Addendum

Page 9, Table 2.1: Quality scale applies to MOS, Impairment scale to Degradation Category Rating (DCR) test[106].

Page 19, Section 2.4.1, Line 12: 'Residual Pulse Excitation' should read 'Regular Pulse Excitation'.

Page 20, Line 20: 'Viterbi coding' should read 'Viterbi decoding'.

Chapter Three, Major Notation Summary:

$Q(.)$	Quantizer Function
g_k	ADPCM Quantizer Gain
Y_k	Quantizer Output Level
S_k	Input Speech Samples
$\hat{S}_{k\|k-1}$	One-step-ahead Prediction
$\hat{S}_{k\|k}$	Reconstructed Speech Samples
z^{-1}	Unit Delay Backward Shift Operator
$A(z^{-1})$	Polynomial in z^{-1} for Predictor Poles
$B(z^{-1})$	Polynomial in z^{-1} for Predictor Zeros

Equations (4.3), (4.10), (4.11), (4.16), (6.1), and (7.4): γ_1 and γ_2 are replaced by γ_1^{-1} and γ_2^{-1} respectively.

Page 109, Figures 5.7 and 5.8: 'Quantization Noise Level' should read 'Signal-to-Quantization Noise Ratio'.

Page 217, Appendix A: Title should read 'A Brief Overview of Information Theory and Entropy Coding'.

Page 233, Appendix D: Title should read 'A Brief Overview of Information Theory and Channel Coding'.

New Techniques in Signal Coding

Craig Robert Watkins
BSc. BE. (QLD) PEng

December 1994

A thesis submitted for the degree of Doctor of Philosophy
of the Australian National University

Department of Systems Engineering
Research School of Information Sciences and Engineering
The Australian National University

Paul,

You leave behind many in grief, both family and friends. However, we can all take comfort in the certainty that our lives are enriched through the times that each of us shared with you. Your characteristic vitality and flair has left its mark on all those you touched. In years your life has been tragically short, but your achievements are testimony to a full and solid innings. May peace be with you.

1994.

Declaration

These doctoral studies were conducted with supervision from Dr. Salvatore (Sam) Crisafulli and Dr. Robert (Bob) Bitmead of the Australian National University, and Dr. Donald McLean of the CSIRO Division of Radiophysics.

The work contained in this thesis, except where explicitly stated, is original research performed by the author under the guidance of Sam, Bob, and Don. This work has not been submitted for a degree at any other university or institution.

Chapter 8 of this thesis concerns research work undertaken while at AT&T Bell Laboratories, Murray Hill, New Jersey. This work was performed under the supervision of Dr. Juin-Hwey (Raymond) Chen in the Speech Coding Department headed by Dr. Rich Cox. The work forms the basis of AT&T patent applications and an AT&T contribution to the ITU-T (International Telecommunication Union – Telecommunications Standardization Sector).

A significant proportion of the research performed for this thesis is contained in patent applications, has been published, or has been submitted to conferences and journals, as listed below.

Standard Contributions and Patent Applications:

[S1] AT&T, "G.728 Decoder Modifications for Frame Erasure Concealment", Contribution to ITU-T SG XV/Q.5, March 1994.

[P1] J.-H. Chen and C. R. Watkins, "Linear Prediction Coefficient Generation During Frame Erasure Or Packet Loss", patent application filed on 14th March 1994.

[P2] R. R. Bitmead, S. Crisafulli and C. R. Watkins, "ADPCM Signal Encoding/Decoding System and Method", two provisional patents lodged on 15th April 1994.

[P4] J.-H. Chen and C. R. Watkins, "Frame Erasure Or Packet Loss Compensation Method", patent application filed on 14th October 1994.

Journal Papers:

[J1] C. R. Watkins, R. R. Bitmead and S. Crisafulli, "Destabilization Effects of Adaptive Quantization in ADPCM", *To appear IEEE Transactions on Speech and Audio Processing*, March 1995.

[J2] C. R. Watkins, S. Crisafulli and R. R. Bitmead, "Practical Kalman Filtering for Speech Coding Applications", *Submitted to IEEE Transactions on Speech and Audio Processing.*

[J3] C. R. Watkins, S. Crisafulli and R. R. Bitmead, "An Entropy Coded ADPCM Speech Coding System for Variable Bit Rate Applications", *Submitted to IEEE Transactions on Speech and Audio Processing.*

[J4] C. R. Watkins, J.-H. Chen, "Improving 16 kb/s G.728 LD-CELP Speech Coder for Frame Erasure and Packet Loss", *in preparation for journal submission.*

Conference Papers:

[C1] C. R. Watkins, S. Crisafulli and R. R. Bitmead, "Reduced Complexity Kalman Filtering for Signal Coding", *International Workshop on Intelligent Signal Processing and Communication Systems*, Sendai, October 1993.

[C2] C. R. Watkins, S. Crisafulli, R. R. Bitmead and R. J. Orsi, "Variable Bit Rate ADPCM via Arithmetic Coding", *IEEE International Conference on Acoustics, Speech and Signal Processing*, Adelaide, April 1994.

[C3] C. R. Watkins and J.-H. Chen, "Improving 16 kb/s G.728 LD-CELP Speech Coder for Frame Erasure Channels", *To appear at IEEE International Conference on Acoustics, Speech and Signal Processing*, Detroit, May 1995.

[C4] C. R. Watkins, S. Crisafulli, and R. R. Bitmead, "A Variable Bit Rate Entropy Coded ADPCM System", *In preparation for submission to 1995 IEEE Speech Coding Workshop*, Annapolis, September 1995.

Canberra, December 1994.

Craig Watkins

Craig Robert Watkins,
Department of Systems Engineering,
Research School of Information Sciences and Engineering,
The Australian National University,
Canberra ACT 0200, AUSTRALIA.

Acknowledgements

I would sincerely like to thank my supervisor, Dr. Bob Bitmead, for his patience and input on both technical and non-technical issues. Dr. Sam Crisafulli, as an advisor and co-supervisor, deserves special thanks for being willing to listen, and provide feedback, to all my crazy ideas over the last three years. Dr. Don McLean has also provided useful input, and I thank him for his more recent attempts at commercialization of some of the work contained in this thesis. On the issue of commercialization, I must not forget the efforts of a number of engineers involved with developmental issues, Dr. Johnson Agbinya, Dr. Rong-Yu Chao, and Haley Jones. Thanks also go to two undergraduate engineering vacation experience students, Robert Orsi, and Raymond Chan, for their work, and the supervision experience they provided.

Dr. Raymond Chen, and Dr. Rich Cox from AT&T Bell Laboratories, Murray Hill, NJ, also deserve many thanks. I am grateful for the opportunity to work at Murray Hill for a 12 week period, and am even more grateful for the opportunity to work on such an interesting, and relevant project. Thanks must also go to Dr. Peter Kroon from Bell Labs, for the interesting discussions we had on occasion.

The three month stay at Bell Labs was part of a six month overseas trip to visit several members of the speech coding community in both university, and industrial research groups. The groups and people visited are far too numerous to mention here. However, I am grateful to everyone I visited for being willing to host me. The funding for these visits came from the International Business Unit of Telstra (OTC Australia), to whom I am most certainly grateful. I believe that the time I spent visiting other researchers was some of the most productive time I have spent during my PhD studies. I was both able to obtain input specific to my speech coding PhD research, and broad engineering knowledge on speech coding, and telecommunications systems. It is this broad perspective that is extremely difficult to obtain through normal university study, whilst being of vital importance to the selection and pursuit of research areas for maximal return. I sincerely hope that even in tough economic times Telstra remains convinced of the long term benefits of these awards, both to the company and to the country as a whole.

I acknowledge the Australian Telecommunications and Electronics Research Board for their postgraduate scholarships, and the Australian Government for their funding of the Cooperative Research Centre (CRC) program and the provision of APRA (Australian Postgraduate Research Award) scholarships. During my time as a PhD student

I have benefited through involvement with the Cooperative Research Centre for Robust and Adaptive Systems. The CRC program is designed to ensure that the research being performed at universities in Australia has every chance of bringing benefits to the Australian economy. As engineering research is concerned with solutions to practical problems, the CRC program provides significant advantages to engineering PhD students and incentives for academics to refocus research towards goals of some benefit to the country. The aim of having academics moving from the sheltered university world into the real world is not without its controversy. However, there are significant advantages for both the universities, and industries involved. Also, academics can not justly claim to be engineers without producing work that might eventually be useful.

Engineering involves wealth creation, rather than wealth shuffling, and support and promotion of engineering should take a high profile. The CRC program is a start, but as with most government programs, criticism can easily be raised that the public sector is not doing enough, and it is too late and ineffective. The private sector also carries part of the burden of blame, as management boards, largely comprised of people with law and economics backgrounds, are slow to accept the need for input from engineering and science (particularly in Australia). A trend towards science and engineering people on company boards is emerging slowly, and those companies that correctly take the initiative will be well placed for future years.

Moving away from the political issues, there are many other people to whom I am grateful. I have had occasion throughout my PhD studies to meet and discuss various issues with a large number of academics, research engineers, and students. From all interactions I was inspired and able to learn. Although no one interaction deserves special mention, the sum total of these contacts can not be underestimated.

Last, but not least, I would like to thank my family for their support, moral and financial. Without both of these I would never even have considered PhD study. With family support the sacrifices necessary for PhD study can appear worthwhile.

I believe I have learned many new, interesting, and useful, things over the course of my PhD studies. Most importantly, the last three years have been a time of great inspiration. Thanks are deserved by every person who has been a part of this. Some know who they are, but many do not.

Abstract

Speech Coding, or the digital representation of speech for communications purposes, consists of many different approaches and applications. Within this thesis some analysis of the very popular ADPCM (Adaptive Differential Pulse Code Modulation) and CELP (Code Excited Linear Prediction) speech coding systems is undertaken. These are traditionally viewed as schemes operating in different regions of the speech coding spectrum. We attempt to show the boundaries between the two different approaches are not black and white, and with techniques such as Kalman Filtering, and variable bit rate coding, the schemes lose a significant proportion of their individual identities.

There are four general issues that need to be addressed by any speech coding system:

- Output speech performance.
- Coder output bit rate.
- Computational complexity.
- Delay introduced to overall system.

Other issues that are very important are: robustness to bit and frame transmission errors; robustness to background noise such as car and babble noise; adequate coding performance for other signals such as music and voice-band data; and performance for multiple talkers and conference calls.

Within this thesis, a number of topics are covered. The first of these considers a theoretical analysis of error recovery and stability trade-offs. We then consider a variable bit rate ADPCM coder which is able to provide good speech quality performance for moderate computational complexity. Kalman filtering techniques are observed to provide significant subjective performance improvement to the variable rate ADPCM system introduced, and reduced complexity approaches to obtaining this benefit are important.

The topic of robustness to frame erasure errors in LD-CELP is also considered within this thesis. Here it is found that minor modifications to the LD-CELP decoder are capable of providing very good frame erasure performance at up to 3% error rates. Some further encoder modifications are useful for providing good performance at very high frame erasure rates such as 10%.

Topics covered in this thesis range from analytical, and quite theoretical, to those

more practical, and applications oriented. However, the overall philosophy of the thesis is to obtain a better understanding of speech coding techniques in the spectrum between ADPCM and CELP.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Thesis Motivation

This PhD thesis addresses a number of useful techniques in signal coding, focusing on speech coding applications. The topics addressed range from the highly practical (and immediately applicable) to the more theoretical (and general) in nature. The thesis also contains a significant element of 'blue-sky' philosophy to complement the basic research results. Except for perhaps with the LD-CELP frame erasure work in Chapter 8, we do not claim to present fully functional speech coding systems within this thesis. Rather, general techniques are investigated and simulations performed where beneficial for 'proof of concept'.

Signal coding is that part of a communications system that represents the incoming signal in a form (often digital) for efficient transmission over a channel (or for storage). The efficiency requirement usually implies high decoded signal quality for a small number of transmitted bits. Other common considerations are implementation complexity (unit expense), and coding delay. Significant engineering trade-offs exist in the design of such systems, with efficient practical solutions being heavily applications dependent. Towards the end of the thesis, we also begin to consider the inclusion of recovery from transmission errors into the 'efficiency' requirement.

In order to meet increased user transmission demand on communications networks, the options of both more bandwidth usage (increased network infrastructure), and better signal coding (improved terminal equipment) must be considered. For situations where bandwidth is limited (such as in radio communications), or very expensive to install (such as for some underground cable situations), the use of improved signal coding is the economically superior option.

The remains of this brief introductory chapter includes a short discussion of research philosophy, a thesis overview, and a summary of original research contributions included in the thesis. Chapter 2 augments this chapter by providing a background overview of speech coding systems, including coverage of a number of the significant speech coding standards.

1.2 Research Philosophy

The pursuit of a PhD is an opportunity to learn how to perform research. As research always involves limited resources to solve problems (engineering research in particular), an extremely important part of learning how to perform research is obtaining as large a background knowledge in the area as possible. This background knowledge assists in problem selection, and some elements of a background discussion nature are included in this thesis for completeness and motivational reasons.

As a firm believer that a PhD thesis is more than just a collection of technical papers, there is also a certain element of a philosophical nature within this thesis. Largely this is evidenced by the further research discussion sections in some of the chapters considering a number of quite 'blue-sky' possibilities. Chapter 10, on applications of the variable bit rate ADPCM system introduced in Chapter 4, would also fall into this category of the more philosophical in nature.

Having said this, however, the first several chapters in this thesis are on fairly concrete signal coding sub-problems. This work has been published, or at least submitted to technical journals and conferences. A number of later chapters discuss research results that do not contain enough depth for separate publication. Chapter 7 includes inconclusive results, and Chapters 9 and 10 contain discussions in preference to concrete research results. However, it is believed that these chapters form a significant and vital part of this thesis, by assisting in tying all the chapters together, and rounding out (and motivating) the work.

1.3 Thesis Overview

Chapter Two:

As much of this thesis is concerned fairly directly with issues related to speech coding, some of the major speech coding systems and standards are reviewed in Chapter 2. This should give a broad background perspective on the topic, covering the system overview issues but avoiding many of the details.

For those readers quite familiar with speech coding, this review section will not present anything new, and can safely be ignored. However, it does attempt to present a unified view across many different systems, and for some readers may assist with motivational insights to sections of the work performed throughout this thesis. Note that where explicit understanding of any parts of the systems is required, they are explained in detail in the body of the thesis.

Chapter Three:

Chapter three consists of a theoretical analysis of ADPCM stability. This analysis takes account of the effect of quantizer adaptation on the stability of the ADPCM system. A number of assumptions are required for the mathematical analysis, but the theory is able to shed some light on the design of adaptive quantizers in ADPCM systems.

Chapter Four:

Chapter four presents a variable bit rate ADPCM system that overcomes the stability problems associated with fixed rate ADPCM systems, as well as exploiting the bursty information nature of speech to save bits through entropy coding. The variable rate ADPCM system forms a large part of this thesis, and concerns relating to it are found in a number of chapters.

Chapter Five:

The fifth chapter is concerned with the use of the Kalman filter in speech coding applications. The Kalman filter is a tool that is useful for a number of tasks in speech coding. The major uses are investigated, and some techniques proposed to control the computational requirement of Kalman filtering, whilst still gaining significant performance benefit.

Chapter Six:

Chapter six presents the results of using Kalman filtering within the proposed variable bit rate ADPCM system from Chapter 4. A 'practical' variable rate ADPCM system is presented at the end of this chapter. Judged by informal listening tests, the output speech quality of this variable rate system at an average rate of 12 kbps is equivalent to that of 16 kbps LD-CELP. The computational complexity for this 12 kbps variable rate ADPCM system is significantly less than that of LD-CELP.

Chapter Seven:

Chapter seven looks at the use of the Kalman filter in CELP. In particular, the use of the

Kalman filter is investigated in FS 1016 4.8 kbps CELP and in CCITT Recommendation G.728 16 kbps LD-CELP.

Chapter Eight:

Chapter eight discusses work performed while at AT&T on the frame erasure problem for wireless communications with LD-CELP. Both a bit-stream compatible version with the existing LD-CELP standard, where only decoder changes are made, and some minor modifications to the encoder are investigated. With decoder changes only excellent performance is obtained for a 3% frame erasure rate, and with encoder modifications, good performance is observed at a frame erasure rate as high as 10%.

Chapter Nine:

Chapter nine investigates the resynchronization problem that appears with the use of Arithmetic Coding in the variable bit rate ADPCM system introduced in Chapter 4. For use in practical systems, resynchronization after bit transmission errors is extremely important, and this is a significant hinderance to the application of the variable bit rate ADPCM approach to a wider variety of uses. Requirements for frame based resynchronization are discussed, but due to the large applications dependent nature of the approaches, no simulation results are presented.

Chapter Ten:

The tenth chapter briefly discusses a number of practical considerations with respect to applications of the variable bit rate ADPCM system introduced in Chapter 4. Some of these are immediate applications, whilst some require the solution of a number of associated research and development problems, and others are very much 'blue sky' type applications.

Conclusion:

The thesis finishes with a conclusion chapter in which the major results are summarized, as well as an indication given of possible areas for future research work.

Glossary:

For the reader's convenience, included at the end of this thesis is a glossary of terms and acronyms that are used within the thesis and that occur frequently in the telecommunications industry.

1.4 Summary of Original Contributions

During the course of research for this PhD thesis, a number of original contributions have been made. These are the subject of patent applications, journal, and conference papers, as previously indicated. A brief description of the original work is listed below.

- **ADPCM Stability Analysis:** A theoretical analysis of ADPCM stability is performed that takes account of quantizer adaptation. The result shows that the rate of quantizer step size decrease is closely linked to stability, and this can be viewed as a theoretical justification for the shape of the multiplier curve for the Jayant 'One-Word Memory' adaptive quantizer.

- **ADPCM and Arithmetic Coding:** A novel variable rate ADPCM speech coding system is proposed that overcomes some ADPCM stability problems, and maximises performance measures such as SNR. Immediate applications are in speech coding for storage purposes, where there are some major advantages of the approach over alternatives.

- **Practical Kalman Filtering:** A number of ways that Kalman filtering techniques can be applied to advantage in speech coding applications are shown, while paying careful attention to the issue of computational complexity. The application of Kalman filtering techniques to the above variable rate ADPCM system and other CELP systems is also investigated.

- **LD-CELP Frame Erasure Recovery:** The problem of lost data frames is considered for where the backwards adaptive LD-CELP coder is used in a mobile communications or packet transmission environment. It is seen that with some modifications to the basic approach, LD-CELP can be made extremely robust to frame erasure errors. This work forms the basis of an AT&T contribution to the ITU-T (International Telecommunication Union – Telecommunications Standardization Sector).

Chapter 2

Speech Coding Systems

2.1 Chapter Motivation

This chapter intends to provide a general background to speech coding, primarily for those readers somewhat unfamiliar with the subject area. Many of the well known and emerging standards are covered briefly, with an attempt made to highlight the upper level systems concepts and the key similarities and differences of the various approaches.

Speech coding is the source coding component of a voice communication system such as the wire-based telephone network or mobile communication networks. Other important system components include channel coding, modulation, network access protocols, and frame synchronization. Many other system components exist which effect speech coder design to various extents, and a good general telecommunications engineering background is extremely valuable for speech coding research. Some elements of this telecommunications background are discussed both within this chapter and throughout the thesis as a whole.

Speech coding is a lossy form of source coding, and has four generic performance issues: transmission bit rate; computational complexity; output speech quality; and coding delay. Other issues are also important, such as robustness to transmission errors, multistage encoding/decoding, and accommodation of non-voice signals such as in-band signalling and voiceband modem data.

The requirement for robustness to transmission errors is significant. Theoretically complete separation of source and channel coding is possible, such that the source decoder receives as input the output of the source encoder without error. Practically, however, this is not possible. To obtain this 'perfect' channel coding is too expensive in terms of delay, computation, and transmission overhead to be useful for most practical speech coding applications. Hence the speech decoder will have to accommodate some transmission errors.

Having made this point, it is important to note that the above four issues are the prime concern from the speech source coding perspective. Different speech coding applications dictate various combinations of ranges of these four parameters, and this also provides a convenient way in which to separate speech coding systems into different groups.

Although there are many other possible categories for speech coding, the following break-up has been chosen for the purposes of this chapter: **(1)** Toll Quality Speech Coding – dealing with telephone quality services, principally over wire based networks; **(2)** Mobile Communications – digital mobile telephone systems; **(3)** Low Rate Communications Quality Voice – primarily for military type applications and satellite systems; and **(4)** PCS (Personal Communications Systems) Standardization – emerging applications for the future.

The next section in this chapter discusses the issue of speech quality measurement, and following this the above four speech coding categories are considered in turn. Towards the end of the chapter a brief summary section is provided with some additional general speech coding references given. Finally the remaining chapters in the thesis are outlined, reiterating and augmenting the thesis overview in Chapter 1.

2.2 Speech Quality Measurement

Transmission bit rate, coder computational complexity, and coding delay are all relatively easy to quantify. However, output speech quality is more difficult to measure and deserves special mention.

Almost invariably speech signals transmitted over telecommunications networks are destined for final processing by the human auditory system. (The most notable exception being that of computer speech recognition algorithms.) Hence it is clear that the final test of speech output quality is how it sounds to the person listening. Unfortunately subjective measures like this are not of much direct benefit to the speech coder. For a subjective measurement to be of any value an obvious requirement is that the subjective evaluation process is formalized in some way.

A number of formal approaches to subjective testing are commonly recognised. Probably the most important of these is the Mean Opinion Score (MOS)[106]. MOS testing involves a group of test subjects rating samples of coded speech (encoded and decoded) on a discrete five point scale. The quality and impairment levels of the five point scale are shown in Table 2.1, and test subjects are generally recruited 'randomly', and given very little instruction on their tasks.

MOS Score	Quality Scale	Impairment Scale
1	Unsatisfactory	Objectionable (very Annoying)
2	Poor	Annoying but not Objectionable
3	Fair	Slightly Annoying
4	Good	Perceptible but not Annoying
5	Excellent	Imperceptible

Table 2.1: Mean Opinion Score Five Point Quality/Impairment Scale

Another formal subjective test strategy is the Diagnostic Rhyme Test (DRT)[92]. This is used primarily for low rate (and lower quality) speech coding systems where intelligibility is the primary concern. Listeners are required to pick which one of a rhyming pair of words was played (both words presented visually). In general only the initial consonants of the words are changed, such that for plosives examples might be: BAM, DAM, PAM, TAM, and KAM[92].

For speech coding, objective measures of quality are required, and standard measurements of SNR (Signal to Noise Ratio) and segmental SNR are often used. However, these must be used with caution, as they do not always give a good indication of the subjective quality. In fact, it is not difficult to obtain a sample of speech which when coded by two different approaches gives a large increase in SNR for one of them, but this is subjectively the inferior of the two. When dealing with similar approaches, such as for tuning a coder parameter, SNR measures are usually a reliable indication of relative performance.

Due to the fact that levels of noise are not perceived equally across the speech spectrum, a more useful objective measure is the perceptually weighted SNR. Other important considerations with such measures is to limit the effect of silence periods between speech utterances on the measure. A threshold on the speech activity level is often used in conjunction with an SNR type of measure.

More complicated objective measures exist, such as those in the papers by Wang, Sekey, and Gersho[182], or Gray and Markel[79]. As a general rule, the more complicated objective measures obtain better performance by exploiting more knowledge about speech perception. However, for complexity reasons such objective measures are useful for evaluating speech coding systems, but not generally of substantial use within the speech coders. Variants of SNR and MSE (Mean Square Error) measures are commonly used[175] either explicitly or implicitly within speech coding systems.

2.3 Toll Quality Speech Coding

Toll quality is generally accepted to be the quality equivalent to an ideal analog wired line connection[106]. This is associated with a Mean Opinion Score (MOS) of around 4, indicating good quality output with (just) perceptible but not annoying impairment. The coders in this section are all rated as toll-quality speech coders, but at decreasing bit rates, and correspondingly increasing computational complexities.

Toll quality coders are primarily designed for use with close integration to existing telephony networks. As such, delays introduced in the coding process are a problem due to hybrid echo. In order to eliminate the need for complicated (and expensive) echo cancellers, the coding delay must be tightly controlled. For PCM and ADPCM coding discussed below, there is no significant delay to give concern. However, in order to maintain performance at lower bit rates, it is generally necessary to increase the coding delay.

Remark 2.1 Higher quality speech than that afforded by toll quality systems is also of significant interest within telecommunications networks. The limitation in bandwidth introduced by the analog telephony system produces substantial degradation in speech quality, and 16 kbps (kilo bits per second) coders with a 7 kHz speech bandwidth have recently been proposed for some applications[81].

2.3.1 PCM Coding

Based on telephony quality speech having a bandwidth between 300 Hz and 3.4 kHz, an 8 kHz sampling rate is standard. The PCM (Pulse Code Modulation) standard for digital speech representation, CCITT[1] Recommendation G.711, uses a non-uniform quantizer characteristic to account for the large dynamic range of speech. Each sample is quantized using 8 bits, resulting in a bit rate of 64 kbps[106].

The non-uniform quantizer characteristic can be considered to be obtained through the use of a non-linearity in cascade with a uniform quantizer (companding), as shown

[1]The International Telegraph and Telephone Consultative Committee (CCITT) was recently renamed ITU-T (International Telecommunication Union – Telecommunication Standardization Sector). For convenience this chapter retains the old terminology (CCITT), in effect during the standardization process for the systems discussed.

in Figure 2.1. However, in practice it is often obtained by high rate quantization followed by a digital piece-wise linear approximation of the non-linearity. Companding using both the A-law and μ-law characteristics is discussed in many texts, including the one by Carlson[27].

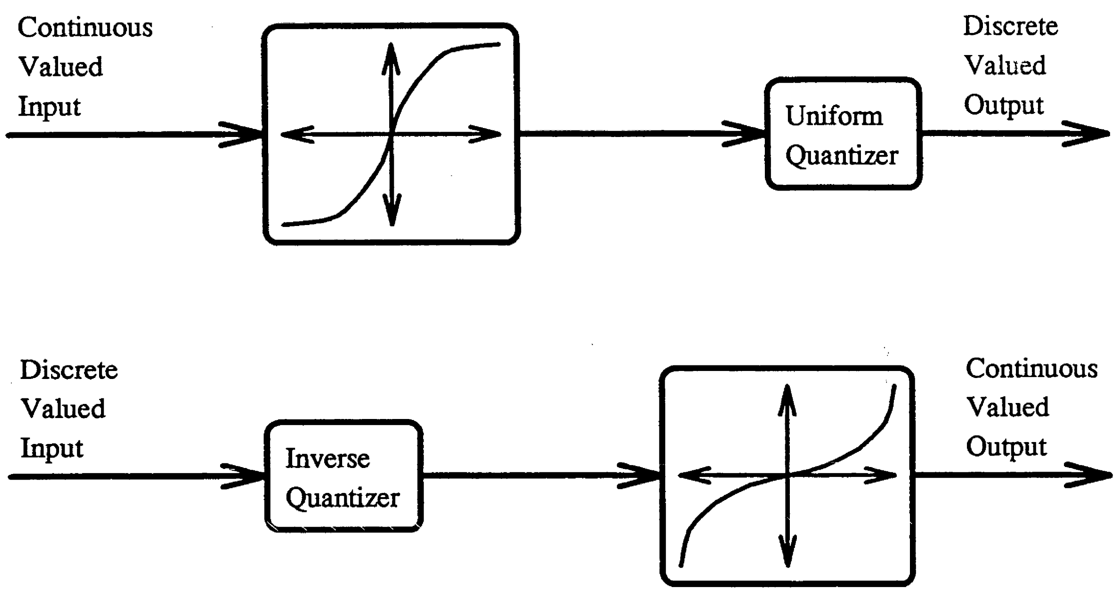

Figure 2.1: PCM Companding Schematic

PCM speech coding is memoryless, and is naturally robust to random bit transmission errors in the sense that a bit error only affects one decoded speech sample.

It is on this rate of 64 kbps that ISDN (Integrated Services Digital Network) telephone lines are based. This standard has received its share of criticism as being too high a basic rate for speech and too low for other applications, but is continuing to be implemented. With fiber optic backbone networks, standards and simplicity of basic structure are important to enable high speed switching. It appears the efficiency of the standard for various applications is of only secondary importance.

2.3.2 ADPCM

CCITT Recommendation G.711 PCM uses non-uniform quantization in an attempt to account for the dynamic range of speech. However, adaptive quantization is able to obtain better performance than the fixed non-uniform quantizer, due to the fact that the variance of the speech signal is only slowly changing with time. Having said this, we note that there is no widely used telecommunications standard based solely on adaptive quantization.

There is, however, an ADPCM (Adaptive Differential Pulse Code Modulation) standard, CCITT Recommendation G.721, based on an adaptive predictor to remove redundancy, and an adaptive quantizer to account for the dynamic range of the prediction residual. The G.721 standard uses a quantizer with 4 bits, thus resulting in a coded speech data stream at 32 kbps.

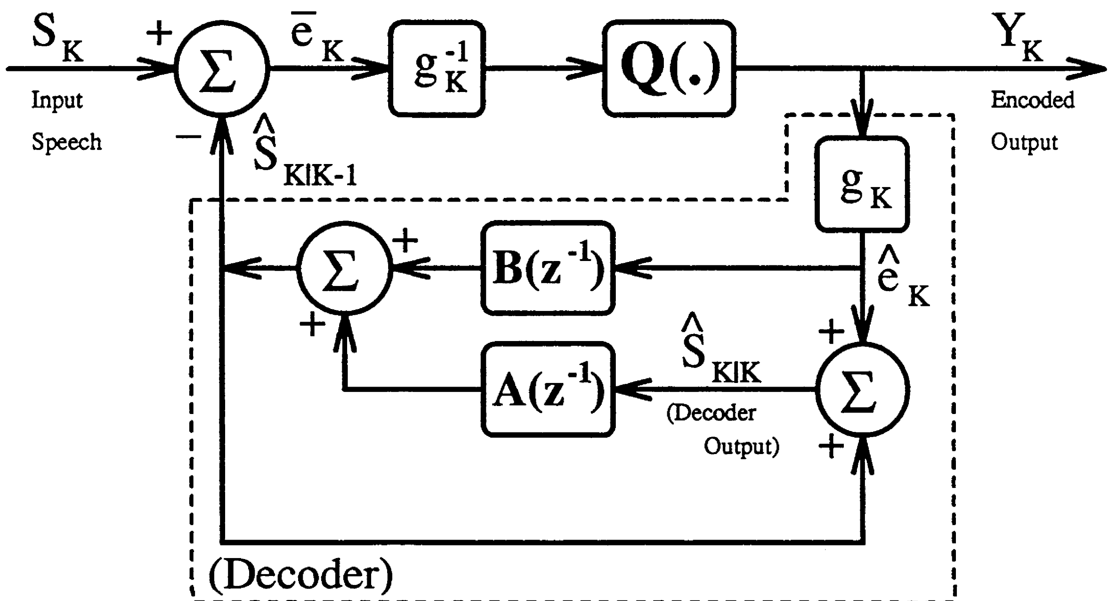

Figure 2.2: CCITT Recommendation G.721 ADPCM Encoder Schematic

Figure 2.2 displays a block diagram of the CCITT Recommendation G.721 32 kbps ADPCM encoder. The quantizer output, Y_k, shown in the diagram is assigned a four bit binary codeword for channel transmission to the decoder. At the decoder the Y_k sequence is reconstructed from the binary transmitted data, and the decoded speech output, $\hat{S}_{k|k}$, is obtained in an identical fashion to in the encoder shown in the figure. Hence a copy of the decoder is inherently contained in the encoder, and the decoder need not be shown explicitly.

Remark 2.2 It is important to note that our definition for the Y_k sequence eliminates the need to display an 'inverse quantizer' in Figure 2.2. This separation of the quantizer (in the general sense) to the two operations of quantization (in the strict sense of limitation to a discrete set of values), and binary codeword allocation is important for work in later chapters. The introduction here is for convenience.

The prediction, $\hat{S}_{k|k-1}$, is formed via the two FIR (Finite Impulse Response) filter structures, $A(z^{-1})$ and $B(z^{-1})$, connected to yield an IIR (Infinite Impulse Response) filter. For G.721 ADPCM $A(z^{-1})$ is of degree two, representing two poles, and $B(z^{-1})$ is of degree six, representing six zeros. Both the quantizer and predictor are adaptive to account for the non-stationary statistics of the input speech. The adaptive quantizer is represented in Figure 2.2 by a fixed quantizer, $\mathbf{Q}(.)$, and an adaptive scaling factor, g_k, which effectively dictates the quantizer step size.

CCITT Recommendation G.721 ADPCM is robust to random bit errors at a rate of 10^{-3}, but produces significantly degraded (and perceptually annoying) speech at an error rate of 10^{-2}. A modified form of the Jayant 'One-Word Memory' adaptive quantizer is used in G.721 ADPCM[106]. From the transmission error perspective, the 'One-Word Memory' is actually an infinite memory, and a decay factor is introduced to help 'forget' previous errors.

The CCITT ADPCM standard is well developed, having been standardized in 1985, and a good general description of ADPCM can be found in the text by Jayant and Noll[106]. An embedded ADPCM approach and separate coders operating at rates from 16 kbps to 40 kbps are included within CCITT standards. Implementation of the G.721 ADPCM standard is now relatively inexpensive due to the existence of ASIC (Application Specific Integrated Circuit) implementations and DSP (Digital Signal Processor) applications notes.

2.3.3 LD-CELP

In 1988 the requirements and objectives for a 16 kbps speech coding standard were approved by the CCITT. The requirements were for a system with effectively the same performance in all areas as 32 kbps ADPCM, and with a very tight bound on the coding delay. At the time many researchers viewed the requirements as practically impossible to meet.

A number of groups persevered with research towards the standard[34, 48, 186], and in May of 1992, the AT&T[2] floating point version of LD-CELP (Low Delay Code Excited Linear Prediction) was officially adopted by the CCITT as the 16 kbps standard G.728[33]. The coder performance is significantly better than that of G.721 ADPCM in some areas. However, this comes at the cost of an extremely high computational requirement.

[2] Formerly American Telephone and Telegraph Company – recently changed to AT&T Corp.

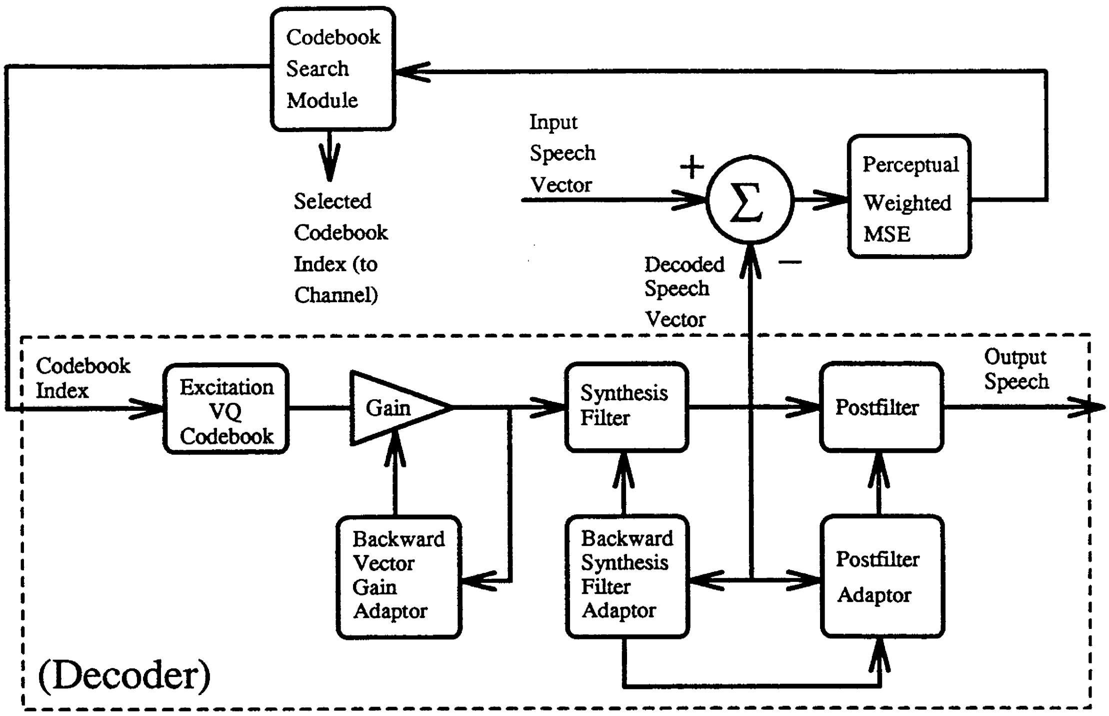

Figure 2.3: CCITT Recommendation G.728 LD-CELP Schematic

The CCITT Recommendation G.728 LD-CELP schematic is shown in Figure 2.3. The diagram minus the postfilter section represents the LD-CELP encoder. As with ADPCM, the decoder is inherently contained in the encoder, but for LD-CELP there is the additional postfilter included at the decoder.

The vector length in LD-CELP is five samples, corresponding to a codebook size of 1024 entries at the 16 kbps code rate. The trained codebook is in gain-shape format, with 128 shape vectors, and 8 gain levels (one gain sign bit, and two gain magnitude bits). Although a number of techniques are used to reduce the amount of computation required for the codebook search, the procedure is an exhaustive search, similar to passing each codebook entry through the synthesis filter to obtain a decoded output vector. A perceptually weighted MSE (mean square error) criterion is used to select the codebook index that gives the least output distortion, and this index is transmitted to the decoder.

To meet the tight low delay constraint, the LD-CELP system is backwards adaptive, with filter parameters obtained from autocorrelation analysis and Levinson-Durbin recursion on windowed previous reconstructed (decoded) speech. In order to maintain current filter parameters, the update procedure is operated once every 20 samples, resulting in a large computation cost. Also to maintain performance in the absence of a pitch predictor, the linear predictor is of a high order (50th order), mainly for the benefit of female speech (with short pitch periods where some element of pitch redundancy can be exploited by the linear predictor).

Recent work on LD-CELP has been targeted at fixed point implementation[35], and the fixed point contribution to the CCITT G.728 standard was made by AT&T in September 1993. This should assist in increasing the applications base for LD-CELP, but the computational requirements are still substantial. A later chapter (Chapter 8) of this thesis discusses some work on the problem of frame erasure that may assist in making LD-CELP a suitable candidate for future PCS (Personal Communications Systems) or FPLMTS (Future Public Land Mobile Telecommunications Systems).

2.3.4 8 kbps Standardization

The terms of reference for production of a toll quality 8 kbps speech coder were formulated by Study Group 15 of the CCITT in 1990, and revised (relaxed) in November 1991. Two coders were proposed as candidates in November 1992. One coder is the CS-CELP (Conjugate Structure CELP) system from NTT (Nippon Telegraph and Telephone)[110, 111, 112, 113], and the other is an ACELP (Algebraic CELP) system from France Telecom/University of Sherbrooke[155, 156]. Both candidate coders met all the CCITT requirements during the qualification phase of the standardization process.

It is generally recognised that for low delay coding some form of backwards adaptation is needed[164, 191], and tight delay requirements make achieving the required rate difficult. However, the 8 kbps standard delay requirements, originally for an encoding frame length of less than 5 ms, were relaxed by the CCITT in November 1991 to less than 16 ms. The other requirements are principally for high quality in error-free conditions, as well as robustness against channel errors, including frame loss and random bit errors.

A block diagram of the NTT CS-CELP system is shown in Figure 2.4. A similar argument to LD-CELP holds with respect to the decoder inherently being contained in the encoder, and hence only one diagram is presented. Key aspects of the CS-CELP proposal are: LSP (Line Spectral Pair) quantization using interframe correlation, preselection of the codebook search, a conjugate structure codebook, and backward

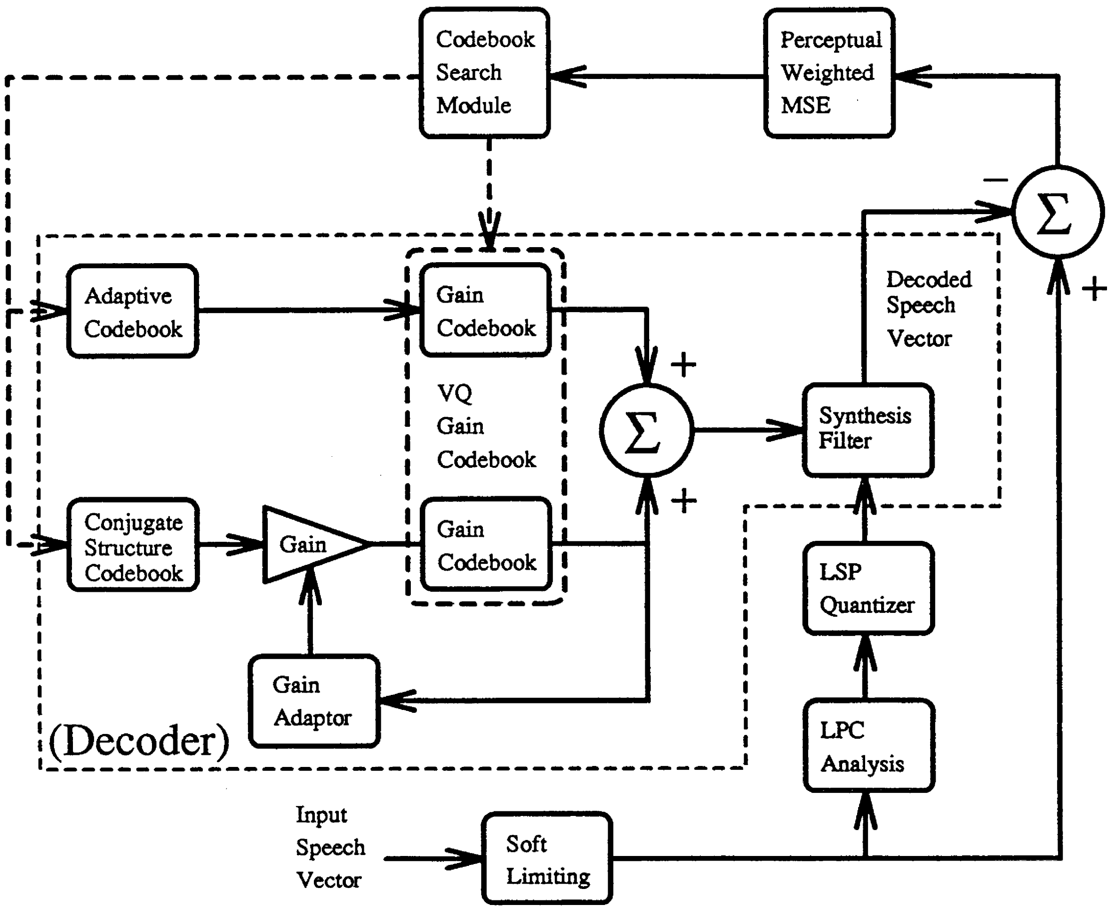

Figure 2.4: NTT 8 kbps CS-CELP Schematic

adaptation of the VQ (Vector Quantization) gain.

The CS-CELP frame length of 10 ms consists of two subframes of 40 samples each (standard 8 kHz sampling frequency). Input speech is soft-limited if above threshold level. LSP parameters are transmitted each frame, and all other parameters are transmitted each sub-frame. The adaptive and fixed-shape codebook are examined sequentially, and the excitation vector is selected that minimises the perceptually weighted error criterion, found through the use of an ARMA (Auto Regressive Moving Average) perceptual weighting filter. The decoded speech is post-filtered at the decoder (not shown) to enhance the perceptual quality.

For robustness considerations, the VQ (Vector Quantization) gain in the NTT system is backwards adaptive, and the pitch parameter is protected by parity. On detection of an error, the pitch from the previous subframe is used.

LPC (Linear Predictive Coder) coefficient analysis is performed on the soft limited input speech. The 10th order line spectral pair prediction residual, after first order MA (Moving Average) prediction, is then quantized via a dual stage VQ, with a split structure at the second stage.

In order to reduce the codebook search complexity for the CS-CELP system, a preselection procedure is used. The codebook search distortion calculation basically involves two separate components, but on the basis of only one of these components it is possible to ignore a number of the codevectors for the next component calculation. It is theoretically possible that this procedure will result in the 'best' codevector being ignored. However, this is very unlikely, and the computational savings are extremely important.

The conjugate structure fixed shape codebook means that an output vector is formed by summing at least two vectors, each stored in a different codebook. The advantage of this approach is improved robustness, reduced memory requirements, and reduced search complexity in combination with preselection.

The ACELP candidate system from France Telecom/University of Sherbrooke is very similar in overall structure to that of the CS-CELP system shown in Figure 2.4. Of course, the details of the two systems differ, and the interested reader should refer to the papers referenced above. The sparse structure of the ACELP codebook allows for efficient searches. In a manner analogous to the preselection used in CS-CELP, the search is performed in a nested fashion, with the requirement that a performance threshold is exceeded on the first stage of the search before proceeding to the next stage. The maximum amount of the codebook search is also limited to 4%, but this results in negligible performance degradation from that of a full search.

