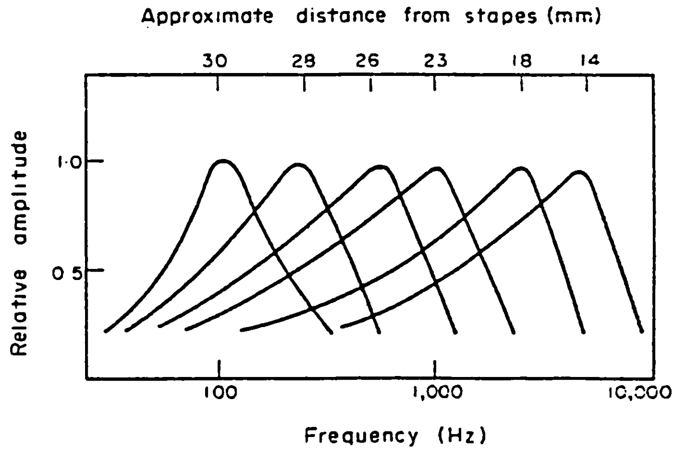


Figure APD2.3 Relative response of various points along the basilar membrane as a function of stimulus frequency intermodulation products which result from non-linearities in the signal path in the ear.

APPENDIX 3

The Ergodic theorem

A3.1 Definition of ergodicity

Consider a set Ω, with a measure P defined upon this. Let ω be a member of the set Ω. Consider a transformation T which operates on members of the set Ω eg

$$\omega_2 = T\omega_1 \qquad \text{APE 3.1}$$

where ω_1, ω_2 are members of the set Ω. This transformation may also be defined on a subset of Ω as follows.

$$\{\omega_4, \omega_3\} = T\{\omega_1, \omega_2\} \qquad \text{APE 3.2}$$

This transformation must have the following properties:

1) $T\Omega \subseteq \Omega$

2) If $\mathfrak{F}$ is the Borel field of subsets of Ω, with members B_i, then if $B_i \in \Omega$ then $T^{-1}B_i \in \Omega$

The transformation is called measure preserving if the measure of the set B_i is equal to the measure of the set $T^{-1}B_i$. That is

$$\int_{\forall \omega \in B_i} dP(\omega) = \int_{\forall \omega \in T^{-1}B_i} dP(\omega) \qquad \forall B_i \qquad \text{APE 3.3}$$

Ergodicity is the study of one class of measure preserving transformations or the study of the ordit of a class of measure preserving transformations. The orbit of a transformation is:

$$\omega,\ T\omega,\ T^2\omega,\ T^3\omega,\ T^4\omega,\ \ldots.$$

We shall give an example of a non-measure preserving transformation and an example of a measure preserving transformation, then the definition of an ergodic transformation or an ergodic source.

Example Let Ω=[0,1]. Let T=1/2 so that for $\omega \in \Omega$, T =ω/2. Let the measure P be the standard integral measure, then the measure for the set

$$\int_a^b \mathrm{d}\omega = b-a \qquad \text{APE 3.4}$$

and the measure for the set T^{-1}(a,b) is

$$\int_{\forall\,\omega:T\omega\in(a,b)} \mathrm{d}\omega = \int_{2a}^{2b} \mathrm{d}\omega = 2(b-a) \qquad \text{APE 3.5}$$

This is therefore not a measure preserving transformation. A portion of the orbit is

$$1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots.$$

(In statistical terms this is a sequence from a non-stationary source)

Example Let Ω =(0,1) and let T=2mod1 so that for all $\omega \in \Omega$, Tω=2ω mod 1. eg. ω=1/4 implies Tω=1/2, ω=3/4 implies Tω=6/4 mod 1=1/2. Let the measure P be the standard integral measure. Then the measure of the set (a,b) is

$$\int_a^b \mathrm{d}\omega = b-a \qquad \text{APE 3.6}$$

The measure of the set T^{-1}(a,b) is

$$\int_{\omega:\,T\omega\in(a,b)} \mathrm{d}\omega = \int_{\frac{a}{2}}^{\frac{b}{2}} \mathrm{d}\omega + \int_{\frac{a+1}{2}}^{\frac{b+1}{2}} \mathrm{d}\omega = \frac{b-a}{2} + \frac{(b+1)-(a+1)}{2}$$

$$= b-a \qquad \text{APE 3.7}$$

This is a measure preserving transformation.

An ergodic source is a source whose output is the orbit of a measure preserving transformation with the following properties.

Let Ω be the outcome set of a source. Let $S \subseteq \Omega$, then if S is invariant under transformation, then S has measure 0 or 1. That is

$$\text{If } S = TS \text{ then } \int_{\omega \in S} d\omega = 1 \text{ or } 0, \text{where } \int_{\forall \omega \in \Omega} d\omega = 1 \qquad \text{APE 3.8}$$

In other words all subsets of Ω which will occur all the time, once an experiment has started, have measure 0 or 1. Measure 1 implies S is different from Ω only on a set of zero measure. This concludes the definition.

A3.2 <u>The ergodic theorem AT3.1</u>

For any function $f(\omega)$ which is L_1 measurable, so that

$$\int f(\omega) dP(\omega) = \bar{f} \qquad \text{APE 3.9}$$

the time average converges to $\bar{f}$ if the transformation T (coordinate shift for a statistical source) or source is ergodic

That is $\lim_{N \to \infty} \frac{1}{N} \sum_{i=0}^{N-1} f(T^i \omega) = \bar{f}$

We shall refer to $\frac{1}{N} \sum_{i=0}^{N-1} f(T^i \omega)$ as $A_N f(\omega)$ or $A_N f$

The proof given here is due to G.D.Birhoff and F.Reisz in 1945. It is called the proof of the INDIVIDUAL ERGODIC THEOREM. An alternative way of stating the theorem is that

$$\lim_{N \to \infty} \int \left| \frac{1}{N} \sum_{i=0}^{N-1} f(T^i \omega) - \bar{f} \right| dP(\omega) = 0 \qquad \text{APE 3.10}$$

<u>Proof</u> This relies on showing that the following are true.

1) The quantities $\frac{1}{N} \sum_{i=0}^{N-1} f(T^i \omega_1)$ converge to $\bar{g}$ for all except for a set of ω values of zero measure.

2) $\|g\|_1 \leq \int |f(\omega)|\, dP(\omega)$ meaning that if $f(\omega) \epsilon L_1$ (it is L_1 measurable) then $|g|$ is finite

3) $\int \bar{g}\, dP(\omega) = \int f(\omega) dP(\omega)$ and hence

$$\bar{g} = \int f(\omega) dP(\omega) = \bar{f} \qquad \text{APE 3.11}$$

The proof of the first point makes up the bulk of the proof of the ergodic theorem. The proof of this requires the statement and proof a distinguished theorem, entitled the Maximal Ergodic Theorem.

Point 1

In going through the proof of point 1, we shall first state and apply the Maximal Ergodic theorem. The proof of which will be given later.

The maximal ergodic theorem:

Let

$$A_N f(\omega) = \frac{1}{N} \sum_{i=0}^{N-1} f(T^i \omega) \qquad \text{APE 3.12}$$

For every ω construct the sequence

$f(\omega)$, $A_2 f(\omega)$, $A_3 f(\omega)$,....$A_N f(\omega)$,.....

Let the supremum of this sequence be

$$\sup_{M \geq 1} A_M f(\omega)$$

for a given ω. Find the subspace $S(\lambda)$ of Ω defined such that for all ω which belong to S

$$\sup_{M \geq 1} A_M f(\omega) \geq \lambda \qquad \text{APE 3.13}$$

The Maximal Ergodic Theorem says that for each λ

$$\lambda P(S(\lambda)) \leq \int_{S(\lambda)} f(\omega) dP(\omega) \qquad \text{APE 3.14}$$

Now we shall use this to show that

$$\lim_{N\to\infty} A_N f(\omega) = \bar{g} \qquad \text{APE 3.15}$$

This is done by showing that the set of all ω for which

$$\{\limsup_{n\to\infty} A_N f(\omega) \geq b\} \quad \text{and} \quad \{\liminf_{n\to\infty} A_N f(\omega) \leq a\}$$

has zero measure for any b>a

Let $A_{ab} = \{\omega : \limsup_{n\to\infty} A_N f(\omega) \geq b \bigcap \liminf_{n\to\infty} A_N f(\omega) \leq a\}$ APE 3.16

Next it is required to show that the set A_{ab} is shift invariant. That is if $\omega \epsilon A_{ab}$ then $T\omega \epsilon A_{ab}$. Suppose that M is finite and there exites an N>M such that

$$A_N f(\omega) = \frac{1}{N} \sum_{i=0}^{N-1} f(T^i \omega) \geq b$$

$$\text{then} \quad \frac{1}{N}\Big(\{\sum_{i=0}^{N-1} f(T^i(T\omega))\} + \{f(\omega) - f(T^N \omega)\}\Big) \geq b \qquad \text{APE 3.17}$$

Thus as N → ∞

$$\frac{1}{N} \sum_{i=0}^{N-1} f(T^i(T\omega)) \geq b \qquad \text{APE 3.18}$$

Suppose that M is finite and there exits some k>M such that

$$A_k f(\omega) = \frac{1}{k} \sum_{i=0}^{k-1} f(T^i \omega) \leq a$$

$$= \frac{1}{k}\Big(\{\sum_{i=0}^{k-1} f(T^i(T\omega))\} + \{f(\omega) - f(T^k \omega)\}\Big) \leq a \qquad \text{APE 3.19}$$

Thus as k → ∞

$$\frac{1}{k} \sum_{i=1}^{k-1} f(T^i(T\omega)) \leq a \qquad \text{APE 3.20}$$

Thus

$$T\omega \in A_{ab}$$

Now by the definition of ergodicity this set should have measure 0 or 1, since for an ergodic source any shift invariant set should have a measure 0 or 1. Next we show the conditions on a,b under which the measure $P(A_{ab})$ is zero or one. We make use of the Maximal Ergodic theorem. Recall that for $S(\lambda)$, the set of all ω such that

$$\sup_{N \geq 1} A_N f(\omega) \geq \lambda$$

$$\lambda P(S(\lambda)) \leq \int f(\omega) \mathrm{d} P(\omega) \qquad \text{APE 3.21}$$

Now if the function $f(\omega)$ is replaced by the function $f(\omega).I_B(\omega)$ where $I_B(.)$ is the indicator function for any set B. That is $I_B(\omega)$ is one for all $\omega \in B$ and zero elsewhere.

Then $$\lambda P(S(\lambda) \bigcap B) \leq \int_{B \bigcap S(\lambda)} f(\omega) \mathrm{d} P(\omega) \qquad \text{APE 3.22}$$

Now let B be the set A_{ab} just considered. We know that if $\omega \in A_{ab}$, that is

$$\omega : \{\limsup_{n \to \infty} A_N f(\omega) \geq b\} \bigcap \{\liminf_{n \to \infty} A_N f(\omega) \leq a\}$$

then ω belongs to the set such that

$$\sup_{M \geq 1} A_M f(\omega) \geq b \text{ or } S(b)$$

Thus $A_{ab} \subseteq S(b)$ and $S(\lambda) \bigcap A_{ab} = A_{ab}$

Hence $bP(A_{ab}) \leq \int_{A_{ab}} f(\omega) \mathrm{d} P(\omega)$

Consider now relpacing the function $f(\omega)$ by $-f(\omega)$ and define A_{ab} as follows. Let $g(\omega) = -f(\omega)$

$$A_{ab} = \{\omega : \limsup_{n \to \infty} A_N g(\omega) \geq (-a) \bigcap \liminf_{n \to \infty} A_N g(\omega) \leq (-b)\} \qquad \text{APE 3.23}$$

Thus A_{ab} is axactly the same as before and

$$-aP(A_{ab}) \leq \int_{A_{ab}} g(\omega) \mathrm{d} P(\omega)$$

$$\text{or } aP(A_{ab}) \geq \int_{A_{ab}} f(\omega) \mathrm{d} P(\omega)$$

$$\text{Thus } aP(A_{ab}) \geq bP(A_{ab}) \qquad \text{APE 3.24}$$

but of course b>a, thus the above statement is a contradiction, unless $P(A_{ab})=0$.

This concludes the proof that

$$\lim_{N\to\infty}\frac{1}{N}\sum_{i=0}^{N-1} f(T^i\omega) \to \bar{g}$$

and $\bar{g}$ is an invariant function of ω. The next steps involve showing that $\bar{g}=\bar{f}$. These are quite straightforward.

Point 2

$$\|\bar{g}\|_1 = \| \lim_{N\to\infty}\frac{1}{N}\sum_{i=0}^{N-1} f(T^i\omega)\| = \int \lim_{N\to\infty} |\frac{1}{N}\sum_{i=0}^{N-1} f(T^i\omega)| \,\mathrm{d}P(\omega) \qquad \text{APE 3.25}$$

By interchanging the integral and the absolute operators, we get;

$$\|\bar{g}\|_1 \le \lim_{N\to\infty}\frac{1}{N}\sum_{i=0}^{N-1}\int |f(T^i\omega)| \,\mathrm{d}P(\omega) \qquad \text{APE 3.26}$$

By the measure invariance of the transformation we have,

$$\|\bar{g}\|_1 \le \lim_{N\to\infty}\frac{1}{N}\sum_{i=0}^{N-1} \|f(\omega)\|_1 = \|f(\omega)\|_1 = \int \|f(\omega)\| \,\mathrm{d}P(\omega) \qquad \text{APE 3.27}$$

By the L_1 measurability of the function $f(\omega)$ therefore, we conclude that $\|\bar{g}\|_1$ is finite.

Point 3

It is required lastly that $\bar{g} = \int f(\omega)\mathrm{d}P(\omega)\bar{f}$

Now

$$\bar{g} = \lim_{N\to\infty}\frac{1}{N}\sum_{i=0}^{N-1} f(T^i\omega) \qquad \text{APE 3.28}$$

Thus

$$\begin{aligned}\bar{g} &= \int \bar{g}\mathrm{d}P(\omega) = \lim_{N\to\infty}\frac{1}{N}\sum_{i=0}^{N-1}\int f(T^i\omega)\mathrm{d}P(\omega)\\ &= \int f(\omega)\mathrm{d}P(\omega)\\ &=\bar{f}\end{aligned} \qquad \text{APE 3.29}$$

This concludes the proof.

A3.3 The Maximal Ergodic theorem AT3.2

Let $$A_N f(\omega) = \frac{1}{N}\sum_{i=0}^{N-1} f(T^i\omega)$$

For every ω construct the sequence

$$f(\omega),\ A_2 f(\omega),\ A_3 f(\omega),\ A_3 f(\omega),\ A_4 f(\omega),\ \ldots.$$

Let the maximum of this sequence be

$$\sup_{M\geq 1} A_M f(\omega)$$

for a given ω. Find the subspace S(λ) of Ω such that for all $\omega \in$ S(λ)

$$\sup_{M\geq 1} A_M f(\omega) \geq \lambda \qquad \text{APE 3.30}$$

The Maximal Ergodic theorem says that for every λ

$$\lambda P(S(\lambda)) \leq \int_{S(\lambda)} f(\omega)\mathrm{d}P(\omega) \qquad \text{APE 3.31}$$

Proof For any λ we can rewrite,

$$\text{as} \quad \sup_{M\geq 1}\frac{1}{M}\sum_{k=0}^{M-1} f(T^k\omega) \geq \lambda \qquad \text{APE 3.32}$$

$$\text{or} \quad \sup_{M\geq 1}\frac{1}{M}\sum_{k=0}^{M-1} [f(T^k\omega)-\lambda] \geq 0 \qquad \text{APE 3.33}$$

replace f(ω)-λ by g(ω). Then it suffices to show that for all functions g(ω)=(f(ω)-λ), if S(λ) is defined thus

$$S(\lambda) = \{\omega : \sup_{M\geq 1}\frac{1}{M}\sum_{k=0}^{M-1} g(T^k\omega) \geq 0\} \qquad \text{APE 3.34}$$

then the following is true.

$$\lambda P(S(\lambda)) \leq \int_{S(\lambda)} f(\omega)\mathrm{d}P(\omega) \qquad \text{APE 3.35}$$

$$\lambda P(S(\lambda)) \leq \int_{S(\lambda)} [g(\omega)+\lambda]\mathrm{d}P(\omega) \qquad \text{APE 3.36}$$

$$\text{thus } 0 \leq \int_{S(\lambda)} g(\omega) \mathrm{d} P(\omega) \qquad \text{APE 3.37}$$

Now we may set about proving this alternative restatement of the theorem.

At this point we have to introduce the idea of m-leaders. Suppose $x_1, x_2, x_3, \ldots, x_n$ is a sequence of real numbers and m<n, then the member x_k is an m leader if either of these quantities is non-negative.

$$x_k \, ; \, x_k + x_{k+1} \, ; \, x_k + x_{k+1} + x_{k+2} \, ; \ldots ; \, x_k + x_{k+1} + \ldots + x_{k+m-1}$$

That is if there exists a $p \leq m$ such that

$$\sum_{i=0}^{P-1} x_{k+i} \geq 0 \qquad \text{APE 3.38}$$

then x_k is an m-leader. Alternatively x_k is an m-leader if

$$\sup_{1 \leq P \leq M} \frac{1}{P} \sum_{i=0}^{P-1} x_{k+i} \geq 0 \qquad \text{APE 3.39}$$

<u>A3.3.1 Lemma AT3.3</u>

The sum of all m-leaders is non-negative.

<u>Proof</u> Consider a sequence $x_{u+1}, x_{u+2}, \ldots$ Let x_a be the first m-leader in this sequence. Let $x_a, x_{a+1}, \ldots, x_{a+p-1}$ be the shortest m-leader sequence such that $x_a + x_{a+1} + \ldots x_{a+p-1}$ is non-negative. p<m. Then every member of x_{a+i-1}, i=1 to p of this sequence is an m-leader. This is because if $x_{a+i} + x_{a+i+1} + \ldots + x_{a+p-1}$ is negative, then $x_a + x_{a+1} + \ldots + x_{a+i-1}$ would be positive and hence $x_a + x_{a+1} + \ldots + x_{a+i-1}$ would be a shorter series which is non-negative. This contradicts the original premise. Thus $x_{a+i} + x_{a+i+1} + \ldots + x_{a+p-1}$ is non-negative and hence x_{a+i} is an m-leader.

Now what we have seen so far is that for x_a, the first m-leader, if we construct the shortest sequence $x_a, x_{a+1}, \dots, x_{a+p-1}$ such that the sum of this is non-negative, then all the members of this sequence are m-leaders. This defines our first p m-leaders, note that their sum is non-negative. Now consider the sequence $x_p, x_{p+1}, \dots$ starting form p, once again, looking for the first m-leader and the shortest sequence that it leads, with a non-negative sum will define the next few m-leaders, once again the sum of all these will be non-negative and so on. This concludes the proof for the lemma.

Recall that we require to show that if

$$S = \{\omega : \sup_{M \geq 1} \sum_{k=0}^{M-1} g(T^k \omega) \geq 0\} \qquad \text{APE } 3.40$$

then

$$0 \leq \int_S g(\omega) \mathrm{d} P(\omega) \qquad \text{APE } 3.41$$

We shall consider initially, the subset

$$S_m = \{\omega : \sup_{1 \leq p \leq M} \sum_{k=0}^{p-1} g(T^k \omega) \geq 0\} \qquad \text{APE } 3.42$$

Now if ω is such that

$$\sup_{1 \leq p \leq M} \sum_{k=0}^{p-1} g(T^k \omega) \geq 0 \qquad \text{APE } 3.43$$

then ω is such that there exist at least one $A_p g(\omega)$ which is non-zero, in other words $g(\omega)$ is an m-leader.