Generally both candidate coders obtain the required performance, and although there is always a desire for lower complexity and lower delays, the 8 kbps standard appears unlikely to experience a radical departure from the two systems mentioned above before the final ratification phase. A significant amount of attention is now focussed on the 4 kbps standardization effort.

2.3.5 4 kbps Standardization

In early 1993, the ANSI (American National Standards Institute) T1 committee endorsed the initiation of work towards the 'Terms of Reference' for a future 4 kbps speech coding standard[6] (as discussed in [53]).

It is recognised that there are two major directions that speech coders are likely to take[53]. The first of these is where power consumption requirements are more important than spectral efficiency. This leads to the consideration of low complexity systems in the 8 to 16 kbps range.

The other approach is where spectral efficiency is of the utmost importance. This is mainly due to terminal proliferation, with applications drive from areas such as visual telephony, personal communications, and satellite based personal systems. This leads to the quest for systems in the 2 to 4 kbps range.

Of significance with the preliminary specifications is that there is no requirement for the system to pass voiceband data, which has been a general requirement for higher rate toll quality systems.

Although perhaps optimistic, initial testing is 'scheduled' for 1996, and standardization in 1998. The push for low rate speech coding is still intense, with some current major driving forces from the LEO (Low Earth Orbit) satellite applications (although these applications do not appear to require toll quality).

2.4 Mobile Communications

Mobile Communications has seen rapid growth over the past ten years, and is well placed to experience even greater growth in the next decade[55]. Although the industry sector is not without its share of problems, such as those of a political nature, and those of somewhat more worrying health concerns[62], it appears that the desire for tetherless personal communications is enough to fuel rapid technology advances in the area.

Speech coding is an important component of mobile communication systems. Of course, there are many other communications and network components, as well as related technology issues, an example of which is battery design. Other system aspects that are somewhat less the realm of engineering, but equally important, are device ergonomics, and marketing approach. Naturally this thesis concentrates on speech coding, however a level of general knowledge about mobile communications systems is important, and assumed of the reader. Some general overview papers are [118, 121]. However, many other informative papers exist.

As a general rule current mobile communication systems do not provide toll quality, due mainly to bit rate considerations. Also, coding delays and coder complexity are often high. An important requirement is also the ability to handle burst errors and frame erasures at the source coding level.

2.4.1 GSM (RPE-LTP)

The European Conference of Post and Telecommunications (CEPT) set up the Groupe Speciale Mobile (GSM) in 1982 to investigate a European wireless mobile system. The GSM system, now commonly taken to mean 'Global System for Mobile', was offered commercially in eight countries at the end of 1992, and in April 1993, 32 operators in 22 countries were committed to the system[7]. The system has also been adopted for use by Telecom (Telstra) Australia, and the two new mobile communications carriers in Australia, Optus and Vodafone.

The performance goal of the GSM system was to obtain an average quality no worse than that of the analog mobile systems. Of course, due to the speech coding distortion, the FM clean channel performance is not matched. However, the system is quite robust to error bursts.

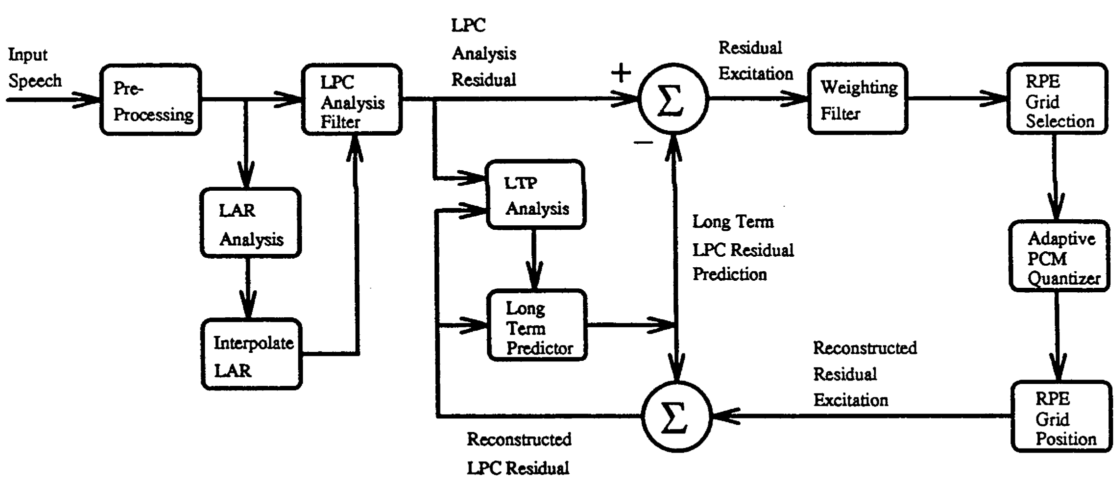

Figure 2.5: RPE-LTP (GSM) Encoder Block Schematic

The GSM system uses a Residual Pulse Excitation coder with Long Term Prediction (RPE-LTP)[120] at 13 kbps. A simplified block schematic of the encoder is presented in Figure 2.5. The RPE-LTP decoder is not shown.

Preprocessing shown in Figure 2.5 involves DC offset compensation, and preemphasis (for numerical considerations). The speech is windowed for each 20 ms frame, and reflection coefficients obtained via Schur recursion (after autocorrelation analysis). The reflection coefficients are converted into the Log Area Ratio (LAR) domain for quantization purposes. The quantized LAR coefficients are then interpolated and converted back to reflection coefficients for use in the LPC analysis filter.

Long Term Prediction (LTP) is used on the LPC analysis residual to attempt to extract long term (pitch) redundancy. The residual excitation is then quantized via the RPE encoding process. RPE grid selection involves downsampling the weighted residual excitation by a factor of 3, and determining which one of four candidate decimation grids (different grid phases) gives rise to the maximum energy. The selected RPE sequence is then quantized via a block adaptive PCM quantizer.

The RPE-LTP decoder uses the transmitted RPE and LTP parameters to obtain the reconstructed excitation signal for the LPC synthesis filter. The synthesis filtering process is performed, and a post-processing deemphasis filter is used to obtain the output decoded speech.

Within the GSM system an additional 9.8 kbps are used for channel coding (over the 13 kbps for speech source coding), with three different levels of protection. The highest level of protection involves the use of a CRC (Cyclic Redundancy Check) code to provide error detection. On the receipt of a frame that has uncorrected bit errors in the 50 most sensitive bits, the frame is discarded, and replaced with the previous frame.

These 50 bits plus the 3 CRC bits and the next 132 most sensitive bits are protected through the use of a rate 1/2 convolutional code. Viterbi coding is thus used at the decoder to correct transmission errors. The remaining 78 bits are left unprotected.

In order to provide robustness to error bursts, frame interleaving and interlacing is used.

As GSM is well developed, there are many references available, covering all aspects of the system, with examples being [38, 83, 135, 166]. To increase system capacity, the standardization process for a half rate coder has been underway for a number of years. Candidate coders are described in the papers [177] and [190], among others.

2.4.2 US and Japanese VSELP Standards

The North American Cellular Telecommunications Industry Association (CTIA) has standardized a Vector Sum Excited Linear Predictive (VSELP) coder for use in digital mobile communications.

The CTIA VSELP[70, 71] system (IS-54) has a speech coding rate of 8 kbps, and

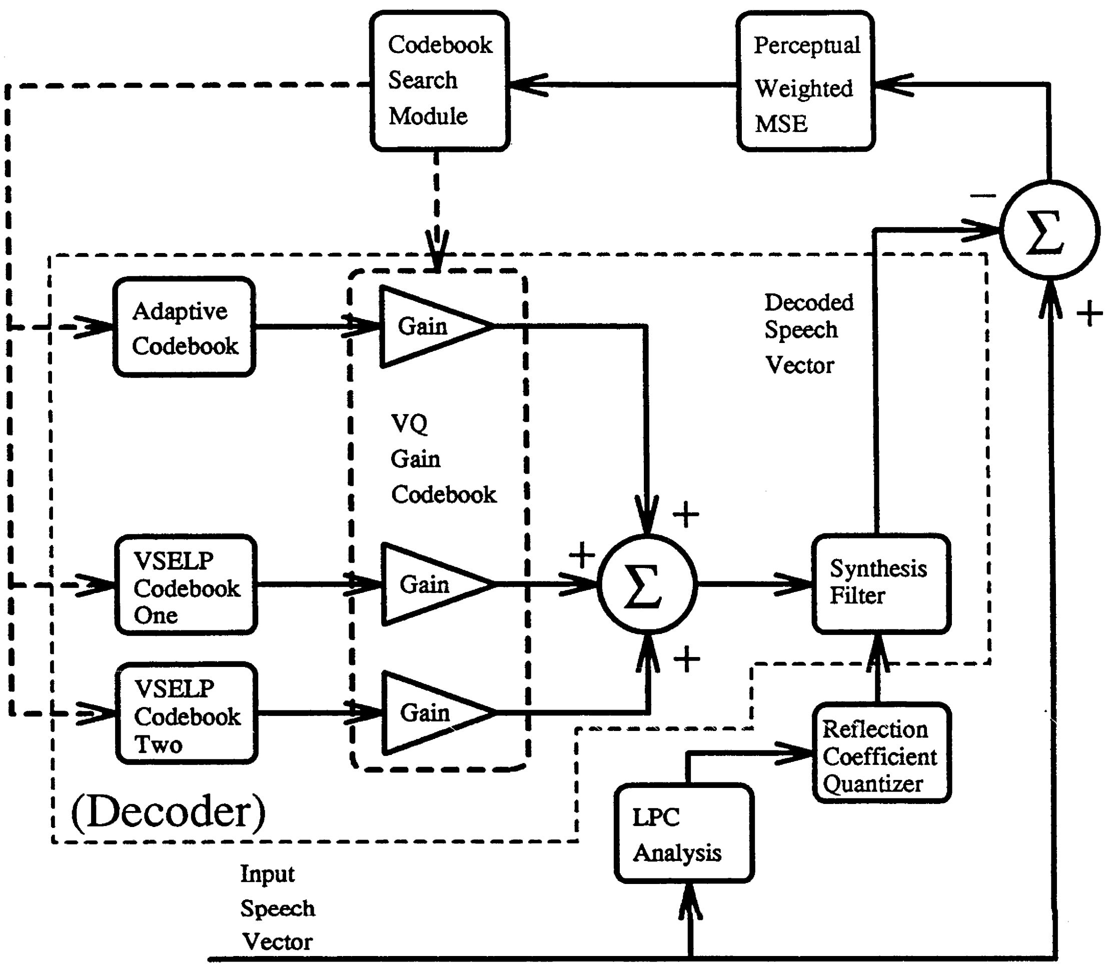

Figure 2.6: VSELP Encoder Block Schematic

channel coding of 5 kbps. A schematic of the VSELP encoder is presented in Figure 2.6. Apart from the obvious lack of codebook searches, the decoder differs from the encoder by the addition of a postfilter on the output of the synthesis filter, and a 'pitch' prefilter on the synthesis filter input (not shown in Figure 2.6).

The pitch prefiltering approach has the advantage of reduced waveform discontinuities in the output compared with the more common pitch postfiltering approach. The smoothing effect of the synthesis filter is important in obtaining this advantage. The adaptive spectral postfilter is similar to spectral postfilters seen in other systems, such as LD-CELP.

The two VSELP codebooks are constructed using basis vectors such that each codebook index bit determines the sign of another basis vector component of the codevector.

In this way, a bit error translates to an inversion of only one basis vector component, and hence produces an output that is 'close' to the actual codevector. The codebook structure also reduces search complexity through an efficient search procedure[70].

Channel coding and missing frame reconstruction strategies used in the IS-54 standard are not discussed in [70, 71], which deal with the VSELP speech coder. However the issue of coder robustness to channel impairments and frame erasures present in the mobile radio transmission environment is very important for mobile communications as there is normally no time to request retransmission of corrupted information. This has received some recent attention in the papers by Yong[192] and Su and Mermelstein[170]. The approach in [170] is claimed to obtain better performance than that in the IS-54 standard, whilst retaining compatibility with the standard.

While the quality of the CTIA VSELP system has been reported to be slightly better than or equivalent to that of the GSM system, the Japanese VSELP standard uses a 6.7 kbps code rate, and has quality that has been reported to be significantly worse than that of GSM[129]. However, this appears to have been based on English speech, and the different characteristics of Japanese speech are likely to have a significant effect on quality assessments.

User subscription to the JDC (Japanese Digital Cellular) system and the CTIA system is also exhibiting substantial differences in patterns. Since commencing operation in April 1993, the JDC system has become rapidly over-subscribed, fuelling research into half rate coding systems. On the other hand, the US VSELP system has experienced a number of hurdles, that appear to have slowed subscription rates. The coder's ability to handle background noise and music has been questioned, and it has even been claimed that the first dual mode terminals (AMPS (Analog Advanced Mobile Phone System) and Digital VSELP) provide unfavourable A/B comparison with the analog system.

There is also some attention being given to the topic of improving the speech quality for mobile communications at the 8 kbps rate. The prospect of obtaining a standard for an Enhanced Fixed Rate Codec (EFRC) at 8 kbps is strengthened by recent developments in 8 kbps toll quality coding, and motivated by PCS type considerations.

2.4.3 PSI-CELP

PSI-CELP (Pitch Synchronous Innovation CELP) has been chosen as the Japanese half rate speech coding standard for mobile communications[126]. The subjective quality of the PSI-CELP coder (3.6 kbps code rate) is claimed to exceed that of the Japanese full-rate VSELP coder (6.7 kbps code rate). A recent paper also describes a 2.4 kbps PSI-CELP coder that has been judged to have superior quality to that of the 4.8 kbps US Federal Standard 1016 CELP coder (discussed below in Section 2.5.1)[130].

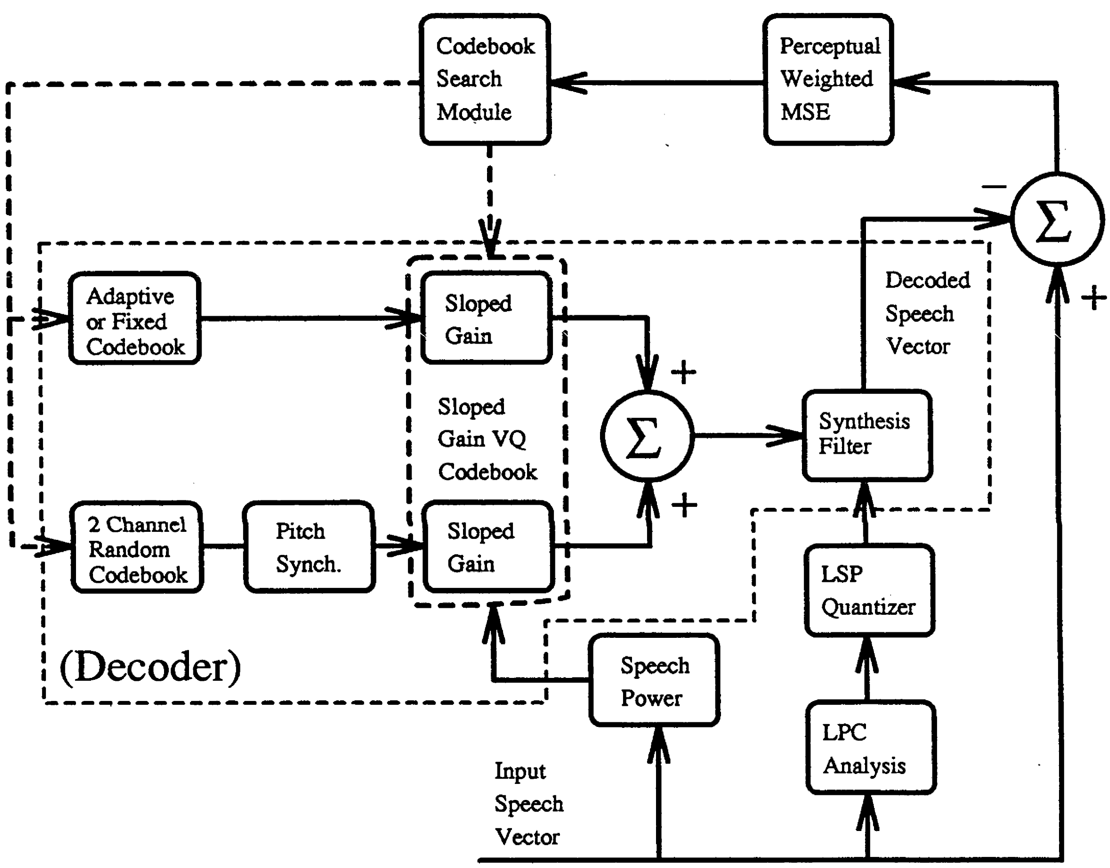

Figure 2.7: PSI-CELP Encoder Block Schematic

At bit rates below 4 kbps, the lengthening of the excitation vector to 8 or 10 ms to improve quantization efficiency, results in perhaps several pitch cycles being included in the vector. Pitch Synchronous Innovation CELP (PSI-CELP) gives pitch periodicity to excitation sources to improve harmonic structures in voiced frames. The area of increased harmonic structure is also targeted for improvement by other coders in the 2-4 kbps range, such as sinusoidal coders[67].

The PSI-CELP coder is shown in block form in Figure 2.7. The coder frame size is 40 ms, consisting of four 10 ms sub-frames. The coder is otherwise fairly similar in structure to the CS-CELP system from above (Figure 2.4), with LPC parameters, adaptive/fixed codevectors, two-channel random codevectors, and gain vectors all being transmitted in the frame. Apart from the pitch synchronization, other noticeable differences to CS-CELP, are that the signal power is also transmitted, and a sloped gain is used. The LPC parameters are represented by 10th order line spectral pairs (LSPs), and coded at 30 bits per frame by a multiple stage vector quantizer with inter-frame moving average (MA) prediction[136].

The pitch synchronizer shown in Figure 2.7 makes use of the pitch period, L, to replicate the first L samples of the codevectors in the random codebook. Hence the harmonic structure in voiced speech is improved through the pitch redundancy introduced both by the adaptive codebook and the pitch synchronized random codebook. Equally important, the excitation during non-stationary speech or non-periodic speech (such as unvoiced), is required to be non-periodic. By switching off the pitch synchronizer this requirement can be obtained for the random codebook, but the adaptive codebook still presents a problem. Hence a switch is used between the adaptive codebook and a fixed codebook such that periodic sections are coded with the adaptive codebook, and non-periodic sections with the fixed codebook.

Phase Adaptive PSI-CELP is also presented in [130] as a means of further improving the pitch synchronization. Incorporating an offset into the pitch synchronization procedure, to allow for the fact that the prediction error waveform also has an element of power localization, is noted to improve performance.

2.4.4 QCELP

Variable rate coders have recently received a significant amount of interest from several groups[28, 56, 137]. Qualcomm's QCELP system is a practical variable bit rate CELP coding system designed for use over a CDMA (Code Division Multiple Access) spread spectrum network[51, 66]. It has been standardized as the North American CTIA CDMA digital cellular standard, and is currently being planned for implementation by a number of regional mobile communications operators in the US, and other countries such as Korea.

TDMA versus CDMA issues have been considered over many years, but the Qualcomm QCELP system is the first practical cellular CDMA system, and has overcome

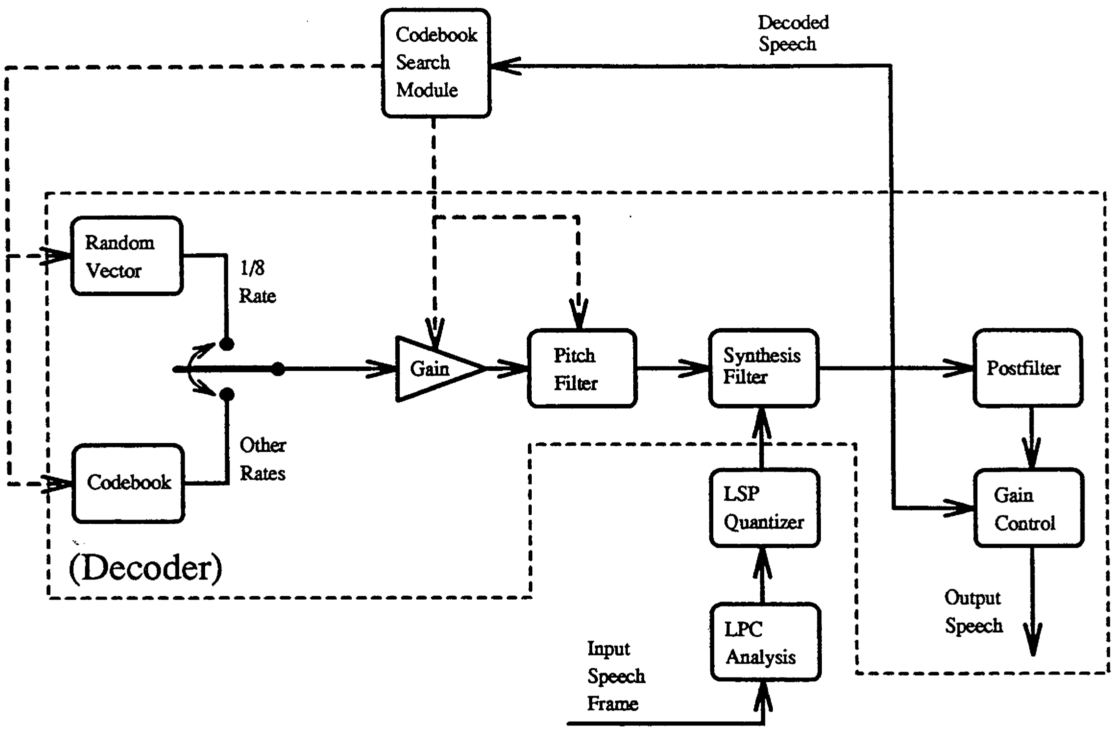

Figure 2.8: QCELP Encoder Block Schematic

many of the widely discussed problems of the spread spectrum approach. In particular, the Qualcomm system has been adequately able to address the issues of the number of users in a cell, and transmission power control. There are many papers discussing TDMA versus CDMA issues, with a good example (at least from the TDMA perspective) being the paper by Adachi *et al.*[1]. Recently some interest in CDMA/TDMA hybrid systems has also begun to immerge[20].

The QCELP speech coding system shown in Figure 2.8 transmits frames at the four different data rates of approximately 8, 4, 2 and 1 kbps. An adaptive algorithm selects the current frame data rate based on an estimate of the difference between the background noise energy, and the current frame's energy. Active speech is typically coded at 8 kbps, and silence and background noise are coded at the lower rates. It has been claimed that MOS tests show that the quality of QCELP coded speech is equivalent to that of 8 kbps VSELP, with an average rate of under 4 kbps maintained during a typical conversation[51].

The basic structure of QCELP is scalable[66], minimizing complexity by allowing the four rates to be implemented in an integrated fashion. The LPC parameters are differentially encoded in the LSP domain, and are more finely quantized at higher coding rates. At the higher rates the pitch and codebook parameters are also updated more frequently. The number of bits, and update frequency per 20 ms frame, used for the LPC, pitch and codebook parameters are shown in Table 2.2.

Coder Data Rate	LPC Parameters		Pitch Parameters		Codebook Parameters	
	Bits Used	Updates per Frame	Bits Used	Updates per Frame	Bits Used	Updates per Frame
8 kbps	40	1	10	4	10	8
4 kbps	20	1	10	2	10	4
2 kbps	10	1	10	1	10	2
$\sim$ 1 kbps	10	1	0	N/A	6	1

Table 2.2: QCELP Bit Allocations for Variable Rate Coding

The pitch predictor uses a single tap, and the pitch gain and pitch lag are quantized with 3 and 7 bits respectively. Only integer lags are used within the range from 17 to 143 samples. The gain sign is transmitted with 1 bit, and the gain magnitude is differentially encoded in the log domain with 2 bits. The length of the codebook is 128, corresponding to 7 bits. During the 1/8 rate frames the pitch gain is set to zero, and the codebook is replaced with a white noise generator to code background noise more efficiently.

Error protection is afforded by frame interleaving and convolutional coding at rate 1/2 on the down-link, and rate 1/3 on the up-link. For full rate frames two CRC (Cyclic Redundancy Check) codes are used for error detection. The first protects the inner 18 most perceptually sensitive bits, and the other protects the entire frame, including the inner CRC bits. Frame erasures are handled by choosing a random codebook vector, setting the pitch gain to one, with the previous pitch lag, and scaling down the LSP parameters. The pitch gain and the LSP predictor parameters are then scaled down at the start of each consecutive frame erasure.

In addition to presenting the QCELP speech coder, the paper by Gardner *et al.*[66] mentions briefly a few of the possible advantages of variable rate coding. Obviously there are advantages associated with voice activity, both in terms of spectral efficiency, and reduced RF (Radio Frequency) and computation power requirements. Other advantages are in the almost seamless introduction of improved speech coder technology, and the possibility for a 'soft capacity' system that could make call blocking and call dropping significantly less probable.

Not specifically mentioned in [66], but also of some interest, is the potential for variable rate systems to better handle 'tricky' coding situations. Areas of considerable interest with current coding systems are adequate handling of background noise, coding of music, and performance for conference calls and with multiple talkers. The ability to switch to a higher rate than which active speech is normally coded may be useful for these and similar 'tricky' situations.

As with the fixed rate CTIA codec, there is also interest in an Enhanced Variable Rate Codec (EVRC), with improved speech quality at the same data rates. Qualcomm is also involved with applying the CDMA variable rate technology to the applications of LEO (Low Earth Orbit) mobile communications and wireless local subscriber loop service. For PCS or FPLMTS type applications a variable rate system with a higher maximum rate and toll quality speech is under consideration by various groups.

2.5 Low Rate Communications Quality Voice

Coders operating at rates of 4.8 kbps and below are generally not toll quality coders. Although, this situation may change over the next few years. Toll quality has effectively been achieved at 8 kbps, and research is targeting this goal at 4 kbps (of significant interest for some PCS style applications). However, currently coders at 4.8 kbps and below are communications quality coders.

The Federal Standard 1016 4.8 kbps coder is a significant standard in this range, and is described briefly below. Other trends for coders in this category of communications quality are then highlighted.

2.5.1 FS 1016 4.8 kbps CELP

In 1988 the US Department of Defense conducted a survey to select a 4.8 kbps coder to supplement its 2.4 kbps LPC-10e coder[178] for use in the third-generation secure telephone unit (STU-III). A coder jointly developed by AT&T Bell Laboratories and the Department of Defense was selected in this survey and became the US Federal Standard (FS) 1016 4.8 kbps coder[25, 26].

FS 1016, shown schematically in Figure 2.9, uses a 30 ms frame size with four 7.5 ms subframes. Linear prediction analysis is performed once per frame, and the 10th order LSP parameters are scalar quantized and transmitted with 34 bits. Linear interpolation of the LSP parameters is used to obtain a set of predictor coefficients for each of the four subframes.

For odd subframes the adaptive codebook index is transmitted using 8 bits to

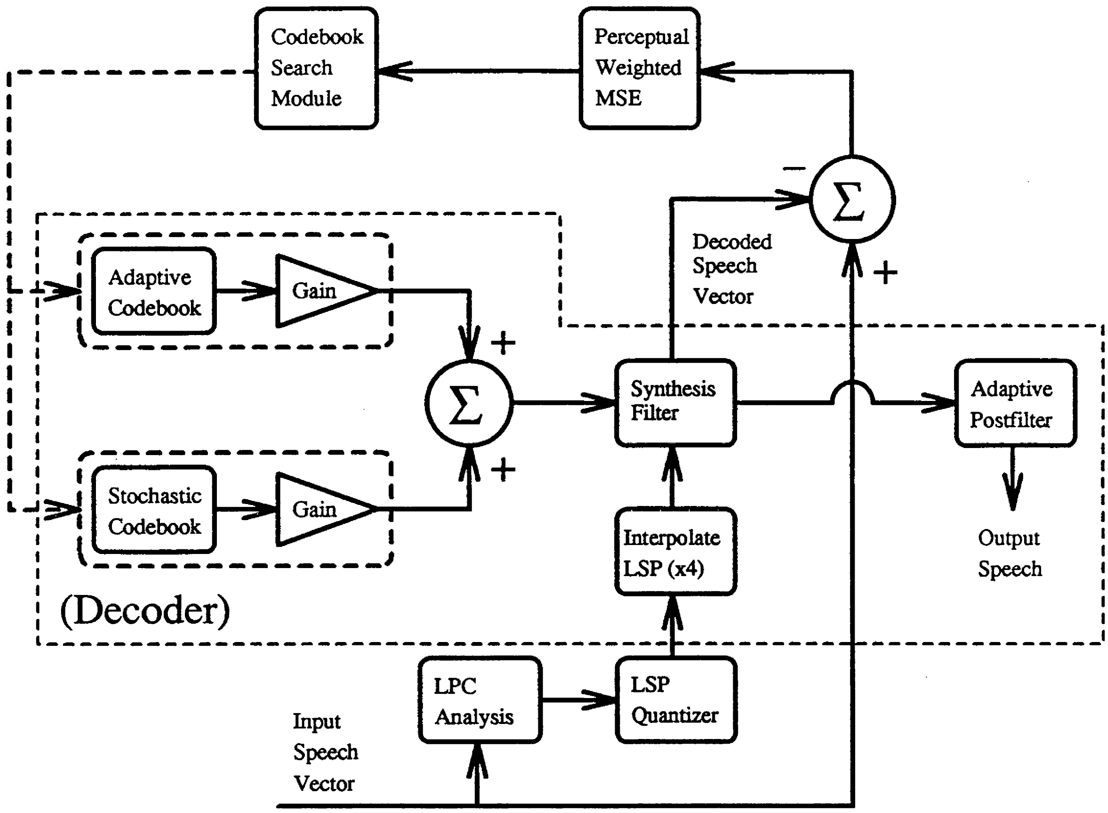

Figure 2.9: FS1016 Simplified Schematic

indicate one of 128 integer or 128 non-integer delays in the range of 20 to 147 samples. For even subframes the delay is differentially encoded with 6 bits relative to the delay in the previous subframe. The adaptive codebook gain is transmitted every subframe using 5 bit non-uniform scalar quantization.

For each subframe a 9 bit index (512 vectors) for the stochastic codebook is transmitted. The stochastic gain is coded with 5 bit non-uniform scalar quantization. To reduce codebook search complexity the codebook is ternary valued (-1,0,+1), sparse, and overlapped. Recent work by Kao and Baras[109] has resulted in further significant reduction in computation by modifying the codebook to form a deterministic structure. This work is not included in the original FS 1016 standard, but is interesting in that a performance improvement is claimed along with the reduction in complexity.

The FS 1016 decoder includes adaptive postfiltering, and adaptive smoothing and stability constraints on the received CELP parameters, which are not shown in Figure 2.9.

2.5.2 2 to 4 kbps Speech Coding

There is a significant amount of interest in speech coding in the bit rate range from 2 to 4 kbps. Driving factors are largely related to mobile communications, and future LEO satellite systems. The desire for a high quality 2.4 kbps speech coder has also recently been announced by the US Government[187].

The high quality of the PSI-CELP Japanese digital half-rate system (3.6 kbps) mentioned above is an indication of the possibilities for rates between 2 and 4 kbps. Other approaches in this range include the work by Shoham[161, 162] on time-frequency interpolation, the Motorola 2.4 kbps work reported recently[59], sinusoidal coding approaches[67], multi-band excitation[153], and many others.

It is likely that the next few years will see significant increases in speech quality for coders operating at these rates. In general coders operating in this range would appear to be highly computationally expensive, and there is also likely to be room for computation efficiency improvements.

2.5.3 Very Low Rate Speech Coding

For some applications, especially in the military arena, very low rate speech coding is required. Very low rate speech coding can have significant advantages for low power or long distance transmission, encryption, security against jamming, and avoidance of transmission detection.

A recent paper[132] presents an 800 bps speech coder with DRT (Diagnostic Rhyme Test) results very close to that of 2.4 kbps LPC-10e coding. As we would expect, the speech quality at this rate is not very good. However, in such applications, speech intelligibility is the more important issue. Unfortunately, current speech coders at 800 bps appear to lose a large amount of the information that identifies the speaker.

2.6 PCS Standardization

The goal of obtaining global and ubiquitous personal wireless communications is fuelling a great deal of serious research effort around the world[40]. Of course, there are many different ideas, and projections of the future, but as far as speech coding is concerned, there are a number of issues which appear reasonably clear for PCS (Personal Communications Systems).

- Speech quality should be high. One quality aim is to be equivalent to that of an ideal connection over the analog telecommunications network (toll quality).
- Computation cost should be low. The major issues here are terminal equipment cost, size, and battery power drain.
- Delay should be small. Excess delay causes significant problems with echo cancellation, and to avoid perceptual problems, minimising delay is important.
- Bit rate should be low. Radio spectrum is a scarce resource, and lower bit rate allows lower transmit power.

The above four requirements are generic to the pursuit of speech coding systems, but there are also some other considerations with PCS type systems. The emergence of 'new' techniques, such as the practical use of CDMA spread spectrum, is likely to have a significant impact on speech coding for PCS. As speech and other sources such as facsimile and video are inherently of variable rate, it appears that PCS systems using CDMA or similar techniques to provide efficient variable rate transmission will find application.

Also appearing to be very important for global ubiquitous mobile communications is the concept of graceful degradation. Digital mobile phones suffer from extremely bursty errors due to rapid fading of the radio carrier. The speech coders must be robust to these problems, and a system such as PCS is likely to require similar robustness, or at least graceful degradation, to other problems such as cell overcrowding, and fringe transmission regions.

Global coverage issues bring us to the subject of LEO (Low Earth Orbit) satellite mobile communications systems. It is likely that one of the most substantial uses for such a system would be to augment coverage areas of the standard cellular mobile networks, for low population areas, or for boating and shipping. Here the issue of multimode terminals (terrestrial cellular and LEO) is likely to be of significant importance. Iridium and the Global Star consortium are both considering LEO satellite systems for mobile communications (among others[154]), and both systems appear likely to fly on the 1997/98 time-frame. Recently Microsoft and McCaw Cellular, through Teledesic, have also announced plans for some form of LEO system.

It is becoming increasingly apparent that speech coders operating in this region of the spectrum must be closely linked to channel coding and modulation issues. Indeed, this is a very important area of research that is likely to see significant attention in the future. Chapter 9 of this thesis discusses the concepts of combined source and channel coding in more detail.

In relation to PCS it is also important to mention the concepts of high tier and low tier systems. The goal of global and ubiquitous personal communications systems is unlikely to be achieved through the use of a single system servicing a population with 'Star Trek' style communications devices. Rather, PCS can be viewed more as refinement, integration, and extension of existing telecommunications systems.

Low tier PCS can most logically be associated with an extension of the current wired telephone network that allows an element of subscriber mobility and flexibility to the service provider. Although several Cordless Telephone (CT) systems have been trialed with only mixed success, there are a number of advantages to a wireless local drop system, either within an office environment, or for wireless local loop service to residential areas. For these applications toll quality appears to be a requirement, and other desirable properties would include low subscriber equipment cost, and long battery life between recharging. Rapid movement of a user, needing many cell hand-offs would generally not be possible.

High tier PCS appears to be more logically associated with the current mobile communications systems. Higher speech quality, and higher subscriber density are natural desires. As mentioned above, the possibility for global coverage via satellite systems is also important.

2.7 Chapter Summary

From the above speech coding system descriptions, it is apparent that there are many different applications for speech and audio coders. Each one of these coders involves different balances of the four basic ingredients of: **(1)** bit rate; **(2)** computational complexity; **(3)** delay; and **(4)** output speech quality. However, there are some common aspects of the coders that should be noted. Generally, knowledge about the characteristics of the source is used in an adaptive fashion to improve coding performance. One issue that is also common is the issue of robustness to channel transmission errors, as the assumption of complete separation of source and channel coding is not valid in practice.

It is also important to realize that a speech coder is only one component of a (typically) complex communications system, and is usually developed in parallel with other system aspects. For mobile communications, such other issues include: channel coding, interleaving, modulation, RF design, antenna diversity considerations, air interface framing strategy, channel equalization, and voice activity detection with discontinuous transmission. Thus a global telecommunications knowledge is useful in understanding the implications for and by speech coding design decisions. This is particularly important for mobile communications.

Other important issues for speech coding design are more related to social issues and usage patterns. For instance, the desire to be able to make conference calls means that speech coders should be able to accommodate multiple speakers. Additional significant issues are the performance in high levels of background noise, coding of music, and binary data transfer.

For readers desiring general or specific telecommunications knowledge, a reasonable sized bibliography is included in this thesis, and should provide a good place to start. Many of the papers and books are referred to above and where relevant throughout this thesis, however, some excellent books are [9, 10, 13, 39, 68, 92, 106, 143, 145, 169]. Several special magazine issues have been found useful [93, 94, 95, 96, 97, 98], and a number of general papers also provide a very good starting point [8, 53, 67, 102, 103, 104, 152, 165]. Finally, the speech coding tutorial review by Spanias[168] is recommended reading.

2.8 Thesis Overview

The remains of the thesis has already been outlined in the introduction chapter (Chapter 1). However, a further outline is given here that may assist in placing the research topics of the thesis in perspective with the speech coding background as presented in this chapter.

Chapter 3 presents a stability analysis applicable to ADPCM with adaptive quantization. CCITT Recommendation G.721 32 kbps ADPCM provides toll quality speech coding, and hence the stability analysis is of only academic interest with respect to the standard. However, the analysis presented does give some useful insight into the Jayant 'One-Word Memory' approach to adaptive quantization, and provides an indication of what might be required to improve ADPCM systems operating at rates below 32 kbps.

Chapter 4 investigates the use of an entropy coded ADPCM approach. This variable bit rate ADPCM system overcomes stability problems of ADPCM systems at low bit rates, and provides performance comparable to some CELP systems at the same average coder output bit rates. A fully functional speech coder is not constructed, but 'proof of concept' is achieved via simulations.

Chapter 5 presents the Kalman filter as a general design tool for speech coding applications. Kalman filtering has not been mentioned in connection with any of the systems discussed in this chapter (Chapter 2) and is seldom used for speech coding. The Kalman filter is presented as an extension to the standard linear predictor that is used in many speech coding applications, and it is shown that the computational cost incurred by the use of the Kalman filter can be managed by a number of simple techniques.

Chapter 6 investigates the integration of the Kalman filter with the variable bit rate ADPCM system introduced in Chapter 4. Through such an approach output speech quality equivalent to that of LD-CELP is claimed with an average coder output bit rate of 8 kbps. Again the focus of simulations is in providing proof of concept, rather than an operational speech coding system.

Chapter 7 attempts to integrate the Kalman filter and CELP speech coding systems. Many of the speech coding systems mentioned in this background chapter have a basic CELP structure. The Kalman filter has been observed to provide significant advantage for the variable bit rate ADPCM system (Chapters 4 and 6), and its application to CELP is worth consideration.

Chapter 8 investigates modifications to CCITT Recommendation G.728 16 kbps LD-CELP for applications where frame erasures exist, such as for PCS or FPLMTS. Minor modifications to G.728 LD-CELP are observed to provide a high level of robustness to frame erasure errors.

Chapters 9 and 10 present no new research results, but discuss a number of important considerations with regard to practical use of the variable bit rate ADPCM system from Chapters 4 and 6.

Chapter 11 discusses possibilities for future research work in the area, and presents conclusions that can be drawn from the research contained in this thesis.

Chapter 3

ADPCM Stability

3.1 Chapter Motivation

Adaptive quantization is used in many ADPCM systems to improve performance. In this chapter we present an analysis dealing with the effect of adaptive quantizers on the stability of ADPCM systems as compared to ADPCM with non-adaptive quantization. A sufficient condition for stability is presented that leads to a relation between adaptation rate and stability. This indicates a price is paid for adaptation in terms of more stringent restrictions on the predictor to guarantee stability. The theorem further indicates that stability is related to the maximum rate of quantizer step size decrease, and can be viewed as a theoretical justification for the shape of the Jayant "One-Word Memory" multiplier curve for adaptive quantization.

3.2 Introduction

Adaptive Differential Pulse Code Modulation (ADPCM) systems are familiar from speech coding applications, and incorporate predictor and quantizer components, at least one of which is adaptive. In the CCITT G.721 ADPCM standard for 32 kilobit per second (kbps) coding of speech both the predictor and quantizer adapt, while other formulations exist in which the predictor alone is adapted and/or the adaptive quantizer is replaced by a fixed quantizer with variable bit rate selection. We pose and study the question of the effect on stability of the adaptive coder due to the adaptation of the quantizer. Note the stability issue that is considered in this chapter is not directly related to encoder/decoder convergence, but is rather the internal stability of the encoder.

The dynamics of ADPCM can be divided into two parts – the stability of the system, treating the time to recover from errors in internal state, and the performance, associated with the steady state behaviour. Our attention here will focus on the stability issues principally and the recovery from errors introduced by initial conditions. However, we do provide some discussion on the performance versus stability trade-off that exists within ADPCM.

To date the only studies of stability have been conducted with non-adaptive quantization, although the detrimental effect of adaptive quantization on stability is well known[22, 23]. A comprehensive stability analysis of the non-adaptive DPCM system is provided in a recent paper by Macchi and Uhl[123]. In their paper, Kennedy and Johnson[115] have provided stability conditions for the ADPCM system which constrain the allowable frequency response of the predictor component. Here we consider the additional effects of adapting the quantizer as in CCITT Recommendation G.721 ADPCM. Since the quantizer adaptation proceeds much faster than the predictor adaptation, we adopt an analysis extending [115] through the application of multiplier theory.

Several authors have noted the existence of stability problems in ADPCM. Iyengar and Kabal [99] recognise that the stability problem is related to the quantizer adaptation, due to the quantization errors cascading through the predictor. Crisafulli *et al.* [45] have noted problems with ADPCM systems caused by the adaptive quantization and adaptive prediction being performed quite separately, without any real account of effects on the other unit. Honda and Itakura [84] have reported experimental observations of stability difficulties with adaptive quantization in ADPCM which are alleviated by replacement with a quantizer of limited step-size adaptation, and adaptive bit rate selection. Our analysis is directed towards explaining this effect and linking the instability to quantizer adaptation design.

The problem conclusion we reach here is that more stringent conditions must be placed on the predictor in order to achieve stability of the overall ADPCM system when used in conjunction with an adaptive quantizer. Connections are made between the parameters of quantizer adaptation, constraints on the predictor frequency response, and performance implications for the ADPCM system.

A block diagram of the ADPCM encoder system studied is presented in Figure 3.1. In this diagram, the adaptive quantizer has been displayed as a time varying gain scaling unit, g_k, associated with a fixed quantizer. All adaptation is 'backwards', ie.

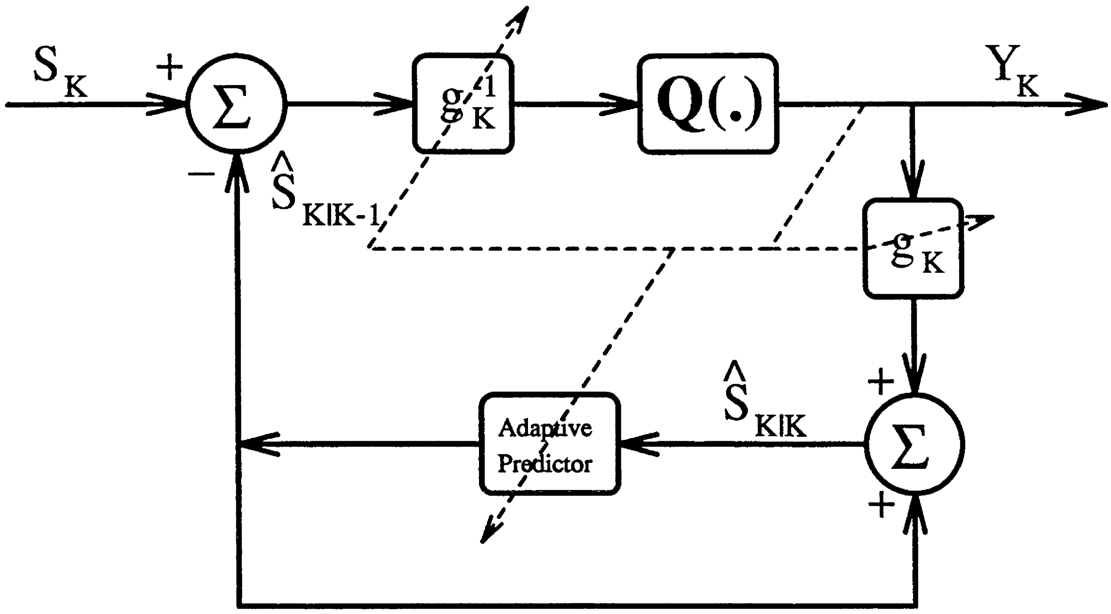

Figure 3.1: Basic ADPCM Encoder Block Schematic

based solely upon the transmitted symbols, Y_k, and not directly on the source signal, S_k. For stability considered in this chapter, only the ADPCM encoder is considered. Standard techniques such as the use of forgetting factors in conjunction with adaptive quantizers are often used to assure convergence of the decoder to the encoder after transmission errors. This topic is not pursued within this chapter.