Thus S_m is the set of all ω such that $g(\omega)$ are m-leaders. Note that the set S_m is an increasing set with m so that $P(S_m)$, the measure of S_m increases monotonically with S_m. Thus

$$\int_{S_m} \mathrm{d} P(\omega) \text{ converges to } \int_S \mathrm{d} P(\omega) \qquad \text{APE } 3.44$$

We shall use the fact that the sum of all m-leaders, for any m, including m=∞, is positive to show that

$$\int_S g(\omega)\mathrm{d}P(\omega) \geq 0 \tag{APE 3.45}$$

Let T_M be the sum of the m-leaders in a sequence $f(\omega), f(T\omega), f(T^2\omega), \ldots, f(T^{N-1}\omega)$. Let $I_{S_M}(\omega)$ be the indicator function for S_M. That is $I_{S_M}(\omega)=1$ for ω S_M and zero otherwise

$$T_M = \sum_{k=0}^{N-1} g(T^k\omega) I_{S_M}(T^k\omega) \tag{APE 3.46}$$

Now we know that T_M is positive. Integrating both sides of the above equation takes us close to the conclusion of the proof

$$\int T_M \mathrm{d}P(\omega) = \sum_{k=0}^{N-1} \int f(T^k\omega) I_{S_M}(T^k\omega)\mathrm{d}P(\omega) \tag{APE 3.47}$$

This gives

$$\frac{T_M}{N} = \frac{1}{N} \sum_{k=0}^{N-1} \int f(T^k\omega) I_{S_M}(T^k\omega)\mathrm{d}P(\omega) \tag{APE 3.48}$$

Let the set S_M be the set of all ω for which f(ω) is an M-leader

then

$$\frac{T_M}{N} = \frac{1}{N} \sum_{k=0}^{N-1} \int_{S_M} f(\omega)\mathrm{d}P(\omega) \tag{APE 3.49}$$

Now

$$\frac{T_M}{N} \geq 0 \Rightarrow \sum_{k=0}^{N-1} \int_{S_M} f(\omega)\mathrm{d}P(\omega) \geq 0$$

$$\text{and} \quad \int_{S_M} f(\omega)\mathrm{d}P(\omega) \geq 0 \tag{APE 3.50}$$

This concludes the proof, since we can allow M to go to ∞.

The proof given here is a combination of the proofs as given by Billingsley and Halmos (Billingsley-(1965) pp 24-29 and Halmos-(1956) pp 18-21)

APPENDIX 4

A4.1 The theorem for the convergence of conditional expectation or probability

First we shall give the underlying definitions

A statistical source is defined by the following three items.

1) The sample set or space. This set Ω is the set of all possible values that a single random variable may take. Ω need not be a finite or countable set.

2) The sigma field of unions of subsets of Ω. For example if a set has a cardinality 3 say, with members ω_1, ω_2 and ω_3, the sigma field $\mathcal{F}$ has the following members;

$$\emptyset ; \omega_1 ; \omega_2 ; \omega_3 ; \omega_1 \bigcup \omega_2 ; \omega_1 \bigcup \omega_3 ; \omega_2 \bigcup \omega_3 ; \omega_1 \bigcup \omega_2 \bigcup \omega_3$$

This field contains the null set ϕ, the whole set Ω and all possible unions of all subsets of Ω. In general for a finite countable set Ω of cardinality C, the sigma field has 2^C members.

The sigma field is called a Borel-Sigma field if the field contains an infinite number of sets.

3) Associated with every member of the field $\mathcal{F}$ is a probability measure P. This is an additive set function such that if Λ_1 and Λ_2 are disjoint sets which belong to $\mathcal{F}$, then $P(\Lambda_1 \cup \Lambda_2)$ is $P(\Lambda_1)+P(\Lambda_2)$. $P(\phi)=0$ and $P(\Omega)=1$.

These three quantities $(\Omega,\mathcal{F},P)$ define a statistical source. Next we define subfields for a source. Suppose a source has a sample space or set Ω. Then $(A_1, A_2, A_3, \ldots, A_n, \ldots)$ is a Ξ decomposition of Ω if the A_i are disjoint and $\bigcup_i A_i = \Omega$. Then a sigma field Ξ of unions of the members of the decomposition may be constructed. The members of the Ξ decomposition are called the ATOMS of the field Ξ.

For example if $\Omega=\{\varpi_1,\varpi_2,\varpi_3,\varpi_4\}$ and the following atoms $\{\varpi_1 \cup \varpi_2, \varpi_3 \cup \varpi_4\}$ are chosen as the Ξ decomposition, then the field Ξ is constructed thus;

$$\emptyset;\ \varpi_1 \bigcup \varpi_2;\ \varpi_3 \bigcup \varpi_4;\ \varpi_1 \bigcup \varpi_2 \bigcup \varpi_3 \bigcup \varpi_4$$

A subfield Ξ of a field $\mathfrak{F}$ both constructed on the sample space Ω is defined as follows. The Ξ decomposition has atoms $\{B_1, B_2, \ldots\}$ with these properties: For any atom B_i, there exist an A_j which is an atom of the field $\mathfrak{F}$ such that $A_j \equiv B_i$. The Ξ decomposition is a courser decomposition of the set or space Ω than the $\mathfrak{F}$ decomposition is.

The books by Kolmogorov-(1933), Doob-(1953) and Billingsley-(1965) together give complete definitions of the terms used in axiomatic probability.

The Conditional expectation of a function $x(\omega)$ with respect to the subfield $\mathcal{G}$ is written as $E(x(\omega)\parallel\mathcal{G})$. $E(x(\omega)\parallel\mathcal{G})$ is defined thus:

1) $E(x(\omega)\parallel\mathcal{G})$ is an integrable or measurable function defined on ω, that is

$$\int_{\omega\in\mathcal{G}} |E(x(\omega)\parallel\mathcal{G})|\,\mathrm{d}P(\omega) < \infty \qquad \text{APE 4.1}$$

2)

$$\int_{\omega\in B_i} E(x(\omega)\parallel\mathcal{G})\mathrm{d}P(\omega) = \int_{\omega\in B_i} x(\omega)\mathrm{d}P(\omega) \qquad \text{APE 4.2}$$

for all B_i which are atoms of $\mathcal{G}$.

$E(x(\omega)\parallel\mathcal{G})$ is generally a 'smooth' approximation to $x(\omega)$

Two examples are; If $\mathcal{G}=\mathfrak{F}$, the largest sigma field associated with Ω, then for the second property above to be satisfied for all atoms of $\mathcal{G}$ means $E(x(\omega)\parallel\mathcal{G})=x(\omega)$

Suppose $\mathcal{G}$ has atoms $\{2, \Omega-2\cap\Omega\}$, where 2 is a member of Ω.

Then a 'version' of $E(x(\omega)\|\mathcal{G})$ is

$$\begin{aligned} E(x(\omega)\,\|\,\mathcal{G}) &= x(2) \quad \text{for} \quad \omega = 2 \\ &= \left\{ \frac{\int_{\forall\omega\neq 2} x(\omega)\mathrm{d}P(\omega)}{\int_{\forall\omega\neq 2}\mathrm{d}P(\omega)} \right\} \quad \text{for} \quad \omega \neq 2 \end{aligned} \qquad \text{APE 4.3}$$

In fact for $\omega=2$, $E(x(\omega)\|\ )$ may take any value provided that

$$\int_{\forall\omega\neq 2} E(x(\omega)\,\|\,\mathcal{G})\mathrm{d}P(\omega) = \int_{\forall\omega\neq 2} x(\omega)\mathrm{d}P(\omega) \qquad \text{APE 4.4}$$

The various allowable functions which may be used for $E(x(\omega)\|\mathcal{G})$ are called 'versions' of $E(x(\omega)\|\mathcal{G})$. If we chose the 'smoothest', then there is a direct link between the conditional expectation as defined above and the usual conditional expectation.

The conditional expectation $E(x(\omega)|A)$ where A is a subset of Ω is defined below;

$$E(x(\omega)|A) = \frac{\int_{\forall\omega\in A} x(\omega)\mathrm{d}P(\omega)}{\int_{\forall\omega\in A}\mathrm{d}P(\omega)} \qquad \text{APE 4.5}$$

This is defined only where $\omega\epsilon$A over which region it is a constant.

Then if we define $\mathcal{G}$ such that A is one of its atoms, then a version of $E(x(\omega)\|\mathcal{G})$ is some function defined when $\omega\epsilon$A so that

$$\int_{\omega\in A} E(x(\omega)\,\|\,\mathcal{G})\mathrm{d}P(\omega) = \int_{\omega\in A} x(\omega)\mathrm{d}P(\omega) \qquad \text{APE 4.6}$$

Letting $E(x(\omega)\|\mathcal{G})$ be a constant over this region gives:

$$E(x(\omega)\,\|\,\mathcal{G}) = \frac{\int_{\omega\in A} x(\omega)\mathrm{d}P(\omega)}{\int_{\omega\in A}\mathrm{d}P(\omega)} = E(x(\omega)|A) \qquad \text{APE 4.7}$$

<u>Conditional probability</u> If we allow $x(\omega)=I_{\mu}(\omega)$, where $I_{\mu}(\omega)=1$ when $\omega=\mu$ and zero elsewhere, then for the case when

Ω is discrete,

$$E(I_\mu(\omega) \parallel \mathcal{G}) = P(\mu \parallel \mathcal{G}) \qquad \text{APE 4.8}$$

where $P(\mu \| \mathcal{G})$ is the conditional probability.

For Ω continuous,

$$\text{Let } x(\omega) = \lim_{\epsilon \to 0} \frac{H_\mu(\omega + \epsilon) - H_\mu(\omega)}{\epsilon} \qquad \text{APE 4.9}$$

where $H_\mu(\omega)$ is a step function and $H_\mu(\omega)=1$ when $\omega \leq \mu$ and 0 when $\omega > \mu$

Then

$$\lim_{\epsilon \to 0} E\left[\frac{H_\mu(\omega + \epsilon) - H_\mu(\omega)}{\epsilon} \parallel \mathcal{G}\right] = \lim_{\epsilon \to 0} \left[\frac{F(\{\mu + \epsilon\} \parallel \mathcal{G}) - F(\mu \parallel \mathcal{G})}{\epsilon}\right] \qquad \text{APE 4.10}$$

where F is the probability distribution function. Of course the right hand side of the last equation is $f(\mu \| \mathcal{G})$ which is the conditional density function. Thus theorems proved for the convergence of conditional expectations apply to conditional probability functions as well.

Properties to conditional expectation

1) If $x(\omega)=a$ everywhere, then $E(x(\omega) \| \mathcal{G})=a$ everwhere.

2) If $x(\omega) \leq y(\omega)$ everywhere, then $E(x(\omega) \| \mathcal{G}) \leq E(y(\omega) \| \mathcal{G})$ everywhere.

3) If a and b are constant then

$$E(\{ax(\omega) + by(\omega)\} \parallel \mathcal{G}) = aE(x(\omega) \parallel \mathcal{G}) + bE(y(\omega) \parallel \mathcal{G}) \qquad \text{APE 4.11}$$

4) $|E(x(\omega) \parallel \mathcal{G})| \leq E(|x(\omega)| \parallel \mathcal{G})$

5) If $\lim_{n \to \infty} x_n(\omega)=x(\omega)$ and $|x(\omega)|<y$ almost everywhere and y is integrable, then $\lim_{n \to \infty} E(x_n(\omega) \parallel \mathcal{G}) = E(x(\omega) \parallel \mathcal{G})$

Proof Let $\quad z_n(\omega) = \sup_{m \geq n} |x_m(\omega) - x(\omega)|$

Then

$$|E(x_n(\omega) \parallel \mathcal{G}) - E(x(\omega) \parallel \mathcal{G})| \le E(|x_n(\omega) - x(\omega)| \parallel \mathcal{G}) \le E(z_n(\omega) \parallel \mathcal{G}) \qquad \text{APE 4.12}$$

Now $E(z_n(\omega)\parallel \mathcal{G})$ be property 2, is non-increasing and therefore must converge to something. Since $E(z_n(\omega)\parallel \mathcal{G})$ is non-negative, if its expectation goes to zero, then $E(z_n(\omega)\parallel \mathcal{G}) \rightarrow 0$.

Then $\int_{\omega \in B_i} E(z_n(\omega) \parallel \mathcal{G}) \mathrm{d}P(\omega) = \int_{\omega \in B_i} z_n(\omega) \mathrm{d}P(\omega)$, B_i are atoms of $\mathcal{G}$

But

$$\int z_n(\omega) \mathrm{d}P(\omega) \rightarrow 0$$

This concludes the proof.

6) If $x(\omega)$ is integrable and the sigma fields $\mathcal{G}_1$ and $\mathcal{G}_2$ are such that $\mathcal{G}_1 \subset \mathcal{G}_2$, then $E(E(x(\omega) \parallel \mathcal{G}_2) \parallel \mathcal{G}_1) = E(x(\omega) \parallel \mathcal{G}_1)$

Proof We note that every atom B_i of $\mathcal{G}_1$ is a subset of some atom A_j of $\mathcal{G}_2$. Recall that

$$\int_{\omega \in A_i} E(x(\omega) \parallel \mathcal{G}_2) \mathrm{d}P(\omega) = \int_{\omega \in A_i} x(\omega) \mathrm{d}P(\omega) \qquad \text{APE 4.13}$$

for all A_i which are atoms of $\mathcal{G}_2$.

Now consider a particular atom B_i of $\mathcal{G}_1$, and suppose that the atoms A_{i1}, A_{i2}, ..., A_{in} make up B_i

Then
$$\int_{\omega \in B_i} E(E(x(\omega) \parallel \mathcal{G}_2) \parallel \mathcal{G}_1) \mathrm{d}P(\omega) = \int_{\omega \in B_i} E(x(\omega) \parallel \mathcal{G}_2) \mathrm{d}P(\omega) \qquad \text{APE 4.14}$$

But for any $A_{ij} \in B_i$,

$$\int_{\omega \in A_{ij}} E(x(\omega) \parallel \mathcal{G}_2) \mathrm{d}P(\omega) = \int_{\omega \in A_{ij}} x(\omega) \mathrm{d}P(\omega) \qquad \text{APE 4.15}$$

Thus

$$\int_{\omega \in A_{ij}} E(E(x(\omega) \parallel \mathcal{G}_2) \parallel \mathcal{G}_1) \mathrm{d}P(\omega) = \int_{\omega \in A_{ij}} x(\omega) \mathrm{d}P(\omega) \qquad \text{APE 4.16}$$

Hence $E(E(x(\omega) \parallel \mathcal{G}_2) \parallel \mathcal{G}_1)$ is a version of $E(x(\omega) \parallel \mathcal{G}_1)$

A4.2 The convergence theorem AT4.1

Suppose that

$$\mathcal{G}_1 \subseteq \mathcal{G}_2 \subseteq \mathcal{G}_3 \subseteq \cdots \subseteq \mathcal{G} \text{ and } \mathcal{G} = \bigcup_{i=1}^{\infty} \mathcal{G}_i \qquad \text{APE 4.17}$$

Then $\lim\limits_{n\to\infty} E(z(\omega) \parallel \mathcal{G}_n) = E(z(\omega) \parallel \mathcal{G})$

for any function $z(\omega)$ which is measurable

Proof

Define $x_n(\omega) = E(z(\omega) \parallel \mathcal{G}_n)$
$\qquad\qquad = E(x_{n+1}(\omega) \parallel \mathcal{G}_n)$

by property 6. Then the process $(x_n(\omega), \mathcal{G}_n)$ for $n>0$ forms a semi-martingale. The martingale convergence theorems therefore apply. That is $\lim\limits_{n\to\infty} x_n(\omega) = x_\infty(\omega)$

The proof for this is very similar to that used for the individual ergodic theorem. The proof makes use of a theorem that serves the same purpose as the maximal ergodic theorem for the ergodic theorem. This theorem will be stated and used, and the proof given later.

Theorem A4.2 Let $x_j(\omega)$, $1 \leq j \leq n$ be a sequence of funtions, which form a semi-martingale. Let λ be any real number. In addition, let the set $S_n(\lambda)$ be the set of all ω values, so that $\max\limits_{1\leq j\leq n}\{x_j(\omega)\} \geq \lambda$

Then

$$\lambda P(S_n(\lambda)) \leq \int_{S_n(\lambda)} x_n(\omega) \mathrm{d}P(\omega) \qquad \text{APE 4.18}$$

Construct, for a given ω, the following sequence.

$x_n(\omega), x_{n+1}(\omega), x_{n+2}(\omega), \ldots.$

Let

$$\mu_S(\omega) = \limsup_{n\to\infty} x_n(\omega)$$
$$\mu_I(\omega) = \liminf_{n\to\infty} x_n(\omega) \qquad \text{APE 4.19}$$

Then define the set A_{ab} as

$$\{\omega : \mu_I(\omega) < a < b < \mu_S(\omega)\}$$

It will be shown that A_{ab} is a set of zero measure. Now it should be noted that $A_{ab} \subseteq \lim_{n\to\infty} S_n(b) = S_\infty(b)$

Let us return to theorem A4.2 Suppose $S_\infty(b) = \{\omega : \lim_{n\to\infty} \max_{1\le j\le n} x_j(\omega) \ge b\}$

Then

$$bP(S_\infty(b)) \le \int_{S_\infty(b)} x_\infty(\omega)\mathrm{d}P(\omega) \qquad \text{APE 4.20}$$

for any measurable $x_j(\omega)$. Suppose $y_j(\omega)=x_j(\omega)I_{A_{ab}}(\omega)$, where $I_{A_{ab}}$ is an indicator function for the set A_{ab}. Then

$$I_{A_{ab}} = 1 \quad \forall\, \omega \in A_{ab}$$
$$= 0 \quad \forall\, \omega \notin A_{ab} \qquad \text{APE 4.21}$$

$$\text{Thus } bP(S_\infty(b) \bigcap A_{ab}) \le \int_{S_\infty(b) \bigcap A_{ab}} x_\infty(\omega)\mathrm{d}P(\omega) \qquad \text{APE 4.22}$$

But since $A_{ab} \subseteq S_\infty(b)$ we have

$$bP(A_{ab}) \le \int_{A_{ab}} x_\infty(\omega)\mathrm{d}P(\omega) \qquad \text{APE 4.23}$$

Now construct the sequence

$-x_1(\omega), -x_2(\omega), -x_3(\omega), \ldots$ Let

$$\mu_S(\omega) = \limsup_{n\to\infty} -x_n(\omega) \text{ and } \mu_I(\omega) = \liminf_{n\to\infty} -x_n(\omega) \qquad \text{APE 4.24}$$

Then define the set A_{ab} as $\{w : \mu_I(\omega) \le -b < -a \le \mu_S(\omega)\}$

A_{ab} is the same set as defined before. Then

$$-aP(S_\infty(-a) \bigcap A_{ab}) \le \int_{S_\infty(-a) \bigcap A_{ab}} -x_\infty(\omega)\mathrm{d}P(\omega) \qquad \text{APE 4.25}$$

But $$A_{ab} \subseteq S_\infty(-a) \text{ and } S_\infty(-a) \bigcap A_{ab} = A_{ab} \qquad \text{APE 4.26}$$

Hence

$$-aP(A_{ab}) \leq \int_{A_{ab}} -x_\infty(\omega)\mathrm{d}P(\omega) \qquad \text{APE 4.27}$$

and

$$aP(A_{ab}) \geq \int_{A_{ab}} x_\infty(\omega)\mathrm{d}P(\omega) \qquad \text{APE 4.28}$$

Thus

$$bP(A_{ab}) \leq \int_{A_{ab}} x_\infty(\omega)\mathrm{d}P(\omega) \leq aP(A_{ab}) \qquad \text{APE 4.29}$$

Thus $P(A_{ab})=0$.