We note that quantizer blocks are often considered to not only perform the quantization operation, but also the operation of coding the quantizer output level in a binary format. Hence an inverse quantizer block that decodes the binary sequence and produces the quantized output is usually required. The operation of the quantization block within this chapter is simply one of restricting the input signal to one of a fixed number of output quantization levels within the same signal domain as the input. As a consequence, no inverse quantization blocks are displayed in the diagrams in this chapter.

The fixed quantizer, $\mathbf{Q}[.]$, in CCITT Recommendation G.721 ADPCM has the following characteristic:

$$\mathbf{Q}[x] \triangleq \sum_{j=-7}^{7} sgn(x + 2j) \tag{3.1}$$

corresponding to 4 bits precision (15 levels). For our analysis we will assume this quantizer characteristic, but could just as easily use another characteristic.

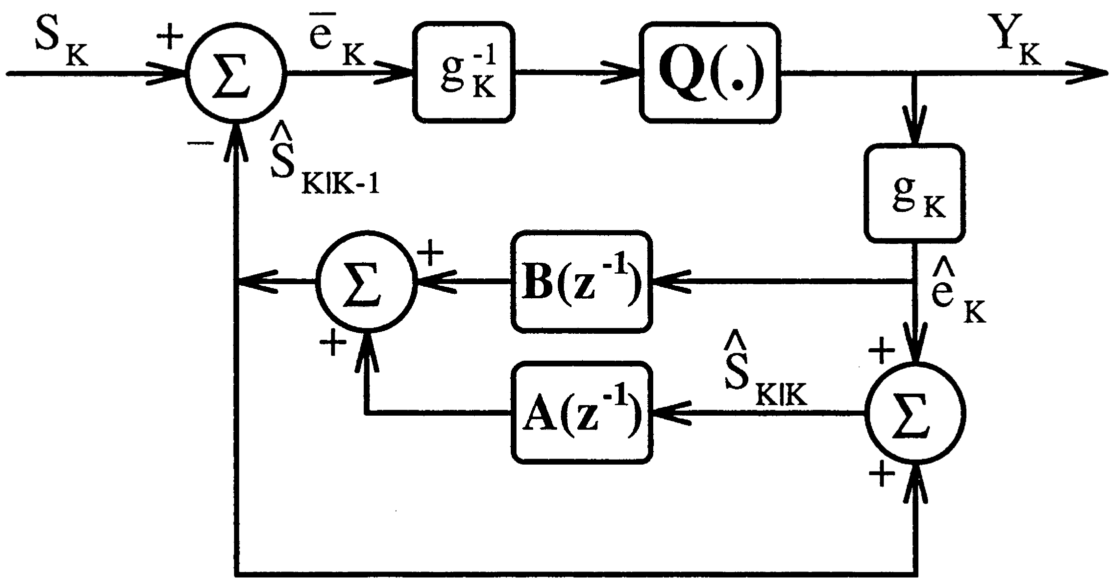

Figure 3.2: Expanded ADPCM Encoder Schematic

Strictly, Figure 3.1 refers to an all-pole predictor. The CCITT ADPCM standard specifies the adaptive predictor to be a linear ARMA filter implemented with the two filter structures as shown in Figure 3.2. The block $A(z^{-1})$ is a polynomial in the delay operator z^{-1} that represents a set of poles, and $B(z^{-1})$ represents a set of zeros. In G.721 ADPCM A has degree 2 and B degree 6. While they are adapted, for our purposes here we shall treat them as fixed. Since (as stated above) the quantizer adaptation is considerably more rapid than predictor adaptation, it is likely to have a more substantial impact on system stability.

In the next section we present the definition of stability applicable to the ADPCM analysis, and introduce the error system framework on which the result is based. The subsequent section brings together Passivity Analysis, Multiplier Theory, and previous results in ADPCM stability to arrive at our stability theorem. The consequences of this theorem are then discussed, and some conclusions drawn.

3.3 Stability of ADPCM Systems

3.3.1 Definition of Stability

An ADPCM system is a signal coder, and as such, has a basic aim to transmit efficiently the encoder input signal to the receiver. Information Theory suggests performing this task by first removing as much redundancy from the signal as possible, and then coding the residual signal efficiently. Unfortunately with a feedback coding system, there is a tendency for any errors in this coding process to explode. This is especially troublesome with low bit-rate implementations, but even for more moderate bit-rates, it is possible to find malicious driving signals that the system cannot handle.

We thus believe that the notion of stability which makes sense in this situation is that previous errors in encoder internal state are filtered out of the system over time. In particular, these initial errors do not lead to an infinite cascade of errors. In this way, we are really dealing with the notion of transient stability of the system. Note that we are appealing to the discrete value set for Y_k here to ensure that asymptotic stability equates with finite time stability. We thus present a formal definition of this concept of stability.

Definition 3.1 *The coding system of Figure 3.2 will be said to be stable if for any initial predictor state there exists a finite positive integer k_0 such that, for a given input sequence $\{S_k\}$, the output sequence $\{Y_k\}$ is independent of the initial state for $k \geq k_0$.*

3.3.2 A Framework for Stability Analysis

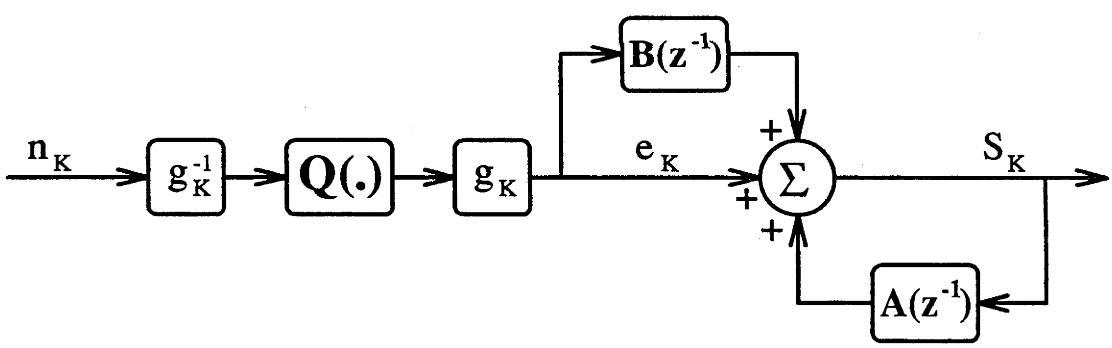

Figure 3.3: Speech Equivalence Class Signal Model

Following the lead of Kennedy and Johnson [115], we pose that the sequence $\{S_k\}$ is generated as a filtered version of a fictitious signal $\{e_k\}$ driven by a white process $\{n_k\}$, as in Figure 3.3. The inclusion of the quantizer in this signal model is somewhat unusual, but justified under the circumstances. It is obvious that if the adaptive quantizer, $g_k\mathbf{Q}[g_k^{-1}(.)]$, were absent from the signal model, then it would be possible to find an $\{e_k\}$ sequence to produce any given $\{S_k\}$ sequence. However, the presence of the quantizer in the coder means that we are only concerned with an 'equivalence class' of

$\{S_k\}$, that produce the same $\{\hat{e}_k\}$ sequence. The addition of the quantizer in the signal model therefore focusses our attention on this equivalence class of $\{S_k\}$ sequences.

Remark 3.1 The use of an equivalence class of $\{S_k\}$ allows us to remove the effects of normal coder distortion from our analysis of stability. This is well justified by the stability Definition 3.1.

Stability is associated with the convergence of $\{\hat{e}_k\}$ in the encoder to $\{e_k\}$ in the fictitious signal model. Thus stability is associated with the rejection of initial disturbances present in the encoder state, as indicated by $\{\hat{e}_k\}$. In order to obtain a stability result, we construct a framework for our analysis in terms of an error system. The first step is to define the two error signals,

$$\tilde{e}_k \triangleq e_k - \hat{e}_k, \qquad \tilde{S}_k \triangleq S_k - \hat{S}_{k|k}. \tag{3.2}$$

Thus, our stability Definition 3.1 can be interpreted as: Given an arbitrary initial predictor state error condition specified by the semi-infinite error sequence $\ldots, \tilde{e}_{-2}, \tilde{e}_{-1}, \tilde{e}_0$ then the system is stable if there exists a finite k_0 such that

$$\tilde{e}_k = 0, \ \ \forall k \geq k_0, \ \ \forall e_k. \tag{3.3}$$

Some simple algebra applied to the various signals in Figure 3.2 and Figure 3.3 leads to the following for the encoded ADPCM signal.

$$\hat{e}_k = g_k \mathbf{Q}\left[g_k^{-1}\left\{e_k + B(z^{-1})\tilde{e}_k + A(z^{-1})\tilde{S}_k\right\}\right] \tag{3.4}$$

In order to proceed, we wish to express $\tilde{S}_k$ in terms of $\tilde{e}_k$. This is done with the use of the following two equalities. From the ADPCM system in Figure 3.2:

$$\hat{S}_{k|k} = \hat{e}_k + B(z^{-1})\hat{e}_k + A(z^{-1})\hat{S}_{k|k}. \tag{3.5}$$

From the signal model in Figure 3.3 we have:

$$S_k = e_k + B(z^{-1})e_k + A(z^{-1})S_k. \tag{3.6}$$

The difference between these two equations leaves us with

$$\tilde{S}_k = \frac{1 + B(z^{-1})}{1 - A(z^{-1})}\ \tilde{e}_k. \tag{3.7}$$

We are now in a position to find an expression for the error signal, $\tilde{e}_k$, as

$$\begin{aligned}
\tilde{e}_k &= e_k - g_k\mathbf{Q}\left[g_k^{-1}\left\{e_k + \left\{B(z^{-1}) + \frac{1 + B(z^{-1})}{1 - A(z^{-1})}A(z^{-1})\right\}\tilde{e}_k\right\}\right] && (3.8)\\
&= e_k - g_k\mathbf{Q}\left[g_k^{-1}\left\{e_k + \left\{\frac{A(z^{-1}) + B(z^{-1})}{1 - A(z^{-1})}\right\}\tilde{e}_k\right\}\right]. && (3.9)
\end{aligned}$$

From this equation we are able to present our error system. This appears in Figure 3.4. The signal w_k shown in this diagram is to allow for the initial error conditions, and without loss of generality can be assumed to be zero for all $k \geq 0$.

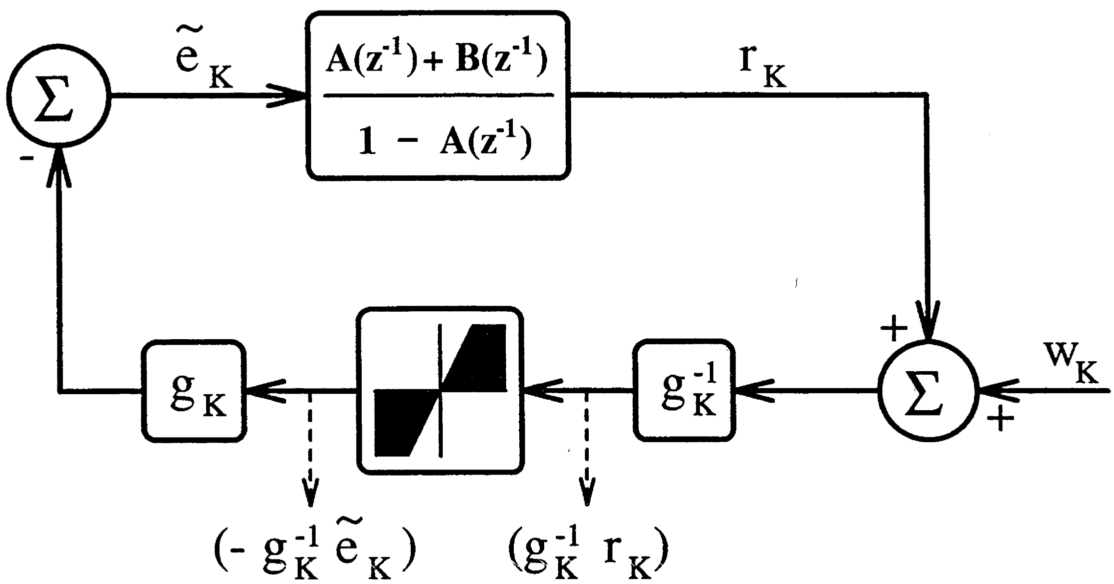

Figure 3.4: Error System Block Diagram

For Figure 3.4 we have defined a filtered error signal,

$$r_k \triangleq \left\{ \frac{A(z^{-1}) + B(z^{-1})}{1 - A(z^{-1})} \right\} \tilde{e}_k, \tag{3.10}$$

representing the memory effect of past errors. Ideally we would like r_k to be zero, or at least small. Intuitively, we require the error signal $\tilde{e}_k$ to be damped by this filtering block to such an extent that we do not get an explosion of the r_k signal, and hence the $\tilde{e}_k$ error signal.

The block representing the conversion of $g_k^{-1} r_k$ to $-g_k^{-1}\tilde{e}_k$ (marked on the diagram) needs some explanation. This is a memoryless sector bounded nonlinear operator that is caused by the quantizer characteristic, given in equation 3.1. Combining and rewriting equations 3.9 and 3.10, we have:

$$\tilde{e}_k = e_k - g_k \mathbf{Q}\left[g_k^{-1}\left\{e_k + r_k\right\}\right]. \tag{3.11}$$

From the model presented in Figure 3.3, we have

$$e_k = g_k \mathbf{Q}\left[g_k^{-1} n_k\right]. \tag{3.12}$$

Thus, for $r_k = 0$, the following results:

$$\tilde{e}_k = e_k - g_k\mathbf{Q}\left[g_k^{-1}g_k\mathbf{Q}\left[g_k^{-1}n_k\right]\right] \tag{3.13}$$

$$= e_k - g_k\mathbf{Q}\left[g_k^{-1}n_k\right] = 0. \tag{3.14}$$

For small values of $g_k^{-1}r_k$, the quantized value $g_k\mathbf{Q}\left[g_k^{-1}\{e_k + r_k\}\right]$ will be unaffected, and thus $g_k^{-1}\tilde{e}_k$ will still be zero. However, once $g_k^{-1}r_k > 1$, then we know that $g_k^{-1}\tilde{e}_k \geq 2$, due to the quantizer characteristic. Similarly, for $g_k^{-1}r_k > 3$, we obtain $g_k^{-1}\tilde{e}_k \geq 4$, and so on. This leads to a nonlinearity between $g_k^{-1}r_k$ and $g_k^{-1}\tilde{e}_k$ that lies between lines of slope 0 and 2.

We next proceed to the consideration of the stability of the nonlinear feedback system depicted in Figure 3.4, noting that its stability coincides with the stability of the ADPCM coder.

3.4 Stability of ADPCM with Adaptive Quantization

3.4.1 The Passivity Theorem

The Passivity Theorem is developed from circuit theory notions. In circuit theory a network is said to be passive if it does not produce energy to external loads other than through initial conditions, and strictly passive if it dissipates energy. In order to formalize these concepts, we first need to introduce some notation following [52]. Let P_T be the truncation operator defined by

$$(P_T x)_k \triangleq \begin{cases} x_k, & if\ k \leq T; \\ 0, & if\ k > T. \end{cases} \tag{3.15}$$

We are then able to introduce the concept of the extended discrete function space l_2^e, defined by

$$x \in l_2^e \Leftrightarrow \|P_T x\| < \infty, \quad \forall T \in Z_+, \tag{3.16}$$

where Z_+ is the set of positive integers. We also need to define the inner product as

$$\langle x, y\rangle \triangleq \sum_{i=0}^{\infty} x_i y_i. \tag{3.17}$$

A euclidean norm is then defined as

$$\|x\| \triangleq \langle x, x\rangle^{1/2} = \left(\sum_{i=0}^{\infty} x_i^2\right)^{1/2}. \tag{3.18}$$

In order to simplify notation used below, we write

$$x_T \triangleq P_T x, \tag{3.19}$$

and

$$\langle x, y \rangle_T \triangleq \langle x_T, y_T \rangle. \tag{3.20}$$

The above notation then allows us to present formal definitions of passivity.

Definition 3.2 *An operator* $\mathbf{H} : l_2^e \mapsto l_2^e$ *is* passive *if* $\exists$ *some constant* β *such that*

$$\langle \mathbf{H}x, x \rangle_T \geq \beta, \quad \forall x \in l_2^e \ \forall T \in Z_+. \tag{3.21}$$

Definition 3.3 *An operator* $\mathbf{H} : l_2^e \mapsto l_2^e$ *is* strictly passive *if* $\exists \delta > 0$ *and* $\exists \beta$ *such that*

$$\langle \mathbf{H}x, x \rangle_T \geq \delta \|x\|_T^2 + \beta, \quad \forall x \in l_2^e \ \forall T \in Z_+. \tag{3.22}$$

We make two remarks connected with passivity. Firstly, from equations 3.21 and 3.22, since $\mathbf{H}x$ is the 'output' of the operator, passivity is very closely allied with the non-negativity of a system. Thus average sign preservation of an operator is related to passivity. Secondly, the constants β in these definitions embody the initial condition effects.

It is also important to note that for a linear time invariant system, $\mathbf{H}(z^{-1})$, strict passivity is equivalent to $\mathbf{H}$ being stable and $\mathbf{Re}\{\mathbf{H}(e^{j\theta})\} \geq 0, \ \forall \theta \in [0, 2\pi)$.

The Passivity Theorem allows a stability result to be obtained for a system consisting of a passive operator and a strictly passive operator connected in a feedback configuration. Desoer and Vidyasagar [52] present a section on the Passivity Theorem, but perhaps a better view for our requirements is presented in the paper by Kennedy, Anderson, and Bitmead [114]. Here we simply present the theorem.

Theorem 3.1 (Passivity) *Consider the feedback system of Figure 3.5. Suppose the operator* $\mathbf{H_1}$ *is strictly passive, and the operator* $\mathbf{H_2}$ *is passive. Further suppose that* u_k *and* v_k *exist and belong to* l_2^e*, then* $y_k \in l_2 \Rightarrow u_k \in l_2$.

3.4.2 Stability without Adaptive Quantization

Kennedy and Johnson present their stability result for ADPCM without consideration of the quantizer adaptation in terms of a frequency response requirement on the $A(z^{-1})$ and $B(z^{-1})$ polynomials. Their theorem follows.

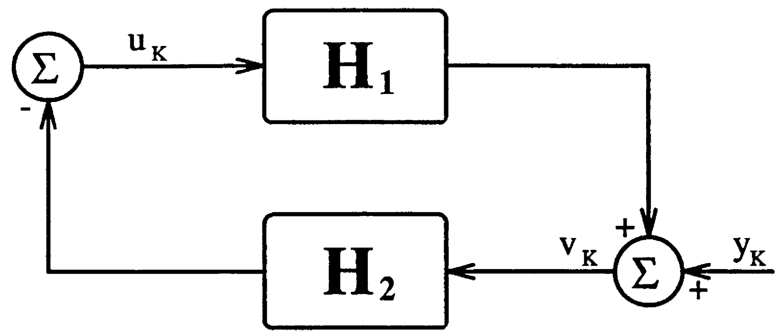

Figure 3.5: Passivity Theorem Block Diagram

Theorem 3.2 *Suppose that*

$$\mathbf{Re}\left\{\frac{A(e^{-j\theta})+B(e^{-j\theta})}{1-A(e^{-j\theta})}\right\} > -\frac{1}{2}, \quad \forall\theta \in [0, 2\pi) \tag{3.23}$$

then the system is stable in terms of Definition 3.1.

This theorem implies that for stability of the coder, there is a restriction on the allowable $A(z^{-1})$ and $B(z^{-1})$ polynomials. These polynomials are part of the signal model in Figure 3.3 and are effectively dictated by the spectral properties of the signal $\{S_k\}$.

Theorem 3.2 is a direct consequence of applying the Passivity Theorem 3.1 to the simplified error system as presented by the authors. Firstly some loop transformations are performed on the system shown in Figure 3.4 resulting in Figure 3.6. We note that here the g_k and g_k^{-1} factors are ignored, but they are shown in Figure 3.6 in order that they can be included in our analysis later.

Note that the net effect of the feedback factor of 0.5 around the sector bounded nonlinearity and the complementary feedforward factor around the linear block in Figure 3.6 is to cancel exactly, with the input and output signals of the non-linearity the same as before. It would appear that the lower block in Figure 3.6 is still a sector bounded nonlinearity, now filling the entire first and third quadrants. The Kennedy and Johnson stability theorem proof is based on this observation, and is outlined below. However we note the existence of a flaw in this approach which we overcome shortly by an alternative theorem proof.

If we take $\mathbf{H_2}$ as this sector-bounded nonlinear gain element, then we have

$$\langle \mathbf{H_2}v, v\rangle_T \geq 0, \quad \forall v \in l_2^e \ \forall T \in Z_+, \tag{3.24}$$

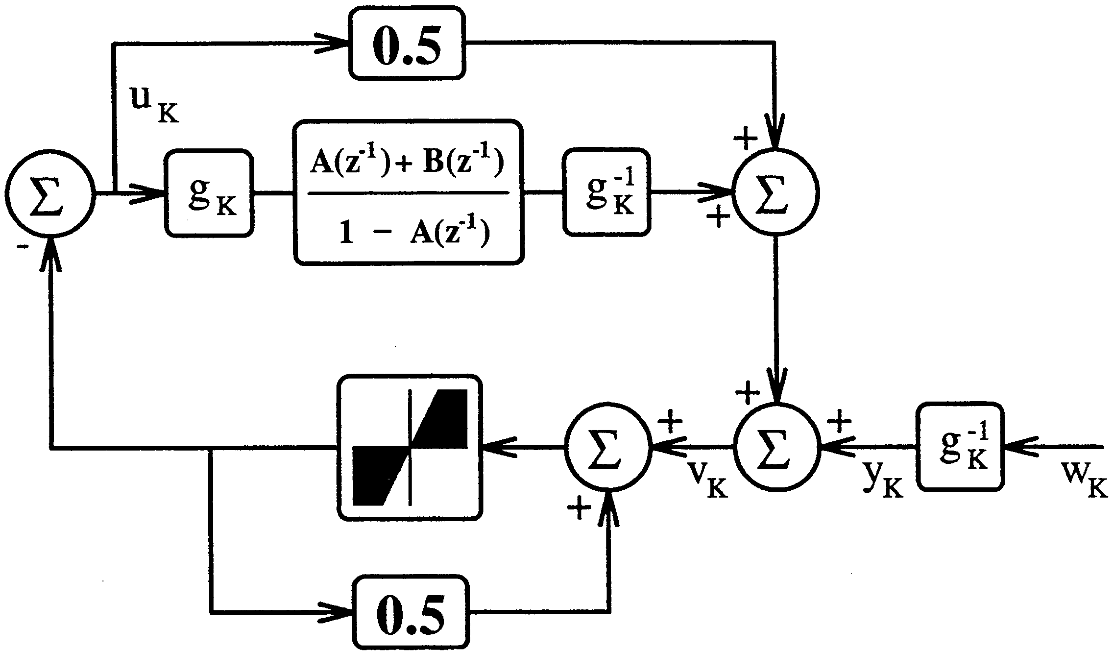

Figure 3.6: ADPCM Passivity Error System Block Schematic

and thus passivity of $\mathbf{H_2}$ by equation 3.21.

The strict passivity of the linear upper block, $\mathbf{H_1} = \left[\frac{A(z^{-1})+B(z^{-1})}{1-A(z^{-1})} + \frac{1}{2}\right]$, can be expressed in terms of a frequency response criterion, as discussed above. Note that the result of the Passivity Theorem 3.1 exactly corresponds to the stability Definition 3.1, as $\tilde{e}_k \in l_2$ implies that $\tilde{e}_k$ tends to zero within a finite time.

The Kennedy and Johnson Theorem 3.2 follows directly. Note that the complementary feedforward and feedback factors in Figure 3.6 result in a more relaxed frequency response criterion in the Theorem.

We wish to extend this result to incorporate the effect of quantizer adaptation, as given by the addition of the g_k and g_k^{-1} gain factors.

3.4.3 Quantizer Adaptation

The adaptive quantizer in the ADPCM system under study has been decomposed into a fixed quantizer, and a time varying positive gain, g_k, as shown in Figure 3.2. In order to proceed further with our analysis, we require more specific knowledge about the gain adaptation.

The Jayant "One-Word Memory" algorithm for quantizer step-size adaptation is well known, and forms the basis of the CCITT G.721 quantizer adaptation rule [46, 101, 106]. The scheme utilizes a table of multiplier values, one of which is selected to update the gain, based upon the previously quantized value. Thus we have:

$$g_{k+1} = M_k \times g_k \qquad a \le M_k \le b \tag{3.25}$$

where M_k is determined from the quantizer output level, Y_k, through the use of a table, and the values in this table are bounded by the constants a and b ($a < 1, b > 1$). Thus we have the range of values possible for the next gain, g_{k+1}, determined by a multiplier, M_k, times the current gain value, g_k.

A typical value for the maximum step size decrease, a, in speech coding is 0.8, and the maximum step size increase, b, is 2.4. As a general rule, all Jayant "One-Word Memory" type adaptive quantization systems have a much more rapid quantizer step size increase than decrease.

3.4.4 Multiplier Theory

Suppose g_k is replaced by a constant scaling, $g_k = \rho^k$, to represent maximum increase or decrease trending in the gain. We consider how this affects the upper block transfer function of Figure 3.6.

Now that we have an expression for g_k in terms of k, we are able to transform the system shown in Figure 3.6 to the condensed block diagram shown in Figure 3.7. This is achieved by first accommodating the multipliers into the error filtering block. Assume we have a system with a transfer function given by the operator $\mathbf{H}(z^{-1}) = h_0 + h_1 z^{-1} + h_2 z^{-2} + \ldots$, then we can write

$$\begin{aligned}
\rho^{-k}\mathbf{H}(z^{-1})\rho^k &= \rho^{-k}\left\{h_0\rho^k + h_1 z^{-1}\rho^{k-1} + h_2 z^{-2}\rho^{k-2} + \ldots\right\} && (3.26)\\
&= h_0 + h_1(\rho z)^{-1} + h_2(\rho z)^{-2} + \ldots && (3.27)\\
&= \mathbf{H}\left((\rho z)^{-1}\right). && (3.28)
\end{aligned}$$

We are thus able to incorporate the quantizer gain, $g_k = \rho^k$, into the error filtering block from Figure 3.6, to give the upper block shown in Figure 3.7.

The passivity condition for the upper block needed to establish coder stability is thus made more stringent for smaller ρ and relaxed for larger ρ. That is, the maximum rate of decay of quantizer gain (quantizer step size) affects the coder stability.

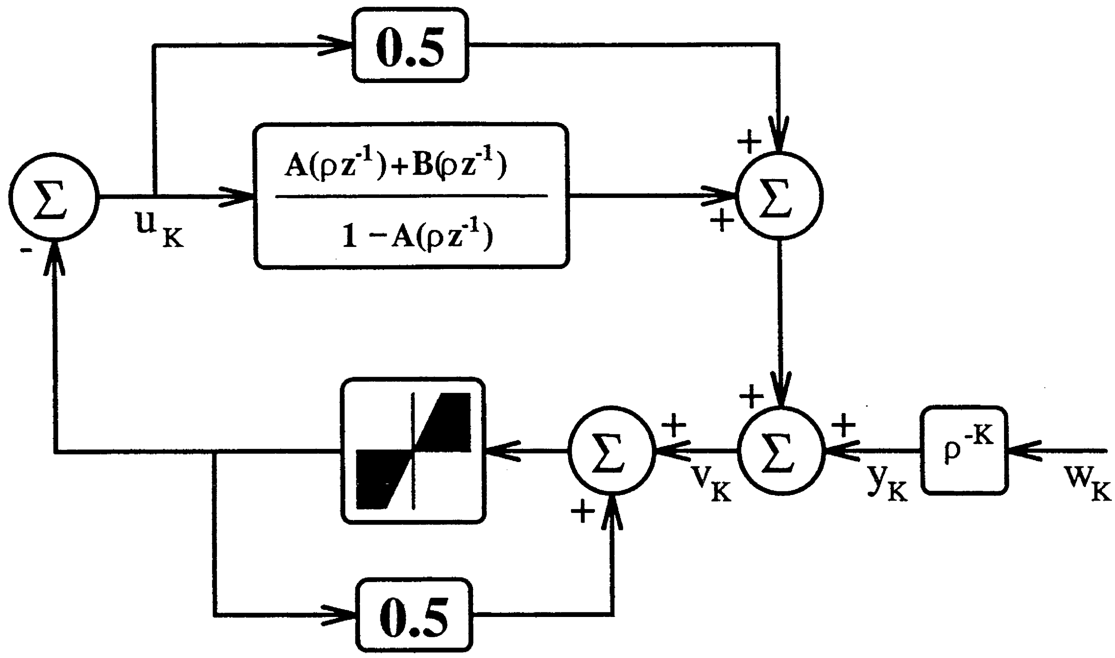

Figure 3.7: Condensed Block Diagram of the Passivity Error System

3.4.5 Stability Theorem

We are now in a position to apply the Passivity Theorem 3.1 directly to the system as presented in Figure 3.7, and arrive at our stability theorem.

In order to apply Theorem 3.1, we now take

$$\mathbf{H_1} = \left[\frac{A((\rho z)^{-1}) + B((\rho z)^{-1})}{1 - A((\rho z)^{-1})}\right] + \frac{1}{2}, \tag{3.29}$$

$$y_k = \rho^{-k} w_k, \tag{3.30}$$

and this leads to our stability theorem.

Theorem 3.3 *Suppose the $A(z^{-1})$ and $B(z^{-1})$ polynomials satisfy the following frequency response criterion*

$$\mathbf{Re}\left\{\frac{A(\rho^{-1}e^{-j\theta}) + B(\rho^{-1}e^{-j\theta})}{1 - A(\rho^{-1}e^{-j\theta})}\right\} > -\frac{1}{2}, \quad \forall\theta \in [0, 2\pi) \tag{3.31}$$

where $\rho < 1$ is the maximum decay factor of the gain g_k, then the ADPCM system is stable in terms of Definition 3.1.

Corollary 3.1 *If the above conditions are met, then for any coder input sequence, the quantizer output sequence, $\{Y_k\}$, tends to a sequence independent of the initial conditions of the coder, in finite time.*

Corollary 3.2 *A larger adaptive quantizer decay, and hence smaller ρ, implies more stringent conditions on the $A(z^{-1})$ and $B(z^{-1})$ polynomials for the system to remain stable.*

3.4.6 Proof of Stability Theorem

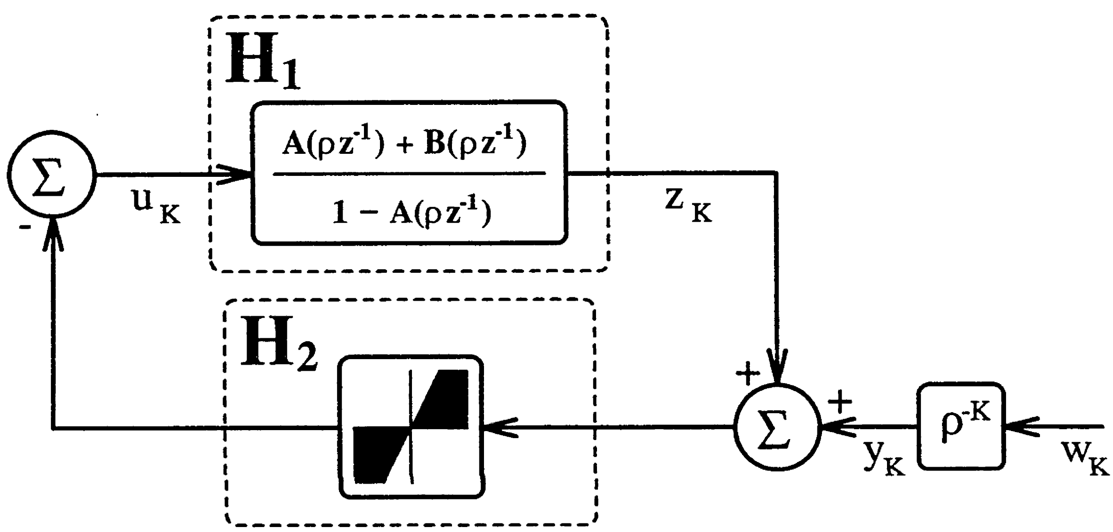

Figure 3.8: Error System Schematic for Alternative Theorem Proof

The theorem proof appears to be a simple augmentation of the proof contained in the Kennedy and Johnson paper[114]. After expressing the system with adaptive quantization in the form shown in Figure 3.7, the modifications to the Kennedy and Johnson proof are quite elementary. Unfortunately an element of mathematical imprecision exists with the proof contained in [114].[1] The identified problem occurs due to the introduction of the feedback factor around the sector bounded non-linearity as shown in Figure 3.6. This feedback factor is introduced to attempt to relax the stability requirement, but strictly results in an ill-defined output from the non-linearity. This is due to the fact that multiple solutions have been inadvertently introduced with the use of the loop transformations.

[1]Many thanks to Prof. Soura Dasgupta, Department of Electrical and Computer Engineering, University of Iowa, for first identifying this issue, and Dr. Rod Kennedy, Australian National University, for discussions.

We avoid the same problem via the use of an alternative approach to the proof utilizing the Popov-Kalman-Yacubovitch Lemma, which can be found in [3, 181]. The discretized version of the lemma states:

PKY Lemma: The system $\mathbf{W}(z)$ with state variable realization

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ z_k &= Cx_k + Du_k \end{aligned} \tag{3.32}$$

is discrete strictly positive real iff there exist real matrices P and L, with P positive definite, and constants ω and γ, such that:

$$\begin{aligned} A^TPA - P &= -LL^T - \gamma^2 I \\ A^TPB &= C^T + \omega L \\ \omega^2 &= 2D - B^TPB \end{aligned}$$

We now consider the system with $\mathbf{H_1}(z)$ strictly proper, as is the case with $\mathbf{H_1}(z)$ in the error system of Figure 3.8.

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ z_k &= Cx_k \end{aligned} \tag{3.33}$$

where $\mathbf{H_1}(z) + D$ is discrete SPR (Strictly Positive Real).

The loop in Figure 3.8 thus has the following dynamical description:

$$x_{k+1} = Ax_k - B\mathbf{H_2}(Cx_k). \tag{3.34}$$

We consider a candidate Lyapunov function for 3.34:

$$V_k = x_k^T P x_k, \tag{3.35}$$

with P from the PKY Lemma with $\mathbf{W}(z) = \mathbf{H_1}(z) + D$.

$$V_{k+1} - V_k = x_{k+1}^T P x_{k+1} - x_k^T P x_k, \tag{3.36}$$

and with substitution for x_{k+1} from equation 3.34:

$$\begin{aligned} V_{k+1} - V_k &= (x_k^T A^T - \mathbf{H_2}(Cx_k)B^T)P(Ax_k - B\mathbf{H_2}(Cx_k)) - x_k^T P x_k \\ &= x_k^T A^T PAx_k - x_k^T P x_k - \mathbf{H_2}(Cx_k)B^T PAx_k \\ &\quad - x_k^T A^T PB\mathbf{H_2}(Cx_k) + \mathbf{H_2}(Cx_k) B^T PB\mathbf{H_2}(Cx_k) \end{aligned}$$

With the use of the results of the PKY Lemma, and some more simple algebraic manipulation,

$$\begin{aligned} V_{k+1} - V_k &= x_k^T(-LL^T - \gamma^2 I)x_k - \mathbf{H_2}(Cx_k)(C + \omega L^T)x_k \\ &\quad - x_k^T\left(C^T + \omega L\right)\mathbf{H_2}\left(Cx_k\right) - \left(\omega^2 - 2D\right)\left[\mathbf{H_2}\left(Cx_k\right)\right]^2 \\ &= x_k^T(-LL^T - \gamma^2 I)x_k - \mathbf{H_2}(Cx_k)Cx_k - x_k^T C^T \mathbf{H_2}(Cx_k) \\ &\quad - \omega\mathbf{H_2}(Cx_k)L^T x_k - \omega x_k^T L\mathbf{H_2}(Cx_k) - (\omega^2 - 2D)[\mathbf{H_2}(Cx_k)]^2 \\ &= -\gamma^2 x_k^T x_k - 2\mathbf{H_2}(Cx_k)Cx_k - (\omega\mathbf{H_2}(Cx_k) + Lx_k)^2 + 2D[\mathbf{H_2}(Cx_k)]^2 \end{aligned}$$

The quantizer construction results in a sector bounded non-linearity for the $\mathbf{H_2}$ block between lines of slope 0 and 2. Hence we have the inequality

$$[\mathbf{H_2}(\alpha)]^2 \leq 2\mathbf{H_2}(\alpha)\alpha. \tag{3.37}$$

This gives us the following inequality for the change in the Lyapunov function:

$$V_{k+1} - V_k \leq -\gamma^2 x_k^T x_k - 2(1 - 2D)\mathbf{H_2}(Cx_k)Cx_k - (\omega\mathbf{H_2}(Cx_k) + Lx_k)^2, \tag{3.38}$$

which, provided $2D \leq 1$, translates to exponential asymptotic stability of 3.34.

Q.E.D.

3.5 Discussion

A direct consequence of the stability result presented in this paper is that we are able to say that it is more difficult to ensure stability for more highly adaptive systems. This corresponds to smaller values of the parameter, ρ, relating to the maximum rate of quantizer step size decrease. Thus we relate the dynamic range capacity of the quantizer to signal correlation properties as present in the predictor polynomials $A(z^{-1})$ and $B(z^{-1})$. Previously known experimentally, this is now theoretically established and linked to the maximum decay rate of the quantizer gain. This formalization of experimental observation allows a stability comparison between differing ADPCM systems.

We note that according to the above Stability Theorem 3.3, an ADPCM system incorporating an adaptive quantizer with no step size decreases is inherently stable. Obviously such a system is likely to have fairly poor performance. In the extreme case, consider a quantizer with only step size increases, and thus the maximum step size decay rate, ρ, is larger than 1. For larger ρ, the system is more inherently stable. However, such a system obviously will not produce good coding performance.

Theorem 3.3 could be viewed as another justification for the fact that adaptive quantization systems similar to the Jayant "One-Word Memory" algorithm scale down the quantizer step size much more slowly than scaling up. Moreover, it implies that when experiencing stability problems with an adaptive quantizer in an ADPCM loop, one should concentrate on reducing the maximum rate of step size decrease. Of course, after the stability problems have been redressed, the performance can then be maximised by adjusting the step size increases.

It would be desirable to view the above stability theorem and see an indication of an 'optimal' ADPCM system. Of course this is not possible. The theorem does indicate that systems with less step size adaptation are more stable. However, such systems may not necessarily obtain good performance. As mentioned in the introduction, Honda and Itakura [84], observed that a system with limited quantizer step size adaptation, and variable bit allocation, had superior stability properties while maintaining performance. Theorem 3.3 seems to explain this stability improvement.

According to the theorem, any ADPCM approach that has limited quantizer step size adaptation will be relatively stable. Alternative approaches to ADPCM that use techniques such as variable bit allocation are thus attractive for further study in order to achieve techniques that exhibit improved stability, without necessarily sacrificing performance. These issues are part of our motivation for the research on variable bit rate ADPCM presented in Chapter 4.

3.6 Stability Conclusions

We have produced a sufficiency theory for stability of ADPCM systems, where we have defined stability as the rejection of previous errors in encoder state. The stability theorem presented gives a frequency response condition on the system to ensure stability.

Although no account is taken of the predictor adaptation, it is noted that the much faster rate of quantizer adaptation implies that it is likely to have the largest effect on stability.

It is seen that the frequency response requirement for stability is more difficult to meet for highly adaptive quantization. However, adaptation is required to achieve performance, and so there is a performance versus stability trade-off.

Also of interest is the fact that stability has been found to depend on the maximum rate of quantizer step size decrease. This appears to be a justification of the shape of the quantizer scaling factor curves used in Jayant "One-Word Memory" type adaptive quantization.

ADPCM structures with some alternative approaches to the basic quantizer step size adaptation idea are under consideration. The theory presented here suggests that step size variation reductions will improve stability, and alternative adaptation approaches could maintain performance. This acts as partial motivation for the variable rate ADPCM speech coding approach in Chapter 4.

Chapter 4

Arithmetic Coding and ADPCM

4.1 Chapter Motivation

CELP (Code Excited Linear Prediction) approaches to speech coding have been comprehensively studied throughout the last decade. It is well known that CELP approaches are capable of producing high quality speech output at bit rates where other approaches such as ADPCM (Adaptive Differential Pulse Code Modulation) tend to perform poorly. Unfortunately, the CELP approach is often extremely computationally expensive, largely due to the necessity of a codebook search. In this sense, CELP approaches are a 'brute-force' attempt at the speech coding task, and do not necessarily represent an optimal engineering solution to the problem. Indeed, recent work by Kao and Baras[109] has shown that in FS1016 4.8 kbps CELP[25, 26], the stochastic codebook can be replaced with a 'deterministic codebook'. This effectively eliminates the need for a codebook search, whilst at the same time providing an improvement in output speech quality.

In this chapter we consider ADPCM as an alternative to CELP for some applications. This is a somewhat unusual approach, as ADPCM and CELP are commonly considered as speech coding systems operating in fairly separate ranges of the speech coding bit rate spectrum. There is good reason for this separation, since ADPCM usually performs very poorly at rates of 16 kbps and less, whilst CELP approaches above this rate often encounter serious computation resource problems. We consider the first of these issues, and attempt to analyse ADPCM at 16 kbps and less. The first problem we encounter is a stability problem within standard ADPCM systems. An analysis of stability has been presented in Chapter 3, and is discussed in the next section, leading to the conclusion that slower adaptation in the quantizer will improve the stability.

Allowing the use of variable bit rate entropy coding within an ADPCM system results in an approach where a fixed uniform quantizer is used in an attempt to maximise SNR (Signal to Noise Ratio) measures. As a result of the fixed quantizer, the stability problems are mitigated. We can also see that the use of entropy coding to maximise SNR results in the effective exploitation of silence periods within the speech. Other benefits of the entropy coding approach are related to the coder design flexibility obtained, while disadvantages of the approach relate to the variable bit rate coder output, and the consequent problems with transmission errors.

To provide a brief overview of the remains of the chapter, the next two sections consider stability issues and the maximisation of SNR measures as motivating forces for the variable bit rate ADPCM system presented in the following section. Some useful enhancements to the basic system are then discussed before proceeding to examine the flexibility and computational considerations of the approach. After this, some further practical issues are covered including a section on possibilities for delayed decision coding, and the chapter finishes with a conclusion section.

4.2 ADPCM Destabilization Effects

As from the previous chapter, fixed rate ADPCM systems such as that shown in block form in Figures 4.1 and 4.2, generally consist of both adaptive quantizers and adaptive predictors. For speech coding, this adaptation is used to account for the wide variations in the input signal statistics.

Consideration of the ADPCM encoder reveals a feedback loop and it is easy to see that, depending on the adaptation strategies for the quantizer and predictor, stability problems may exist. Several authors have noted the existence of stability problems, including [45, 84, 99], and some studies of stability have been undertaken [115, 123, 183].