Therefore for each ω the superior and inferior limits have the same value for the sequence $x_i(\omega)$.

Theorem A4.3

Let $\{x_i(\omega), 1 \leq j \leq n\}$ be a semi-martingale and let λ be a real number not neccessarily positive. Let $S(\lambda) = \{\omega : \max_{1\leq j\leq n} x_j(\omega) \geq \lambda\}$

Then $\lambda P(S(\lambda)) \leq \int_{S(\lambda)} x_n(\omega)\mathrm{d}P(\omega)$

Proof

Recall that the set $S(\lambda)$ is the set of all ω where there exists at least one $x_j(\omega)$, $1 \leq j \leq n$, such that $x_l(\omega) \geq \delta$. Let the set Λ_k be the set of ω values so that there exists an $x_j(\omega)$, $1 \leq j \leq n$, which is greater than λ and the first one that is greater than λ is $x_j(\omega)$.

$$\Lambda_k = \{\omega : x_k(\omega) \geq \lambda \text{ and } x_l(\omega) < \lambda, \ \forall\, 1 \leq l < k\} \qquad \text{APE 4.30}$$

Then $\bigcup_{k=1}^{n} \Lambda_k = S(\lambda)$ and the Λ_k are disjoint.

Then

$$P(S(\lambda)) = \sum_{k=1}^{n} \int_{\Lambda_k} \mathrm{d}P(\omega) \qquad \text{APE 4.31}$$

and

$$\lambda P(S(\lambda)) = \sum_{k=1}^{n} \int_{\Lambda_k} \lambda \mathrm{d} P(\omega) \qquad \text{APE 4.32}$$

Now for each set Λ_k, $x_k(\omega) \geq \lambda$, therefore for each Λ_k

$$\int_{\Lambda_k} \lambda \mathrm{d} P(\omega) \leq \int x_k(\omega) \mathrm{d} P(\omega) \qquad \text{APE 4.33}$$

Thus

$$\lambda P(S(\lambda)) \leq \sum_{k=1}^{n} \int_{\Lambda_k} x_k(\omega) \mathrm{d} P(\omega) \qquad \text{APE 4.34}$$

The functions

$x_1(\omega)=E(z(\omega) \| \mathcal{G}_1)$, $x_2(\omega)=E(z(\omega) \| \mathcal{G}_2)$, ...$x_n(\omega)=E(z(\omega) \| \mathcal{G}_n)$ are all only as fine as the number of atoms in the respective $\mathcal{G}_i$ allow. The consequence for the sets Λ_i is that each Λ_i contains an integer number of atoms of $\mathcal{G}_i$. This is because of the courseness of the functions $E(z(\omega) \| \mathcal{G}_i)$. If there exists some ω such that $x_i(\omega)=E(z(\omega) \| \mathcal{G}_i) \geq \lambda$ then these ω will by definition be an integer quantity of atoms of $\mathcal{G}_k$. Let the atoms of $\mathcal{G}_k$ which belong to Λ_k be B_{ki} .

Then

$$\begin{aligned} \int_{\Lambda_k} x_k(\omega) \mathrm{d} P(\omega) &= \int_{\Lambda_k} E(z(\omega) \parallel \mathcal{G}_k) \mathrm{d} P(\omega) \\ &= \int_{\Lambda_k} E(E(z(\omega) \parallel \mathcal{G}_k) \parallel \mathcal{G}_n) \mathrm{d} P(\omega) \\ &= \int_{\omega \in B_{ki}} E(z(\omega) \parallel \mathcal{G}_n) \mathrm{d} P(\omega) \qquad \text{APE 4.35} \end{aligned}$$

for every atom B_{ki} of $\mathcal{G}_k$ that belongs to Λ_k. Thus

$$\sum_{k=1}^{n} \int_{\Lambda_k} x_k(\omega) \mathrm{d} P(\omega) = \int_{\forall B_j} E(z(\omega) \parallel \mathcal{G}_n) \mathrm{d} P(\omega) \qquad \text{APE 4.36}$$

B_j are the atoms of $\mathcal{G}_k$. Also since $\mathcal{G}_k$ is a subfield of $\mathcal{G}_n$ we may integrate over the atoms of A_i of the $\mathcal{G}_n$ decomposition of Ω instead.

Thus

$$\lambda P(S(\lambda)) = \int_{\forall A_i} E(z(\omega) \parallel \mathcal{G}_n) \mathrm{d} P(\omega)$$

$$\lambda P(S(\lambda)) = \int x_n(\omega) \mathrm{d} P(\omega) \qquad \text{APE 4.37}$$

These theorems and their proofs may be found in Billingsley-(1965) pp106-122 and in Doob-(1953)'s chapter on Martingales.

APPENDIX 5

The rate and distortion bounds for Gaussian sources with imprecisely specified distribution parameters

A5.1 The transition probability function for coding a Gaussian source to attain the rate-distortion bound

Let the distortion measure be the square difference measure

$$d(x,y) = (x-y)^2 \qquad \text{APE 5.1}$$

where x is the source symbol and y is the reproduction symbol. The following equations solve the constrained optimisation problem of finding the conditional density function which achieves the minimum rate for a distortion bound.

$$1 = \int_{\pm\infty} f(x)\exp\{-\rho d(x,y)\}\mathrm{d}x \qquad \text{APE 5.2}$$

$$\frac{p(x)}{f(x)} = \int_{\pm\infty} q(y)\exp\{-\rho d(x,y)\}\mathrm{d}y \qquad \text{APE 5.3}$$

$$p(y|x) = \frac{q(y)f(x)}{p(x)}\exp\{-\rho d(x,y)\} \qquad \text{APE 5.4}$$

f(x) is a Lagrange multiplier function and a Lagrange multiplier governing respectively conditions APE5.6 and APE5.7.

$$\begin{aligned}\text{Rate} = R(d^*) &= \int p(x)\int p(y|x)\ln[\frac{p(y|x)}{\int p(y|u)p(u)\mathrm{d}u}]\mathrm{d}y\,\mathrm{d}x \\ &= \int p(x)\int p(y|x)\ln[\frac{p(y|x)}{q(y)}]\mathrm{d}y\,\mathrm{d}x \qquad \text{APE 5.5} \\ \text{where } 1 &= \int p(y|x)\mathrm{d}y \quad \forall x \qquad \text{APE 5.6} \\ \text{and } d^* &\geq \int p(x)\int p(y|x)d(x,y)\mathrm{d}y\,\mathrm{d}x \qquad \text{APE 5.7}\end{aligned}$$

Let

$$p(x) = \frac{1}{\sigma_x\sqrt{2\pi}}\exp-\frac{(x-\mu_x)^2}{2\sigma_x^2}, \qquad q(y) = \frac{1}{\sigma_y\sqrt{2\pi}}\exp-\frac{(y-\mu_y)^2}{2\sigma_y^2} \qquad \text{APE 5.8}$$

The derivations for equations APE5.2 and APE5.3 may be obtained from Berger-(1970) pages 88 to 90. In Gallagher-(1968) it is shown that the supposition that f(x) is a constant is consistent with the equations APE5.2, APE5.3, APE5.6 and APE5.7. In fact f(x) should be $\sqrt{\rho/\pi}$. supposing that q(y) is Gaussian with distribution as follows

$$q(y) = \frac{1}{\sigma_y\sqrt{2\pi}}\exp\{-\frac{(y-\mu_y)^2}{2\sigma_y^2}\} \qquad \text{APE 5.9}$$

Inserting this into APE5.3 allows the evaluation of μ_y and σ_y in terms of the μ_x, σ_x and ρ.

Thus

$$\sigma_y = \sqrt{\frac{2\rho\sigma_x^2 - 1}{2\rho}} \qquad \text{APE 5.10}$$

$$\mu_y = \mu_x \qquad \text{APE 5.11}$$

$$q(y) = \sqrt{\frac{\rho}{\pi(2\rho\sigma_x^2 - 1)}}\exp\{-\frac{\rho}{2\rho\sigma_2 - 1}(y-\mu_x)^2\} \qquad \text{APE 5.12}$$

now $\rho = \frac{1}{2d^{\bullet}}$ APE 5.13

(Gallagher-(1968) page 476)

$$p(y|x) = \sqrt{\frac{\sigma_x^2}{2\pi d^{\bullet}(\sigma_x^2 - d^{\bullet})}}exp\{-\frac{\sigma_x^2}{2d^{\bullet}(\sigma_x^2 - d^{\bullet})}(y-A)^2\} \qquad \text{APE 5.14}$$

$$\text{where } A = \mu_x\frac{d^{\bullet}}{\sigma_x^2} + (1 - \frac{d^{\bullet}}{\sigma_x^2}) \qquad \text{APE 5.15}$$

A5.2 The rate if the source statistics are (μ,σ) and the transition density function presumes statistics $(\hat{\mu},\hat{\sigma})$

The rate bound is

$$R(d^{*}) = \int p(x) \int \hat{p}(y|x) \ln[\frac{\hat{p}(y|x)}{\int \hat{p}(y|u)\hat{p}(u)\mathrm{d}u}]\mathrm{d}y\,\mathrm{d}x$$

$$= \int p(x) \int \hat{p}(y|x) \ln[\frac{\hat{p}(y|x)}{\hat{q}(y)}]\mathrm{d}y\,\mathrm{d}x$$

where

$$\frac{\hat{p}(y|x)}{\hat{q}(y)} = \frac{\hat{f}(x)}{\hat{p}(x)} \exp\{-\rho d(x,y)\}$$