Toll-quality speech coding is achieved by ADPCM at 32 kbps (CCITT Recommendation G.721), but an ADPCM system at 24 kbps shows a significant decrease in performance over the 32 kbps system. Simply dropping the rate to 16 kbps can give an ADPCM system with extremely poor performance, with evidence of severe stability problems being observed in the decoded output speech. Jayant and Noll[106] show it is possible to tune the adaptive quantizer effectively to eliminate this stability problem, but the cost is that the performance of the 16 kbps ADPCM system is extremely poor. Tabulations of 'loading factors' are provided by Jayant and Noll, to be used with adap-

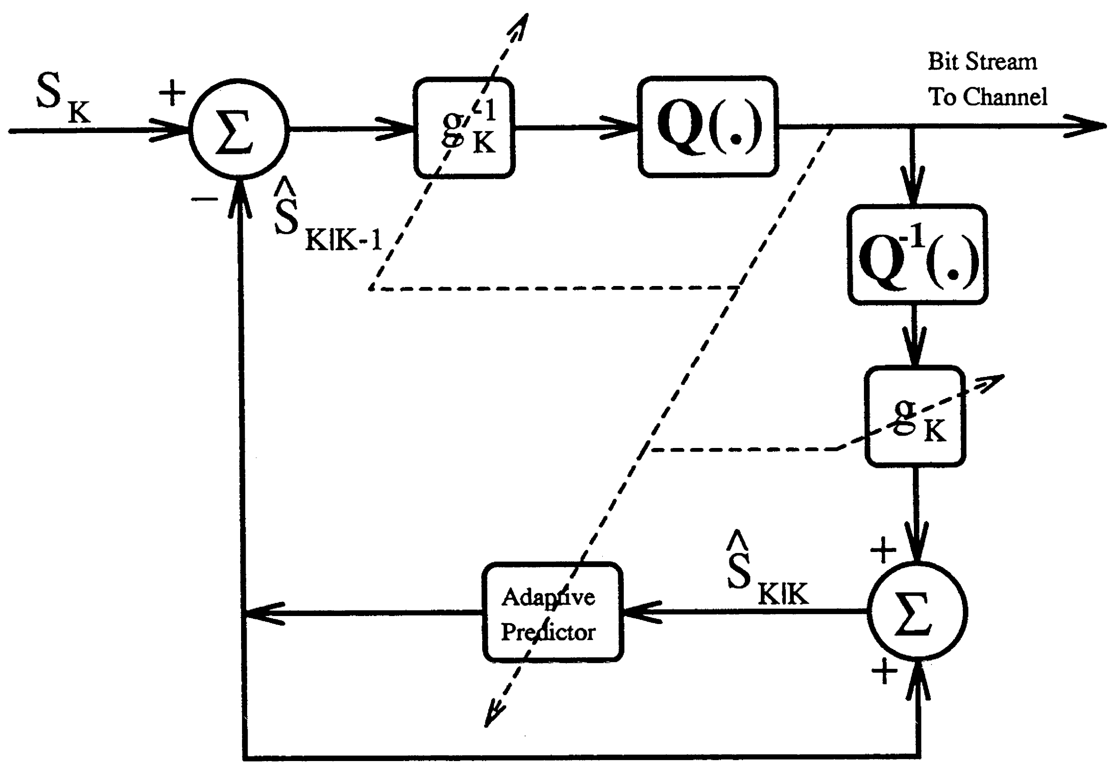

Figure 4.1: Fixed Rate ADPCM Encoder Block Schematic

tive quantizers of various numbers of bits to give a reasonable stability/performance compromise.

Chapter 3 has shown how the stability of the ADPCM loop is closely linked to the rate of quantizer step size decrease. Slower adaptation in the quantizer step size is shown to improve stability, but with standard fixed rate ADPCM systems, this usually comes at a performance cost.

Remark 4.1 Chapter 3 dealt with a stability analysis of CCITT G.721 32 kbps ADPCM which incorporates a pole-zero predictor. For an all-pole predictor, as considered in this chapter, the results are still valid, as this is simply a special case of the pole-zero predictor where no zeros are present. Using the notation from the previous chapter, $B(z^{-1}) = 0$.

In this chapter we investigate the approach of effectively eliminating the instability discussed in Chapter 3 by the use of a quantizer with fixed step size. The next section considers the use of entropy coding to improve the performance of the fixed quantizer ADPCM system.

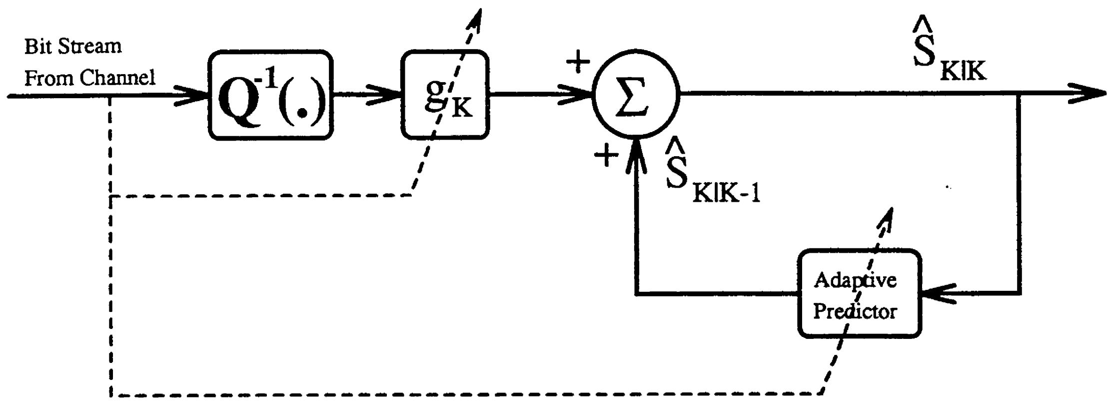

Figure 4.2: Fixed Rate ADPCM Decoder Block Schematic

4.3 Entropy coding and SNR

Variable bit allocation (for fixed output bit rate coding) has been proposed for use in many systems[50, 105, 193], and more recently variable rate coders have received a significant amount of attention[28, 51, 56, 66, 137]. The ability to use variable rate entropy coding within a speech coder can result in significant performance improvements albeit at the possible expense of delay. We consider in this section the maximization of SNR measures. While SNR measures are not necessarily particularly useful when assessing speech quality, it is a good place to start.

Remark 4.2 An overview of Information Theory and entropy coding is presented in Appendix A. This appendix attempts to provide a brief overview of this area of source coding, and references are given for those readers interested in more detail. Fixed rate codes, Huffman codes, Block Huffman codes, Arithmetic Coding, and Quasi-Arithmetic Coding are all introduced, with the common elements of the approaches outlined.

Within a predictive coding system such as ADPCM, the quantization errors added to the prediction difference signal become the quantization errors in the reconstructed speech. Therefore, in order to minimise the error, or noise, power in the reconstructed speech, it is sufficient to minimise the noise power in the prediction difference signal.

We assume that the prediction difference signal is uncorrelated, since the aim of the predictor is to remove the correlation (whiten or remove redundancy) from the speech signal, and we assume that it performs this task well. (Later, and in particular with the introduction of the Kalman Filter in Chapter 5, it is shown that this is strictly not the case, but the assumption is commonly made, and is adequate for the present considerations.) Due to changes in input signal statistics, and dynamic range, it is logical to assume that the prediction difference signal will exhibit a considerable range of variance values over time. This fact is the usual motivation for an adaptive quantizer.

We further assume that the prediction difference signal can be modelled by a Laplacian distribution, and in Appendix E we attempt to display that a constant step size uniform quantizer has the effect of maximising the SNR of the prediction difference signal. Later in the chapter we also consider the maximisation of SNR of a perceptually weighted signal, and segmental SNR measures.

Remark 4.3 The entropy coded uniform quantizer is known to be optimal in the high rate situation[77], and has also been shown to obtain effectively the same performance as the optimal quantizer even when the high rate assumption is violated[57]. Appendix E considers the effect of quantizer input variance changes, and whether quantizer step size modifications are required to maximise SNR.

Intuitively, we also expect to see a considerable SNR gain from the variable rate ADPCM system over standard fixed bit rate approaches, since we would expect a large advantage from exploiting silence periods within the speech. However, it is important to note that the approach investigated in this chapter is significantly different from a simple silence exploitation approach.

4.4 Arithmetic Coding ADPCM – System Concept

The stability problems with fixed rate ADPCM systems are related to the fact that the fixed rate quantizer has conflicting requirements. It must have levels spaced closely together to obtain good coding performance, but must have levels spaced far apart to ensure stability by being able to track rapid changes in input signal statistics. For lower bit rate systems, these two requirements become increasingly difficult to meet simultaneously.

Lifting the restriction of a fixed rate system is one approach that can help to relieve the stability problems. Moving away from a sample-by-sample ADPCM approach to a vector basis CELP approach is another. CELP approaches have been well studied and extremely successful in practice, but codebook searches are required and these imply a significant computation cost. This chapter is concerned with a variable bit rate ADPCM approach that can be viewed as an alternative to CELP (for some applications), and avoids the computational cost of the codebook search.

Our proposed system, introduced briefly in [185], is presented in block form in Figure 4.3. Here it can be seen that the basic feedback loop of ADPCM is intact. However, the quantizer within the loop is now non-adaptive with many levels (practically infinite) to avoid overload distortion, and adaptive Arithmetic Coding is used as post-processing on the quantizer output data stream. We are thus dealing with an instantaneous variable bit rate system. This system we refer to as Arithmetic Coding ADPCM (AC-ADPCM).

Remark 4.4 It is important to note that the 'infinite' number of levels in the quantizer certainly does not imply zero granular distortion, only that the quantizer does not have any overload regions. Thus the quantizer output entropy is finite.

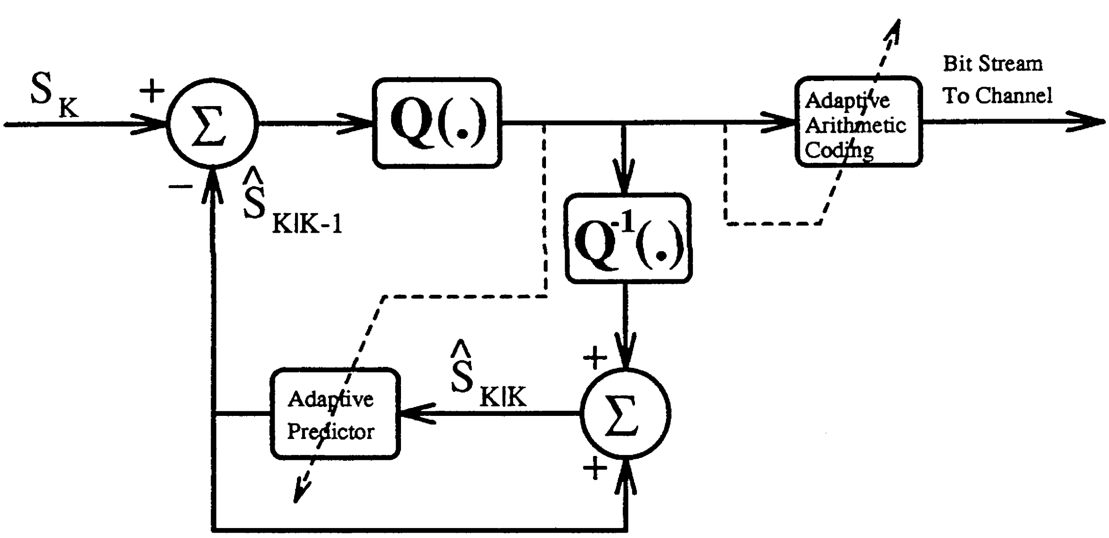

Figure 4.3: Arithmetic Coding ADPCM Encoder Block Diagram

Arithmetic Coding[188], introduced briefly in Appendix A, is a practically optimal entropy coding scheme. It is used here (Figure 4.3) to encode the quantizer output with only the number of bits required by information theoretic considerations, based on the probability of that quantization level being used. The prime difference between Arithmetic Coding and Huffman coding is that it does not suffer from the disadvantage of requiring each source symbol to be encoded with an integral number of bits, and hence better coding performance is possible. Another important concern is the ease with which adaptive symbol distributions are handled.

The system shown in Figure 4.3 is similar in some ways to one investigated by Howard and Vitter[86, 87] for DPCM compression of images. However, there are some significant differences for an ADPCM system for coding speech. Obviously the predictor in Figure 4.3 is adaptive to account for variations in input signal statistics. The Arithmetic Coding block is likewise adapted to account for changing distributions of the prediction difference signal.

A quantizer with no overload distortion means there exists ample ability to track rapid changes in input signal statistics, without feedback of large errors into the predictor. This feedback process has already been noted as the cause of instability with adaptive quantization schemes such as the Jayant 'One-Word Memory' algorithm[106] in fixed rate ADPCM.

As well as obtaining an advantage over fixed rate ADPCM systems through the elimination of the stability problems, the AC-ADPCM system also has the advantage of exploiting inactive periods of input, and highly predictable sections, to reduce the average bit rate. Of course, it is dependent upon the application as to what extent this type of saving is possible or desirable, due to restrictions on output bit rate flexibility. However, throughout this thesis, only the basic approach is considered, with applications issues being of secondary importance (Chapter 10).

The linear predictor shown in Figure 4.3 is updated in a backwards adaptive manner, using auto-correlation analysis, and Levinson-Durbin recursion, similar to CCITT Recommendation G.728 16 kbps LD-CELP. Hence (initially at least) we use a 50th order predictor, which is updated every 20 samples (2.5 ms).

Remark 4.5 The choice of using similar system components as in LD-CELP is certainly significant. We wish to draw a performance comparison between the AC-ADPCM system and LD-CELP. This is a logical approach considering that LD-CELP and ADPCM are both low delay coders, but as noted they usually operate in different ranges of the speech coding spectrum. Toll quality coding is provided by LD-CELP at 16 kbps, but 32 kbps must be used to obtain equivalent quality with a standard fixed rate ADPCM system. We would like to ascertain how the variable rate system performs at the lower rates.

For the purposes of our initial AC-ADPCM system, the arithmetic coding block shown in Figure 4.3 uses source symbol probabilities based on the assumption that the input to the quantizer can be modelled as a Laplacian distribution. Adaptation of the

Arithmetic Coding block occurs via the use of a backwards adaptive variance estimate[1] in conjunction with the Laplacian distribution.[2] We do not attempt to justify the Laplacian assumption here, but note that if the actual distribution is not Laplacian, this implies that we are able to improve coding performance. Later discussions (Section 4.6) will consider the use of trained distribution tables to provide the probabilities required by the arithmetic coding.

The first point we note after having constructed the basic system as above is that it does, in fact, practically eliminate the stability problems that occur within fixed rate ADPCM. We are thus able to code at very low bit rates, with somewhat graceful degradation in output quality as the bit rate is decreased. The smooth degradation in SNR and segmental SNR measures as the average bit rate is reduced is shown in Figure 4.4. The curves displayed are obtained via simulations with an input file consisting of four male and four female sentences, with only slight pauses between each. Although the input file length of just over 22 seconds would appear small for statistical type considerations, it is sufficient to allow general trends and basic comparisons to be made.

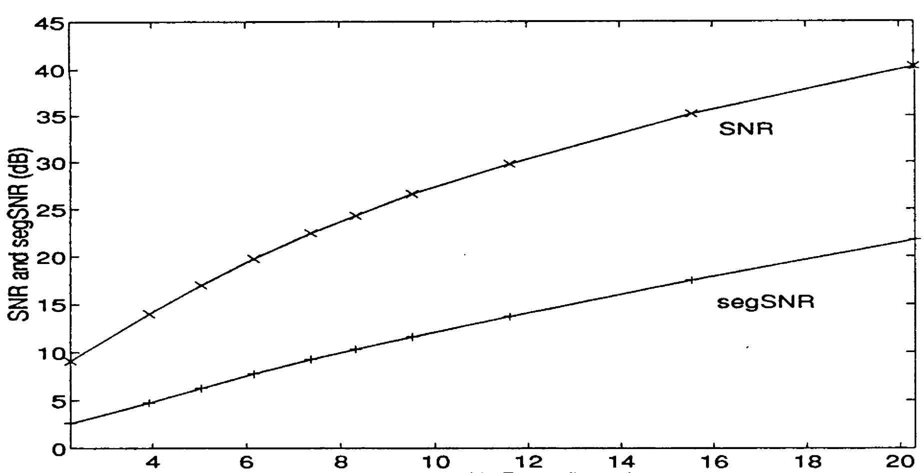

Figure 4.4: SNR and Segmental SNR versus Average Bit Rate

The general trend of the AC-ADPCM system for graceful degradation with decreas-

[1]An estimate of the form $\hat{\sigma}_k^2 = 0.97\hat{\sigma}_{k-1}^2 + 0.03Y_k^2$ is used for the variance, where Y_k is the quantizer output $\mathbf{Q}\left[S_k - \hat{S}_{k|k-1}\right]$.

[2]Note that the adaptive quantizer can be viewed as implicitly containing a variance estimate[37], and hence the variance estimate is not a new ADPCM system component required in Figure 4.3.

ing average bit rate is clear from Figure 4.4. Also from the graph we are able to note the very high SNR values obtained. In fact, the SNR does not fall below 20 dB until the rate is reduced to around 6 kbps. It should be clear that the high SNR values are a result of effectively maximising SNR through the use of the constant step size uniform quantizer (Appendix E), and that SNR is not always a good indication of speech quality. The segmental SNR figures shown on the graph perhaps give a slightly more reliable indication of the subjective performance.

In order to discuss further the performance of the Arithmetic Coding ADPCM approach, and in particular to be able to draw some comparisons with LD-CELP, we present some selected results in Table 4.1.

Coder:	Bit Rate:	SNR (dB):	segSNR (dB):
AC-ADPCM	~ 16 kbps	35.61	17.82
AC-ADPCM	~ 12 kbps	29.79	13.69
AC-ADPCM	~ 8 kbps	23.70	9.93
AC-ADPCM	~ 4 kbps	14.34	4.92
LD-CELP	16 kbps	15.93	18.19
LD-CELP (no PW)	16 kbps	17.40	19.56

Table 4.1: AC-ADPCM SNR Values for Selected Rates

From Table 4.1 it is clear that while the SNR obtained with the AC-ADPCM approach is significantly better than that obtained by LD-CELP, the segmental SNR values are lower. This indicates the importance of basing performance comparisons of the two differing approaches on subjective criteria. Also included in the table are values for an LD-CELP system in which the perceptual weighting filter has been removed. As the basic AC-ADPCM system has no perceptual weighting (at this stage) this provides a fairer comparison. Postfiltering is also not used in either the basic AC-ADPCM system, or for the LD-CELP results here.

Remark 4.6 It is important to realise that the bit rates listed for the AC-ADPCM system in Table 4.1 are approximate only. The actual bit rates are dependent upon the entropy, which is related to the quantizer step size. Hence by adjusting the quantizer step size it is possible to obtain average bit rates very close to the nominal values in the table. In order to obtain 'fair' comparisons, where a bit rate is indicated as ~ 16 kbps, the actual bit rate will be slightly less than this nominal value.

Informal listening tests confirm that although the SNR values are much higher for AC-ADPCM than LD-CELP, the subjective performance at the 16 kbps average rate is very close to that of LD-CELP. For low quality audio output (as might be expected within a standard telephone handset), it is impossible to distinguish a significant performance difference between 16 kbps average rate AC-ADPCM and LD-CELP either with or without perceptual weighting.

AC-ADPCM at an average rate of 12 kbps is still very difficult to distinguish from LD-CELP with the use of the low quality audio output. However, it does appear that the level of audible background noise in the AC-ADPCM system at this rate is higher than that of LD-CELP. Significant levels of quantization noise are audible in the AC-ADPCM output at 8 kbps, and this can easily be judged as inferior in output quality to that of both LD-CELP with and without perceptual weighting. The output quality at the 4 kbps average rate is quite poor, with some loss of unvoiced speech, and substantial levels of audible quantization noise.

Remark 4.7 In some sense these comparisons are unfair, as we are dealing with an average bit rate for the AC-ADPCM system over speech with some periods of silence. An LD-CELP system that has been modified with the addition of Voice Activity Detection (VAD) and Discontinuous Transmission (DTX) in silence periods could be expected to have an average rate less than 16 kbps. However, such schemes must necessarily be conservative in their activity detection, and most performance gain is achieved during long periods of silence, such as are present in a two way conversation. The input used here is eight sentences with slight pauses between, and in this case it is believed that the comparison with LD-CELP is reasonable.

For high quality audio output, the results of the comparisons above are a little different. Even with an average rate of 16 kbps, the performance of AC-ADPCM is judged to be significantly inferior to that of LD-CELP. This is due to the existence of a non-negligible amount of audible quantization noise, which is not substantially noticeable with low quality audio. The quantization noise level is significantly higher at the 12 kbps average rate, and at 8 kbps the unvoiced speech experiences some perceptually annoying degradation. The performance at 4 kbps is fairly poor with both large levels of quantization noise audible during active speech sections, and some unvoiced sections effectively completely missing.

Remark 4.8 The degradation in unvoiced speech indicates an area for our attention. However, this should not cause great concern, as it is common for speech coders at 8 kbps and below to treat voiced and unvoiced speech sections differently. We return to this problem briefly towards the end of the next section.

Having noted that the subjective performance of AC-ADPCM at a 16 kbps average rate is inferior to that of LD-CELP, we point out that we are dealing with an extremely simple system and these early results are very encouraging. As noted the performance is far superior to that of 16 kbps fixed rate ADPCM, and we have eliminated the requirement for a codebook search as in LD-CELP. The reasonable performance obtained down to rates of around 8 kbps is also quite promising. The next section considers a number of possibilities for performance improvements to the basic AC-ADPCM system.

4.5 AC-ADPCM System Improvements

Several techniques widely used in signal, and particularly speech, coding can be applied to the basic AC-ADPCM system to improve performance. As these are well understood techniques, and secondary to the main (somewhat academic) thrust of this chapter, we only provide brief discussions of their application. Where appropriate we attempt to highlight any differences in the application of these techniques to the AC-ADPCM system over standard speech coding situations.

Throughout this section we particularly concentrate on the AC-ADPCM system at a 16 kbps average rate, and attempt to improve the performance to closer match that of LD-CELP.

4.5.1 Quantization Dither

In any practical speech coding system the introduction of quantization noise is a necessary evil. However, the properties of the quantization noise introduced should be carefully controlled to minimise the perceptual degradation in the output. An elementary requirement is that the quantization noise should be 'white'.

For high rate scalar quantization, the assumption that the quantization noise is white may be fairly accurate. However, at an average rate of 16 kbps (for 8 kHz sampled speech), this assumption needs to be investigated. Figures 4.5 to 4.7 show quantization noise auto-correlation functions. The quantization noise auto-correlation data was calculated from 10000 samples (over a second) of coder output data from the AC-ADPCM system with a 20th order predictor (speech data includes voiced,

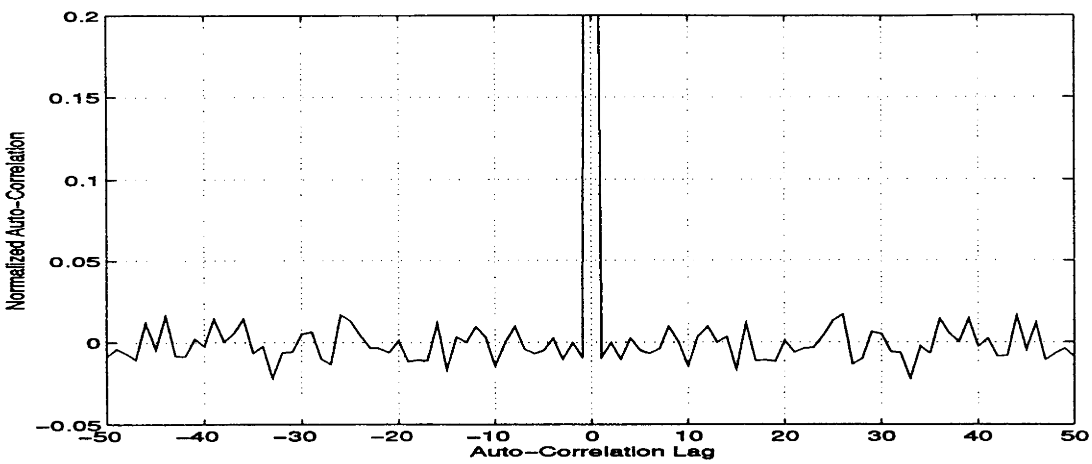

Figure 4.5: Auto-correlation of White 'Quantization' Noise

unvoiced and small silence periods). Only the first 50 auto-correlation lags are shown in the diagrams, and of these the first few delays are of most current significance to us. It is important to note that the graphs have been plotted using normalisation of the variance to one. In order to display the detail, the variance peaks are not shown on the graphs.

Figure 4.5 shows the auto-correlation function for white pseudo-random 'quantization' noise. This graph is designed to display the type of performance we desire, with only small auto-correlation values observed. Figure 4.6 presents the auto-correlation of the quantization noise for the 16 kbps average rate AC-ADPCM system. A number of values around the central variance peak can be seen to be somewhat larger than those in Figure 4.5, indicating a degree of non-whiteness of the quantization noise. This is displayed even more markedly in Figure 4.7 relating to the AC-ADPCM system at an 8 kbps average rate. However, even at 8 kbps, the non-whiteness of the quantization noise would appear to be relatively minimal.

One technique that is often used to improve the properties of the quantization noise is that of introducing a pseudo-random dither[106]. The dither signal may be subtracted at the decoder if synchronization of the encoder and decoder random generators exist. For applications where this is impossible, the addition of a dither that is not removed at the decoder may still give important subjective improvements, even with the additional quantization noise power incurred.

A number of simulations were performed with the addition and subtraction of a

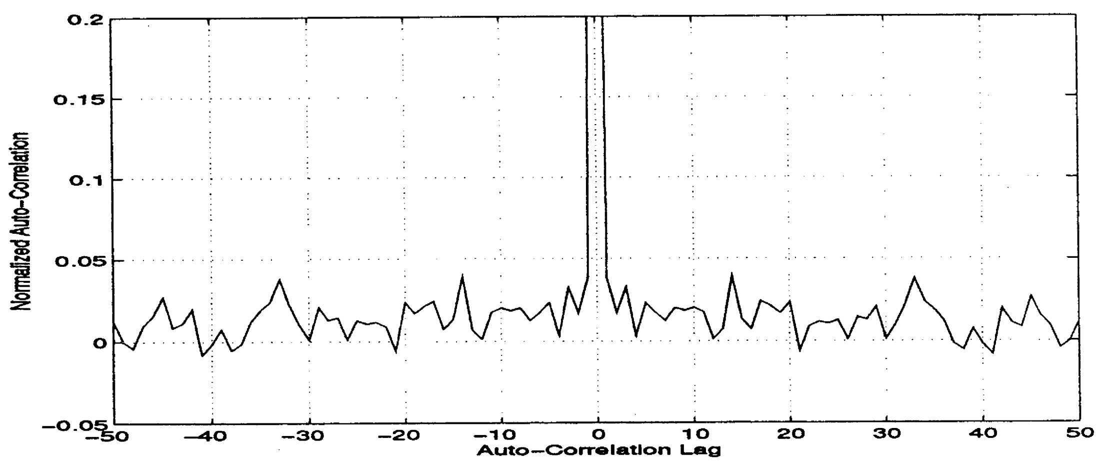

Figure 4.6: Auto-correlation of Quantization Noise for 16 kbps AC-ADPCM

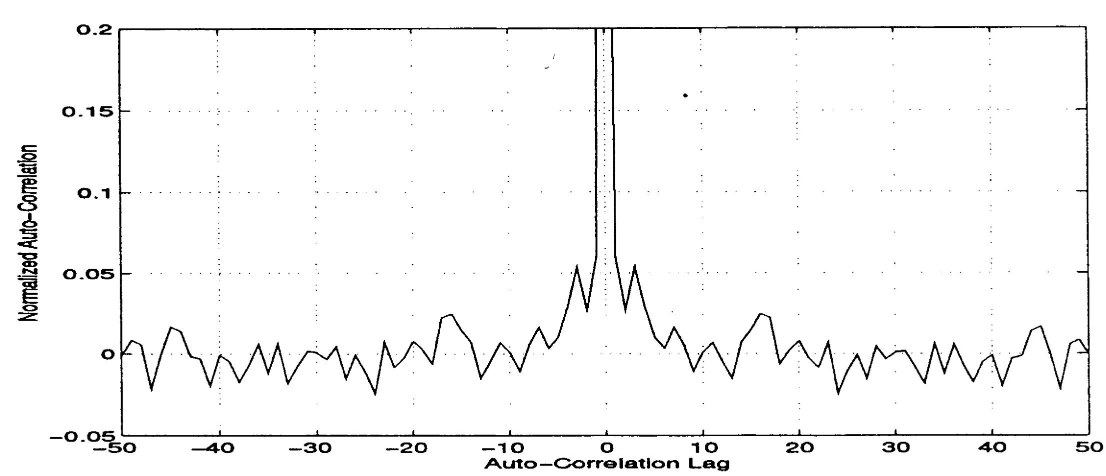

Figure 4.7: Auto-correlation of Quantization Noise for 8 kbps AC-ADPCM

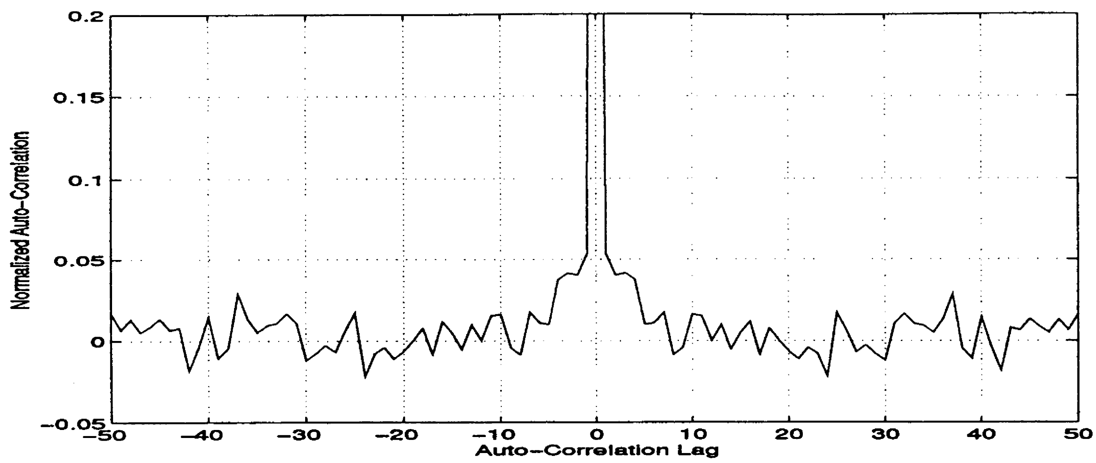

Figure 4.8: Auto-correlation of Dithered Quantization Noise for 8 kbps AC-ADPCM

pseudo random dither signal. The results of these simulations appear to display minimal improvement in perceptual output quality (judged by informal listening tests). The auto-correlation function for the dithered quantization noise in 8 kbps AC-ADPCM is displayed in Figure 4.8, showing negligible differences to where no dither was used (Figure 4.7).

Not subtracting the dither at the decoder significantly increases the overall quantization noise level, and produces a substantial subjective performance degradation. Hence this does not appear to be a valid option for the AC-ADPCM system.

The fact that even with the reasonably coarse quantization we have observed no substantial non-whiteness of the quantization noise, and no advantage to dithering, is not completely surprising. In a fairly loose sense, a similar type of result was observed with respect to the optimality of the uniform quantizer in maximising SNR measures[57]. Due to the minimal effects of dithering, and the fact that it requires synchronization of a pseudo random noise source between encoder and decoder, we provide no further consideration of the approach within this thesis.

4.5.2 Perceptual Weighting

A more sophisticated requirement on the quantization noise introduced during coding is that it should be spectrally shaped relative to the signal spectrum to exploit the masking property of the signal spectral peaks. The papers by Gibson[73] and Makhoul and Berouti[124] both include discussions of quantization noise spectral shaping via the use of Noise Feedback Coding (NFC) in ADPCM and APC (Adaptive Predictive Coding) systems. Atal[8] also discusses the use of quantizer noise spectral shaping, and investigates the control of the error spectrum at low bit rates.

Many CELP approaches, such as LD-CELP[33, 34], perform spectral shaping of the quantization noise by selecting a codevector from the codebook on the basis of a perceptually weighted MSE (Mean Square Error) criterion. We would like to show that the minimisation of a perceptually weighted MSE in CELP is equivalent to a noise feedback coding approach in ADPCM or APC.

Remark 4.9 It is important to note that the use of the terms ADPCM and APC are in the most general sense. In this way, both systems may incorporate both short and long term (pitch) predictors, multi-level quantizers, perceptual weighting, and postfiltering. The distinction between the two schemes is purely that ADPCM is backwards adaptive, and APC is forwards adaptive. Likewise the use of the term NFC is in regards to a general principle, and not a specific coding system.

For those readers unfamiliar with the general principle of noise feedback coding, rather than consulting the references mentioned above, an elementary description follows. Within a system employing noise feedback, the encoder quantizes the prediction difference signal with the addition of a filtered version of the previous quantization error. Using the notation defined in Chapter 3,

$$Y_k = \mathbf{Q}\left[S_k - \hat{S}_{k|k-1} + H(z^{-1})\left(S_{k-1} - \hat{S}_{k-1|k-1}\right)\right], \tag{4.1}$$

where $H(z^{-1})$ is the relevant noise feedback filter.

Figure 4.9 shows a portion of a CELP coder relevant to performing the minimum perceptually weighted MSE codebook search. The prediction $\hat{S}_{k|k-1}$ is added to the (quantized) excitation sample $\hat{e}_k$ from the codebook to form the reconstructed value $\hat{S}_{k|k}$. This is then subtracted from the input sample S_k, and the difference, $\tilde{e}_k$ is filtered by the perceptual weighting filter to form the perceptually weighted difference sample P_k.

Remark 4.10 The notation used here is almost identical to that used in Chapter 3, with the exception being that we are no longer dealing with a class of input signals as for the stability analysis. Hence the error signal $\tilde{e}_k$ is not defined as in equation 3.2, but rather is defined as: $\tilde{e}_k \triangleq \bar{e}_k - \hat{e}_k$, where $\bar{e}_k = S_k - \hat{S}_{k|k-1}$ (from Figure 3.2).

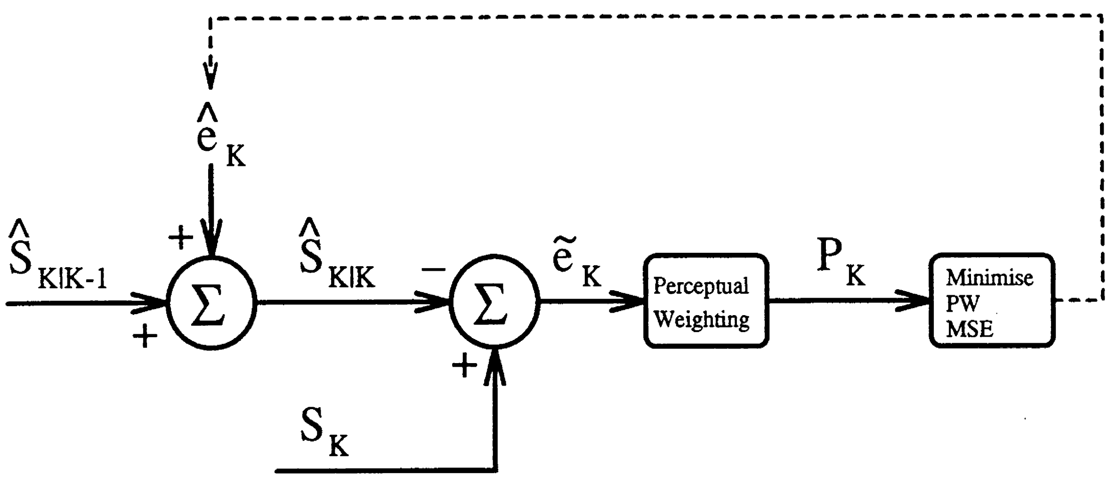

Figure 4.9: Partial Schematic of CELP Codebook Search Procedure

The above process is repeated for samples at time $(k+1)$ through to $(k+n-1)$, where there are n samples in the codevector. The (perceptually weighted) MSE for the codevector is then calculated as

$$MSE = \frac{\sum_{i=k}^{k+n-1} (P_i)^2}{n}. \tag{4.2}$$

The codevector corresponding to the minimum (perceptually weighted) MSE is then selected.

In LD-CELP and many other systems, the perceptual weighting filter has the form:

$$W(z^{-1}) = \frac{1 - A(z^{-1}/\gamma_1)}{1 - A(z^{-1}/\gamma_2)}, \tag{4.3}$$

where $A(z^{-1})$ is a polynomial in the backward shift operator z^{-1}, corresponding to the all-pole predictor used for speech. The factors γ_1 and γ_2 are used for bandwidth expansion of the poles and zeros to form $W(z^{-1})$, and are chosen in LD-CELP to be 0.9 and 0.6 (respectively). LD-CELP also specifies $A(z^{-1})$ to be 10th order and derived from LPC analysis on the unquantized (clean) speech, which is possible as the perceptual weighting filter is not used in the decoder.

The ADPCM system can be (loosely) viewed as a CELP system with a codebook of size $n = 1$. Hence the ADPCM quantizer is seen as being equivalent to selecting the excitation sample $\hat{e}_k$ giving rise to the least output distortion (in a perceptually weighted MSE sense).

From (4.3), and noting that $A(z^{-1})$ is a strictly proper polynomial in the delay operator z^{-1}, we can express the transfer function $W(z^{-1})$ in the form of a semi-infinite sum:

$$W(z^{-1}) = 1 + h_1 z^{-1} + h_2 z^{-2} + h_3 z^{-3} + \ldots, \tag{4.4}$$

with appropriate definitions for $h_i, i = 1..\infty$.

We thus obtain the expression for the perceptually weighted difference,[3]

$$\begin{aligned} P_k &= (S_k - \hat{S}_{k|k}) + h_1(S_{k-1} - \hat{S}_{k-1|k-1}) + h_2(S_{k-2} - \hat{S}_{k-2|k-2}) + \ldots \qquad (4.5) \\ &= (S_k - \hat{S}_{k|k-1} - \hat{e}_k) + h_1(S_{k-1} - \hat{S}_{k-1|k-1}) + h_2(S_{k-2} - \hat{S}_{k-2|k-2}) + \ldots \end{aligned}$$

The task of the quantizer is thus to minimise $(P_k)^2$, which is performed by selecting the excitation as:

$$\hat{e}_k = \mathbf{Q}\left[(S_k - \hat{S}_{k|k-1}) + h_1(S_{k-1} - \hat{S}_{k-1|k-1}) + h_2(S_{k-2} - \hat{S}_{k-2|k-2}) + \ldots\right], \tag{4.6}$$

where $\mathbf{Q}(.)$ represents the standard quantizer.

Defining $W^1(z^{-1})$ in terms of $W(z^{-1})$ as:

$$W^1(z^{-1}) = W(z^{-1}) - 1, \tag{4.7}$$

the quantization operation is expressed as

$$\hat{e}_k = \mathbf{Q}\left[(S_k - \hat{S}_{k|k-1}) + W^1(z^{-1})(S_k - \hat{S}_{k|k})\right], \tag{4.8}$$

which is identical in structure to the noise feedback coding approach shown in equation 4.1 (with $W^1(z^{-1}) = z^{-1}H(z^{-1})$).

The choice of $W^1(z^{-1})$ needs some clarification. Some algebra reveals

$$\begin{aligned} W^1(z^{-1}) &= W(z^{-1}) - 1 \qquad (4.9) \\ &= \frac{1 - A(z^{-1}/\gamma_1)}{1 - A(z^{-1}/\gamma_2)} - 1 \qquad (4.10) \\ &= \frac{A(z^{-1}/\gamma_2) - A(z^{-1}/\gamma_1)}{1 - A(z^{-1}/\gamma_2)}. \qquad (4.11) \end{aligned}$$

A slightly different way to view this error filtering process is to break down the quantization (or scalar codebook search) process into two separate stages. For the first stage we have an *a priori* error signal $\tilde{e}_{k|k-1} = S_k - \hat{S}_{k|k-1}$ $(= \bar{e}_k)$, which is obtained from Figure 4.9 by noting that the *a priori* quantizer value, $\hat{e}_{k|k-1}$ is zero. The second

[3]Note that the semi-infinite sum is simply the result of the pole-zero structure of the perceptual weighting filter producing an Infinite Impulse Response (IIR) filter. The filter is of course able to be simply realized in pole-zero form.

stage involves the selected quantizer, or *a posteriori*, value $\hat{e}_{k|k}$ $(= \hat{e}_k)$ being used to obtain the *a posteriori* error, $\tilde{e}_{k|k} = S_k - \hat{S}_{k|k}$ $(= \tilde{e}_k)$.

When expressed in this way, it is clear that the quantizer output is selected as

$$\hat{e}_k = \mathbf{Q}\left[P_{k|k-1}\right] = \mathbf{Q}\left[\tilde{e}_{k|k-1} + W^1(z^{-1})\tilde{e}_{k|k}\right], \tag{4.12}$$

with the obvious definition for $P_{k|k-1}$. Note that $P_{k|k-1}$ does not depend on $\tilde{e}_{k|k}$, since $W^1(z^{-1})$ has no term in z^0. Hence the ADPCM or APC noise feedback coding approach is fundamentally identical to the CELP perceptual weighting approach, with no codebook search (quantizer search) being required for the former.

Approximate Average Bit Rate	No Perceptual Weighting			With Perceptual Weighting		
	SNR (dB)	Segmental SNR (dB)	Bit Rate (Entropy)	SNR (dB)	Segmental SNR (dB)	Bit Rate (Entropy)
16 kbps	35.61	17.83	1.963	34.92	17.33	1.938
12 kbps	29.78	13.70	1.430	29.12	13.28	1.407
8 kbps	23.70	9.93	0.993	23.07	9.58	0.967
4 kbps	14.33	4.92	0.504	13.63	4.59	0.480

Table 4.2: SNR values for AC-ADPCM with Perceptual Weighting

Table 4.2 displays the SNR values obtained with and without perceptual weighting. Similar to LD-CELP, a 10th order perceptual weighting filter was used (with identical parameters to LD-CELP for comparison purposes), updated every 20 samples (2.5 ms). A decrease in SNR of about 0.7 dB, and 0.5 dB in segmental SNR is observed through the use of perceptual weighting. However, it is interesting to note that this is offset slightly by the reduced bit rates observed. For the average rate of close to 16 kbps, equalizing the bit rates by decreasing the quantizer step size for the system with perceptual weighting produces 35.15 dB SNR and 17.49 dB segmental SNR, regaining about 0.2 dB of the difference.

Remark 4.11 The entropy values (in bits/sample) listed in Table 4.2 correspond closely to the average number of bits per sample required by the AC-ADPCM coding process. As the speech is sampled at 8 kHz, it is a simple matter of multiplying by 8000 to obtain the average bit rate in kilo bits per second (kbps). Note that the tabulated values are actually entropy values, as for simulations at this stage the entropy of the quantizer output bit stream is simply measured based on the Laplacian distribution assumption, and no Arithmetic Coding is actually performed. Arithmetic Coding is practically optimal entropy coding, with an incurred bit rate overhead of only a negligible amount

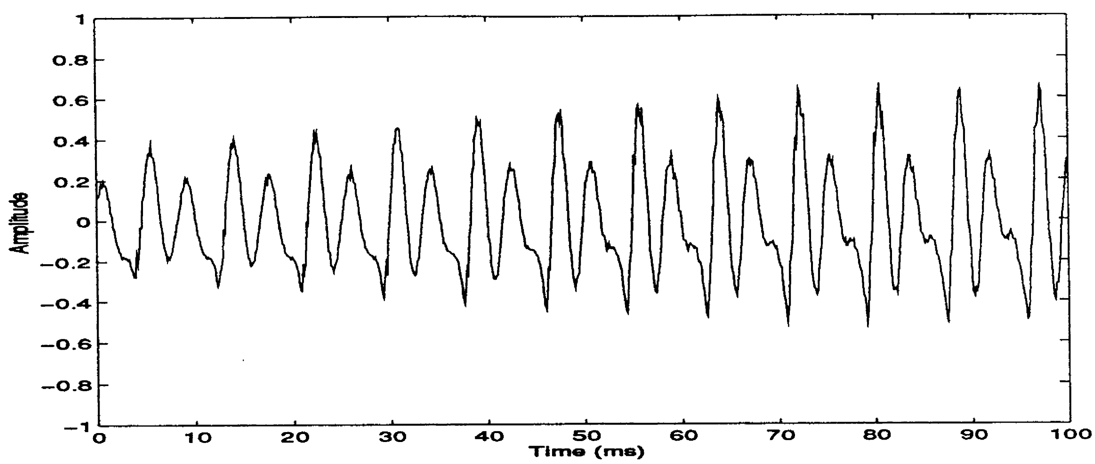

Figure 4.10: Sample 100 ms Waveform Segment for Voiced Speech

above that indicated by the entropy. Hence the use of entropy values (for simplicity) is a reliable indication of the actual average bit rate.

Informal listening tests indicate only minor difference in subjective performance at the 16 kbps average rate. However, at average rates of 12 and 8 kbps, the subjective performance improvement through the use of perceptual weighting is significant.

4.5.3 Pitch Prediction

An analysis of the speech signal reveals a significant amount of redundancy both in short term (formant) structure, and in longer term (pitch) structure for voiced speech. (A 100 ms sample of voiced speech waveform is presented in Figure 4.10.) Pitch periods are commonly within the range of 35 to about 140 samples, and may take both integer, and non-integer values (as there is nothing magical about the 8 kHz sampling frequency in relation to the pitch).

The fact that a cursory examination of a speech waveform by the human eye often reveals a very large amount of pitch redundancy in the speech signal does not imply that it is a simple matter for this redundancy to be exploited at the speech coding level. A significant problem is that the pitch period is quite non-stationary, and even a small difference between the estimated and actual pitch periods can lead to a significant performance degradation for the pitch predictor.

Hence pitch period estimation is an extremely important topic, and it should be

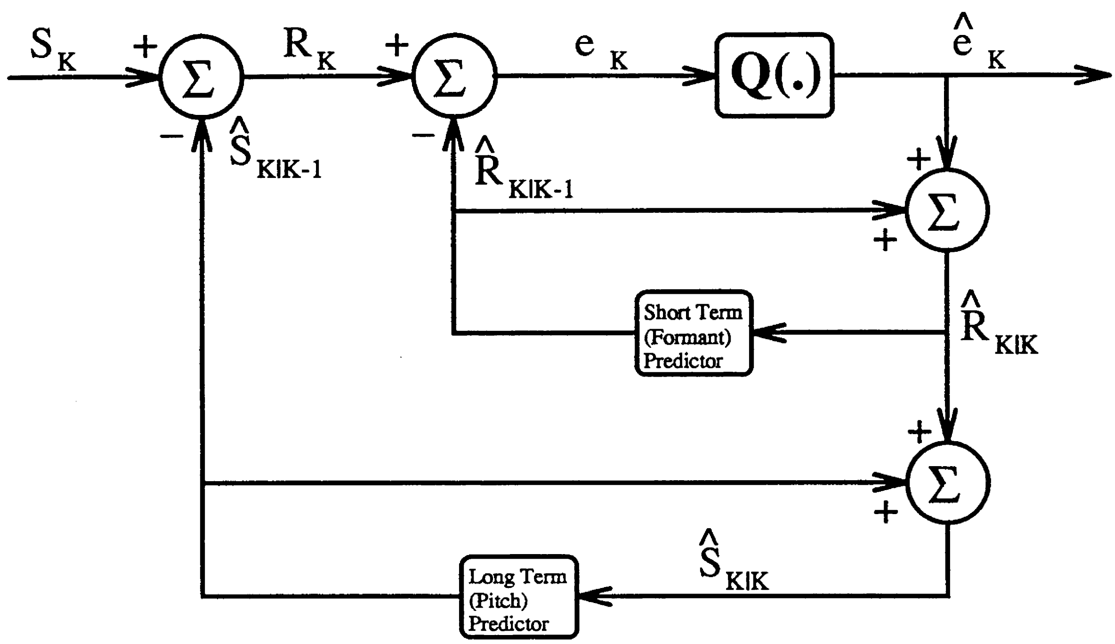

Figure 4.11: APC or ADPCM Encoder with Pitch Prediction

clear that forwards adaptation of the pitch period would normally imply better pitch prediction performance. Pitch prediction has thus been used to advantage in many forwards adaptive systems, such as CELP systems, and others such as APC and similar systems[73, 124]. A good general paper on stability and performance of pitch predictors is the one by Ramachandran and Kabal[149].

Figures 4.11 and 4.12 display how long term prediction is incorporated into an APC or ADPCM system. In this configuration, a single tap pitch predictor would have the

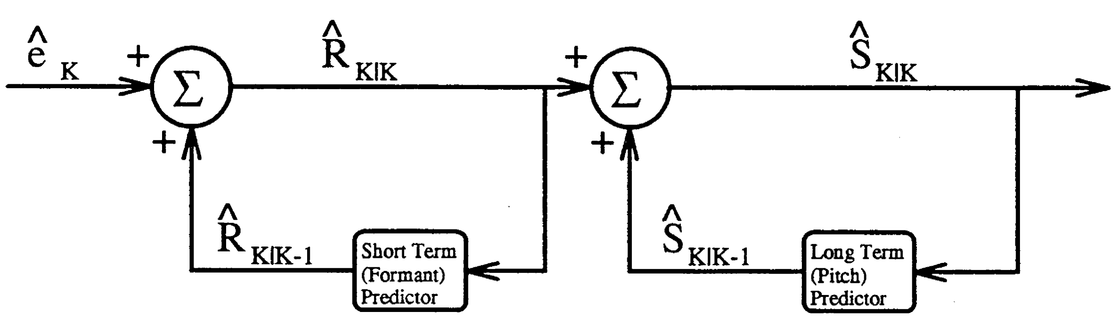

Figure 4.12: APC or ADPCM Decoder with Pitch Prediction

form:

$$\hat{S}_{k|k-1} = b_1 \hat{S}_{k-kp|k-kp}, \tag{4.13}$$

where kp is the pitch period, and b_1 is the single predictor tap coefficient. As previously, the short term (formant) predictor is of the form:

$$\hat{R}_{k|k-1} = a_1 \hat{R}_{k-1|k-1} + a_2 \hat{R}_{k-2|k-2} + \ldots + a_N \hat{R}_{k-N|k-N}, \tag{4.14}$$

for an Nth order predictor with coefficients $a_i, i = 1..N$.

Generally the pitch period and predictor coefficient parameters are all adaptive. Consequently it is somewhat more difficult to exploit pitch redundancy in backward adaptive systems such as ADPCM, rather than in forward adaptive systems. However, a useful paper on backward adaptive pitch prediction is the one by Pettigrew and Cuperman[140]. In this paper, both block adaptive and recursive algorithms are discussed, and a hybrid approach presented, where the pitch periods and pitch taps are initialized on a block basis, and recursively updated during the blocks in order to track time variations.

Although we do not implement a pitch prediction strategy such as the one in [140], we do expect there to be a significant performance improvement in doing so. The issue of the robustness to transmission errors of backwards adaptive pitch prediction has been mentioned in [33], and would be of some concern. However, this is not enough to stop investigation of a similar approach to [140], as the interest would initially be somewhat academic, since AC-ADPCM already has very poor transmission error performance. (The transmission error resynchronization problem will be discussed in Chapter 9.)

One possibility that appears to be worth further consideration with the AC-ADPCM approach is the option of incorporating both forward and backward adaptation into the pitch prediction process. CELP schemes such as QCELP[51, 66] already attempt to reduce the number of bits transmitted via the use of cyclically transmitting the full pitch period, and then just an offset, which can be viewed as an elementary form of backward adaptation. Transmitting only the offset to the pitch period calculated via backwards adaptation can be viewed as an increment over the existing approach.

The inclusion of some backwards pitch adaptation may not be particularly useful for forwards adaptive CELP systems. However, for current low delay backwards adaptive systems some benefits may exist to the use of a forwards adaptive pitch prediction component. A small increase in delay (less than 5 ms) may be found to result in an increase in pitch prediction performance. It is unlikely that the corresponding delay would be tolerated for current low delay systems such as LD-CELP. However the effect on a variable rate system such as AC-ADPCM might be less significant. This is related to the fact that the AC-ADPCM output bit stream would most likely be buffered to some extent, even if destined for a variable rate channel such as in a CDMA system.

Apart from the expected prediction gain advantage with the use of a pitch predictor in AC-ADPCM, we might also expect that the output instantaneous bit rate is smoothed to some extent. Currently we observe large peaks in the bit rate corresponding to peaks in the excitation signal at pitch period intervals. Another characteristic of the basic AC-ADPCM approach that may need to be considered in relation to pitch prediction is the effect of the coarse excitation structure on the prediction performance. Some recent work by Taniguchi, Johnson and Ohta[172] concerns a similar issue in sparse codebook CELP.

In summary, it is reasonable to expect some performance improvement with the use of existing backwards adaptive pitch prediction approaches, such as the one introduced in [140], and further improvements may be possible with judicious use of the flexibility afforded by the AC-ADPCM approach. Of course these possible further improvements would be significantly applications dependent. Pitch prediction within the AC-ADPCM system is thus left as a topic for future research work.

4.5.4 Adaptive Postfiltering

Adaptive postfiltering is used in a number of coding systems such as LD-CELP to obtain perceptual improvements in the output speech quality. The application of standard postfiltering approaches to the AC-ADPCM system is worth further consideration.

Chen *et al.*[34] note that an adaptive postfilter in general attempts to attenuate spectral valleys and emphasize spectral peaks in the decoded speech. Although this has the effect of introducing some slight distortion into the speech signal, the perceived coding noise is reduced.

Postfiltering in LD-CELP consists of cascaded short and long term components. The transfer function of the long term postfilter is given by

$$H_l(z^{-1}) = g_l(1 + bz^{-kp}), \tag{4.15}$$

where kp is the calculated pitch period, and the coefficients b and g_l are calculated from the the optimal tap weight of a single tap pitch predictor, as discussed in [34].

The short term postfilter transfer function is

$$H_s(z^{-1}) = \frac{1 - A(z^{-1}/\gamma_1)}{1 - A(z^{-1}/\gamma_2)} \left[1 + \gamma_3 k_1 z^{-1}\right], \tag{4.16}$$

where $A(z^{-1})$ represents the 10th order predictor coefficients obtained by halting the 50th order Levinson-Durbin recursion at an intermediate stage, and k_1 is the first reflection coefficient obtained via the same analysis. The parameters γ_1, γ_2, and γ_3 control the amount of postfiltering, and are tuned as discussed in [34].

Approximate Average Rate	No Postfilter		With Postfilter	
	SNR (dB)	Segmental SNR (dB)	SNR (dB)	Segmental SNR (dB)
16 kbps	34.92	17.33	18.68	10.62
12 kbps	29.12	13.28	18.56	9.36
8 kbps	23.07	9.58	17.91	7.86
4 kbps	13.63	4.59	13.40	4.56
LD-CELP	15.93	18.19	15.44	16.05

Table 4.3: SNR values for AC-ADPCM with Postfiltering

Table 4.3 presents the results of applying to AC-ADPCM an identical adaptive postfilter to that used in LD-CELP. It is clear from the table that a substantial drop in SNR figures occurs with the use of the postfilter in AC-ADPCM. Shown in the table are values for LD-CELP with and without postfiltering for comparison reasons.

Remark 4.12 Note that all values in Table 4.3 refer to systems utilizing perceptual weighting, and the AC-ADPCM figures have been obtained from a system with a 10th order predictor updated every 2.5 ms. This predictor strategy is simply to reduce computation requirements, while attempting to maintain similarity of AC-ADPCM and LD-CELP for reasons of comparison.

Informal listening tests again indicate only marginal improvement at the 16 kbps average rate, which is likely to be connected with the high performance already observed at that rate. However, substantial subjective performance improvements are observed at the 12, 8, and 4 kbps rates through the use of postfiltering.

The subjective performance of 12 kbps AC-ADPCM with perceptual weighting and postfiltering is judged to be very similar to that of LD-CELP. Of course, some differences in artifact characteristics are observed, but the only significant difference would appear to be a slightly higher level of audible quantization 'hiss' in the AC-ADPCM system.

Performance at 8 kbps is also quite reasonable, and although the performance at 4 kbps is certainly not good, it may be adequate for some purposes.

4.5.5 Quantizer Step-Size Updates

The concept of a constant quantizer step-size came from observation of stability problems related to quantizer adaptation, and the effective minimisation of MSE, or maximisation of SNR, via the basic AC-ADPCM approach. Hence the motivation for AC-ADPCM has been somewhat academic in nature, and it has already been mentioned that SNR is not necessarily a particularly good measure to maximise.

Noting that segmental SNR is usually considered a better measure with respect to speech coding, and the fact that degradation in unvoiced speech has been noted in AC-ADPCM at rates below 8 kbps, it may seem logical to consider a quantizer step size update strategy. Unlike traditional fixed rate ADPCM systems, only limited step size update on a relatively long time scale would appear applicable. This should imply that the stability problems discussed in Chapter 3 are still mitigated.

As good performance is already observed at 12 kbps, and reasonable performance at 8 kbps, a quantizer step size variation approach would only appear to become an important consideration at below the average rate of 8 kbps. This is not in the range of our current concern, as we are attempting to provide 'proof of concept' for between the rates of 8 and 16 kbps. Hence the topic is left as an area for future research.

Remark 4.13 At rates of 8 kbps and below, separate treatment of voiced and unvoiced speech would appear useful, and any further work in this area should certainly not be limited to a simple quantizer step size variation procedure.

4.5.6 Kalman Filter Application

The application of the Kalman Filter to AC-ADPCM is an important part of this thesis. A logical place for discussion of this topic would be at the present location in the chapter. However, in order to break the thesis into more manageable chapters, the discussion of the use of the Kalman Filter in AC-ADPCM has been deferred until Chapter 6. Chapter 5 introduces the Kalman Filter in general terms and provides an overview of its use in speech coding applications.

4.6 Design Flexibility and Computation Issues

We have mentioned above how the output from the AC-ADPCM system compares to that from the CCITT G.728 LD-CELP standard. As noted, this is perhaps an unfair comparison, as an LD-CELP variant with VAD (Voice Activity Detection) and DTX (Discontinuous Transmission) would probably be of more use. However, this would still imply some additional 'place holder' transmission, and extra complexity.

Another very important practical concern is the issue of computational complexity. The AC-ADPCM approach avoids the codebook search required with CELP systems, which can represent a significant computation saving. However, it would seem that there is no significant difference between the two schemes as far as the other elements of the system are concerned, such as the backwards adaptation of the predictors.

This is not necessarily true, as the simplicity of the AC-ADPCM system results in significant design flexibility in trading off computational complexity against output quality and average bit rate. CELP approaches are often able to trade some complexity for output quality, but it is quite difficult to have any significant flexibility in the output bit rate. (A notable exception to this the QCELP system[51, 66], but even this has flexibility limitations.)

Variable rate coding has received a significant amount of recent interest[28, 50, 51, 56, 65, 66, 69, 133, 137, 173], and the AC-ADPCM variable bit rate approach also has another important advantage. It may be useful to have a quality/bit rate control 'knob', where the quality can be increased on demand, of course at a cost of a higher average bit rate. Other applications may utilize the quality 'knob' for the system to demand a lower bit rate, and hence lower output quality. The whole issue of bit rate control stems from these considerations, but we do not investigate them in any depth here.

For the rest of this section we ignore the use of quantization dither, perceptual weighting, pitch prediction, postfiltering, and quantizer step size updates. Instead we approach the computation issue by performing some simulations of the effect of different predictor orders, predictor update frequency, and implementation of the Arithmetic Coding.

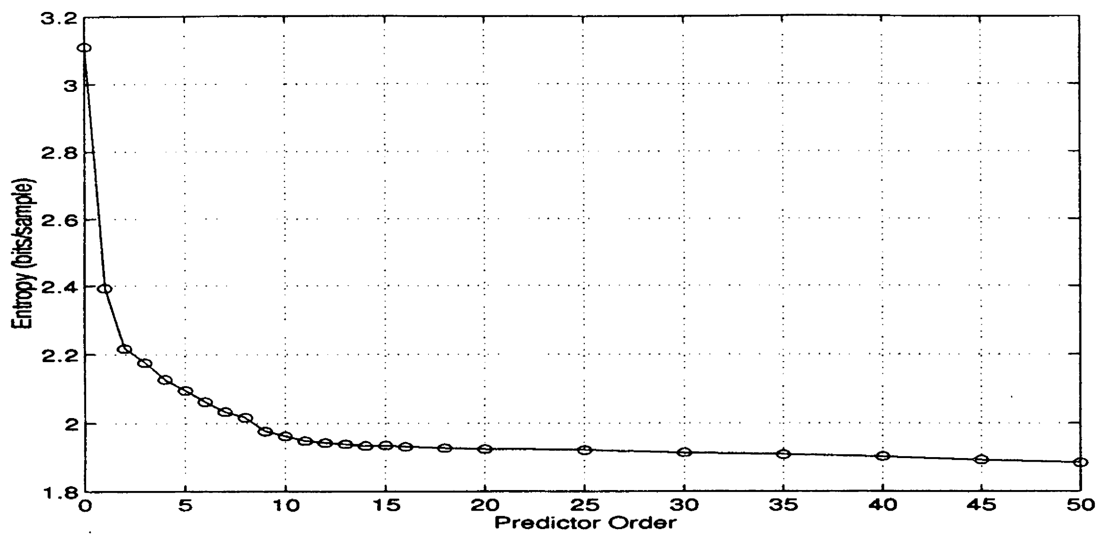

Figure 4.13: Effect of Predictor Order on AC-ADPCM Entropy

4.6.1 Prediction Order

In order to make 'fair' comparisons of performance between our AC-ADPCM proposal, and LD-CELP, we have have used a high order predictor within the ADPCM loop. The LD-CELP standard specifies a 50th order predictor, which Chen and Cox[33] report as offsetting the performance degradation in female speech caused by the absence of pitch prediction (which was removed for channel error robustness considerations).

Within AC-ADPCM any degradation in predictor performance caused by reducing the predictor order does not directly translate into a degradation in the output speech performance. Instead, the output bit rate is increased, which indirectly corresponds to a decrease in speech quality for the same average bit rate. The additional flexibility afforded in trading-off predictor computational complexity directly against average bit rate is an important advantage of the AC-ADPCM system.

The graph displayed in Figure 4.13 shows the effect of the predictor order on the entropy (average bit rate per sample) of the AC-ADPCM system. The graph has been obtained with a fixed quantizer step size, and hence fixed output quality. For all predictor orders, the output SNR is 35.6 dB, and the segmental SNR is 17.8 dB, corresponding to an average rate under 16 kbps with a 50th order predictor. The simulations performed used over 22 seconds of speech, consisting of 4 male and 4 female sentences.

Predictor Order	Average Bit Rate (Entropy)		
	~ 16 kbps	~ 12 kbps	~ 8 kbps
50	1.885	1.362	0.932
40	1.903	1.376	0.950
30	1.914	1.387	0.961
20	1.925	1.403	0.970
15	1.935	1.406	0.976
10	1.963	1.430	0.993
5	2.095	1.545	1.077

Table 4.4: Effect of Predictor Order in AC-ADPCM

From the graph it is clear that significant gain is achieved by increasing the predictor order to about 10. After this, very little decrease in average bit rate is observed. It is expected that the 50th order predictor would give significant advantage for particular sections of speech, such as high pitched female speech, but as this is only a small percentage of the total, the overall effect on bit rate is minimal.

The figures in Table 4.4 confirm the fact that the bit rate increase corresponding to the 10th order predictor is only a few percent. The computation decrease related to this is substantial, and hence a significant amount of flexibility exists for implementation of AC-ADPCM. Again, the rate of approximately 16 kbps corresponds to SNR of 35.6 dB, and segmental SNR of 17.8 dB, while 12 kbps gives 29.8 dB and 13.7 dB, and the 8 kbps rate produces 23.7 dB SNR and 9.9 dB segmental SNR.

4.6.2 Predictor Update Frequency

A similar argument to the issue of prediction order holds with respect to predictor update frequency. Due to the direct trade-off between predictor efficiency and average bit rate, a significant amount of computation can be saved for a small increase in average bit rate.

Table 4.5 presents the results of increasing the predictor update period from the 2.5 ms period used in LD-CELP to 20 ms. Again, it can be seen that only a few percent increase in the bit rate occurs. This may well be worth the cost if silicon implementation expense is high. Note that these figures have been obtained by performing the predictor update similar to LD-CELP, but less often. Tuning of the update procedures for the higher update periods might be expected to increase performance further.

Table 4.6 summarizes the results of Tables 4.4 and 4.5, and shows that in going from a backward adaptive 50th order predictor updated every 2.5 ms (as in LD-CELP)

Predictor Update Period (ms)	Average Bit Rate (Entropy)		
	~ 16 kbps	~ 12 kbps	~ 8 kbps
2.5	1.963	1.430	0.970
5.0	1.974	1.439	0.999
7.5	1.983	1.450	1.008
10.0	1.997	1.459	1.015
12.5	2.012	1.473	1.026
15.0	2.025	1.482	1.035
17.5	2.036	1.498	1.042
20.0	2.048	1.505	1.055

Table 4.5: Effect of Predictor Update Frequency in AC-ADPCM

Predictor Update Period (Order)	Average Bit Rate (Entropy)		
	~ 16 kbps	~ 12 kbps	~ 8 kbps
2.5 ms (50th Order)	1.885	1.362	0.932
20.0 ms (10th Order)	2.048	1.505	1.055

Table 4.6: Effect of Predictor Order and Update Frequency

to a 10th order predictor updated every 20 ms, the average bit rate increase is in the range of 10%.

Another way of measuring the performance decrease incurred by the lower order predictor with less frequent updates is by equalizing the bit rates and comparing objective and subjective performance of the two approaches. The SNR and segmental SNR figures for this situation are displayed in Table 4.7.

Approximate Average Bit Rate	50th Order Predictor 2.5 ms Update Period		10th Order Predictor 20 ms Update Period	
	SNR	segSNR	SNR	segSNR
16 kbps	37.28	19.14	35.60	17.82
12 kbps	31.56	14.90	29.78	13.69
8 kbps	25.40	10.93	23.69	9.92

Table 4.7: SNR Measures for Selected Predictor Orders and Update Frequencies

From Table 4.7, a degradation in SNR measures of between 1.0 dB and 1.8 dB is observed. Considering the computational saving made with the lower order predictor updated less often, this decrease might appear acceptable for some applications. Subjective comparisons of the two predictor strategies were difficult due to only minor differences in the output speech. However, there is no reason to suggest that there is not a small subjective degradation, in line with the small degradation in objective measures.

Remark 4.14 A 20 ms predictor update period for a backwards adaptive predictor is quite substantial, and the fact that only moderate system performance decrease has been observed is important. This is testimony to the high level of design flexibility within the AC-ADPCM approach.

Remark 4.15 For a 20 ms predictor update period, and a 10th order predictor, it would be sensible to consider a forwards adaptive system, such as the AC-APC system already mentioned. The reason for this is due to the non-stationarity of the speech signal giving poor quality backward adaptive predictor coefficients for the long update period. By the end of a 20 ms frame the predictor coefficients are at least 20 ms old, and probably longer due to computation balancing considerations, and the fact that the window used to obtain the predictor coefficients is often centred significantly back from the end of the last frame. Ten predictor coefficients can be quantized efficiently for a forwards adaptive system, but some additional delay will be incurred through the use of forwards adaptation. Hence the 'optimal' engineering solution to the question of forwards versus backwards adaptation is application dependent.

Both considerations of prediction order and predictor update frequency are closely related, and allow a direct trade-off between computation and average bit rate. The flexibility of the AC-ADPCM system need not be restricted to the simple approach of choosing some fixed predictor order and update frequency. It is not difficult to imagine a system where both predictor order and update frequency exhibit some amount of dynamic variation in response to changing signal conditions. However, if the variation of these parameters is to occur in a backwards adaptive fashion, care must be taken to ensure the adaptive parameters do not exacerbate problems due to transmission errors.[4]

Remark 4.16 Since any DSP (Digital Signal Processor) or VLSI (Very Large Scale Integration) hardware implementation must be capable of coding at the maximum or peak computation level[5], the saving from such a variable prediction approach is not likely to be in terms of silicon device expense. Rather, savings in device power

[4]This comment is somewhat academic at this stage, as the AC-ADPCM system would appear to have catastrophic error performance in its current form (due to the use of Arithmetic Coding). The resynchronization problem of AC-ADPCM is discussed further in Chapter 9, and assuming a satisfactory solution of the basic resynchronization issue, the effect of further adaptive parameters on resynchronization must be analyzed.

[5]For applications where coding delay is not of significant concern, the average computation level dictates the implementation cost.

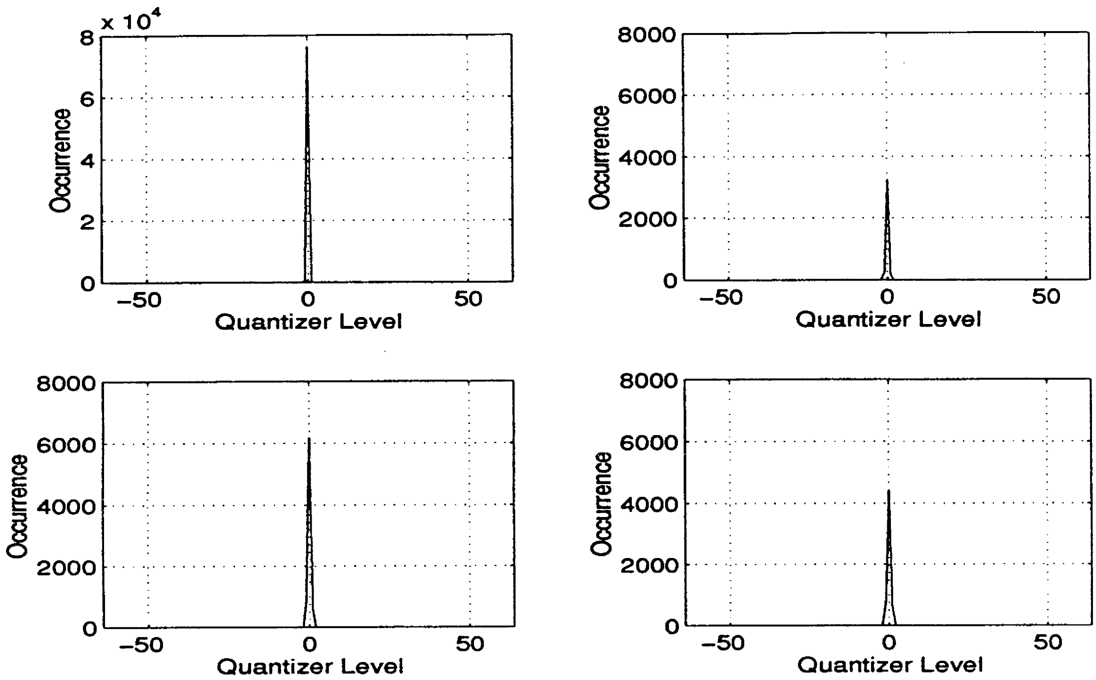

Figure 4.14: Histograms for Variance Bins 1 to 4

consumption could be expected, which would be particularly useful in a portable unit where battery drain is a problem. Further discussion on similar issues is relegated to the chapter on AC-ADPCM applications, Chapter 10.

4.6.3 Arithmetic Coding Implementation

The AC-ADPCM system presented in Figure 4.3 involves Arithmetic Coding of a quantized prediction difference signal based on an adaptive model for the distribution of the quantizer output source. The assumption was made that the quantizer output could be modelled by a Laplacian distribution, with time-variation in the distribution variance.

The graphs presented in Figures 4.14 to 4.17 show the quantizer output level distributions obtained from using 16 different variance bins at the 12 kbps average rate (variance increases from left to right, top to bottom). The histograms were obtained from coding with a fixed quantizer step size over about 22.5 seconds of speech (just over 180000 samples). The quantizer output is represented by integer values in the range from -64 to 64. Note that the occurrence scale changes from the first variance bin (small variance values) to the last variance bin (large variance values).

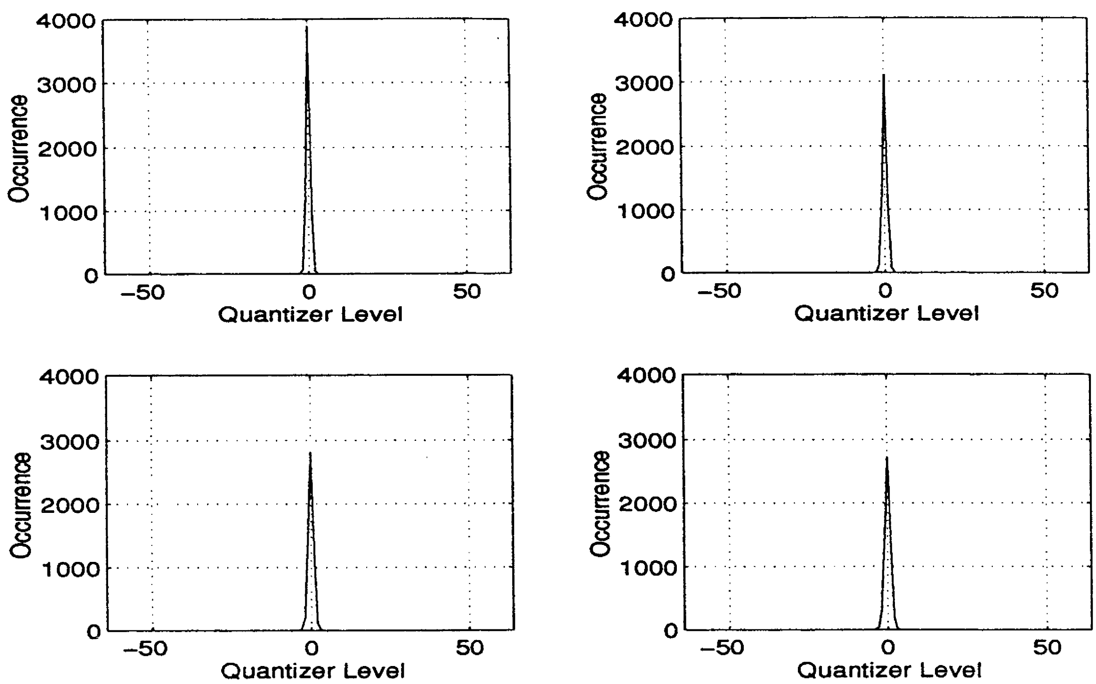

Figure 4.15: Histograms for Variance Bins 5 to 8

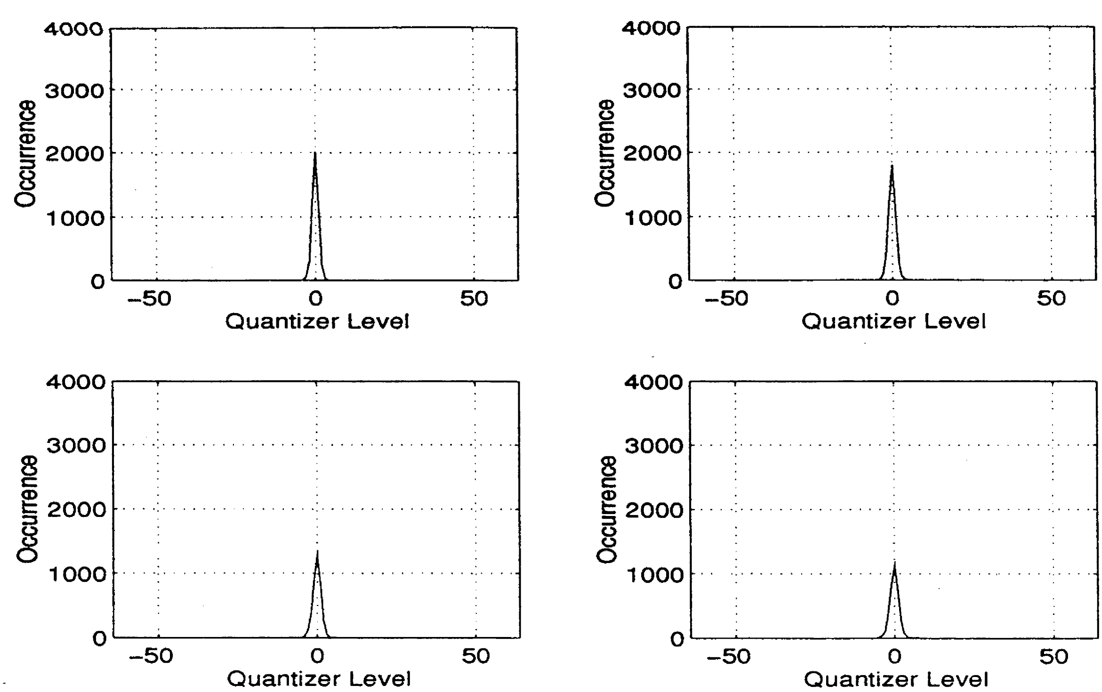

Figure 4.16: Histograms for Variance Bins 9 to 12

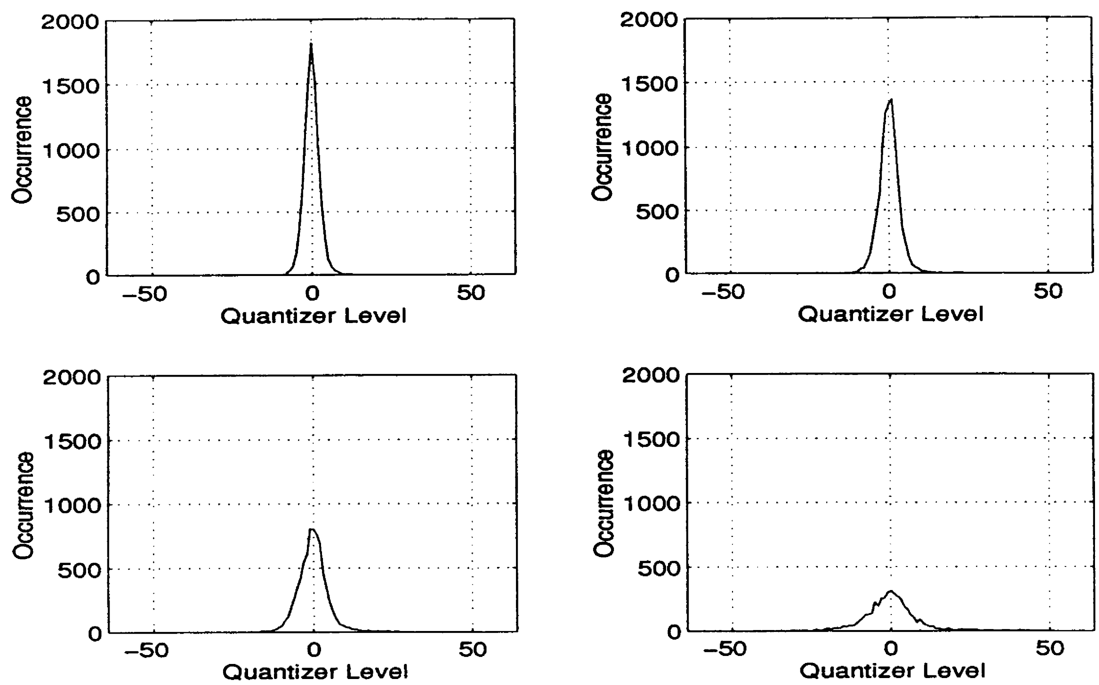

Figure 4.17: Histograms for Variance Bins 13 to 16

Although over 100 quantizer levels are shown on the diagrams, it is clear that not all levels are used. In fact, for the distributions shown in Figures 4.14, 4.15, and 4.16, it appears that only a maximum of 16 levels have non-zero occurrence values. For the higher variance bins shown in Figure 4.17, more levels are used, but the overall frequency of quantizer levels far away from the zero level is only minimal.

Observation of the actual quantizer output distribution reveals that the Laplacian assumption appears valid for the low to medium variance range, although the distributions appear to exhibit somewhat sharper peaks than the basic Laplacian distribution. (Note that for the lower variance bins the plotted quantizer level distributions appear triangular, when they are in fact closer to a delta function.) In the upper variance range, and especially for bins 15 and 16, the distribution appears to become more gaussian in nature.

As most samples do not fall in the upper variance range, the effect of the non-optimality of the Laplacian distribution assumption on overall performance might seem limited. However, the use of a trained distribution approach to the Arithmetic Coding does seem worth consideration. In addition to the assumption validity argument, the simple approach of determining the source probabilities via integration of the Laplacian distribution involves many complex computations that may be avoided through the use of a tabular approach.

We use 16 different tables (Figures 4.14 to 4.17) for the Arithmetic Coding symbol probabilities, corresponding to the 16 different variance levels. The tables do not occupy a large amount of storage, as the table corresponding to the lowest variance has only three significant values, moving up to about 64 entries in the largest variance table.

AC-ADPCM	Average Bit Rate (Entropy)			
Probabilities:	$\sim$ 16 kbps	$\sim$ 12 kbps	$\sim$ 8 kbps	$\sim$ 4 kbps
Laplacian Assumption	1.963	1.430	0.993	0.504
Trained Tables	1.919	1.396	0.968	0.485

Table 4.8: Tabular Approach versus Laplacian Distribution Assumption

The figures in Table 4.8 have been obtained by training the probability distribution tables on the same 22.5 seconds of speech that is then coded. Normally this approach would cast serious doubts as to the reliability of the tabulated figures. However, the trained distributions in Figures 4.14 to 4.17 are observed to be extremely regular, leading to the conclusion that the training approach has resulted in very good generalization, and hence the displayed figures can be assumed reliable.

The results obtained through the use of the Laplacian distribution assumption and the training approach agree very closely, indicating the validity of the original assumption. Due to computational considerations, a tabular approach to obtaining the probabilities would be preferred, since this avoids the exponential evaluations required for integration of the Laplacian distribution. Also some benefit may be possible with the tabular approach in terms of error performance, but no consideration of this issue is provided within this chapter.

4.7 Further AC-ADPCM Considerations

Well known techniques of spectral smoothing and bandwidth expansion have been found useful in assisting with robustness to numerical accuracy in fixed point LD-CELP[32], and should certainly also be considered for any fixed point implementation of AC-ADPCM. It is also possible that there may be some advantage to the use of the techniques even for high precision arithmetic, due to the coarse quantization noise structure. Especially at low bit rates, we expect the coarse quantization noise to result in a fairly poor quality reconstructed speech signal used for backwards adaptation of the predictor coefficients. In this situation, the 'smoothing' effect of spectral smoothing and bandwidth expansion may be of advantage.

Another 'somewhat standard' speech coding approach is that of residual driven adaptation for the predictor coefficients. This has been found to assist with robustness considerations in some systems, such as ADPCM. Recent work by Nam and Gibson[134] has extended the use of residual driven adaptation approaches to lower rate systems (without significant loss of quality) via the use of a smoothed residual driven approach. Of course, for such an approach to assist with robustness to transmission errors in AC-ADPCM, the Arithmetic Coding block must first be made robust to errors. This issue has been mentioned above, and is discussed in Chapter 9.

Of some concern with AC-ADPCM for any potential application is the requirement for some form of input level tracking. Obviously a fixed quantizer step size is of limited use with input level fluctuations, as low level input will be badly distorted, and high level input will result in an extremely high output bit rate. Fixed rate coding systems such as LD-CELP to some extent automatically track the input signal level, which does not occur in the elementary AC-ADPCM system. Fortunately it can be assumed that the input level changes occur relatively slowly (or perhaps more accurately, the input level is relatively constant, with rapid jumps, although this may be somewhat applications dependent), and hence standard Automatic Gain Control (AGC) techniques can be incorporated to adjust the quantizer step size accordingly.

Remark 4.17 It is important to note the difference between the Automatic Gain Control approach for tracking of the input level and the earlier discussion on quantizer step size adaptation for maximisation of objective measures such as segmental SNR. Obviously the two problems are related, as they both involve adaptation of the quantizer step size. However, they are quite distinct in terms of the time-scales involved. Quantizer step size adaptation for maximisation of objective measures would need to occur on the order of tens to hundreds of milliseconds maximum, while input level tracking is more likely to occur over tens of seconds.

Related to the issue of input level tracking is the topic of output bit rate control, or loosely, output level tracking. Both the user and system (or network) may desire to control the bit rate (from cost considerations), and hence the output speech quality. Normally user requirements would be expected to relate to long term average bit rate or quality considerations, while that of the system could be related to shorter term capacity considerations. Again, control is effected by altering the quantizer step size after monitoring the average output bit rate.

Another small concern that is application dependent is the issue of sampling frequency. All the simulations performed in this thesis are with speech that has been sampled at 8 kHz, which is common for telephony. However, a number of applications such as that of speech storage discussed in Chapter 10 do not necessarily require this sampling rate. Hence consideration of speech sampled at 6.4 kHz for example, may be important.

4.8 Delayed Decision Coding

CELP coding, tree coding, and trellis coding are all forms of delayed decision coding[63, 106]. A significant amount of research attention has been given by some groups to the problem of delayed decision coding using tree codes[54, 63, 75, 99, 125]. Mano and Moriya[125] discuss tree coding use in CELP, although our interest is mainly in its application to ADPCM or APC systems, where the use of the delayed decision can be viewed as an alternative approach to obtaining some of the advantage that is obtained via a CELP system.

Dunham and Ghosh[54] consider the use of tree coding in delta modulation and DPCM. Their conclusions are that small delays can improve performance to some extent, although it is noted that the performance for multibit DPCM is small. Gibson and Chang[75] discuss the use of a multi-tree coding approach within both ADPCM and APC like systems. They consider systems operating at rates of 16 kbps, 12 kbps, and 9.6 kbps, and note that the 9.6 kbps backwards adaptive multi-tree coder substantially out-performs an APC coder, whilst having a delay less than 2 ms. Spectral distortion and granular noise levels are noted to be significantly less than in APC systems. Recently published work from Woo and Gibson[189] also discusses a tree coding approach, directed towards 8 kbps low delay speech coding.

Foodeei and Kabal[63] consider a tree coder as a potential candidate for 16 kbps low delay toll quality coding. The basic tree coder was presented in the paper by Iyengar and Kabal[99], and is a delayed decision coder based on generalized ADPCM. Importantly the authors note that the use of a stochastic tree gives performance improvement over the deterministic tree obtained as a simple extension of ADPCM. Training of the stochastic innovations dictionary was found to further improve performance, analogous to trained codebook usage in CELP systems. The overall coder performance was found to be very close to that of an early LD-CELP system.

Through the variable bit rate ADPCM approach discussed within this chapter, we have seen significant performance improvements over 16 kbps fixed rate ADPCM. The performance of AC-ADPCM at an average rate of 12 kbps is already seen to be approaching that of LD-CELP. It is reasonable to expect that a delayed decision tree coding approach similar to that in [99] would further improve performance significantly. There are thus strong possibilities for future work on this topic.

Some work on trellis coded quantization of memoryless sources and entropy constrained trellis coded quantization has also been presented in the papers by Marcellin and Fischer[128], Fischer and Wang[61], and Marcellin[127]. This approach has been used for subband coding of images in the paper by Joshi, Crump, and Fischer[107], and parts of the approach may also be found to integrate with the AC-ADPCM system.

Remark 4.18 By the use of Kalman smoothing at the AC-ADPCM decoder (to be discussed in Chapter 6), we are implicitly using a form of delayed decision on the output speech sample. Of course this is quite different to the use of a delayed decision at the encoder. However it appears that the two techniques are related, at least in a general sense. Unfortunately we provide no further analysis on this topic, but it is indicated as a potential area for future research.

4.9 Chapter Conclusion

The use of variable rate ADPCM via Arithmetic Coding has been shown to eliminate practically the stability problems apparent with fixed bit rate ADPCM systems at lower bit rates than the 32 kbps G.721 ADPCM standard.

The variable bit rate AC-ADPCM system is also able to exploit inactive periods during speech utterances, to give a relatively low average bit rate. Although quite reasonable performance is achieved at an average rate of 16 kbps, the basic AC-ADPCM system provides an output that is not toll quality, due the the presence of significant levels of audible quantization noise.

Standard techniques of perceptual weighting and postfiltering have been shown to be applicable to AC-ADPCM, and result in significant subjective performance improvements. The performance at the 16 kbps average rate has been judged by informal listening tests to be equivalent to that of LD-CELP, and the output at 12 kbps is very close to the quality of LD-CELP, with a slightly higher level of audible quantization noise. The performance at 8 kbps is also quite good, and although not good, the performance at 4 kbps is promising.

Other techniques such as pitch prediction and quantizer step size updates have also been discussed as possible methods for obtaining further performance improvements. However, these have been left as topics for further research.

A large amount of design flexibility has been observed within the AC-ADPCM approach, and trading-off computation via predictor optimality against bit rate is seen to be worth attention. It has been shown that starting with a 50th order predictor updated every 2.5 ms, and decreasing the predictor order and update frequency to 10th order and 20 ms results in only around 10% increase in output bit rate. Of course, for the latter, forwards adaptation becomes a more worthwhile consideration (as a small additional transmission requirement can provide better predictor coefficients, often with only a small delay increase), giving an AC-APC system.

The use of trained distribution tables to reduce the complexity of the Arithmetic Coding has also been considered. It was shown that the use of as little as 16 tables (for different output variance) gives Arithmetic Coding performance almost identical to, and slightly better than, the use of the Laplacian distribution assumption.

The use of delayed decision coding has also been discussed, and indicated as an area of significant potential for future research.

In the next two chapters, we present, and investigate the use of Kalman prediction. Chapter 5 presents the Kalman filter as a general design tool for use in speech coding applications, and Chapter 6 considers AC-ADPCM with the application of Kalman filtering. It is shown that the Kalman filter practically eliminates the audible quantization noise present at an average bit rate of 12 kbps, and results in high speech quality at this rate. At an average rate of about 12 kbps, the output speech quality with the use of Kalman filtering is judged to be equivalent to that of LD-CELP at 16 kbps, and at an average rate of 8 kbps, the speech quality is below LD-CELP, but very good considering the system simplicity.

The motivation for consideration of AC-ADPCM has been somewhat academic, as it is perhaps a little extreme both to attempt to maximise SNR in a speech coding system, and to use completely non-adaptive quantization to eliminate stability problems.