Thus

$$R(d^{*}) = \int p(x) \int \hat{p}(y|x) \ln\frac{\hat{f}(x)}{\hat{p}(x)}\mathrm{d}y\,\mathrm{d}x - \rho \int p(x) \int (x-y)^2 \hat{p}(y|x)\mathrm{d}y\,\mathrm{d}x$$

$$= \int \frac{p(x)}{\hat{p}(x)} f(x) \ln\frac{\hat{f}(x)}{\hat{p}(x)} \int \hat{q}(y) \exp\{-\hat{\rho}(x-y)^2\}\mathrm{d}y\,\mathrm{d}x - \rho \int \frac{p(x)}{\hat{p}(x)} f(x) \int (x-y)^2 \exp\{-\hat{\rho}(x-y)^2\}\mathrm{d}y\,\mathrm{d}x$$

For the Gaussian source

$$R(d^{*}) = \sqrt{\frac{1}{2\pi d^{*}}} \int \frac{p(x)}{\hat{p}(x)} \int \hat{q}(y) \{\ln\frac{\hat{f}(x)}{\hat{p}(x)} - (x-y)^2\} \exp\{-\frac{(x-y)^2}{2d^{*}}\}\mathrm{d}y\,\mathrm{d}x$$

By substituting the following into the above equation;

$$p(x) = \frac{1}{\sigma_x\sqrt{2\pi}} \exp\{-\frac{(x-\mu_x)^2}{2\sigma_x^2}\}$$

$$\hat{p}(x) = \frac{1}{\hat{\sigma}_x\sqrt{2\pi}} \exp\{-\frac{(x-\hat{\mu}_x)^2}{2\hat{\sigma}_x^2}\}$$

$$\hat{q}(y) = \sqrt{\frac{1}{2\pi(\hat{\sigma}_x^2-d^{*})}} \exp\{\frac{(y-\mu_x)^2}{2(\hat{\sigma}_x^2-d^{*})}\}$$

one obtains

$$R(d^{*}) = \frac{1}{2\hat{\sigma}_x^2}(1-\frac{d^{*}}{\hat{\sigma}_x^2})\{(\sigma_x^2-\hat{\sigma}_x^2)-(\mu_x-\hat{\mu}_x)^2\} + \frac{1}{2}\ln\frac{\hat{\sigma}_x^2}{d^{*}}$$

A5.3 The distortion if the source statistics are (μ,σ) and the transition density function presumes statistics $(\hat{\mu},\hat{\sigma})$

The distortion D_{tot} resulting from the use of inappropriate statistics and hence the conditional density function p(y|x) is

$$D_{\mathrm{tot}} = \int p(x) \int \hat{p}(y|x) d(x,y) \mathrm{d}y \, \mathrm{d}x$$

$$= \int \frac{p(x)}{\hat{p}(x)} \int \{ \sqrt{\frac{1}{2\pi d^{*}}} \hat{q}(y) \exp\{-\frac{1}{2d^{*}}(x-y)^2\} (x-y)^2 \mathrm{d}y \, \mathrm{d}x$$

where

$$\hat{q}(x) = \sqrt{\frac{1}{2\pi(\hat{\sigma}_x^2 - d^{*})}} \exp\{-\frac{(y-\hat{\mu}_x)^2}{2(\hat{\sigma}_x^2 - d^{*})}\}$$

$$p(x) = \frac{1}{\sigma_x \sqrt{2\pi}} \exp\{-\frac{1}{2}\frac{(x-\mu_x)^2}{2\sigma_x^2}\}$$

$$\hat{p}(x) = \frac{1}{\hat{\sigma}_x \sqrt{2\pi}} \exp\{-\frac{1}{2}\frac{(x-\hat{\mu}_x)^2}{2\hat{\sigma}_x^2}\}$$

With some algebraic manipulation the above reduces to

$$d^{*} + (\frac{d^{*}}{\hat{\sigma}_x^2})^2 \{ (\sigma_x^2 - \hat{\sigma}_x^2) + (\mu_x - \hat{\mu}_x)^2 \}$$

REFERENCES

I. E. Abdou and W. K.Pratt, "Quantitative design and Evaluation of Enhancement/Thresholding edge detectors." IEEE Proceedings, Vol 67 No 5, pp753-763, May 1979.

H. Abut, R. M. Gray and G. Robodello, "Vector quantisation of speech and speech-like waveforms." IEEE Transactions on Acoustics Speech and Signal Processing, Vol ASSP-30, pp423-425, June 1982

S. Ahmadi, "Low bit rate digital speech signal processing systems." PhD Thesis, Imperial College, University of London, 1980.

N. Ahmed and K. R. Rao, "Orthogonal transforms for digital signal processing" Springer-Verlag 1975.

N. Ahmed, T. Natarajan and K. R. Rao, "Discrete Cosine Transform." IEEE Transactions on Computers, Vol C-23 p90-93, Jan 1974.

W. A. Ainsworth, "Mechanisms of speech recognition." Pergamon Press Ltd., 1976.

G. B. Anderson and T. S. Huang, "Piecewise Fourier transformation for picture bandwidth compression." IEEE Transactions on Commun., Vol COM-19, pp133-140, April 1971.

J. B. Anderson and C.W. Law, "Real number convolutional codes for speech-like quasi-stationary sources." IEEE Transactions on Info. Theory, Vol IT-23, pp778-782, Nov 1977.

H. C. Andrews, J. Kane and W. K. Pratt, "Hadamard transform image coding" IEEE Proceedings, Vol 57, pp58-68, January 1969.

L. Arena and G. Zarone, "Facsimile encoding by patterns." International conference on Digital Signal Processing, pp189-195, Florence, Italy, Aug/Sept 1978.

B. S. Atal, "Predictive coding of speech at low bit rates." IEEE Transactions on Commun., Vol COM-30, pp601-614, April 1982.

B. S. Atal and S. L. Hanauer, "Speech analysis and synthesis by Linear Prediction of the speech waveform." Journal of the Acoustic Society of America, Vol 50, pp637-655, Aug 1971.

B. S. Atal and R. Remde, "A new model of LPC excitation for producing natural sounding speech." IEEE International Conference on Acoustics Speech and Signal Processing, pp614-617, Paris 1982.

B. S. Atal and M. R. Shroeder, "Linear prediction analysis of speech based on a pole-zero representation." Journal of Acoustic Society of America, Vol 64 No 5, pp1310-1318, Nov 1978.

T. P. Barnwell, "Windowless techniques for LPC analysis." IEEE Transactions on Acoustics Speech and Signal Processing, Vol ASSP-28, pp421-427, August 1980.

T. Berger, "Rate Distortion Theory, a mathematical basis for Data Compression." Prentice-Hall Englewood Cliffs, N.J. 1971

M. Berouti and J. Makhoul, "Improved techniques for adaptive predictive coding of speech". IEEE International Communications Conference, pp12A 1.1-12A 1.5, 1978.

P. Billingsley, "Ergodic theory and Information." John Wiley and Sons 1965

L. Brillouin, "Science and information theory" Academic Press, New York, 1965

J. P. Burg, "A new analysis technique for time series data." Proceedings of the NATO Advanced Study Institute on Signal

Processing with emphasis on Underwater Acoustics, Aug 12-23, 1968.

A. Buzo, R. M. Gray, A. H. Gray,Jr. and J. D. Markel, "Speech coding based upon Vector Quantisation." IEEE Transactions on Acoustics Speech and Signal Processing, Vol ASSP-28, pp562-574, Oct 1980.

S. J. Campanella and G. S. Robinson, "A comparison of transformations for digital speech processing." IEEE Transactions on Commun. Tech. Vol COM-19, pp1045-1049 Dec. 1971.

J. C. Candy and R. H. Bosworth, "Methods of designing differential quantisers based on a subjective evaluation of edge busyness." Bell Systems Technical Journal, Vol 51, pp1495-1516, Sept 1972.

E. C. Cherry, "On human communication: a review, a survey and a criticism." 3rd Ed., MIT press 1978.

E. C. Cherry, M. P. Barton and M. H. Kubba, "An experimental study of the possible bandwidth compression of a visual image signal." IEEE Proceedings, no 51, p1507-1517, Nov 1963.

D. J. Conner, R. F. W. Pease and W. G. Scholes, "Television coding using 2-dimensional spacial prediction." Bell Systems Technical Journal, Vol 50, pp1049-1061, March 1971.

R. E. Crochiere, "On the design of Sub-band coders for low bit rate speech communications." Bell Systems Technical Journal, Vol 56, pp747-770, May/June 1977.

R. E. Crochiere, R. V. Cox and J. D. Johnston, "Real time speech coding." IEEE Transactions on Commun., Vol COM-30, pp621-633, April 1982.

R. E. Crochiere, S. A. Weber and J. L. Flanagan, "Digital Coding of

Speech in Subbands." Bell Systems Technical Journal, Vol 55, pp1064-1085, Dec 1976.

P. Cumminsky, N. S. Jayant and J. L. Flanagan, "Adaptive quantisation in differential PCM coding of Speech". Bell Systems Technical Journal, pp1119-1145, 1973.

E. E. David, M. R. Shroeder, B. F. Logan and A. J. Prestigiacomo, "Voice Excited Vocoders for practical speech bandwidth reduction." International Symposium on Info. Theory, Brussels-August 1962, ppS101-S105.

L. D. Davisson, "Universal Noiseless Coding." IEEE Transactions on Info Theory, Vol IT-19 no 6, pp783-745, Nov 1973.

F. DeJager, "Delta Modulation, a method for PCM transmission 1-unit code." Philips Research Reports, Dec 1952, pp442-466.