However, the performance of AC-ADPCM appears to be quite good considering the elementary stage of development. Immediate applications for the scheme are in speech storage. Here the variable bit rate, and the Arithmetic Coding bit error problems do not constitute a significant disadvantage. Other potential applications include speech compression for ATM networks[117, 133], and variable rate mobile telephony. Some applications for the AC-ADPCM approach are discussed in Chapter 10.

Chapter 5

Practical Kalman Filtering in Signal Coding

5.1 Chapter Motivation

Kalman filtering techniques are widely known and used in various electrical engineering pursuits. However, they are not widely used for signal coding applications. We view the Kalman filter as a design tool, and we see that there are many ways a Kalman filter can be tuned to suit engineering applications such as practical signal coding systems. For much of this chapter, we are strictly dealing with mathematically sub-optimal Kalman filtering approaches. These approaches provide much of the performance gain of the Kalman filter, but at a reduced computational cost.

We investigate the problems of input noise filtering, signal prediction, and smoothing, and where possible attempt to show the connections between these three problems. Simulation results are presented to illustrate the effectiveness of the techniques discussed.

This chapter also attempts to display the fact that in some circumstances where only a very small amount of additional computation is available, it may still be possible to obtain a substantial performance advantage through the use of a mathematically sub-optimal Kalman filtering approach.

5.2 Introduction

We investigate the use of Kalman filtering techniques in signal coding applications. For a general introduction to linear filtering theory and related topics, the reader may find it useful to refer to the comprehensive paper by Kailath[108], or the text by Haykin[82].

The primary aim of the filtering techniques presented here is to take account of the known signal and noise properties to remove noise from the signal as efficiently as possible. As part of this efficiency, we not only take account of the output quality from the filtering (usually in the mean-square error sense), but also the computational cost required to perform the filtering. The engineering trade-offs involved are extremely important for any practical applications, yet are often given only cursory mention (if at all) in the published literature on Kalman filtering.

The Kalman filter has a number of potential uses within speech (or other signal) coding systems. Some of these applications have been presented in various papers. Examples are the papers by Crisafulli *et al.*[43], discussing the use of Kalman filtering in an adaptive quantizer, and that by Ramabadran and Sinha[147], presenting the use of Kalman filtering techniques within the speech prediction loop.

We attempt to cover some of the major applications of the Kalman filter to signal coding in this chapter. We start with a general presentation of Kalman filtering from a smoothing perspective, as the use of the Kalman filter to enhance noisy signals is quite widely accepted. From this smoothing approach, we examine the use of Kalman filtering techniques within the prediction loop in signal coding, and extend this to the consideration of reconstructed output smoothing. Simulations are performed to complement the basic theory, initially using the assumption of white quantization noise within the coding loop.

The effect of coloured noise on the Kalman filtering methods is investigated briefly. From here we deal a little more with engineering practicalities, by proposing a method of exploiting smoothing properties to obtain most of the performance gain of Kalman filtering, for only a fraction of the computational cost.

The remainder of the chapter is concerned with other issues and uses of Kalman filtering, such as downsampling, input noise filtering, and transmission error recovery.

5.3 The All-pole Signal Model

Many signals such as speech have distinct characteristics, and from a statistical perspective, exhibit large amounts of correlation. From a coding or filtering perspective, this correlation can be used to significant advantage. For these tasks of coding or filtering, we typically model the signal as filtered white noise. The all-pole, or autoregressive

(AR), signal model is often used for speech, and is a general model that also finds many other applications. From Crisafulli *et al.*[44], the all-pole signal model we introduce is:

$$S_k = \left(\frac{1}{1 - \sum_{i=1}^{N} a_i z^{-i}}\right) w_k, \tag{5.1}$$

where S_k is the signal, w_k represents the excitation sequence (white noise), and a_i are the filter coefficients.

Remark 5.1 For a stationary signal, the 'optimal' set of a_i coefficients can be computed efficiently from the signal autocorrelation matrix through the use of the Levinson-Durbin Algorithm[68]. Most 'real-world' signals are non-stationary, however these can usually be considered to be piecewise stationary, allowing practical use of algorithms such as that of Levinson-Durbin. It is important to note that the degree of stationarity, as defined by the period over which the signal can be assumed to be effectively stationary, is extremely important for practical reasons. The issue of forwards versus backwards adaptation is covered briefly later in the chapter.

Equation 5.1 can be written as

$$S_k = a_1 S_{k-1} + a_2 S_{k-2} + a_3 S_{k-3} + \cdots + a_N S_{k-N} + w_k, \tag{5.2}$$

which, in turn, can be expressed in state space form as

$$X_k = F X_{k-1} + \bar{w}_k \tag{5.3}$$

$$S_k = H X_k \tag{5.4}$$

where

$$F = \begin{pmatrix} a_1 & a_2 & \cdots & a_{N-1} & a_N \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}; \tag{5.5}$$

$$X_k = \begin{pmatrix} S_k \\ S_{k-1} \\ S_{k-2} \\ \vdots \\ S_{k-N+1} \end{pmatrix}; \quad \bar{w}_k = \begin{pmatrix} w_k \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix};$$

$$H = [1 \ 0 \ \cdots \ 0].$$

In this formulation, the state vector, X_k, consists of N samples, from the kth sample back to the $(k-N+1)$th sample. F is the system matrix, H is the output vector, and $\bar{w}_k$ is the excitation vector, the form of which is closely related to the structure of F.

For signal coding purposes, a prediction is obviously required, and for smoothing, we use the signal model and form a prediction in order to attempt to remove redundancy from the signal so the noise is able to be filtered more efficiently. From 5.2, the best prediction of the sample S_k, given the samples up to time $(k-1)$, is

$$\hat{S}_{k|k-1} = a_1 S_{k-1} + a_2 S_{k-2} + a_3 S_{k-3} + \cdots + a_N S_{k-N}. \tag{5.6}$$

In state-space form, this is simply

$$\hat{S}_{k|k-1} = HFX_{k-1}. \tag{5.7}$$

Unfortunately, this prediction cannot be used effectively for signal coding due to the fact that the decoder does not have exact copies of the samples S_{k-1} to S_{k-N}. (A similar argument also holds for the smoothing application.) Within a standard ADPCM framework, the decoder does have reconstructed estimates of the input speech samples. As the aim of the signal coding process is to produce high quality reconstructed estimates of the input signal at the decoder, it is usually assumed that these reconstructed values, $\hat{S}_{k-1|k-1}, \ldots, \hat{S}_{k-N|k-N}$ are relatively faithful copies of the input samples, and thus the following linear prediction is used:

$$\begin{aligned} \hat{S}^{LP}_{k|k-1} &= a_1 \hat{S}^{LP}_{k-1|k-1} + a_2 \hat{S}^{LP}_{k-2|k-2} + a_3 \hat{S}^{LP}_{k-3|k-3} && (5.8) \\ &\quad + \cdots + a_N \hat{S}^{LP}_{k-N|k-N} \\ \hat{S}^{LP}_{k|k} &= \hat{S}^{LP}_{k|k-1} + \mathbf{Q}[S_k - \hat{S}^{LP}_{k|k-1}]. && (5.9) \end{aligned}$$

Shown also (equation 5.9) is the method for obtaining the reconstructed values, where $\mathbf{Q}[S_k - \hat{S}^{LP}_{k|k-1}]$ is the best available *a posteriori* estimate of the prediction error which is provided by the quantized difference signal in the case of ADPCM or by the code book entries in the case of CELP. Note that in the above the superscript LP has been introduced to denote the standard Linear Prediction approach. This prediction can also be expressed in state-space form:

$$\hat{S}^{LP}_{k|k-1} = HF\hat{X}^{LP}_{k-1|k-1}, \tag{5.10}$$

where

$$\hat{X}^{LP}_{k-1|k-1} = \begin{pmatrix} \hat{S}^{LP}_{k-1|k-1} \\ \hat{S}^{LP}_{k-2|k-2} \\ \hat{S}^{LP}_{k-3|k-3} \\ \vdots \\ \hat{S}^{LP}_{k-N|k-N} \end{pmatrix} \quad (5.11)$$

is now a state estimate vector internal to the Linear Predictor. To introduce some more notation which shall be useful shortly, we define the above to be the *a posteriori* state estimate vector, and the *a priori* state estimate vector as:

$$\hat{X}^{LP}_{k|k-1} = \begin{pmatrix} \hat{S}^{LP}_{k|k-1} \\ \hat{S}^{LP}_{k-1|k-1} \\ \hat{S}^{LP}_{k-2|k-2} \\ \vdots \\ \hat{S}^{LP}_{k-N+1|k-N+1} \end{pmatrix}. \quad (5.12)$$

Using this definition, equation 5.10 is equivalent to

$$\hat{S}^{LP}_{k|k-1} = H\hat{X}^{LP}_{k|k-1}, \quad (5.13)$$

where $\hat{X}^{LP}_{k|k-1}$ is found from $\hat{X}^{LP}_{k-1|k-1}$ via

$$\hat{X}^{LP}_{k|k-1} = F\hat{X}^{LP}_{k-1|k-1}. \quad (5.14)$$

To determine the new *a posteriori* state estimate, $\hat{X}^{LP}_{k|k}$, based on the observation, $\mathbf{Q}[S_k - \hat{S}^{LP}_{k|k-1}]$, and the *a priori* state estimate vector, $\hat{X}^{LP}_{k|k-1}$, we have:

$$\hat{X}^{LP}_{k|k} = \hat{X}^{LP}_{k|k-1} + H^T\mathbf{Q}[S_k - \hat{S}^{LP}_{k|k-1}]. \quad (5.15)$$

This directly corresponds to equation 5.9. The use of this notation will assist with understanding the differences between the standard Linear Predictor and the Kalman Filter, presented in the following section.

Remark 5.2 For zero quantization noise within the ADPCM system, the reconstructed estimate, $\hat{S}^{LP}_{k-1|k-1}$ would be exactly S_{k-1}. However, for any practical system, this is not the case.

For signal coding, the decoder has no access to the original input signal and hence is unable to obtain directly the a_i coefficients relevant to the input speech. Forwards adaptation schemes compute the a_i coefficients based on the input speech and transmit these to the decoder in a quantized form. Both the encoder and decoder then use the quantized coefficients in the prediction process. Backwards adaptation is based on the fact that the reconstructed signal available at both the encoder and decoder can be used to produce a set of coefficients for the prediction process. If the signal is 'fairly stationary' (or frequent coefficient updates are used), and the reconstructed signal is close to the input signal, then it is reasonable to expect that the backwards adaptation approach would perform well. Certainly it eliminates the need for transmission of additional side information for the coefficients.

As signal coding is concerned with transmitting a signal using a finite number of bits, it is perhaps optimistic to believe that the reconstructed estimates, $\hat{S}^{LP}_{k-1|k-1}$, would not have significant errors compared to S_{k-1}. Especially in low bit rate applications, due to coarse quantization, the reconstructed samples could be expected to be far from the input samples. Given this situation, we might well expect to obtain better performance by accounting for the fact. The Kalman Filter is known to be capable of effectively utilizing quantization noise statistics to provide improved performance.

5.4 The Kalman Filter

Crisafulli *et al.*[44] discuss how the Kalman Filter can be applied to speech coding, and note that the performance gain is due to the prediction filter using smoothed estimates of past speech samples. In their work, they show that the Kalman filter is a logical extension of the standard linear predictor, complementing previous work by Gibson[74] that uses the Kalman filter to remove some element of the quantization noise in an ADPCM system. We follow very closely, and extend, the approach in [44].

From the state-space formulation in equations 5.3 and 5.4 above, we can view the problem of linear prediction as one of state estimation for the state X_k. The Kalman Filter can be applied to give the 'optimal' state estimation for the linear gaussian system[4]. From [44], the Kalman Filtering equations relevant to the signal coding application, based on the all-pole signal model are:

$$\hat{X}^{KF}_{k|k} = \hat{X}^{KF}_{k|k-1} + K_k \mathbf{Q}[S_k - \hat{S}^{KF}_{k|k-1}] \tag{5.16}$$

$$\hat{X}^{KF}_{k+1|k} = F\hat{X}^{KF}_{k|k}, \tag{5.17}$$

where K_k is the Kalman gain vector, given by

$$K_k = P_k H^T \left[H P_k H^T + R_k \right]^{-1} \tag{5.18}$$

and

$$P_{k+1} = FP_kF^T - FP_kH^T\left[HP_kH^T + R_k\right]^{-1}HP_kF^T + Q_k, \tag{5.19}$$

is a Riccati Difference Equation (RDE) which recursively calculates the error covariance matrix, P_k. The quantity

$$R_k = E[n_k^2] = \sigma_{nk}^2 \tag{5.20}$$

is the variance of the measurement noise, n_k, related to the quantized measurement of the prediction error signal, $\mathbf{Q}[S_k - \hat{S}_{k|k-1}^{KF}]$, and

$$Q_k = E[\bar{w}_k\bar{w}_k^T] = \begin{pmatrix} \sigma_{wk}^2 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \tag{5.21}$$

is the excitation noise variance matrix. Note the use of the KF superscript to distinguish between the Kalman Filtering and standard Linear Prediction approaches.

Remark 5.3 In the above development we have introduced the notion of a quantization noise, seemingly implying a restriction to the coding application. This is not the case, as in the standard filtering or smoothing applications, the 'quantization' measurement $\mathbf{Q}[S_k - \hat{S}_{k|k-1}^{KF}]$ would simply refer to an observation of $[S_k - \hat{S}_{k|k-1}^{KF}]$, and the 'quantization' noise is actually the observation noise, due to the fact that the available signal is the 'clean' signal plus noise, $S_k + n_k$. (Where n_k is defined as the observation noise: $n_k \triangleq (S_k - \hat{S}_{k|k-1}^{KF}) - \mathbf{Q}[S_k - \hat{S}_{k|k-1}^{KF}]$.)

Remark 5.4 Equation 5.16 is often referred to as a measurement update, as it updates the state estimate based on the observation. Likewise, equation 5.17 is a time update, updating the state estimate based on the system matrix, F, only. It is important to note that the difference between the Kalman Filter approach and the standard Linear Predictor is related to the use of the observations. By comparing equations 5.16-5.17 to equations 5.14-5.15, we see that the time updates are identical in nature, and the measurement updates differ by the use of the Kalman gain vector, K_k. From equations 5.16 and 5.15 we observe that replacing the Kalman gain vector K_k by the vector $H^T = [1 \;\; 0 \;\; \ldots \;\; 0]^T$ we obtain exactly the standard linear predictor.

Having chosen the system matrix, F, in the above form (equation 5.5), we have the following for the state estimate vector for the Kalman Filter

$$\hat{X}^{KF}_{k|k} = \begin{pmatrix} \hat{S}^{KF}_{k|k} \\ \hat{S}^{KF}_{k-1|k} \\ \hat{S}^{KF}_{k-2|k} \\ \vdots \\ \hat{S}^{KF}_{k-N+1|k} \end{pmatrix}. \tag{5.22}$$

Thus the previous speech sample estimates are present in the state vector to various smoothing lags. It is then a simple matter of reading out the smoothed reconstructed sample from this state vector. Another way of writing 5.16, 5.17 and 5.22 is

$$\begin{aligned} \hat{S}^{KF}_{k|k-1} &= a_1\hat{S}^{KF}_{k-1|k-1} + a_2\hat{S}^{KF}_{k-2|k-1} \\ &\quad + a_3\hat{S}^{KF}_{k-3|k-1} + \cdots + a_N\hat{S}^{KF}_{k-N|k-1} \\ \hat{S}^{KF}_{k|k} &= \hat{S}^{KF}_{k|k-1} + K^1_k\mathbf{Q}[S_k - \hat{S}^{KF}_{k|k-1}] \\ \hat{S}^{KF}_{k-1|k} &= \hat{S}^{KF}_{k-1|k-1} + K^2_k\mathbf{Q}[S_k - \hat{S}^{KF}_{k|k-1}] \\ &\vdots \\ \hat{S}^{KF}_{k-N+1|k} &= \hat{S}^{KF}_{k-N+1|k-1} + K^N_k\mathbf{Q}[S_k - \hat{S}^{KF}_{k|k-1}], \end{aligned} \tag{5.23}$$

where K^i_k is the ith entry of the Kalman gain vector at sample k, K_k. This can be compared with the Linear Predictor in equation 5.8.

The Kalman predictor takes better account of the measurement (quantization or observation) noise in the signal coding situation, by recognising that the reconstructed sample $\hat{S}^{KF}_{k|k}$ is not a perfect estimate of the input sample S_k. The Kalman Filter exploits the correlation between samples, given by the all-pole signal model, and noise statistics information, to obtain smoothed estimates of the input samples for use in the prediction, hopefully resulting in better predictions.[1] For zero mean white gaussian noise, the Kalman filter is the optimal predictor, and for non-gaussian quantization noise (zero mean white) the Kalman filter results in the best linear predictor.

Remark 5.5 For the case of zero measurement noise, both the Linear Predictor, and Kalman Filter estimates, $\hat{S}^{LP}_{k|k}$ and $\hat{S}^{KF}_{k|k}$, are perfect estimates of the input sample S_k. In this situation, it is easy to see that the linear prediction given by equation 5.8 is identical to the optimal prediction based on the model, as presented in equation 5.6. As we are claiming that the Kalman filtering approach gives the optimal linear prediction,

[1] Appendix C concerns a justification of the reduced order Kalman filtering approach presented in Section 5.9 and, as an intermediate result, shows that the sample estimate error covariance decreases with smoothing lag.

it is reassuring to note that for the case of zero measurement noise, the Kalman filter collapses to the standard Linear Predictor. This is shown in Appendix B, and has previously been discussed in [42].

Linear Predictors for speech coding typically range from order 10 to order 50 (as in LD-CELP), and for input noise filtering or speech enhancement, model orders of from 4 to 20 are often seen. An inspection of the Kalman Filter reveals that the Riccati equation (equation 5.19) involves matrix multiplications of $(N \times N)$ matrices, where N is the model order. Hence the computation cost is order N^3, $O(N^3)$. Certainly this is a significant problem for large values of the Kalman Filter order, N. A closer observation of the particular Riccati equation described above shows that, although there are matrix multiplications involved, the computation cost is only $O(N^2)$. This is due to the nature of the model matrix, F, which only has one row of elements that require multiplications, and the rest of the rows only involve shifts.

For Kalman Filtering with a 50th order predictor, the computational cost of the update equations run every sample (8 kHz sampling) is near 120 Mips (Millions of instructions per second). Comparing this with the total cost of the already computation hungry LD-CELP algorithm at around 40 Mips, it is readily seen that Kalman Filtering is infeasible in this situation. For a 10th order predictor, the cost is 5 Mips, which is more manageable. The cost for a 5th order predictor is only 1.2 Mips.

Remark 5.6 For some systems, the computational complexity of the Kalman Filtering approach can be reduced even further through the use of steady state approximations to the solution of the Riccati equation[42]. The validity of this approach obviously depends heavily on the particular application. A brief coverage of these issues will be given in Section 5.10.

5.5 Input Noise Filtering and Speech Enhancement

As presented above, the Kalman Filter can be used to smooth or filter out noise in speech signals. This is an important application area for the Kalman filter, and the scheme has received significant attention for use in enhancement of degraded voice recordings, enhancement prior to the use of speech recognition algorithms, and background input noise filtering prior to speech coding. This section presents the use of the Kalman Filter in a general sense, and acts as a foundation for the next two sections that are more specific to the speech coding application.

Examples of papers discussing the general application of smoothing to speech processing are those by Chang and Gibson[30] and Rabiner, Sambur, and Schmidt[144]. The paper by Paliwal and Basu[138] specifically concerns the use of the Kalman filter for speech enhancement.

The Kalman Filter state estimate vector, presented in equation 5.22, consists of estimates of the last N speech samples to various smoothing lags. As only a small delay is incurred by each additional smoothing lag, and the smoothing performance generally increases with lag, the smoothed samples are normally extracted from the state estimate vector after the maximum smoothing lag. For Kalman smoothing with the all-pole signal model, the use of different smoothing lags does not imply any additional computation, and the only cost is the delay. (Subsequent discussions will consider smoothing in more detail.)

Hence to obtain the smoothed estimate of the sample at time k, we must wait until time $k+N-1$, were we have the following state estimate vector

$$\hat{X}^{KF}_{k+N-1|k+N-1} = \begin{pmatrix} \hat{S}^{KF}_{k+N-1|k+N-1} \\ \hat{S}^{KF}_{k+N-2|k+N-1} \\ \hat{S}^{KF}_{k+N-3|k+N-1} \\ \vdots \\ \hat{S}^{KF}_{k|k+N-1} \end{pmatrix}. \tag{5.24}$$

The smoothed estimate, $\hat{S}^{KF}_{k|k+N-1}$, is then simply read from the state estimate vector.

When faced with an input signal on which we would like to perform Kalman smoothing, we need estimates of the parameters used by the Kalman filter. Speech signals are highly non-stationary, and hence an adaptive approach must be taken for the production of the parameters of the all-pole model. For speech it is extremely unrealistic to expect any additional information about the model parameters, other than what is obtained from the noisy speech input file or stream. Hence for input noise (background noise) filtering we do not consider the use of 'actual' model parameters obtained from the clean speech signal, as this is not available.

For the simulations discussed below, we use backwards adaptive model parameters based on auto-correlation analysis and Levison-Durbin recursion on windowed previous filtered speech. For the use of the Kalman Filter, we use an estimate of the excitation noise variance, $\hat{\sigma}^2_{wk}$, and an input noise (observation or measurement noise) estimate, $\hat{\sigma}^2_{nk}$. The problem of estimation of the input noise variance deserves special mention, and is given this in the next subsection.

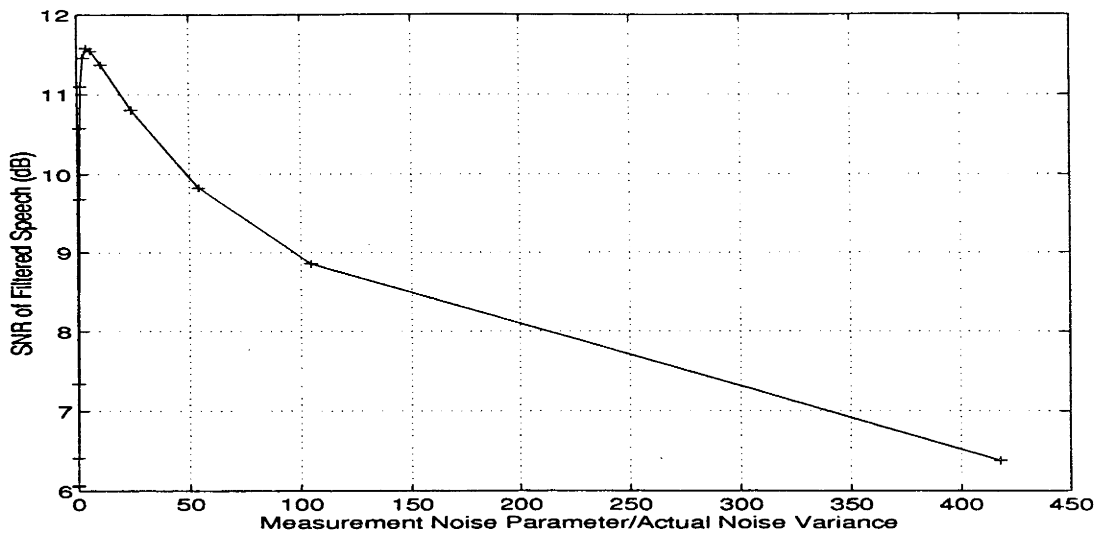

Figure 5.1: Filtered Speech SNR versus Ratio of $\hat{\sigma}^2_{nk}$ to σ^2_{nk}

5.5.1 Measurement Noise Variance Parameter Estimation

Speech signals are inherently highly bursty in nature, having long silence periods between words and utterances, and while listening to another speaker in a two-way conversation. Average speech activity levels for one talker are often quoted to be around 40% of the total time of a conversation. The fact that the signal is zero for 60% of the time leaves a large opportunity to observe only the noise present, and obtain parameter estimates for the noise.

Although we do not propose to implement such an approach, it is worth considering the effect of a mismatch between the actual noise variance, σ^2_{nk}, and the estimate of the variance, $\hat{\sigma}^2_{nk}$, which is used as a parameter to the Kalman Filter. Figures 5.1 and 5.2 display the results of simulating such a parameter mismatch. The input signal consists of 22 seconds of speech with four male and four female speakers. A white gaussian noise has been added to the speech signal to provide an overall SNR of 6dB.

Figure 5.1 displays the SNR of the filtered speech against the ratio of the estimated noise variance to the actual noise variance. This graph shows that a mismatch of over an order of magnitude (with the estimate larger than the actual value), is possible before a large degradation in SNR occurs. Also, at all times the filtered output is better in terms of SNR than the input (6dB SNR).

Unfortunately Figure 5.1 does not clearly show the detail for the smaller variance

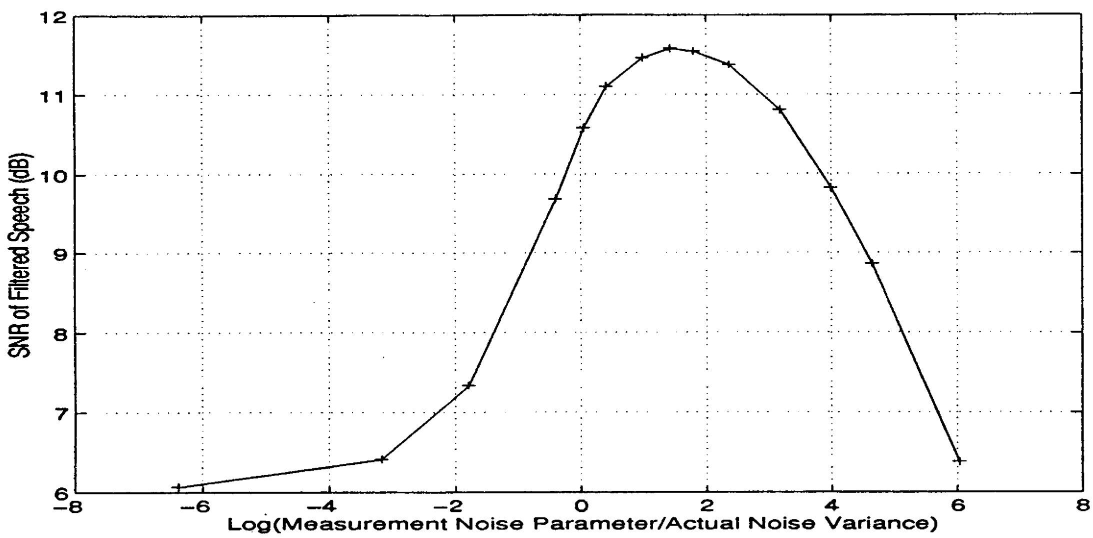

Figure 5.2: Filtered Speech SNR versus Log Ratio of $\hat{\sigma}^2_{nk}$ to σ^2_{nk}

ratios that are more common in practice. For this reason Figure 5.2 presents the same information with a logarithmic scale (natural log) for the variance ratio. From this graph it is seen that the peak in filtered speech SNR does not correspond to where the variance estimate is equal to the actual variance (zero on the log-ratio scale). Rather, the peak in SNR appears to occur for a ratio where the estimated variance is five times larger than the actual variance.

This observation can be related to the inadequacy of the AR (all-pole) speech model, and robustness considerations of the Kalman filter[4], and suggests that it is much better to err on the high side for the noise variance estimate. However the SNR peak being so distant from its expected location needs some explanation. For speech, SNR is not always a good indication of output quality, and in this case the situation is no different. As the input noise variance estimate is increased from zero to the actual noise variance value, we generally observe increasing subjective quality of the filtered speech output. Further increasing the noise variance estimate by a factor of two or three results in some slight speech quality degradation, and once we approach an order of magnitude in the variance ratio, the speech starts to become severely muffled. This muffling appears to indicate the Kalman filter being too conservative, and filtering out a large part of the excitation, especially in the high frequencies. Hence an order of magnitude difference in the noise variance is not recommended, but small deviations (especially on the high side) have only a minimal effect.

The above simulations were performed with a backwards adaptive 10th order signal model, and a backwards adaptive variance estimator for the process noise variance parameter, $\hat{\sigma}^2_{wk}$. With the correct value for the measurement noise variance parameter, σ^2_{nk}, the performance of the Kalman smoother was observed to be quite significant, both in terms of SNR measurements and in terms of subjective quality. However, a high level of residual noise is observed within the speech silence periods, and perceptually this is very annoying due to a small 'swirling' artifact introduced by the Kalman filter.[2] Closer analysis reveals that due to the backwards adaptive procedure used to calculate the process noise, although the actual process noise variance, σ^2_{wk}, is zero during the speech silence periods, the estimated process noise variance, $\hat{\sigma}^2_{wk}$, is non-zero.

Incorporating a threshold into the calculation of the process noise parameter, $\hat{\sigma}^2_{wk}$, provides a significant performance improvement. We choose this threshold such that when the process noise value obtained from the backwards adaptive variance estimator is less than the measurement noise parameter, σ^2_{nk}, the process noise parameter, $\hat{\sigma}^2_{wk}$, is set to zero. Note that for this approach some care must be taken not to inadvertently set the measurement noise parameter, σ^2_{nk}, to zero (or a very small value for finite precision arithmetic), as this will result in ill-conditioned computation for the inverse operation in equation 5.18.

For the speech input signal from above, with a constant noise level of 6 dB below the average speech power, the Kalman smoother utilizing this threshold approach obtained an SNR value of 11.30 dB, and a segmental SNR value of -6.37 dB. This represents an SNR improvement over not using the threshold of 0.72 dB, and a segmental SNR improvement of 1.36 dB. The corresponding subjective improvement was significant, with the noise level during silence periods greatly reduced, and consequently the swirling effect made much less noticeable.

5.5.2 Effect of Noise Input Level

It is useful to consider the performance of the Kalman Filter for a variety of input noise levels. Figure 5.3 presents the filtered output speech SNR for input noise levels relative to the signal power of between 42 dB and -15 dB. For the simulations shown in this graph, the same 22.5 second speech input file was used, and again the process noise

[2]The swirling artifact appears to indicate some form of slow-scale instability caused by independent adaptation of several parameters. Although this is a valid topic for our attention, we do not give it consideration as the use of a threshold on the process noise variance estimate effectively eliminates the problem.

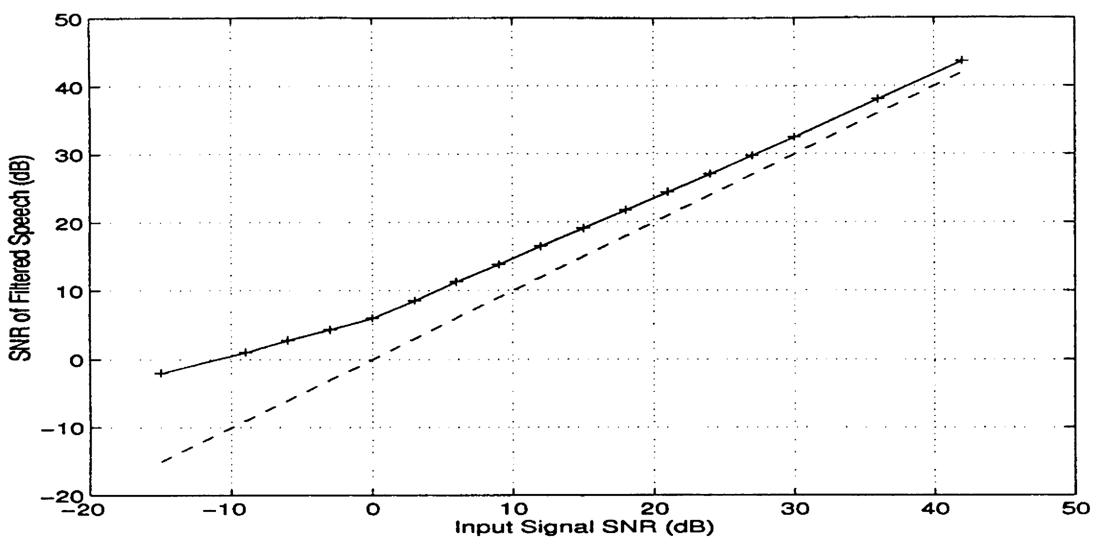

Figure 5.3: Filtered Speech SNR versus Input Noise SNR

parameter, $\hat{\sigma}^2_{wk}$, was obtained from a variance estimator with threshold incorporation. The measurement noise parameter used, $\hat{\sigma}^2_{nk}$, was the actual variance value of the noise.

The dashed line on the graph in Figure 5.3 is at a slope of one, and helps to visualize the trend in the Kalman filter performance. It is clear that in terms of SNR improvement, the Kalman filter provides more significant benefits for higher levels of input noise. For speech, an input signal to noise ratio of better than 30 dB is unusual, but values are included on the graph to display the gradual decrease in SNR improvement. Likewise, speech intelligibility is lost completely for the higher noise levels, so they are not seriously considered, except perhaps in some speech enhancement applications.

It should be clear that the Kalman filter can provide significant filtering improvements across a wide range of input conditions, and is thus a valuable tool. The question of the cost incurred by the use of Kalman Filtering, generally in terms of computation, has yet to be fully considered.

5.5.3 Waveform Plots

In order to assist with conceptualizing the performance gain achieved by Kalman smoothing, some waveform plots are provided in Figures 5.4 to 5.6. Figure 5.4 presents the smoother input waveform, which is a 250 ms segment of speech plus noise taken

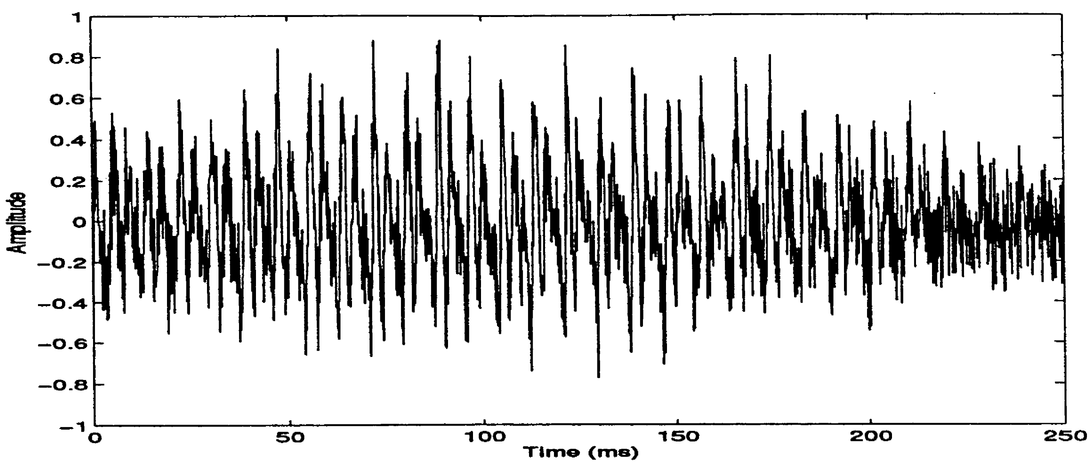

Figure 5.4: Kalman Smoother Input Waveform

from the file used for the above simulations. The observation or background noise is a constant additive white gaussian noise at a level of 0 dB relative to the average speech signal power over the whole file.

Figure 5.5 shows the output waveform from the Kalman smoother, and this can be compared with the clean speech signal in Figure 5.6.

From observation of the Kalman smoother output in Figure 5.5, it is apparent that there is still a substantial amount of noise present relative to the clean speech signal in Figure 5.6. However, it should also be clear that the smoother output is a significant improvement over the very high noise level present in Figure 5.4.

5.5.4 Model Order for Kalman Filtering

The choice of model order used in the Kalman filtering process deserves further mention. Obviously we expect a higher order model to provide improved performance, but this comes at a computation cost. An engineering trade-off exists between these two issues, and it is useful to have some idea of the general considerations involved.

From Table 5.1 it is clear that even for very high levels of input noise, where we observe large SNR improvements through the use of the Kalman Filter, there is little gain from using more than a very small order signal model. This is an important observation, and forms part of the consideration for the reduced complexity Kalman filtering approach introduced in Section 5.9.

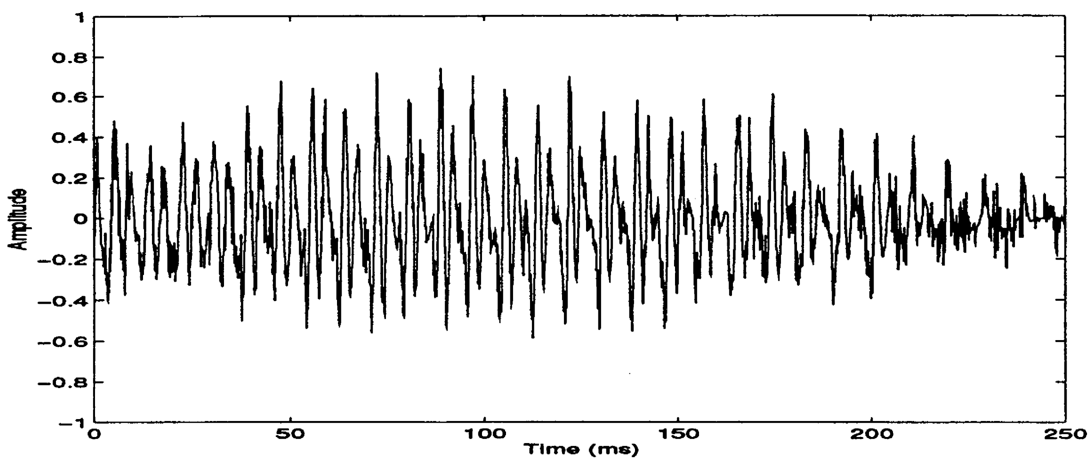

Figure 5.5: Kalman Smoother Output Waveform

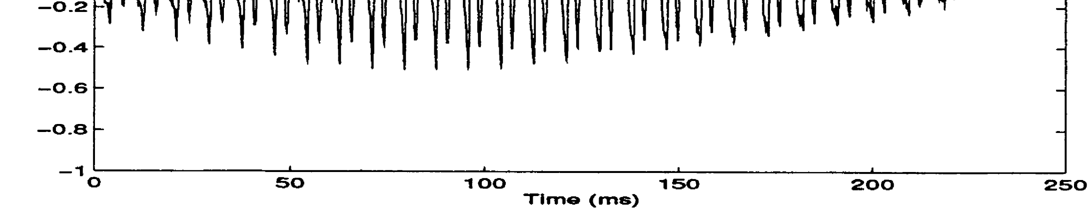

Figure 5.6: Original Waveform

Kalman Filter Order	6 dB Input Noise		-6 dB Input Noise	
	Filtered Output SNR (dB)	Output Segmental SNR (dB)	Filtered Output SNR (dB)	Output Segmental SNR (dB)
1	10.35	-6.74	1.92	-11.42
2	11.11	-6.36	2.29	-11.47
3	11.19	-6.33	2.40	-11.50
4	11.23	-6.34	2.55	-11.53
6	11.29	-6.33	2.74	-11.45
8	11.29	-6.37	2.80	-11.59
10	11.30	-6.37	2.85	-11.59

Table 5.1: SNR versus Kalman Filter Order for Various Noise Levels

5.6 Kalman Prediction

In speech coding, there is a requirement to filter quantization noise to produce a high quality reconstructed signal. This is a problem very similar in nature to the input noise filtering or speech enhancement problems discussed in the previous section. For the assumption of white quantization noise, these problems are effectively identical, and a coloured quantization noise assumption is also very similar to the coloured input noise scenario (Section 5.8).

Two useful papers considering the general topic of prediction in an ADPCM system are those by Honig and Messerschmitt[85] and Gibson[72]. One paper specifically discussing the use of the Kalman filter in DPCM prediction is that of Pirani and Scagliola[141].

The filtering benefits of the Kalman Filter can also be exploited within a signal encoder to obtain improved prediction. This differs from the previous problem by the absence of any allowable delay to utilize for a smoothing benefit. However, the Kalman Filter does allow us to form the prediction

$$\hat{S}^{KF}_{k|k-1} = H\hat{X}^{KF}_{k|k-1}, \tag{5.25}$$

where $\hat{X}^{KF}_{k|k-1}$ is the *a priori* Kalman filter state estimate vector:

$$\hat{X}^{KF}_{k|k-1} = \begin{pmatrix} \hat{S}^{KF}_{k|k-1} \\ \hat{S}^{KF}_{k-1|k-1} \\ \hat{S}^{KF}_{k-2|k-1} \\ \vdots \\ \hat{S}^{KF}_{k-N+1|k-1} \end{pmatrix} \tag{5.26}$$

Hence some smoothing benefits are effectively being captured in the prediction formed by the Kalman filter. (Compare the above state estimate with that from the standard linear predictor shown in equation 5.22.)

We thus exploit the smoothing properties of the Kalman Filter to produce a higher quality prediction than that produced by the standard linear predictor. Simulation results are shown in Table 5.2, obtained with the use of gaussian white noise injected to represent quantization noise. For the simulation results presented in this table, and in Figure 5.7, a constant quantization noise level has been used.

Prediction Filter Order	Prediction Gain (dB)			
	6 dB Quantization Noise		-6 dB Quantization Noise	
	Kalman Filter	Linear Predictor	Kalman Filter	Linear Predictor
1	4.637	4.880	1.930	1.833
2	5.803	5.638	2.136	1.948
3	6.018	5.943	2.306	1.938
4	6.277	6.114	2.455	1.912
6	6.809	6.651	2.666	1.870
8	6.949	6.760	2.673	1.710
10	7.130	6.876	2.648	1.505

Table 5.2: Prediction Gain versus Filter Order for Selected Noise Levels

From the table it is clear that some additional prediction gain is afforded via the use of the Kalman filter. In Figure 5.7 the prediction gain of the Kalman filter is shown with the solid curve, and the standard linear predictor is shown with a dashed curve. The graph displays clearly that the additional Kalman filter prediction gain relative to the standard linear predictor is greatest for larger levels of quantization noise. Although not readily observable from Figure 5.7, the prediction gain for the Kalman filter is slightly worse than that of the standard linear predictor for very low levels of quantization noise. However, the effect does not appear significant.

Perhaps a more usual assumption with regard to quantization noise is that the noise power is relative to the input signal power. This type of scenario can be assumed to exist where adaptive quantization and similar types of coding approaches are used, incorporating quantization noise scaled relative to the input signal power. A comparison of prediction gains for this scenario (with white gaussian noise injected) is shown in Figure 5.8. From the graph, the general trends in prediction gain can be seen to be very similar to those observed in Figure 5.7.

An even more realistic assumption is that the quantization noise level is fixed in

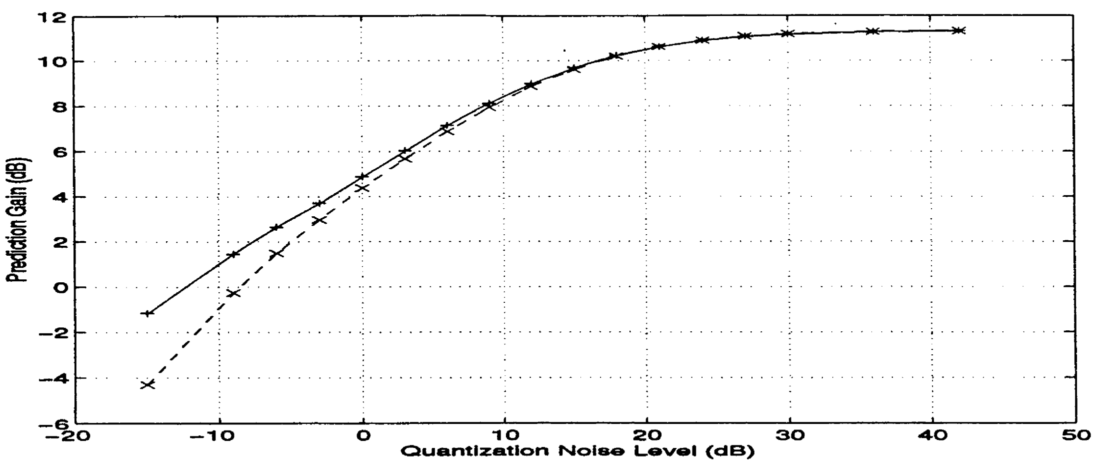

Figure 5.7: Prediction Gain for Kalman Predictor (solid line) and Standard Linear Predictor (dashed line) versus Quantization Noise Level (Relative to Average Signal Level)

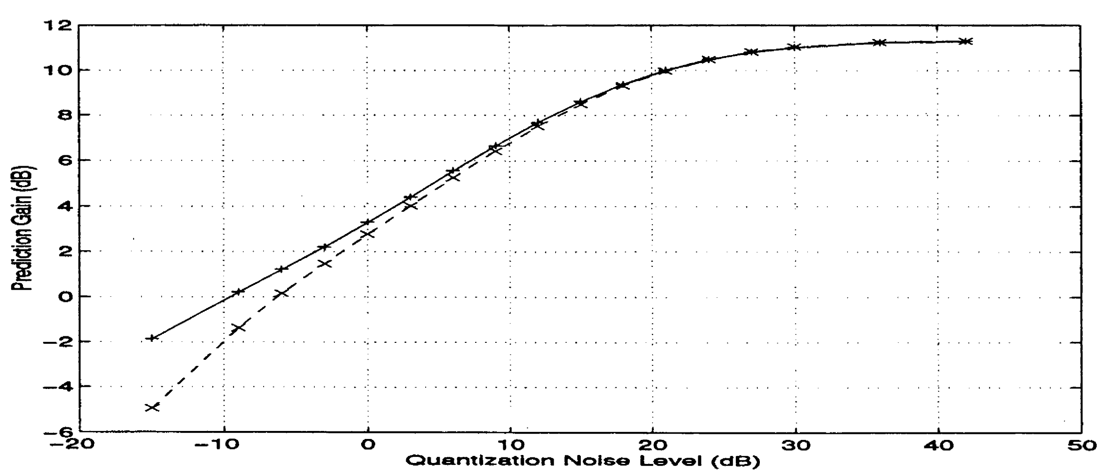

Figure 5.8: Prediction Gain for Kalman Predictor (solid line) and Standard Linear Predictor (dashed line) versus Quantization Noise Level (Relative to Input Signal Level)

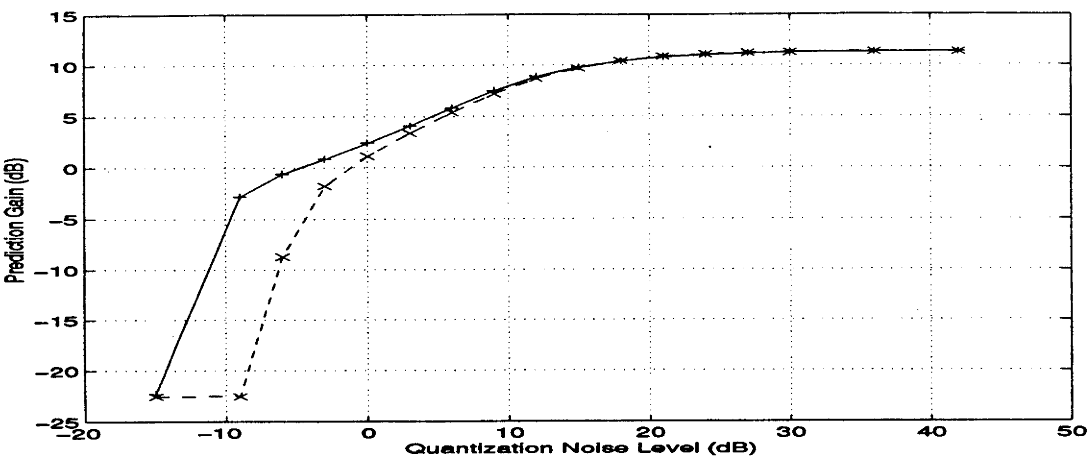

Figure 5.9: Prediction Gain for Kalman Predictor (solid line) and Standard Linear Predictor (dashed line) versus Quantization Noise Level (Relative to Prediction Error Level)

relation to the prediction difference signal level. This is a valid assumption for many predictive coding systems such as ADPCM and CELP approaches. Simulation results for this assumption are presented in Figure 5.9.

Figure 5.9 shows some significantly different characteristics to Figures 5.7 and 5.8, and is worthy of further discussion. For low levels of quantization noise, the prediction gains obtained are very similar to those shown in the previous graphs. However, for higher quantization noise levels some 'strange' behaviour is apparent. The prediction gain for the standard linear predictor rapidly drops, while that of the Kalman predictor drops, but considerably more slowly.

This behaviour can be traced to the adaptation strategy for the quantizer (or for the current simulations, the variance adaptation for the synthetic 'quantization' noise), and is a stability problem of the type discussed in Chapter 3. Large prediction errors result in large quantizer errors fed back into the predictor, causing a form of instability. Although the additional prediction gain afforded via Kalman prediction in general appears rather small, it is clear from Figure 5.9 that some significant advantages can be obtained where the feedback loop introduces stability problems.

Remark 5.7 The 'instability' discussed above from Figure 5.9 is somewhat artificial, as the high level 'quantization' noise is actually significantly larger than the prediction difference signal that is being 'quantized'. For signal coding this would be a highly unusual situation, with quantization noise normally being limited in size to the signal being quantized (zero bit quantization). The simulations at this stage are performed with white noise injected to simulate quantization noise, and hence the extremely high quantization noise situation should be ignored. However, the 'instability' trend is still apparent before this, and would seem to be important.

Ignoring this last issue with regard to the connection with stability for adaptive quantization, it would appear that only a small prediction gain is obtained via the use of the Kalman filter for prediction. Hence, its use within a signal coding system must be questioned, as the computation cost is high, yet the performance benefit obtained is only small.

It has already been observed that the performance gain obtained from the Kalman filter for smoothing the effects of additive noise is significant. The next section considers the use of the Kalman filter in providing an improved reconstruction signal in a predictive coding system. However, before moving to this section, one final issue must be mentioned with the use of the Kalman filter within the prediction loop.

An important issue with all speech prediction is how the predictor coefficients are obtained. For all the curves above we obtained the predictor coefficients in a backwards adaptive fashion via auto-correlation analysis of the reconstructed output speech signal. The use of a forwards adaptive approach initially resulted in extremely poor performance, but the addition of an appropriate level of White Noise Correction (WNC) to desensitise the predictor to the noise present in the reconstructed speech signal was found to improve performance significantly. This observation of better performance with more 'conservative' prediction is somewhat analogous to the better SNR figures obtained with overestimates of the Kalman filter measurement noise parameter, $\hat{\sigma}^2_{nk}$, in Figure 5.2.

The standard linear predictor was then observed to obtain slightly better performance with forwards adaptive parameters, as might be reasonably expected. However, it is important to note that better performance was not obtained via the use of forwards adaptation with the Kalman filtering approach. The reason for this is somewhat unclear, but not appearing to be of great significance, as with the predictor update period of 20 samples (2.5 ms), a large improvement through the use of forwards adaptation is not expected (since the backwards adaptive coefficients can be assumed to provide an adequate model).

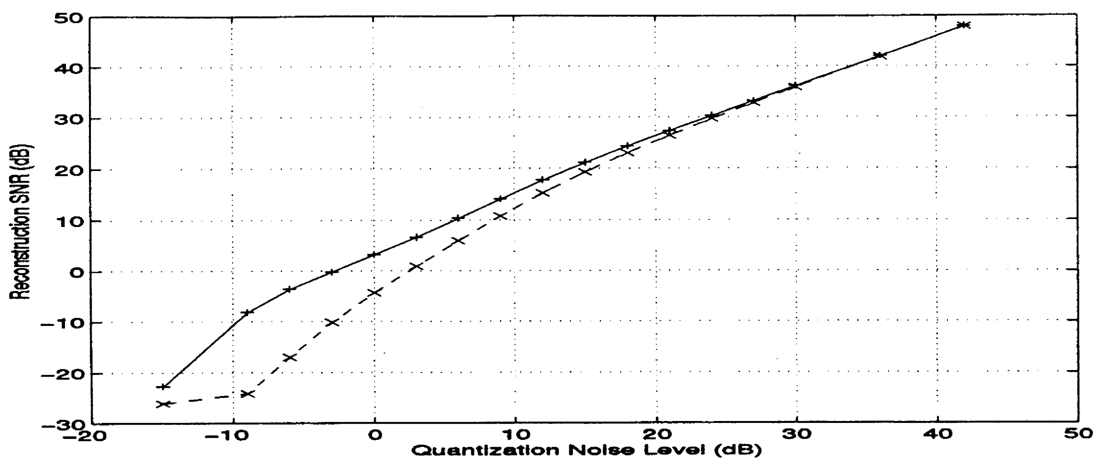

Figure 5.10: Kalman Filter Smoothed Reconstruction SNR (solid line) versus Quantization Noise Level (Standard Linear Prediction Reconstruction – dashed line)

5.7 Reconstructed Output Smoothing

Although only relatively small additional prediction gains have been observed through the use of the Kalman filter for prediction within a speech coding system, the smoothing benefit of the Kalman filter can be exploited to significant advantage in the signal decoder. The smoothing advantage comes at the cost of a small (usually insignificant) additional delay. However, it has been shown in [44], that the smoothed reconstruction signal can be obtained from the Kalman filter predictor at no additional computational cost.

Figure 5.10 shows the SNR of the smoothed reconstruction signal within the Kalman filter, $\hat{S}^{KF}_{k|k+n}$, compared to the reconstructed signal from the standard linear predictor, $\hat{S}^{LPC}_{k|k}$, (dashed curve). In terms of SNR for a 'middle' range of quantization noise, 3 or 4 dB improvement is observed (a number of values from the graph are reproduced in Table 5.3 for convenience). For higher levels of quantization noise, even larger SNR improvements are obtained with the use of the Kalman filter smoothed output. Hence, although the Kalman filter was observed to give only minimal additional prediction gain over the standard linear predictor, its smoothing benefit can be exploited for the production of a significantly better reconstructed output signal.

From Table 5.3 and Figure 5.10 it is clear that a substantial advantage exists for the use of the Kalman filter within signal coding systems to filter quantization noise. An obvious question that appears due to the small prediction gain observed is whether

Quantization Noise Level (dB)	Standard Predictor		Kalman Filter	
	Reconstructed Output SNR (dB)	Output Segmental SNR (dB)	Smoothed Output SNR (dB)	Output Segmental SNR (dB)
0	-4.26	-4.94	3.24	1.56
3	0.98	0.60	6.65	5.69
6	6.00	5.81	10.36	9.91
9	10.79	10.60	14.07	13.72
12	15.28	14.93	17.76	17.23
15	19.32	18.96	21.14	20.32
18	23.05	22.01	24.34	23.13

Table 5.3: Linear Predictor and Smoothed Kalman Filter SNR for various Noise Levels

the Kalman filter is of any significant advantage in the encoder. Perhaps the smoothing advantage of the Kalman filter in the decoder can be obtained without the extra computational complexity incurred by Kalman filtering within the encoder. Further discussion on this topic is deferred until Chapter 7, where the Kalman filter is applied to CELP speech coding systems.

5.8 Coloured Noise and the Kalman Filter

The basic Kalman filter development relies on the assumption of white gaussian measurement noise. There are several common situations where the white noise assumption is not particularly valid, and a coloured noise Kalman filter would be useful. Simple examples are for filtering of background noise in a car or helicopter.

A paper by Brown and Sage[24], and more recent papers by Koo, Gibson, and Gray[76, 119] show how the basic Kalman filter approach can be modified to suit the coloured noise assumption, by a simple augmentation of the state vector. A paper by Chang and Yang[31] also discusses the use of the coloured noise Kalman filter, but for the application of filtering coloured quantization noise in DPCM systems.

Obviously the coloured noise Kalman Filter requires a noise model, and to adapt this in a backwards fashion at the same time as the predictor is updated is an extremely difficult optimization problem. Fortunately for most applications simple noise models, even first order models, are able to provide significant benefit. Techniques do exist to update the noise models, often exploiting silence periods during speech to perform this operation.

This section follows closely the work of Gibson *et al.* in [76], but deals specifically with the all-pole source model presented in equation 5.1. A first order model is assumed for the coloured noise, as this appears to be useful for modelling car noise, and can be easily extended to higher order if required.

From equations 5.3 and 5.4, the state space model for the system is:

$$X_k = FX_{k-1} + Gw_k \tag{5.27}$$

$$S_k = HX_k + v_k \tag{5.28}$$

where v_k is defined as the observation or measurement noise, and the column vector $G = [1, 0, \ldots, 0]^T$.

A first order model for coloured v_k is

$$v_k = b_1 v_{k-1} + \eta_k, \tag{5.29}$$

where η_k is a white process, and b_1 is the first order AR coefficient for the coloured noise.

Hence the augmented system is:

$$\begin{bmatrix} X_k \\ v_k \end{bmatrix} = \begin{bmatrix} F & 0 \\ 0 & b_1 \end{bmatrix} \begin{bmatrix} X_{k-1} \\ v_{k-1} \end{bmatrix} + \begin{bmatrix} G & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} w_k \\ \eta_k \end{bmatrix}, \tag{5.30}$$

$$S_k = [H \;\; 1] \begin{bmatrix} X_k \\ v_k \end{bmatrix}, \tag{5.31}$$

and with relevant definitions this becomes

$$\bar{X}_k = \bar{F}\bar{X}_{k-1} + \bar{G}\bar{w}_k, \tag{5.32}$$

$$S_k = \bar{H}\bar{X}_k. \tag{5.33}$$

A transformation is then used to derive a reduced order optimal filter. Due to the structure of the system matrix F, we choose a transformation matrix of the form:

$$T = \begin{bmatrix} 1 & 0 & \ldots & 0 & 1 \\ 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 & 1 \end{bmatrix}. \tag{5.34}$$

This results in the transformed system

$$\tilde{X}_k = \tilde{F}\tilde{X}_{k-1} + \tilde{G}\bar{w}_k, \tag{5.35}$$

$$S_k = \tilde{H}\tilde{X}_k, \tag{5.36}$$

where

$$\tilde{F} = T\bar{F}T^{-1} = \left[\begin{array}{cccc|c} & & & & b_1 - a_1 \\ & F & & & -1 \\ & & & & 0 \\ & & & & \vdots \\ \hline 0 & 0 & \ldots & 0 & b_1 \end{array}\right], \tag{5.37}$$

$$\tilde{G} = T\bar{G} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \tag{5.38}$$

$$\tilde{H} = \bar{H}T^{-1} = [1 \;\; 0 \;\; \ldots \;\; 0], \tag{5.39}$$

$$\tilde{X}_k = T\bar{X}_k = \begin{bmatrix} S_k + v_k \\ S_{k-1} \\ \vdots \\ S_{k-N+1} \\ v_k \end{bmatrix}. \tag{5.40}$$

The structure of the state vector, $\tilde{X}_k$, is important. The observations are of the signal plus coloured noise, $S_k + v_k$, which appears as one element of the state vector. Hence the output vector, H, only has one non-zero element. This simplifies the computation required by the Riccati Difference Equation (RDE) 5.19. It should thus be clear that the incorporation of a coloured noise model into the Kalman filtering process is a relatively simple matter.

The coloured noise Kalman filter has been included in this chapter for reasons of completeness. The simplicity of the approach over the standard white noise assumption Kalman filter is important. However, we provide no simulation results specific to the use of the coloured noise assumption filter. The papers by Koo, Gibson, and Gray[76, 119] and Chang and Yang[31] discuss the use of the coloured noise assumption Kalman filter for both input noise filtering, and filtering of quantization noise within DPCM coding systems. The interested reader is advised to consult the references for further coverage of the coloured noise Kalman filter.

Remark 5.8 There is little doubt as to the validity of the coloured noise assumption for some input noise filtering applications[76, 119]. However, for filtering of quantization noise, the approach taken by Chang and Yang[31] needs to be carefully considered. In the studies in [31], the simulations involved some 'ideal' situations of synthetic input signals and quantization noises. Here it was seen that some significant advantage can be obtained through the use of the coloured noise Kalman filter. For the AC-ADPCM system introduced in Chapter 4, significantly coloured quantization noise was not observed. Hence the approach appears application dependent.

5.9 Complexity Reduction using Smoothing Properties

We noted above that the Kalman Filter obtains an advantage over the standard linear predictor through the use of smoothed sample estimates. Smoothing theory tells us that most of the smoothing gain is to be obtained in the first few smoothing lags[2, 44]. Since this is the case, we can obtain most of the advantage of Kalman Filtering through smoothing to say 4 or 5 lags, rather than continuing to the full order, possibly 50 lags.

As proposed in our paper [184], we present a reduced complexity (and consequently mathematically sub-optimal) Kalman Filter approach, that uses smoothed estimates in the predictor up to a relatively small lag of n samples. Our prediction is based on

$$\begin{aligned}
\hat{S}^{KFn}_{k|k-1} &= a_1\hat{S}^{KFn}_{k-1|k-1} + a_2\hat{S}^{KFn}_{k-2|k-1} + \cdots \\
&\quad + a_n\hat{S}^{KFn}_{k-n|k-1} + a_{n+1}\hat{S}^{KFn}_{k-n-1|k-2} \\
&\quad + a_{n+2}\hat{S}^{KFn}_{k-n-2|k-3} \\
&\quad + \cdots + a_N\hat{S}^{KFn}_{k-N|k-N+n-1} \\
\hat{S}^{KFn}_{k|k} &= \hat{S}^{KFn}_{k|k-1} + K^1_k\mathbf{Q}[S_k - \hat{S}^{KFn}_{k|k-1}] \\
\hat{S}^{KFn}_{k-1|k} &= \hat{S}^{KFn}_{k-1|k-1} + K^2_k\mathbf{Q}[S_k - \hat{S}^{KFn}_{k|k-1}] \\
&\vdots \\
\hat{S}^{KFn}_{k-n+1|k} &= \hat{S}^{KFn}_{k-n+1|k-1} + K^n_k\mathbf{Q}[S_k - \hat{S}^{KFn}_{k|k-1}] \\
\hat{S}^{KFn}_{k-n|k} &= \hat{S}^{KFn}_{k-n|k-1} \\
&\vdots \\
\hat{S}^{KFn}_{k-N+1|k} &= \hat{S}^{KFn}_{k-N+1|k-1}.
\end{aligned} \tag{5.41}$$

The superscript KFn has been introduced to denote our sub-optimal reduced complexity Kalman Filtering approach using smoothing to n lags. After n lag smoothing, the sample estimates are simply stepped back in time at each successive sampling instant for the remainder of the N sample state estimate.

The reduced complexity Kalman Filter state estimate vector has the form

$$\hat{X}_{k|k}^{KFn} = \begin{pmatrix} \hat{S}_{k|k}^{KFn} \\ \hat{S}_{k-1|k}^{KFn} \\ \vdots \\ \hat{S}_{k-n+1|k}^{KFn} \\ \hat{S}_{k-n|k-1}^{KFn} \\ \hat{S}_{k-n-1|k-2}^{KFn} \\ \vdots \\ \hat{S}_{k-N+1|k-N+n}^{KFn} \end{pmatrix}, \tag{5.42}$$

where the subscript notation reflects the extent to which smoothing is performed (smoothing is performed for the first n lags, after which the estimates are simply stepped back in time). Compare this state estimate vector with equation 5.8 for the standard linear predictor, and equation 5.23 for the Kalman Filter, and it can be seen that we now have a 'hybrid' type of approach.

Smoothing to n lags only can be viewed as being equivalent to assuming that only the top left hand $(n \times n)$ block of the error covariance matrix is non-zero.[3] Thus, we have an error covariance matrix of the form

$$P_k = \left(\begin{array}{c|c} P_k^{KFn} & 0 \\ \hline 0 & 0 \end{array} \right), \tag{5.43}$$

where P_k^{KFn} is an $(n \times n)$ matrix.

With P_k of this form, the Riccati equation 5.19, gives

$$P_{k+1} \approx \left(\begin{array}{c|c} P_{k+1}^{KFn} & 0 \\ \hline 0 & 0 \end{array} \right), \tag{5.44}$$

where P_{k+1}^{KFn} is found from the nth order Riccati Difference Equation:

$$\begin{aligned} P_{k+1}^{KFn} &= F^{KFn} P_k^{KFn} (F^{KFn})^T \\ &\quad - F^{KFn} P_k^{KFn} (H^{KFn})^T \left[H^{KFn} P_k^{KFn} (H^{KFn})^T + R_k \right]^{-1} H^{KFn} P_k^{KFn} (F^{KFn})^T \\ &\quad + Q_k^{KFn}, \end{aligned} \tag{5.45}$$

with F^{KFn} and Q_k^{KFn} defined as the top left $(n \times n)$ blocks from the F and Q_k matrices (from equation 5.5), and H^{KFn} defined as the vector of size n from the left of the H vector. In fact, for P_k as in equation 5.43, the top left $(n \times n)$ block of P_{k+1} is exactly P_{k+1}^{KFn}, and the only non-zero elements in P_{k+1} are in the top left $(n+1) \times (n+1)$ block.

[3] We omit any attempt at an intuitive justification for this assumption, as we shortly introduce an improved assumption.

If equation 5.43 is a good approximation for P_k obtained during full Kalman Filtering, then we would expect the non-zero elements outside the $(n \times n)$ top left block of P_{k+1} to be close to zero, and hence

$$P_{k+1} = \left(\begin{array}{c|c} P_{k+1}^{KFn} & 0 \\ \hline 0 & 0 \end{array} \right) \tag{5.46}$$

is a good approximation for P_{k+1}. We thus are able to reduce the computational cost of Kalman Filtering, through a sub-optimal approach, by simply updating the $n \times n$ error covariance matrix P_k^{KFn}. For small values of the measurement noise, R_k, (relative to the process noise Q_k) we note that the cost-performance compromise is likely to be heavily in favour of a very small order for P_k^{KFn}, while larger R_k suggests a higher order matrix approximation may be necessary.

Remark 5.9 For any given value of R_k, the choice of order n of the matrix P_k^{KFn} provides for a 'continuum' between standard linear prediction, and full Kalman Filtering. The standard linear predictor corresponds to a 0th order P_k^{KFn} matrix, and an Nth order matrix gives the full Kalman Filter.

We gave no intuitive justification for the the choice of P_k as in equation 5.43. However, we have noted that if this structure for P_k is a good approximation, then P_{k+1} should be very close to that given in equation 5.44. The motivation for choosing a P_k approximation of this form is to exploit the computational savings achieved by using the reduced order RDE as in equation 5.45. Likewise, we may consider alternative approximations to P_k, other than that given in equation 5.43.

An approximation of the form

$$P_k = \left(\begin{array}{c|c} P_k^{KFn^*} & 0 \\ \hline 0 & \sigma_{SR}^2 I \end{array} \right), \tag{5.47}$$

results in $P_{k+1}^{KFn^*}$ being found from the 'modified' reduced order Riccati equation:

$$\begin{aligned} P_{k+1}^{KFn^*} &= F^{KFn} P_k^{KFn^*} (F^{KFn})^T \\ &- F^{KFn} P_k^{KFn^*} (H^{KFn})^T \left[H^{KFn} P_k^{KFn^*} (H^{KFn})^T + R_k \right]^{-1} H^{KFn} P_k^{KFn^*} (F^{KFn})^T \\ &+ Q_k^{KFn} + \sigma_{SR}^2 \sum_{i=n+1}^{N} a_i^2 \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix}, \end{aligned} \tag{5.48}$$

where the only difference to the Riccati Difference Equation in 5.45 is the final term in σ_{SR}^2.

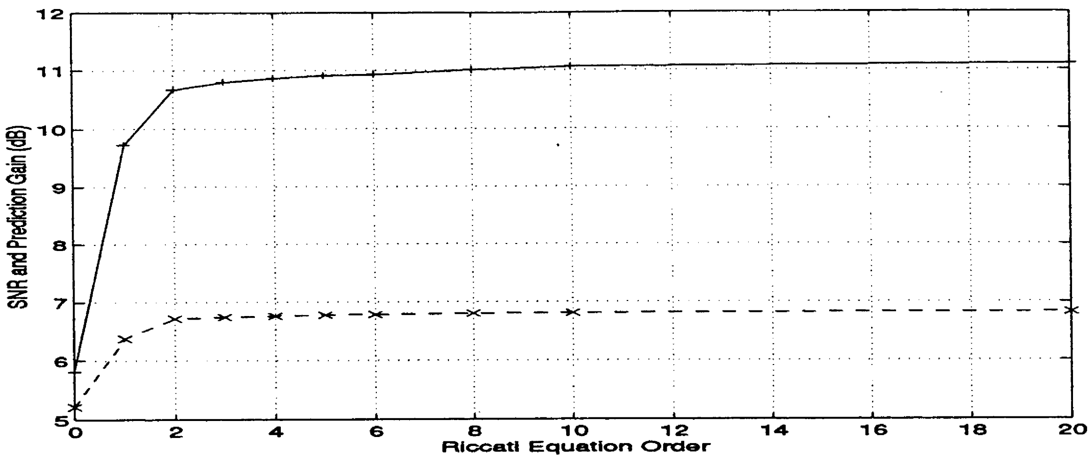

Figure 5.11: Smoothed Signal SNR (solid line) and Prediction Gain (dashed line) for Reduced Order Kalman Filter Approach

The above approximation for P_k reflects the fact that although we are only smoothing to n lags, there are residual uncertainties in the samples from lag $(n+1)$ to lag N. These uncertainties contribute to the *a priori* prediction error variance, as given by the term, $\sigma_{SR}^2 \sum_{i=n+1}^{N} a_i^2$. The additional computation required to take account of this term is minimal, as the summation only needs to be performed once per model update.

The value of the residual uncertainty, σ_{SR}^2, is not really another introduced parameter that we need to choose, as it can be calculated in a straight-forward manner from the RDE. (The σ_{SR}^2 value can be found from the bottom right hand value in the reduced order error covariance matrix, $P_k^{KFn^*}$.)

It may also be useful to consider the effect of an approximation where some of the close off-diagonal terms outside of the top left hand block are non-zero. However we do not pursue this type of issue any further.

The simulation results presented in Figure 5.11 are obtained with the use of a 50th order predictor, and a reduced order Riccati equation obtained by assuming that only the top left hand block of the error covariance matrix is non-zero. The performance for the full 50th order Kalman filter is not shown in the figure, as the huge computational expense required to generate this value makes it inhibitive. In Figure 5.11, the smoothed output SNR is shown with the solid curve, while the Kalman filter prediction gain is presented with the dashed curve. Note that a Riccati equation order of zero corresponds to the standard linear predictor.

Figure 5.11 shows clearly that most of the performance gain of the Kalman filter is obtained via the use of only a very small number of significant (non-zero) values in the Riccati equation. Hence although the full order Kalman filtering is inhibitive for high order prediction, it is possible to obtain most of the performance benefit of Kalman filtering for only a fraction of the computational cost. For a 50th order predictor, as the Riccati equation complexity is order n^2, a 4th order non-zero block translates to less than 1% of the computation required by the full 50th order Kalman filter. From the graph, for 2nd to 4th orders, around 90% of the performance of the full Kalman filter is achieved.

Riccati Equation Order	50th Order Predictor		Reduced Order Predictor	
	Smoothed Output SNR (dB)	Prediction Gain (dB)	Smoothed Output SNR (dB)	Prediction Gain (dB)
1	9.73	6.38	6.39	3.04
2	10.67	6.72	7.44	3.20
3	10.80	6.75	8.24	3.88
4	10.86	6.76	8.78	4.43
6	10.92	6.78	9.55	5.20
8	11.00	6.80	9.79	5.36
10	11.06	6.81	10.00	5.53

Table 5.4: Reduced Order Approach compared to Reduced Order Predictor Approach

Table 5.4 presents the results of a comparison between the reduced order Riccati equation approach, and a standard reduced order approach, where the predictor order is also decreased. As the majority of the computation is concerned with the Riccati equation, both approaches involve similar overall computation requirements. (The computation is strictly not identical, due to the extra computation required for the higher order predictor coefficients.) However, the performance obtained via the 'hybrid' reduced order Riccati equation approach is significantly better than a standard reduced order Kalman filter approach (with reduced order prediction).

Remark 5.10 A very similar approach to that above could be applied to the coloured noise Kalman filter discussed in the previous section. However, it is likely that the additional correlation introduced by the coloured noise may imply the need for a slightly higher reduced order to obtain similar performance.

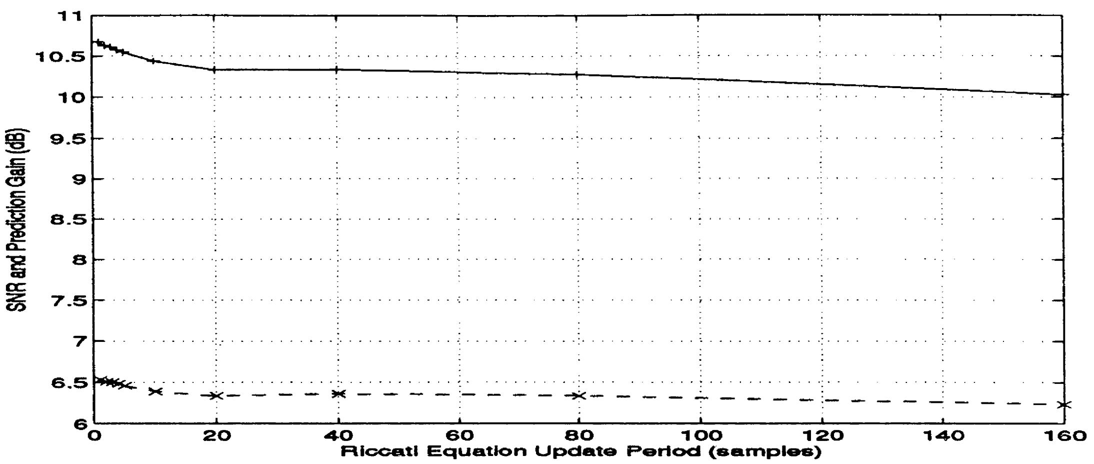

Figure 5.12: SNR (solid line) and Prediction Gain (dashed line) versus Riccati Equation Update Period

5.10 RDE Update Frequency and Steady State Solutions

The reduced order Riccati equation approach presented in the previous section has been observed to decrease significantly the computational requirements for the use of the Kalman filter. With a high order predictor (up to 50th order for speech coding), full order Kalman filtering is practically impossible. The use of a Riccati equation of say 4th order makes Kalman filtering possible, but the Riccati equation updated every sample still consumes a significant amount of computational resources.

Ignoring the problems associated with AR model updates, it is apparent that the inputs to the Kalman Filtering equations of the process and measurement noises do not change quickly. Both these parameters are in general obtained from variance estimators, implying only slow progression in values. Hence it may be reasonable to assume that only updating the Riccati equation once every few samples would lead to a relatively insignificant performance degradation. As the matrix calculations contribute significantly to the overall computation, this approach may lead to substantial complexity reduction for only a small decrease in performance.

Figure 5.12 presents the results of only periodically updating the Riccati equation for the reduced order approach, where a 50th order predictor is used, but only a 4th order Riccati equation. The smoothed output SNR is shown with the solid line, and the prediction gain is given with the dashed curve. The predictor coefficients were updated every 80 samples, or 10 ms.

From the graph in Figure 5.12, it is clear that some degradation in performance is incurred via the use of only periodic Riccati equation updates. However, only updating the error covariance once in an 80 sample frame results in a substantial computational saving over that of performing updates every sample. This must be considered in assessing the fact that about 0.4 dB SNR degradation is obtained in this case. Even smaller SNR degradations can be obtained for higher updates frequencies, and a significant computation saving can still be made over where updates are performed every sample.

Shown in Figure 5.12 are values for updates only every 160 samples. This translates to only one Riccati equation update every two predictor updates, and for the 4th order Riccati equation implies only a very small increase in computation over that for the standard linear predictor. However, it is interesting to note that even for this situation, a substantial increase in performance over the standard linear predictor is observed. Compare with the values on the graph an SNR for the linear predictor of 5.14 dB, and a prediction gain of 4.54 dB.

Remark 5.11 The curves displayed in Figure 5.12 have been obtained through simulations where synthetic white quantization noise has been injected at a level of 6 dB relative to the prediction difference signal. Previously this simulation approach was introduced as being applicable for representation of quantization noise in predictive coding systems. Of course, this is application dependent, and in the next chapter the Kalman filter is applied to the AC-ADPCM system introduced in Chapter 4. For this coding system, due to the fixed quantizer step size, the quantization noise can be assumed to be at a constant level over the entire speech signal. Hence, it might be expected that the behaviour of the general approaches studied here will be somewhat different to observations made in the next chapter.

As we often use the assumption that speech is piecewise stationary, it makes sense to consider the effect of a piecewise stationary solution to the Riccati equation 5.19. This approach would seem to be especially relevant for CELP applications. There are a significant number of approaches for obtaining steady-state solutions to Riccati Difference Equations, and it is beyond the scope of this thesis to discuss these techniques. The interested reader is advised to consult the numerous books on the subject, such as that edited by Bittanti, Laub, and Willems[18].

For the low order Riccati equation used with our approach introduced in Section 5.9, it might be feasible to implement a pseudo steady-state solution by iterating the

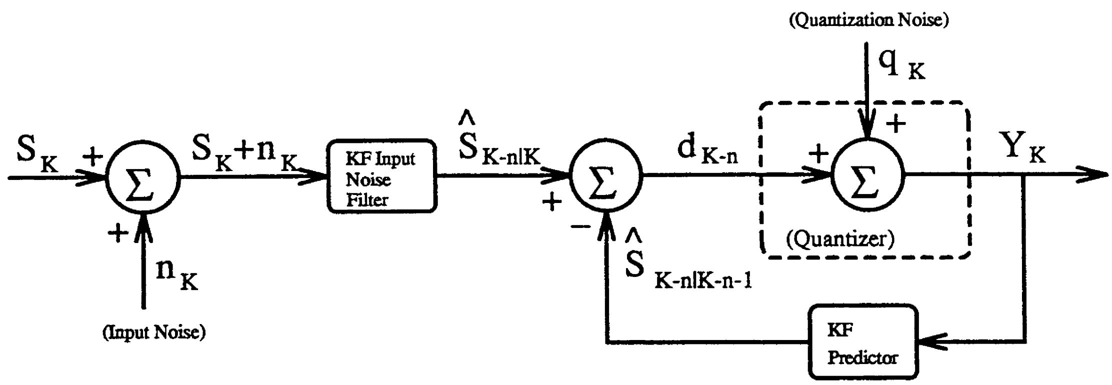

Figure 5.13: System Schematic for Input Noise and Prediction Kalman Filtering

Riccati equation a number of times until very little change is observed. This would be performed on a periodic basis, usually on a time-scale related to the predictor update period.

Within Appendix B it is shown that for the case of zero measurement noise, the Riccati equation reaches steady-state after a number of iterations equal to the order of the matrices involved. Unfortunately this has not been shown to be the case for non-zero measurement noise. However, a number of simulations displayed that the assumption of the steady state solution being reached after n iterations is reasonable. Hence, this simple approach can easily be used for obtaining a pseudo steady-state solution. However, no simulations are presented within this chapter for the pseudo steady state Riccati equation approach.

5.11 Coder Input Noise Filtering

The Kalman Filter has been seen to be a useful component of a speech coder, as well as for filtering of coder input noise. A simplified schematic of a speech coding system is shown in Figure 5.13, where Kalman filtering is used both for filtering of the input noise, n_k, and the quantization noise, q_k.

An important issue that we have tried to address within this chapter is the close relationship between the two problems of input noise filtering, and quantization noise filtering. As both problems involve Kalman filtering, and operate very similarly, it is valid to ask whether the Kalman filtering operations can be combined (at least partially) in order to reduce computational requirements.

For small levels of input noise, filtering both the input and quantization noise can be performed efficiently by the use of the Kalman filter within the prediction loop. As far as the Kalman filter is concerned, it is simply filtering the 'quantization' noise of $n_k + q_k$, where n_k is the (background) input noise and q_k is the quantization noise. However, the speech plus input noise is being quantized, and for high levels of input noise this implies wasted transmission bandwidth (wasted bits). Hence Kalman filtering for the two tasks should be separated to some extent in order to provide good performance.

The reduced order Kalman filter approach can be used for both filtering operations. However, even assuming the same system model for both filters, the measurement noise processes are different for the two different applications. It is common to assume that the background (input) noise parameters change relatively slowly, while for a predictive coding system, the quantization noise can be assumed to vary with the level of the prediction difference signal.

It would thus appear very difficult to exploit any similarity in the Kalman filters to obtain computational savings. However, there may be applications dependent considerations involved. In any case, the reduced order Kalman filtering approach presented in Section 5.9 is likely to be of significant use. Also note that the fact that all the performance gain of the Kalman smoother is obtained within only a few samples means that the Kalman input filtering does not incur much additional delay. (Of course the additional computation requirements for input noise filtering does not effect the decoder in any way.)

5.12 Downsampling, Embedded Coding, and Error Bursts

Recovery from transmission errors and robustness to situations such as downsampling and other embedded coding approaches is an important practical issue in signal coding systems used for several network and other applications. The common thread with these problems is that the encoder is not usually able to adjust in the same way as the decoder, as the information is not available to the encoder as to which section has been lost or discarded.

Embedded coding and error bursts can be handled in a logical fashion by the Kalman filter, by simply setting the measurement noise variance parameter to the appropriate value. Embedded coding approaches often involve received data at a coarse resolution, due to transmission errors or bit stealing from the fine detail information. During these periods the Kalman filter measurement error variance, σ_{nk}^2, is simply set to the larger value relevant to the coarse resolution measurement.

Error bursts, such as might occur with frame erasures, are also accommodated in a straight-forward fashion by the Kalman Filter. During the detected error burst period, where no received data is deemed to be useful, the measurement noise variance parameter is increased to account for this fact.

Remark 5.12 During an error burst we assume no measurement information is received at the signal decoder. Hence it would seem that a logical choice for the measurement error variance parameter, σ_{nk}^2, is infinity, $\sigma_{nk}^2 = \infty$. However, this is too conservative, as although no information is received, within coding systems there is almost always some *a priori* information available about the measurement. In the simplest case, the expected distribution of the measurement is known. Hence taking the unknown measurement to be zero, we know that the variance of the measurement noise will be equal to the variance of the measurement distribution, $\sigma_{nk}^2 = \sigma_{wk}^2$, where σ_{wk}^2 is the excitation noise variance. Using signal redundancy not exploited by the signal encoder can potentially further reduce the measurement noise variance, although this is beyond the scope of this chapter.

Remark 5.13 The Riccati equation 5.19 is often split into a time update equation and a measurement update equation (as in Appendix C). For the case of a missing measurement, the time update is often performed without the associated measurement update. This is equivalent to setting σ_{nk}^2 to infinity. Although this approach appears to be frequently used, it is non-optimal according to the considerations of the previous remark.

The handling of a periodic disturbance such as downsampling can be performed in a slightly different manner to that of embedded coding, and error bursts. The approach we use is to augment the state space system description, such that the system equations are only updated once for each (downsampled) time instant.

For downsampling by a factor of two, the state vector becomes

$$X_{2k} = \begin{pmatrix} S_{2k} \\ S_{2k-1} \\ S_{2k-2} \\ \vdots \\ S_{2k-N} \end{pmatrix}. \tag{5.49}$$

Note that there are now $N+1$ elements in the state vector, and observe the use of the time index k, with the $2k$ implying that for each time increment, the speech samples are shifted by two in the state vector.

For the system matrix, F, as before, we now have an augmented system

$$X_{2k} = F_2 X_{2(k-1)} + \bar{w}_{2k}, \tag{5.50}$$

where

$$F_2 = \begin{pmatrix} a_1^2 + a_2 & a_1a_2 + a_3 & \cdots & a_1a_{N-1} + a_N & a_1a_N & 0 \\ a_1 & a_2 & \cdots & a_{N-1} & a_N & 0 \\ 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 \end{pmatrix};$$

$$\bar{w}_{2k} = \begin{pmatrix} w_{2k} & a_1w_{2k-1} \\ 0 & w_{2k-1} \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{pmatrix}.$$

It is important to note that the above equations are simply a different representation of the system from that presented in equations 5.3 and 5.4. The actual system has not been altered. This may become clearer by representing F_2 in the form:

$$F_2 = \left(\begin{array}{ccccc|c} & & & & & 0 \\ & & F^2 & & & \vdots \\ & & & & & 0 \\ \hline 0 & \cdots & 0 & 1 & 0 & 0 \end{array}\right). \tag{5.51}$$

Hence the matrix F_2 is simply an augmented version of the squared system matrix, F, representing two time updates. As the all-pole system is simply a source model, this fact is important. Downsampling is a process that occurs in the coding and transmission process, and not in the source. Downsampling thus refers to the fact that we now have only one measurement for every two samples.

For a downsampled coding system, the simplest form of this measurement is simply a quantized representation of w_{2k}. Note that the downsampled system could equally well have been constructed by quantizing w_{2k-1}.

This form of downsampling is simple and is often used, but is not necessarily optimal. The basic requirement of downsampling is to provide one observation over a number of samples. A linear combination of a number of samples is one possible measurement – another is a 'complex' measurement, such as an average magnitude, and some slope. Similarities between 'downsampled' ADPCM approaches and CELP approaches should now be somewhat apparent. Some CELP approaches can certainly be viewed as complex measurements at a much lower frequency. We leave further discussion on this topic until Chapter 7.

5.13 Chapter Conclusion

Within this chapter, following the work in [44], the Kalman filter has been introduced as a logical extension of the standard linear predictor. The Kalman filter is seen to provide a performance improvement by implicit accommodation of information about measurement noise statistics.

This chapter has also attempted to show the connections between the three issues of input noise filtering, signal prediction, and smoothing of quantization noise to obtain better reconstructed signals. It has been seen that significant smoothing gain can be obtained for no increase in computational complexity over that of the Kalman predictor, and for only a small additional delay.

The problems of parameter estimation for the measurement noise variance and the predictor coefficients were also briefly discussed. In most simulations, performance trends were displayed via simulations at different noise levels, and with different parameter settings.

A novel approach for Kalman filter computation reduction has been introduced involving full order prediction, but smoothing to a few lags only. Via this approach it is possible to obtain over 90% of the performance improvement of the Kalman filter with as little as 1% of the computational requirements.

Further complexity reductions have been considered through the use of less frequent updates of the Riccati difference equation, and pseudo steady-state solutions.

The Kalman filter has also been discussed in terms of obtaining improved performance in various situations such as downsampling, and embedded coding.

In summary, the Kalman filter is observed to be an extremely valuable tool for speech enhancement and coding applications. Although the complexity of full Kalman filtering often makes it inhibitive, several mathematically sub-optimal techniques are able to be used to minimise the computation. It is then possible that significant performance gains can be achieved with minimal additional computation over that of standard linear prediction approaches.

One final point with respect to the use of Kalman filtering in general signal coding applications is related to its parallelization potential. The Kalman filter can consume a large amount of computation resources on a single processor machine, as used in research simulation, or standard DSP implementation. However, the Riccati equation is only loosely coupled to the other operations involved in signal coding, such as windowing, auto-correlation analysis, etc. This fact can be exploited for implementation on a multi-processor system, or for custom VLSI (Application Specific IC's – ASIC's). Another important consideration for VLSI implementation is the possibility for very low precision arithmetic. Within the Riccati equation, the ratio of the measurement noise to the process noise is significantly more important than the actual values. Hence scaling can be used to minimise the arithmetic precision required.

Chapter 6

AC-ADPCM with Kalman Filtering

6.1 Chapter Motivation

The previous two chapters, have discussed first a novel variable rate ADPCM coding scheme (using Arithmetic Coding), and then the use of Kalman filtering to smooth, and attempt to improve signals such as speech corrupted by measurement noise in the form of both input (background) and quantization noise.

The AC-ADPCM system introduced in Chapter 4 is conceptually an extremely simple system, and has been observed to provide good speech output quality in the bit rate range of 8 kbps to 16 kbps. With perceptual weighting and postfiltering the subjective performance of the AC-ADPCM system at an average rate of 12 kbps has been judged by informal listening tests to be comparable to that of LD-CELP.

Unfortunately, at the 12 kbps average rate the level of audible quantization noise within the AC-ADPCM system is noticeably higher than that of LD-CELP. An investigation into the application of the Kalman filter to the AC-ADPCM system is made within this chapter. It is found that the (reduced complexity) Kalman filter approach introduced in Chapter 5 can significantly improve the subjective performance of the AC-ADPCM system with only minimal additional computational requirements.

6.2 Kalman Filter Application

From Chapter 5, it has been seen that the Kalman filter can be used as a replacement for the standard linear predictor, and provides a smoothed output signal (with a small delay lag) at no additional cost over that of the Kalman filter predictor. Within this section we investigate the simple replacement of the standard linear predictor in the AC-ADPCM system with the Kalman filter predictor. No perceptual weighting or postfiltering is used at this stage.

Riccati Equation Order	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)	Entropy (bits/sample)
0 (LPC)	23.76	9.97	11.43	0.932
1	24.10	10.15	11.39	0.937
2	24.70	10.52	11.38	0.932
3	24.75	10.56	11.38	0.932
4	24.84	10.61	11.38	0.933
6	24.88	10.64	11.37	0.935

Table 6.1: Reduced Order Kalman Filter Approach for AC-ADPCM

Table 6.1 displays the effect of the reduced order Riccati equation assumption in conjunction with the AC-ADPCM system. In Chapter 5 it was noted that very little gain is achieved through the use of other than very small Riccati equation orders, and due to computation considerations, it is not worthwhile to consider larger Riccati equation orders than that shown in the table. (Note that the 0th order Riccati equation is simply the standard linear predictor.)