J. J. Dubnowski and R. E. Crochiere, "Variable rate coding of speech." Bell Systems Technical Journal, March 1979, pp577-600.

R. L. Dobrushin, "General formulation of Shannon's main theorem in Information Theory." Mathematical Society of America Translations, Vol 33, Series 2, pp323-438, 1963.

J. L. Doob, "Stochastic processes." John Wiley and Sons Inc. 1953.

P. Elias, "Predictive coding." IRE Transactions on Info. Theory, Vol IT-1, pp16-33, March 1970.

D. Esteban and C. Galland, "32kB/s CCITT compatible split band coding scheme." IEEE International Conf. on Acoustics Speech and Signal Processing, Tulsa-Oklahoma, pp320-325, April 1980.

R. M. Fano, "Transmission of information." MIT Press, Cambridge, Mass. and John Wiley, New York, 1961.

G. Fant, "Acoustic theory of Speech production." Mouton and Co., 'S-Gravenhage, 1960.

H. G. Fehn and P. Noll, "Tree and Trellis coding of speech and stationary speech like signals". IEEE ICASSP, Denver 1980, pp547-551.

H. G. Fehn and P. Noll, "Multi-path Search coding of Stationary signal with applications to Speech". IEEE Transaction on Commun., Vol COM-30, no 4, p687-701, April 1982.

T. R. Fischer and R. M. Dicharry, "Vector quantiser design for memoryless Gaussian, Gamma and Laplacian sources." IEEE Transactions on Commun. Vol COM-32, no. 6 pp1065-1069 Sept 1984.

J. L. Flanagan, M. R. Shroeder, B. S. Atal, R. E. Crochiere, N. S. Jayant, J. M. Tribolet, "Speech coding." IEEE Transactions on Commun., Vol COM-27, pp710-739. April 1979.

E. Forgey, "Cluster analysis of multivariate data: Efficiency vs. interpretability of classifications." (abstract), Biometrics, vol 21 p768, 1965.

G. D. Forney, "The Viterbi algorithm." Proceedings IEEE, Vol. 61 no3, pp268-275, March 1973.

J. R. Fram and E. S. Deutsch, "On the quantitative evaluation of Edge Detection schemes and their comparison with human performance." IEEE Transactions on Computers, Vol C-24, no 6, pp616-628, June 1975.

E. D. Frangoulis, "Orthogonal transform methods for speech coding." PhD Thesis, Imperial College, University of London, 1978.

J. W. Fussel, "The Karhunen-Loeve transform applied to the Log-Area-Ratios of a Linear Predictive Speech coder.", Proceedings of the International Conference on Acoustics Speech and Signal Processing, pp36-39 Denver 1980.

R. G. Galagher, "Information Theory and Reliable Communication." John Wiley and Sons, New York, 1968.

A. Gercho, "On the structure of vector quantisers." IEEE Transactions on Info. Theory, Vol IT-28, no 2, pp157-166, March 1982.

A. Gersho and B. Ramamurthi, "Image coding using vector quantisation." Proceedings IEEE International Conference on Acoustics Speech and Signal Processing. Paris-March 1982. pp428-431.

J. I. Gimlett, "Use of Activity Classes in adaptive transform image coding." IEEE Transactions on Commun. Vol COM-23, pp785-786, July 1975.

R. C. Gonzalez and P. A. Wintz, "Digital image processing." Addison-Welsey 1977.

W. M. Goodall, "Telephony by Pulse Code Modulation". Bell Systems Technical Journal, Vol. 26, p395, July 1947.

W. M. Goodall, "Television by Pulse Code Modulation." Bell Systems Technical Journal, Vol-30, pp33-49, Jan 1951.

D. N. Graham, "Image transmission by 2-dimensional contour coding."

IEEE Proceedings Vol 55, p336-346, March 1967.

C. Grauel, "Sub-band coding with adaptive bit allocation." Signal Processing, a EURASIP journal, Vol 2 No 1, pp23-30, Jan 1980.

R. M. Gray, "Time invariant Trellis coding of Ergodic discrete-time sources with a fidelity criterion", IEEE Transactions on Info. Theory, Vol IT-23, no 1, pp71-74, Jan 1977.

R. M. Gray, A. Buzo, A. H. Gray and Y. Matsuyama, "Distortion measures for Speech Processing." IEEE Transactions on Acoustics Speech and Signal Processing, Vol ASSP-28, no 4, pp367-376, August 1980.

R. M. Gray, A. H. Gray, G. Robodello and J. E. Shore, "Rate-distortion speech coding with a minimum discrimination information distortion measure." IEEE Transactions on Info. Theory, Vol IT-27, pp708-721 1981.

A. H. Gray and J. D. Markel, "Quantisation and Bit allocation in Speech processing." IEEE Transactions on Acoustics Speech and Signal Processing, Vol ASSP-24, no 6, pp459-473, Dec 1976.

J. N. Gupta and P. A. Wintz, "A boundary finding algorithm and its applications." IEEE Transactions on Circuits and Systems, Vol CAS-22, pp351-362, April 1975.

A. Habibi, "A survey of adaptive image coding techniques." IEEE Transactions on Commun., Vol COM-25, pp1275-1284, Nov 1977.

P. R. Halmos, "Lectures in Ergodic Theory." Chelsea Publishing Company, New York, 1956.

C.W. Harrison, "Experiments with linear prediction in television."

Bell Systems Technical Journal, Vol 31, pp764-783, July 1952.

M. H. L. Heckler and N. Guttman, "Survey of methods for measuring speech quality." Journal of Audio Engineering Society, Vol 15, pp400-403, 1967.

J. M. Heinz and K. N. Stevens, "On the properties of voiceless fricative constants." Journal of the Acoustic Society of America. Vol 34 pp589.

W. H. Higgins, "A note on the auto-correlation analysis of speech sounds." Journal of the Acoustic Society of America, Vol 26, pp790-792 Sept 1954.

J. N. Holmes, "A survey of methods for the encoding of speech signals." Radio and Electronics Engineer, Vol 52 no 6, pp267-277, June 1982.

D. A. Huffman, "A method for the construction of minimum redundancy codes." Proceedings IRE, Vol 40, pp1098-1101, 1952.

A. K. Jain, "Image data compression, a review." IEEE Proceedings, Vol 69, pp349-389, March 1981.

A. K. Jain, "Advances in mathematical models for image processing." IEEE Proceedings, Vol 69, pp502-528, May 1981.

V. K. Jain and R. Hangartner, "Efficient algorithm for multi-pulse LPC analysis of speech." Proceedings IEEE International Conference on Acoustics Speech and Signal Procesing. San-Diego, March 1984, pp1.4.1-1.4.4.

N. S. Jayant, "Adaptive Delta Modulation with a one bit memory." Bell Systems Technical Journal, pp321-342, March 1970.

F. Jelinek, "Tree coding of memoryless discrete time sources with a fidelity criterion." IEEE Transactions on Info Theory, Vol IT-15, no 5, pp584-590, Sept 1969.

B. Julesz, " A method of coding TV signals based on Edge Detection." Bell Systems Technical Journal, no 38, pp1001-1020, July 1959.

H. Kitajima, T. Saito and T. Kuroba, "Comparison of the Discrete Cosine and Fourier transforms as possible substitutes for the Karhunen-Loeve transform." Transactions of the IECE of Japan. Vol E-60, no 6, pp279-283, Jan 1977.

A. N. Kolmogorov, "Foundations of the theory of probability." Chelsea Publishing Co., New York, 1950.

H. J. Landau and D. Slepian, "Some experiments in picture processing for bandwidth compression." Bell Systems Technical Journal, Vol 50 pp1525-1540 May/June 1971.

J. O. Limb and C. B. Rubinstein, "On the design of quantisers for DPCM codes: A functional relationship between visibility, probability and masking." IEEE Transactions on Commun., Vol COM-26, pp573-578, May 1978.

Y. Linde, A. Buzo and R. M. Gray "An algorithm for vector quantiser design." IEEE Transactions on Commun., Vol COM-28, pp84-95, Jan 1980.

D. T. Magill, "Adaptive speech compression for packet communication systems." Proceedings Nat., Telecommun., Conf., pp29D1-29D5 Nov 1973.

P. A. Maragos, R. W. Shafer and R. M. Mersereau, "Two-dimensional linear prediction and its application to adaptive predictive coding of images. "IEEE Transactions on Acoustics, speech and signal processing Vol-ASSP-32 no 6, pp1213-1229 Dec 1984.

Y. Matsuyama and R.M. Gray, "Universal Tree encoding for speech". IEEE Transactions on Info. Theory, Vol. IT-27, no 1, pp31-40, Jan 1981.

Y. Matsuyama and R. M. Gray, "Voice coding and Tree encoding systems based upon Inverse Filter matching." IEEE Transactions on Commun., Vol COM-30, pp711-720, April 1982.

B. McMillan, "The basic theorems of Information Theory." Annals of Mathematics and Statistics, Vol 24, pp196-219, June 1953.

A. N. Netravali, "On quantisers for the DPCM coding of picture signals." IEEE Transactions on Info. Theory, Vol IT-23, no 3, pp360-370, May 1977.

A. Netravali and J. O. Limb, "Picture coding: A survey." IEEE Proceedings, Vol 68 pp366-406, March 1980.

K. N. Ngan, "Adapative transform coding of video signal." IEE Proceedings Part F, Vol 129 No 1, pp28-40, Feb 1982.

B. M. Oliver, J. R. Pierce and C. E. Shannon, "The philosophy of PCM". Proceedings of IRE, pp1324-1331, Nov 1948.

P. E. Papamichalis and T. P. Barnwell, "A Dynamic Programming approach to variable rate speech transmission." Proceedings of International Conference on Acoustics Speech and Signal Processing, pp28-31, Denver 1980.

P. E. Papamichalis and T. P. Barnwell, "Variable rate speech compression by encoding subsets of the PARCOR coefficients." IEEE Transactions on Acoustics Speech and Signal Processing. Vol ASSP-31, no 3 pp 706-713 June 1983.

A. Parker, S. T. Alexander and H. J. Trussel, "Low bit rate speech enhancement using a new method of multiple impulse excitation." Proceedings IEEE International Conference on Acoustics Speech and Signal Processing, San-Diego, March 1984, pp1.5.1-1.5.4.

A. Perez, "Extensions of the Shannon-McMillan limit theorem to more general stochastic processes." Transactions of the third Prague Conference on Information Theory, Statistical Decision Functions and Random Processes, pp545-574, June 1962.

J. R. Pierce and E. E. David, "Man's world of sound." Doubleday and Co. New York, 1958.

W. K. Pratt, P. J. Capitant, W. H. Chen, E. R. Hamilton and R. H. Wallis, "Combined symbol matching Facsimile data compression system." IEEE Proceedings, Vol 68, no 7,pp786-796, July 1980.

W. K. Pratt, W. H. Chen and L. R. Welch, "Slant transform image coding." IEEE Transactions on Commun., Vol COM-22, pp1074-1093, August 1974.

L. R. Rabiner, M. J. Cheng, A. E. Rosenberg and C. A. McGonegal, "A comparative performance study of several pitch detection algorithms." IEEE Transactions on Acoustics, Speech and Signal Processing, Vol ASSP-24, no 5 pp 399-418, Oct 1976.

K. R. Rao, M. A. Narasimhan and W. J. Gorzinski, "Processing image data by hybrid techniques." IEEE Transactions on Systems Man and Cybernetics, Vol SMC-7 no 10, October 1977.

K. R. Rao, M. A. Narasimhan and K. Revuluri, "Image data processing by Hadamard-Haar transform." IEEE Transactions on Computing, Vol C-24, pp888-896, Sept 1975.

F. M. Reza, "An introduction to Information Theory." McGraw Hill, 1961.

A. H. Robinson and E. C. Cherry, " Results of a prototype television bandwidth compression scheme." IEEE Proceedings, Vol-55, pp356-364, March 1967.

G. Robodello, R. M. Gray and J. P. Burg, "A multirate voice digitiser based upon Vector Quantisation." IEEE Transactions on Commun., Vol COM-30, pp721-727, April 1982.

J. E. Rose, J. F. Brugge, D. F. Anderson, J. E. Hind, "Pattern activity in single auditory nerve fibres of a Squirrel Monkey." Hearing Mechanisms in Vertibrates, pp144, Chirchill-London.

A. Rozenfeld and A. C. Kac, "Digital picture processing, Vols I and II." Academic Press, New-York, 1982.

C. E. Shannon, "A mathematical theory of Communication, part 1." Bell Systems Technical Journal, Vol 27, pp379-423, 1948.

C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion." IRE National Convention Record, Part 4, pp142-163, 1959.

M. R. Schroeder, "Correlation techniques for speech bandwidth compression." Journal of Audio Engineering Society, Vol 10, pp163-166 1962.

M. R. Schroeder, "Vocoders: analysis and synthesis of speech." IEEE

Proceedings, Vol 54, no 5, pp720-733, May 1966.

M. R. Schroeder and E. E. David, "A vocoder for transmitting 10kc/s speech over a 3.5kc/s channel." Acoustica, Vol 10, pp35-43, 1960.

C. P. Smith, "Voice communications method using pattern matching for data compression." (Abstract from program of 65th meeting of the Acoustic Society of America, New York, May 15-18, 1963) Journal of Acoustic Society of America, Vol 35, p805 1963.

L. C. Stewart, R. M. Gray and Y. Linde, "The design of trellis waveform coders." IEEE Transactions on Commun., Vol COM-30, no 4, pp702-709, April 1982.

M. R. Sumbar, "An efficient Linear Prediction Vocoder." Bell Systems Technical Journal, Vol 54, pp1613-1723, Dec 1975.

H. Tanaka and A. Leon-Garcia, "Efficient Run Length Encodings." IEEE Transactions on Info. Theory, Vol. IT-28, no 6, pp880-890, Nov. 1982.

M. Tasto and P. A. Wintz, "Image coding by adaptive block quantisation." IEEE Transactions on Commun. Vol COM-19 No 6, pp957-971, Dec 1971.

J. M. Tribolet and R. E. Crochiere, "Frequency domain coding of speech." IEEE Transactions on Acoustics Speech and Signal Processing, Vol ASSP-27, pp512-530, October 1979.

J. M. Tribolet, P. Noll, B. McDermot and R. E. Crochiere, "A comparison of four low bit rate waveform coders." Bell Systems Technical Journal, March 1979, pp699-712.

T. J. Ulrich and T. N. Bishop, "Maximum Entropy spectral analysis and auto-regressive decomposition." Res. Geophysics and Space Physics, Vol 13, pp183-200, Feb 1975.

C. K. Un and D. H. Cho, "Hybrid companding Delta Modulation with a variable rate sampling." IEEE Transactions on Commun., COMM-30, no 4, pp593-599, April 1982.

C. K. Un and H. S. Lee, "A study of the comparative performance of Adaptive Delta Modulation systems." IEEE Transactions on Commun., Vol COMM-28, pp96-101, Jan 1980.

V. F. Vanlandingham and J. F. Bogdanski Jr, "An adaptively sampled delta modulator." IEEE ICASSP, Denver 1980, p543-545.

V. R. Viswanathan, A. L. Higgins and W. H. Russel, "Design of a robust Baseband LPC coder for speech transmission over 9.6kB/s nosy channels." IEEE Transactions on Commun., Vol COM-30, pp663-673, April 1982.

V. R. Viswanathan, J. D. Makhoul, R. M. Shwartz and A. W. F. Huggins, "Variable frame rate transmission, A review of methodology and application to narrow band LPC speech coding." IEEE Transactions on Commun., Vol COM-30, pp674-685, April 1982.

A. J. Viterbi and J. K. Omura, "Trellis encoding of memoryless discrete-time sources with a fidelity criterion." IEEE Transactions on Info. Theory, Vol IT-20, pp325-331, 1974.

A. J. Viterbi and J. K. Omura, "Principles of Digital Communication and Coding." McGraw-Hill, 1979.

P. J. Wilson, "Frequency domain coding of speech signals." PhD Thesis, Imperial College, University of London, 1983.

S. G. Wilson and S. Hussain, "Adaptive tree encoding at 8000bits/s with a frequency-weighted error criterion." IEEE Transactions on Commun., Vol COM-27, no 1, pp165-170, Jan 1979.

P. A. Wintz, "Transform picture coding." IEEE Proceedings, Vol 60 no 7, pp809-820, July 1972.

J. K. Yan and D. J. Sakrison, "Encoding of images based upon a two-component source model." IEEE Transactions on Commun., Vol COM-25, pp1315-1322, Nov 1977.

B. Yegnanarayana, "Speech analysis by pole-zero decomposition of short time spectra." Signal Processing, a EURASIP journal, Vol 3 No 1, pp5-17 Jan 1981.

R. Zelinski and P. Noll, "Adaptive transform coding of speech signals." IEEE Transactions on Acoustics, Speech and Signal Preocessing, Vol ASSP-25, pp299-309, Aug 1977.

R. Zelinski and P. Noll, "Approaches to adaptive transform speech coding at low bit rates." IEEE Transactions on Acoustics Speech and Signal Processing, Vol ASSP-27, pp89-95, Feb 1979.

W. Zschunke, "DPCM picture coding with adaptive prediction." IEEE Transactions on Commun., Vol COM-25, no 11, pp1295-1302, Nov 1977.

J.B. Anderson and J.B. Bodie, "Tree encoding of speech." IEEE Transactions on Info Theory, Vol IT-21, July 1975, pp379-387.

Y. Huang and P.M. Shulthiess, "Block quantisation of correlated Gaussian random variables." IEEE Transactions on Commun., Sept 1963, pp289-296.

N.S. Jayant, "Digital coding of speech waveform; PCM, DPCM and DM quantisers." IEEE Proceedings, Vol-62, no5, May 1974 pp611-631.

N.S. Jayant and S.A. Christiansen, "Tree encoding of speech using the (M-L]-Algorithm and adaptive quantisation." IEEE Transactions on Commun., Vol COMM-26, Sept 1978, pp1376-1379.

N.S.Jayant and P.Noll. "Digital coding of waveforms." Prentice-Hall Inc. Englewood Cliffs, New Jersey 1984.

F.Jelinek and J.B.Anderson, "Instrumentable Tree encoding for information sources." IEEE Transactions on Info. Theory, Vol IT-17, Jan 1977, pp118-119.

Y. Linde and R.M. Gray, "A fake process approach to data compression." IEEE Transactions on Commun., Vol COMM-26, June 1978 pp840-846.

S.P. Lloyd, "Least-square quantisation in PCM." IEEE Transactions on Info Theory, Vol IT-28, no2 March 1982, pp129-137.

J. Max, "Quantising for minimum distortion." IRE Transactions on Info. Theory, Vol IT-6, March 1960, pp7-12.

J.W. Modestino and V. Bhaskaran, "Robust 2-dimensional Tree encoding of images." IEEE Transactions on Commun. Vol COM-29, no12, Dec 1981, pp1786-1798.

W.K. Pratt, "Digital Image Preocessing." John Wiley 1978.

A.H. Reeves, French Patent no852183, 3rd Oct 1938.

B. Steele, "Delta modulation systems" John Wiley, 1975.

S.G. Wilson and S. Hussain, "Adaptive tree coding of speech at 8000bits/sec with a frequency weighted error criterion." IEEE Transactions on Commun. Vol COM-27 no1, Jan 1979, pp165-170.