The values presented in Table 6.1 relate to an AC-ADPCM system with a 50th order predictor updated every 20 samples (2.5 ms). The reduced order Riccati equation update is run every sample, and the average coder output bit rate is slightly under 8 kbps (as seen from the entropy values in the table). As with all previous simulations, the speech input used to obtain the results in the table is 22.5 seconds consisting of four male and four female sentences.

From the table we observe little performance gain for a Riccati equation order in excess of 2, which is in agreement with the results of the previous chapter. For reasons of conservatism, at this stage we choose to concentrate on the 4th order Riccati equation, and use this for the Kalman filtering results in the remainder of the chapter. No significant subjective performance difference was observed by informal listening tests with 4th order or 6th order Riccati equations. The same can probably be said of the 2nd order and 4th order Riccati equations, but the 4th order is selected as this is further away from the knee of the curve shown in Figure 5.11.

Table 6.2 compares the use of the 4th order Riccati equation Kalman filter in AC-ADPCM to that of the standard linear predictor (again a 50th order predictor is used,

Approximate Average Bit Rate	Standard Linear Predictor			Kalman Filter		
	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)
16 kbps	35.60	17.82	12.03	36.05	18.26	12.01
12 kbps	29.78	13.68	11.89	30.52	14.23	11.84
8 kbps	23.76	9.97	11.43	24.84	10.61	11.38
4 kbps	14.43	4.94	9.18	15.68	5.56	9.17

Table 6.2: AC-ADPCM Performance with Kalman Filtering

backwards adapted every 20 samples). From the table it is clear that although there are some improvements in SNR values with the use of Kalman filtering, these are relatively minor, especially for the higher rate situations. However, the subjective performance improvement observed is substantially higher than that indicated by the objective measures shown in the table. In fact, the 12 kbps average rate system with Kalman filtering is observed to be comparable in output speech quality to that of the 16 kbps average rate without Kalman filtering.

The performance at the 4 kbps average rate is noted to be fairly poor, but the use of the Kalman filter does provide some important perceptual improvement. At all other rates, including the 16 kbps average rate where only minimal objective improvement is observed, significant subjective performance improvements are observed through the use of the Kalman filter.

6.3 Further Computation Trade-Offs

A substantial barrier to wider use of the Kalman filter is the large computational cost involved due to the matrix multiplications required in the Riccati equation. For practical applications it is often important to exploit as much symmetry or similarity as we can from the computation process to reduce the cost of the calculations. This often comes for the price of a mathematically non-optimal solution, but the engineering cost-performance trade-off is the key design issue.

Chapter 5 introduced a reduced order Riccati equation approach where it was recognised that the top left hand block of the error covariance matrix was the most significant. This was retained, and the rest of the matrix assumed to be zero. Further computational savings are made by recognising that certain elements of the system matrix, F, are always zero. Hence when we multiply by F, we do not waste time multiplying by zero elements.

The above example of avoiding multiplications by zero deals with no performance loss. A similar example where there is an associated performance loss is to round a small value to zero, which is effectively what has been done with the reduced order Riccati equation approach. Another possibility is to group a number of very similar elements together, and assign them one value so that less arithmetic is required.

Issues such as the above are extremely important for engineering development work, and are not really of concern for the research contained in this thesis. However, what is important is to show that the use of Kalman filtering with the AC-ADPCM scheme is feasible as far as computation is concerned. The reduced order Riccati equation approach is an extremely important factor in reducing the Kalman filter computational requirements, but another important component is the exploitation of the slow variation in Riccati equation parameters to reduce computational requirements. Again, this topic has been discussed in general terms in Chapter 5, and here we wish to examine its specific application to AC-ADPCM.

Riccati Equation Update Period	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)	Entropy (bits/sample)
1	24.84	10.61	11.38	0.933
2	24.77	10.57	11.36	0.934
3	24.66	10.52	11.35	0.936
4	24.62	10.50	11.34	0.935
5	24.56	10.48	11.32	0.936
6	24.48	10.45	11.33	0.937
8	24.40	10.40	11.32	0.938
10	24.34	10.38	11.30	0.939
20	23.59	10.20	11.24	0.939
40	23.45	10.12	11.24	0.940
∞ (LPC)	23.76	9.97	11.43	0.932

Table 6.3: Riccati Equation Update Period in AC-ADPCM

Table 6.3 displays the effect of computation reduction via mathematically sub-optimal Kalman filtering where the Riccati equation is updated only periodically. For the results shown in the table a 50th order predictor was once again used, with updates every 20 samples (2.5 ms). The Riccati equation used was 4th order, and the average coder output bit rate is just under 8 kbps. Note that a Riccati equation update period of one corresponds to the 'full reduced order' Kalman filter with Riccati equation updates every sample, and an infinite update period is equivalent to the standard linear predictor (as the Kalman gain vector is initialized to $[1 \;\; 0 \;\; \ldots \;\; 0]^T$).

From the table it can be seen that a Riccati equation update period of five samples gives only a small degradation in objective performance measures over the Kalman filter with updates every sample. The minimal reduction in objective measures appears to be echoed by minimal changes in subjective output quality.

Note that for a Riccati equation update period of 20 samples (equal to predictor update period), apart from segmental SNR all objective measures indicate poorer performance to that of the standard linear predictor. However, this is not the case, and a large proportion of the subjective improvement of the full Kalman filter is maintained. The performance of the system with a 40 sample Riccati equation update period is also better than the standard linear predictor, although some artifacts appear in the output speech.

To reduce computational complexity, but not risk significant performance degradation, we conservatively choose to concentrate on performing the Riccati equation updates every five samples. We have already noted above the minimal performance degradation obtained, and remark that the computation saving is significant, as we only require 20% of the 'full update' Riccati equation computation.

Approximate Average Bit Rate	Standard Linear Predictor			Kalman Filter		
	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)
16 kbps	35.60	17.82	12.03	35.48	18.08	12.01
12 kbps	29.78	13.68	11.89	30.05	14.04	11.84
8 kbps	23.76	9.97	11.43	24.56	10.48	11.32
4 kbps	14.43	4.94	9.18	15.50	5.47	9.09

Table 6.4: AC-ADPCM Performance with Kalman Filtering and Five Sample Riccati Equation Update Period

Table 6.4 shows the comparison of the performance of the standard linear predictor to that of the Kalman filter with 4th order Riccati equation and five sample update period. Very little subjective performance degradation is observed compared to the results for the full update Riccati equation, presented in Table 6.2 and discussed above.

6.4 A 'Practical' AC-ADPCM System

The reduction in computational complexity afforded by the use of the reduced order Riccati equation with periodic updates means that the Kalman filter can be used in a number of applications to provide performance improvements with a manageable level of computational cost.

With the addition of the 'reduced complexity' Kalman filter, the AC-ADPCM system provides perceptually improved performance, making it worth consideration for a number of applications. However, the above simulations still used a 50th order predictor updated every 20 samples. The flexibility of the AC-ADPCM system observed in Chapter 4 can be exploited to reduce the computational complexity significantly. We consider the performance of the AC-ADPCM system with a 15th order predictor (chosen somewhat arbitrarily from the results presented in Figure 4.13), updated every 80 samples. As above, we use a 4th order Riccati equation, updated once every 5 samples.

Approximate Average Bit Rate	Standard Linear Predictor			Kalman Filter		
	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)
16 kbps	35.60	17.82	10.94	35.47	18.08	10.92
12 kbps	29.79	13.70	10.81	30.04	14.03	10.76
8 kbps	23.70	9.93	10.38	24.55	10.48	10.31
4 kbps	14.34	4.92	8.39	15.54	5.51	8.28

Table 6.5: AC-ADPCM Performance for Reduced Order Predictor

Note that the results presented in Table 6.5 are almost identical to those presented in Table 6.4, with the exception of lower prediction gain values. The similar SNR performance is a consequence of the same AC-ADPCM quantizer step size values being used in both cases. The difference in prediction gains is evidence of the lower order predictor used for the simulations in Table 6.5. Naturally the average bit rates of Table 6.5 are also somewhat larger than those presented in Table 6.4. (The actual bit rates for Table 6.5 are very close to the nominal values shown in the table, while those in Table 6.4 are smaller by approximately 5%.)

Remark 6.1 For reasons of comparison and simplicity we have used an identical windowing structure to LD-CELP for the backwards adaptive auto-correlation coefficient calculation. The window strategy should strictly be tuned for the 80 sample update period, and doing so may provide a significant performance benefit. However, this has not been performed as part of the work carried out for this thesis.

6.5 Perceptual Weighting

The Kalman filter has been observed to reduce significantly the effects of quantization noise through exploiting smoothing. Perceptual weighting shapes the quantization noise spectrum to give improved subjective performance by exploiting the perceptual masking properties of the signal spectral peaks. These two approaches can be combined in a speech coding system to give further performance improvements.

The Kalman filter is optimal for the assumption of gaussian white measurement (quantization) noise, and for arbitrary distributions of white noise, the Kalman filter is the best linear estimator[4]. A section in Chapter 4 discussed the validity of the white noise assumption for the coarse quantization in the AC-ADPCM system, and it appears that the noise is substantially white in nature. The Kalman filter has been observed to give some objective SNR improvement for the AC-ADPCM application, and a significant level of subjective improvement.

For the perceptually weighted AC-ADPCM system, the white quantization noise assumption is no longer true. Actually the quantization noise is still white, but it is added to a filtered speech signal (as a perceptually weighted difference signal is quantized) and not the speech difference signal directly. Hence the Kalman filter should really take account of this fact. However, a first order approximation would be to assume that the white noise assumption Kalman filter will still be valid for the perceptually weighted system. Simulations were performed using this assumption by directly applying the Kalman filter to the perceptually weighted AC-ADPCM system.

Approximate Average Bit Rate	Standard Linear Predictor			Kalman Filter		
	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)
16 kbps	34.98	17.37	10.94	34.95	17.61	10.92
12 kbps	29.19	13.31	10.80	29.40	13.62	10.77
8 kbps	23.17	9.63	10.38	23.82	10.06	10.25
4 kbps	13.70	4.62	8.27	14.82	5.19	8.11

Table 6.6: AC-ADPCM Performance with Perceptual Weighting and Kalman Filtering

Remark 6.2 The results in Table 6.6 correspond to the use of a 15th order predictor backwards adapted every 80 samples, perceptual weighting updated every 80 samples by auto-correlation analysis on the clean speech, and the Kalman filter with a 4th order Riccati equation updated every 5 samples.

From Table 6.6 it appears that very little gain is achieved by the use of the Kalman filter over the standard linear predictor where perceptual weighting is used. However, the subjective performance does not reflect this fact, with a significant improvement observed through the use of the Kalman filter.

For the 12 kbps and 8 kbps average rates, the performance of the standard linear predictor with perceptual weighting has been observed to be better than that without perceptual weighting. The additional performance improvement through the use of Kalman filtering is quite substantial. The performance improvement with the use of Kalman filtering and perceptual weighting is also noted to be significantly better than that of Kalman filtering only.

With perceptual weighting and Kalman filtering, the performance at the nominal 16 kbps rate can be rated as excellent, while that at the 12 kbps rate can be judged (by informal listening tests) to be roughly equivalent to LD-CELP. The performance at the 8 kbps average rate is also very good.

Work by Sinha[163] considers the incorporation of perceptual weighting into the Kalman filter. This is done by setting the Kalman filter measurement vector, H, (defined in Chapter 5) to the truncated impulse response of the perceptual weighting filter. For adaptive perceptual weighting (as is common), the measurement vector should strictly be denoted H_k, with the subscript k referring to the kth sample. It is thus seen that the perceptually weighted Kalman filter can be obtained with no change in basic structure to the Kalman filtering equations (equations 5.16 to 5.19) in Chapter 5. However, the fact that the measurement vector, H_k, has more than one non-zero element implies a slight increase in computation required by the Riccati equation (equation 5.19) and the Kalman gain calculation (equation 5.18).

For perceptual weighting as in LD-CELP, the ARMA filter structure is of the form:

$$W(z^{-1}) = \frac{1 - A(z^{-1}/\gamma_1)}{1 - A(z^{-1}/\gamma_2)}, \tag{6.1}$$

where $A(z^{-1})$ is a polynomial in the backward shift operator z^{-1}, corresponding to the all-pole predictor used for speech, and the bandwidth expansion factors, γ_1 and γ_2, are specified to be 0.9 and 0.6 respectively.

In order to implement the perceptually weighted Kalman filtering approach of Sinha, we require MA (Moving Average) filter coefficients for the measurement vector H_k. These are obtained from the truncated impulse response of $W(z^{-1})$. For $W(z^{-1})$ as given above, we can calculate the impulse response as:

$$\begin{aligned} W(z^{-1}) &= h_0 + h_1 z^{-1} + h_2 z^{-2} + h_3 z^{-3} + \ldots, \qquad (6.2) \\ h_0 &= 1, \\ h_1 &= -0.3a_1, \end{aligned}$$

$$\begin{aligned} h_2 &= -0.18a_1^2 - 0.45a_2, \\ h_3 &= -0.108a_1^3 - 0.378a_1a_2 - 0.513a_2, \\ h_4 &= \ldots, \end{aligned}$$

where $A(z^{-1}) = \sum_{i=1}^{N} a_i z^{-i}$. Hence, for the 4th order Riccati equation, the measurement vector becomes

$$H_k = [h_0 \;\; h_1 \;\; h_2 \;\; h_3]\,. \tag{6.3}$$

Noting that the non-zero values in the measurement vector, H_k, result in some increase in computation, we can extend the reduced complexity considerations of the previous chapter to taking only a small number of non-zero H_k values. As we have already noted significant Kalman filter performance gains without the use of the perceptually weighted Kalman filter, it is perhaps logical not to expect an additional gain of the same order using the truncated impulse response for H_k. Hence the use of only an additional one or two non-zero H_k terms may seem appropriate. However, it was found that significant additional subjective performance gains are actually obtained through the use of the above measurement vector, H_k, and hence the additional computation does appear to be warranted.

Approximate Average Bit Rate	Standard Linear Predictor			Kalman Filter		
	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)
16 kbps	34.98	17.37	10.94	34.58	17.40	10.93
12 kbps	29.19	13.31	10.80	29.04	13.43	10.79
8 kbps	23.17	9.63	10.38	23.60	9.96	10.33
4 kbps	13.70	4.62	8.27	14.74	5.14	8.16

Table 6.7: AC-ADPCM with Truncated Impulse Response Perceptual Weighted Kalman Filtering

Table 6.7 presents SNR values for the truncated impulse response measurement vector approach to perceptually weighted Kalman filtering. Coder parameters identical to those for the results in Table 6.6 were again used. Note that as we are dealing with maximisation of perceptually weighted performance criteria, the SNR measures presented in the table are not a reliable indication of relative subjective quality. In fact, the SNR results in Table 6.7 would indicate inferior Kalman filter performance to the values in Table 6.6, but this is certainly not the case. Hence some informal listening tests have been made.

Informal listening tests indicate that there is a significant performance gain in using the truncated impulse response measurement vector for the Kalman filter. Using the simple measurement vector, $H = [1 \;\; 0 \;\; \ldots \;\; 0]$, obtains an important subjective performance improvement over the standard linear predictor, and the additional improvement with the measurement vector from equation 6.3 is of a similar magnitude.

It is important to note that by using the measurement vector of equation 6.3, we are simply adjusting the Kalman filter to match the actual system with perceptual weighting. From Chapter 4, it was observed that a perceptual weighting approach similar to that used in LD-CELP can be obtained via a Noise Feedback Coding type of approach within the AC-ADPCM system. Within Chapter 4 it was seen that the coder must choose the excitation (or measurement) as

$$\hat{e}_k = \mathbf{Q}\left[(S_k - \hat{S}^{LPC}_{k|k-1}) + h_1(S_{k-1} - \hat{S}^{LPC}_{k-1|k-1}) + h_2(S_{k-2} - \hat{S}^{LPC}_{k-2|k-2}) + \ldots\right], \quad (6.4)$$

which for Kalman filtering, would incorporate smoothing into estimates of past samples to obtain

$$\hat{e}_k = \mathbf{Q}\left[(S_k - \hat{S}^{KF}_{k|k-1}) + h_1(S_{k-1} - \hat{S}^{KF}_{k-1|k-1}) + h_2(S_{k-2} - \hat{S}^{KF}_{k-2|k-1}) + \ldots\right]. \quad (6.5)$$

Truncating this to N samples, and regrouping terms, we can express the excitation as

$$\begin{aligned} \hat{e}_k \;&=\; \mathbf{Q}\Big[(S_k + h_1 S_{k-1} + \ldots + h_{N-1} S_{k-N+1}) \\ &\quad -(\hat{S}^{KF}_{k|k-1} + h_1 \hat{S}^{KF}_{k-1|k-1} + \ldots + h_{N-1}\hat{S}^{KF}_{k-N+1|k-1})\Big]. \end{aligned} \quad (6.6)$$

With the definitions in Chapter 5 for the state vector X_k and the *a priori* Kalman filter state estimate vector $\hat{X}_{k|k-1}$,

$$X_k = \begin{pmatrix} S_k \\ S_{k-1} \\ \vdots \\ S_{k-N+1} \end{pmatrix}, \quad (6.7)$$

$$\hat{X}_{k|k-1} = \begin{pmatrix} \hat{S}^{KF}_{k|k-1} \\ \hat{S}^{KF}_{k-1|k-1} \\ \vdots \\ \hat{S}^{KF}_{k-N+1|k-1} \end{pmatrix}, \quad (6.8)$$

the excitation can be simply expressed as

$$\hat{e}_k = \mathbf{Q}\left[H_k X_k - H_k \hat{X}_{k|k-1}\right]. \quad (6.9)$$

From the above equation it should be clear that the measurement we have in the perceptually weighted system is exactly that given by the truncated impulse response measurement vector (within the truncation error). Hence the approach of Sinha[163] is ensuring that the Kalman filter equations match the actual system as closely as possible.

6.6 Postfiltering

Adaptive Postfiltering was also noted as providing significant performance benefits for the AC-ADPCM system in Chapter 4. A logical progression is to apply postfiltering to the AC-ADPCM system in addition to perceptual weighting and Kalman filtering. Table 6.8 presents the objective measures obtained when postfiltering is applied to the AC-ADPCM system from Table 6.7. The predictor is 15th order, updated every 80 samples. The Riccati equation is 4th order updated every 5 samples, and the perceptual weighting filter is a 10th order ARMA structure, with coefficients obtained from autocorrelation analysis on clean speech every 80 samples. The postfilter is identical to that used in LD-CELP, accommodating for the fact that the predictor updates only occur every 80 samples.

Approximate Average Bit Rate	Standard Linear Predictor			Kalman Filter		
	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)
16 kbps	18.48	10.69	10.94	18.47	10.72	10.93
12 kbps	18.37	9.39	10.80	18.33	9.42	10.79
8 kbps	17.86	7.92	10.38	17.84	7.99	10.33
4 kbps	13.45	4.61	8.27	13.94	4.89	8.16

Table 6.8: Objective Measure for Postfiltering and Kalman Filtering in AC-ADPCM

As above, the objective measures displayed in Table 6.8 show very little, if any, performance improvement for the Kalman filter over that of the standard linear predictor. Also, a large degradation in objective measures is observed between those values in Table 6.7, where no postfiltering is used, and the values in Table 6.8. However, in terms of subjective performance, there is a significant advantage with the use of the Kalman filter, and the output speech quality with the use of postfiltering is a substantial improvement over the output speech quality without postfiltering.

We claim that the performance of the AC-ADPCM system in terms of output speech quality at the 12 kbps average bit rate is equivalent to that of 16 kbps CCITT Recommendation G.728 LD-CELP. We make this claim on the basis of informal listening tests on standard test sentences. The primary test file consists of just over 22.5 seconds of speech, including four male sentences and four female sentences, with slight pauses between. As discussed in Chapter 4, we believe that this comparison is 'fair' as although Voice Activity Detection (VAD) and Discontinuous Transmission (DTX) could be used with LD-CELP, we would only expect these techniques to give a substantial reduction in average bit rate during long periods of silence.

Remark 6.3 Due to the absence of a codebook search, the much lower predictor order, and the much longer update periods in the AC-ADPCM system, the computational complexity required is significantly less than that of LD-CELP. This is an important consideration in assessing the overall utility of the AC-ADPCM approach.

The subjective quality of the 8 kbps average rate coder output from the simulations performed for Table 6.8 is judged to be inferior to that of LD-CELP, but is very good considering the bit rate and complexity. The subjective performance at the 16 kbps average rate is noticeably better than LD-CELP, while the output at 4 kbps is still quite poor.

Approximate Average Bit Rate	Standard Linear Predictor			Kalman Filter		
	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)	SNR (dB)	Segmental SNR (dB)	Prediction Gain (dB)
14.5 kbps	19.12	10.39	12.03	19.11	10.27	12.01
10.4 kbps	18.73	9.02	11.88	18.73	8.92	11.84
8.0 kbps	—	—	—	17.96	7.85	11.54
6.8 kbps	17.19	7.16	11.34	17.16	7.08	11.21
3.0 kbps	11.04	3.63	8.20	11.06	3.56	8.01

Table 6.9: Objective Measures for the High Complexity AC-ADPCM Version

Table 6.9 presents objective measures for the AC-ADPCM system with predictor order and update periods as in LD-CELP. Hence the predictor is 50th order, updated every 20 samples, the postfilter is updated every 20 samples, and the perceptual weighting filter is a 10th order ARMA structure also updated every 20 samples. The Riccati equation is 4th order and updated every 5 samples. As no codebook search is required in the AC-ADPCM system, the overall complexity is no more than that of LD-CELP, even with the Kalman filtering computation.

As noted above, the objective measures must be used with caution. Significant subjective performance is observed with the use of the Kalman filter, although this is not indicated by the tabulated figures. The simulations for Table 6.9 were performed with the same quantizer step size values as used for above simulations (Table 6.8). The better prediction (evidenced by the higher prediction gain values) afforded by the 50th order predictor updated every 20 samples, implies that with the same quantizer step sizes, the output bit rates are somewhat less than the nominal values shown in the previous table.

Judging by our informal listening tests, we rate the subjective performance at the 10.4 kbps average rate to be slightly better than that of LD-CELP, and the performance at the 6.8 kbps rate to be noticeably worse. In Table 6.9 the results of an additional simulation are shown where an average bit rate of very close to 8 kbps is obtained. We rate the subjective performance of the speech output at this 8 kbps average rate to be roughly equivalent to that of 16 kbps LD-CELP. Careful listening reveals that the quantization noise characteristics of the 8 kbps AC-ADPCM scheme and 16 kbps LD-CELP vary slightly, and hence an informal subjective comparison is not easy. However, with a reasonable level of accuracy we believe the judgement of rough equivalence is fair.

6.7 Bark Spectral Distortion

The objective measures presented in Tables 6.6 to 6.9 display very little, if any, performance improvement with the use of the Kalman filter. However, due to prior Kalman filtering discussion in this chapter and in Chapter 5, there should be little doubt that significant subjective performance improvements are actually obtained.

SNR and segmental SNR values have been tabulated above, but noted as being of variable utility. Without considerable expense of formal subjective testing, informal subjective comparisons would appear the only significant option. Unfortunately these informal tests are also often quite unreliable. Within this section we consider the use of an additional objective measure that is intended to provide a better correlation with subjective quality than the SNR measures used previously.

Within the previous two sections we made some claims on the subjective performance of the AC-ADPCM system (with Kalman filtering) compared to that of LD-CELP. Based on our informal listening tests we claim that the performance of AC-ADPCM is roughly equivalent to that of LD-CELP for an average rate of as low as 8 kbps (high computation version of AC-ADPCM). For speech coding this claim is quite significant, and the question must be asked as to whether we are biasing our subjective comparisons. Of course we are not biasing the comparisons intentionally[1], but some favouritism could still be introduced on a subconscious level.

Fortunately some more advanced objective quality measures are available than simple SNR measures. The paper by Wang, Sekey, and Gersho[182] presents an objective measure of Bark Spectral Distortion (BSD), which is reported to obtain strong correlation with Mean Opinion Score (MOS) measures. The BSD measure presented in [182] involves a number of techniques to account for knowledge about human auditory perception. Hence to obtain the measurement is significantly more computationally expensive than for SNR values, but we might expect a greater degree of utility from BSD.

Approximate Average Bit Rate	Bark Spectral Distortion		
	No Postfiltering	No Postfiltering or Perceptual Weighting	Postfiltering and Perceptual Weighting
14.5 kbps	0.0011	0.0015	0.0036
12.0 kbps	0.0019	0.0010	0.0044
10.4 kbps	0.0026	0.0015	0.0051
8.0 kbps	0.0047	0.0027	0.0073
6.8 kbps	0.0070	0.0040	0.0096
LD-CELP	0.0050	0.0049	0.0071

Table 6.10: Bark Spectral Distortion (BSD) Measures for (High Complexity) AC-ADPCM

Table 6.10 presents BSD measures for AC-ADPCM[2] From the table it is clear that the BSD measure has limitations. In almost all cases, the performance without postfiltering or perceptual weighting is observed to be better than where perceptual weighting has been used, which in turn is better than where both perceptual weighting and postfiltering are used. (Better performance is indicated by lower BSD figures.) We know that this is not the case, and that perceptual weighting and postfiltering both give significant subjective performance improvements.

For the BSD measures where postfiltering is used, and where postfiltering is not used but perceptual weighting is, we observe monotonic improvement in BSD as the average bit rate of the AC-ADPCM system is increased. We further observe roughly equivalent BSD values for the 8 kbps average rate AC-ADPCM system and LD-CELP. This is in direct agreement with the results of our informal listening tests.

[1] As far as the reader is concerned, the used-car salesman scenario is a possibility, or even an expectation!

[2] Many thanks to Youngkyu Choi of the CSIRO Division of Radiophysics for the implementation of the BSD measure from [182].

Approximate Average Bit Rate	Bark Spectral Distortion	
	Without Postfiltering	With Postfiltering
15.7 kbps	0.0007	0.0034
11.5 kbps	0.0018	0.0045
9.5 kbps	0.0029	0.0056
8.0 kbps	0.0045	0.0069
LD-CELP	0.0050	0.0071

Table 6.11: Bark Spectral Distortion (BSD) Measures for (Low Complexity) AC-ADPCM

Table 6.11 presents results for the AC-ADPCM system with lower order prediction and larger update periods. (Predictor order 15, with updates every 80 samples.) Here again the BSD measure is observed to produce some strange results. Note that the BSD measures for the 8 kbps average rate are lower than those given in Table 6.10, although a significant subjective performance degradation is observed.

A further remark on the Bark Spectral Distortion measure is that it also indicates slightly better performance with the use of the standard linear predictor, rather than the Kalman filter. Hence the reliability of the measure is observed to be suspect, and although the results in Table 6.10 agree with our judgements of subjective quality compared to LD-CELP, care must be taken with the BSD measure.

Remark 6.4 Note that most of the problems noted above with the BSD measure are also problems for other measures such as SNR. (Observe the large SNR decrease between the values in Table 6.7 without postfiltering, and those in Table 6.8 where adaptive postfiltering has been used.) We have performed no analysis as to the relative usefulness of BSD with respect to SNR, and note that both measures (and all objective measures) must be used with caution in speech coding.

Speech Coder Variant:	Without Postfiltering			With Postfiltering		
	BSD	SNR (dB)	SegSNR (dB)	BSD	SNR (dB)	SegSNR (dB)
11.5 kbps Low Complexity	0.0018	29.04	13.43	0.0045	18.33	9.42
8.0 kbps High Complexity	0.0047	23.10	9.93	0.0073	17.96	7.85
16 kbps LD-CELP	0.0050	16.16	18.09	0.0071	15.00	15.67

Table 6.12: Objective Measure Comparisons for Equivalent Subjective Quality Systems

Table 6.12 summarizes the objective performance measures for the both the low and high complexity AC-ADPCM versions. We claim that both the AC-ADPCM versions shown in the table display equivalent subjective performance to that of LD-CELP. We have a reasonable level of confidence is this claim for the high complexity version of AC-ADPCM, as the strong similarity of coder components has allowed us to perform fairly accurate comparisons. For the high complexity AC-ADPCM version we have noted that the performance of the 10.4 kbps average rate system is subjectively better than LD-CELP, while that of the 6.8 kbps average rate is subjectively worse. For the low complexity AC-ADPCM version, our subjective comparisons appear a little less accurate, but we believe that claiming the performance at the 11.5 kbps average rate is equivalent to LD-CELP is conservative.

From the table we observe that in terms of both BSD and SNR, the AC-ADPCM versions obtain equivalent, or significantly better performance than LD-CELP. (Note that higher SNR values indicate improved performance while lower BSD values are associated with the same.) In terms of segmental SNR the AC-ADPCM system is noted as achieving significantly worse performance than LD-CELP. It is important to note that LD-CELP and AC-ADPCM are quite different coding systems, and in some sense AC-ADPCM can be viewed as maximising SNR, while the fixed bit rate of LD-CELP implies that it will still retain a moderately high SNR value in silence periods. The use of a segmental SNR measure that incorporates a threshold for periods of silence would be another objective measure to consider. However, this has not been tabulated above. Instead, a plot of segmental SNR versus time is shown in Figure 6.1.

From Figure 6.1 it is clear that the AC-ADPCM output has very high segmental SNR during high amplitude period of voiced speech. During unvoiced speech (lower amplitude) and silence, the segmental SNR is considerably lower. The segmental SNR profile shown in Figure 6.1 is a consequence of the constant step size quantizer used in AC-ADPCM. We do not suggest that this SNR profile is optimal in any way. In fact, we have previously indicated the rapid degradation of unvoiced speech at lower bit rates to be cause for consideration of variable quantizer step sizes to track voiced/unvoiced changes. Note that the segmental SNR values shown in the figure correspond to the AC-ADPCM system at an average rate of 8 kbps which we claim to be subjectively equivalent to LD-CELP in output speech quality.

In summary, the real test of output speech quality is in terms of subjective performance. The objective measures such as those displayed in Table 6.12 can be used to obtain some important indications, but should be used with caution. We indeed claim that the three coder variants shown in the table all achieve an equivalent level of subjective performance.

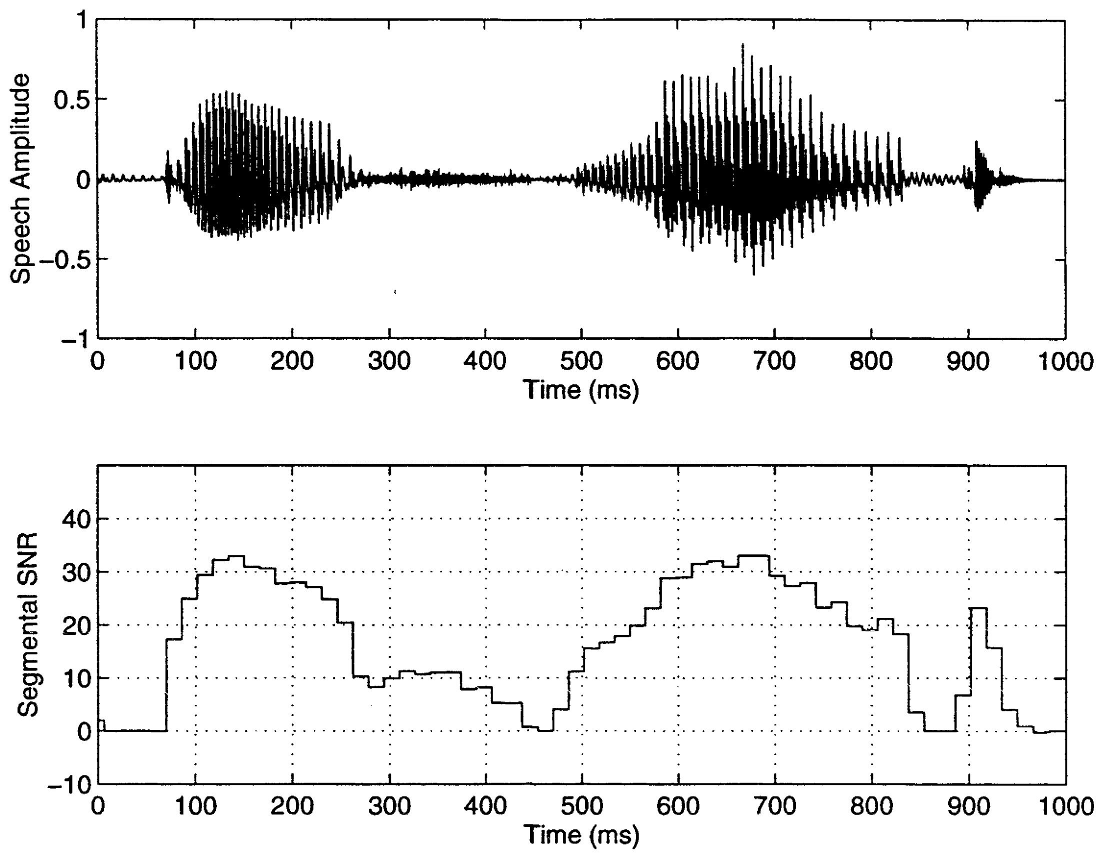

Figure 6.1: One Second Speech Input Segment and AC-ADPCM Output Segmental SNR (for 8 kbps Average Rate)

6.8 Output Bit Rate Profile

Throughout Chapter 4 and much of this chapter we have introduced and discussed a variable bit rate ADPCM system using Arithmetic Coding. Potential applications for such a system are discussed in Chapter 10, but from the above performance comparisons it is clear that on the basis of performance alone, the scheme should not have any difficulty finding applications. However, as far as practical applications are concerned, there are a number of factors that are important in addition to speech output quality.

Computational complexity is often a critical issue, but for the AC-ADPCM system we are able to obtain quality equivalent to that of LD-CELP at an average rate of 12 kbps for a system with a 15th order predictor, and with parameter updates only once every 80 samples (10 ms).

Robustness to bit transmission errors is obviously an important consideration for the AC-ADPCM system, and this will be discussed further in Chapter 9. An important issue that has not yet been discussed is the topic of the output bit rate profile. Knowing how the bit rate varies with time (with input signal) is important for two main reasons. The first is to assist with understanding how the AC-ADPCM system will perform when faced with input with different activity levels, and the second is determine important issues when considering application for communication purposes. Here the variable rate output is usually fed into packets for transmission over a multiple access network. Important considerations are the peak to average rates observed.

The first observation of the bit rate profile we make is that it exhibits a tremendous variation from sample to sample. Isolated peaks of samples requiring 30 bits for transmission occur infrequently. These high peaks are an artifact of the Laplacian distribution approach to calculating symbol probabilities giving very little codespace to symbols far from the mean. With a tabular approach to the Arithmetic Coding, even improbable symbols can be assigned probabilities large enough so the peaks are significantly reduced without detrimentally affecting the average bit rate.

As the large sample to sample variation in (instantaneous) bit rate is not of great significance, we consider a short term average bit rate. Figure 6.2 displays one second of speech input together with the 10 ms average output bit rate from the AC-ADPCM coder. The dashed line shown on the plot is the average bit rate for the second of speech shown (1.7 bits/sample, 13.6 kbps), and the long term average bit rate over the entire 22.5 seconds of input speech is 1 bit/sample, or 8 kbps.

From Figure 6.2 it is clear that even with a relatively low overall average of 8 kbps, shorter term bit rate averages for the AC-ADPCM system can easily reach as high as 32 kbps (4 bits/sample). Hence, any practical use of AC-ADPCM for communication purposes is likely to require a channel with large variations in possible data rates, or incur large buffering delays.

If the use of techniques such as delayed decision coding and pitch prediction are able to improve performance to the extent of providing the current performance at a long term average rate of closer to 4 kbps, the large variation in output data rate may not be as significant (less than 16 kbps maximum rate rather than 32 kbps).

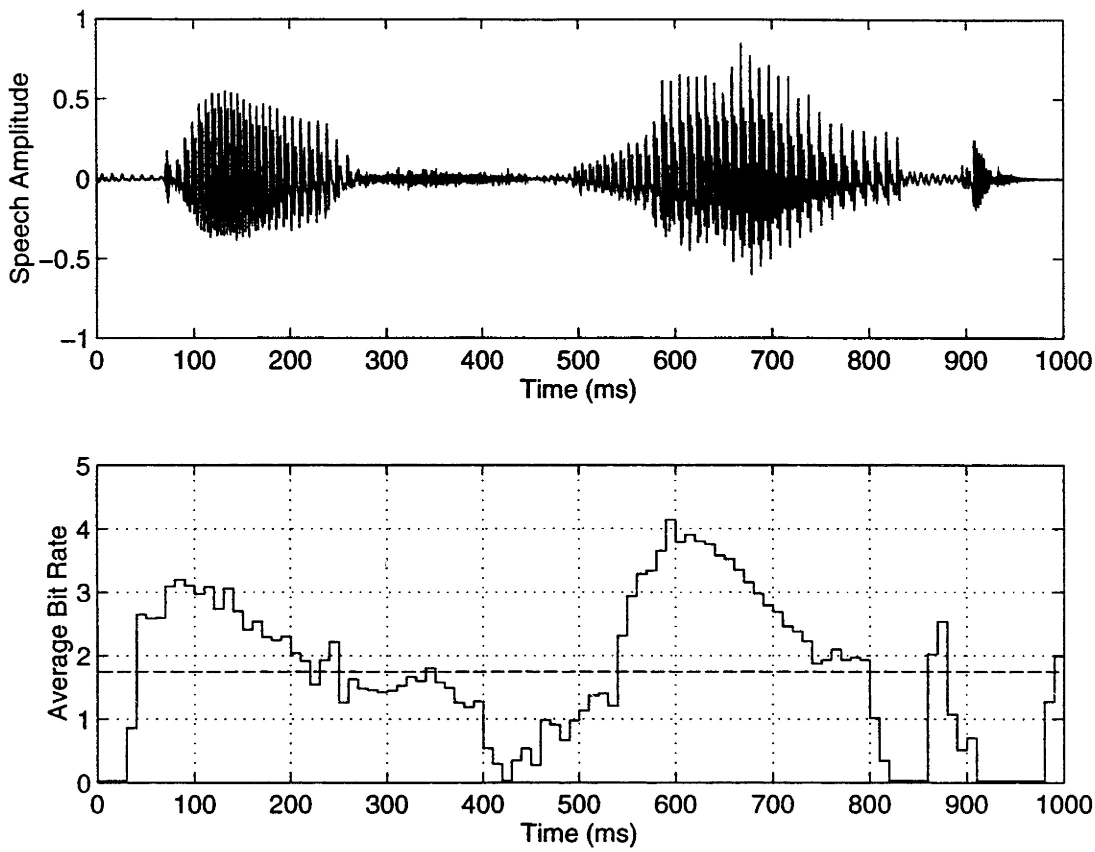

Figure 6.2: One Second Speech Input Segment and Output Bit Rate

6.9 Further Work

This chapter has discussed the incorporation of the Kalman filtering techniques from Chapter 5 into the AC-ADPCM system introduced in Chapter 4. No significantly new research work has been presented in this chapter, and hence a section discussing further work appears a little out of place. However, in light of the results presented in this chapter, we feel it is important to re-iterate some of the ideas for further work that were mentioned in the previous two chapters.

Very good performance has been observed with the AC-ADPCM system operating with an average bit rate of 16 kbps and below. The use of Kalman filtering techniques significantly improves the performance of the basic AC-ADPCM system. Other techniques of pitch prediction and delayed-decision coding are definitely worth serious future consideration, and may provide important performance improvements.

The performance of the AC-ADPCM system degrades significantly with average bit rates much below 8 kbps. Non-voiced speech sections begin to disappear at these rates. Techniques of slowly adaptive quantizer step sizes, or non-uniform quantization should be considered as possible means of remedying this situation.

Techniques such as pitch prediction and non-uniform quantization may also significantly assist with decreasing the peak to average output bit rate ratio. The peak to average ratio becomes important when dealing with communications applications rather than storage applications (further discussion on applications is relegated to Chapter 10). In communications applications probably a more important problem with the AC-ADPCM system that needs attention is the issue of robustness to channel transmission errors. The topic of AC-ADPCM resynchronization is covered briefly in Chapter 9.

6.10 Chapter Conclusion

The Kalman filter provides significant subjective performance improvements when applied to the AC-ADPCM system from Chapter 4. The Kalman filter can also be applied in parallel with other standard techniques of perceptual weighting and postfiltering to give a substantial combined benefit.

The computation required by the Kalman filter can be significantly decreased through the use of the reduced order Riccati equation approach, and by only periodically updating the Riccati equation to exploit the slow variation in parameters. The use of such a practical Kalman filtering approach gives a significant performance improvement with a fairly manageable computational requirement.

With perceptual weighting and Kalman filtering the performance of the basic AC-ADPCM system is surprisingly good considering the simplicity of the approach. The incorporation of adaptive postfiltering results in a practical speech coding system which can be used to make comparisons of output speech quality with other speech coding systems such as LD-CELP.

For the low complexity version of AC-ADPCM presented, where a 15th order predictor is used, updated every 80 samples, we obtain good performance at average output bit rates in the range of 8 kbps to 16 kbps. Judged by informal listening tests, we regard the performance at an approximate average bit rate of 12 kbps to be equivalent to that of LD-CELP at a 16 kbps rate. The performance of AC-ADPCM at 8 kbps is also quite good, but substantial degradations are observed in the output at the 4 kbps average rate.

To obtain this AC-ADPCM system operating with an 80 sample update period, we have simply modified coder routines used for LD-CELP, where updates are performed every 20 samples. A more sophisticated approach would be to redevelop the coder components specifically for the 80 sample update period. A significant performance improvement might be expected from this approach, and it is left as a topic for future development work.

Increasing the complexity of the AC-ADPCM system to be roughly equivalent to that of LD-CELP by the use of a 50th order predictor, and 20 sample update periods, we obtain significantly improved performance. For this higher complexity version of AC-ADPCM we regard the performance at the 8 kbps average bit rate to be roughly equivalent to that of LD-CELP.

Throughout this chapter we have attempted to be 'fair' with our subjective comparisons, but it is important to note that LD-CELP is a well-tuned system while the AC-ADPCM approach is not. Further developmental work may provide further performance improvements, but the comparisons in this chapter have certainly provided us with 'proof of concept'.

Based on the good performance obtained by the computationally relatively simple AC-ADPCM scheme, it is logical to consider potential applications for the approach. This is the topic of Chapter 10.

Chapter 7

Kalman Filter use in CELP

7.1 Chapter Motivation

Kalman filtering is known to be a useful tool for some speech coding applications. Chapter 5 has shown the utility of Kalman filtering techniques for dealing with measurement noises, both in the form of background or input noise, and coder quantization noise. Chapter 6 considered a particular application of the Kalman filter to the AC-ADPCM system introduced in Chapter 4. Here the Kalman filter was observed to obtain substantial subjective performance improvements.

CELP (Code Excited Linear Prediction) approaches to speech coding are common for bit rates below 16 kbps. Two well known CELP speech coding standards are LD-CELP[33, 34, 35] and FS 1016 4.8 kbps CELP[25, 26]. Both of these coders have been discussed briefly in Chapter 2. Within this chapter we consider the application of Kalman filtering techniques to both the LD-CELP and FS 1016 coders.

For LD-CELP, the Kalman filter is considered for both the purpose of quantization noise filtering, and to assist with obtaining improved performance when channel bit errors are present. We observe that these two problems are closely related, with both being 'measurement noises' (as were the problems of background noise filtering and quantization noise filtering). The Kalman filter is also applied to FS 1016 CELP, and a deterministic codebook version of FS 1016 introduced by Kao and Baras[109].

7.2 Quantization Noise Filtering in LD-CELP

Quantization noise filtering already occurs in the LD-CELP coder in the form of perceptual weighting, and both short and long term adaptive postfiltering[33, 34]. In the previous chapter it was shown that the additional quantization noise filtering afforded by the Kalman filter was quite significant for the AC-ADPCM system. Although LD-CELP already gives toll quality speech output, with high quality audio equipment an audible level of quantization noise is observed. Hence we attempt to reduce this audible quantization noise through the application of the Kalman filter.

There are a number of possible ways that the Kalman filter can be applied to LD-CELP. The first possibility is application as a form of postfilter, after the normal LD-CELP decoding operation. Intuitively it would appear that the Kalman filter should be used between the LD-CELP decoder synthesis block, and the normal adaptive postfiltering block. Previously we have used the Kalman filter in both the encoder as a replacement for the standard linear predictor, and in the decoder as a replacement for the standard synthesis filter operation. The close association of the Kalman filter and the standard synthesis filter would thus appear logical. No investigation was made with the use of the Kalman filter on the postfiltered output.

Parameters for the estimated process noise variance, $\hat{\sigma}_{wk}$, and measurement noise variance, $\hat{\sigma}_{nk}$, are required by the Kalman filter. For the LD-CELP system, we can assume the process noise variance is obtained with the use of a standard variance estimation procedure on the gain-scaled excitation. From analysis of the trained gain-shape LD-CELP codebook, it is not directly clear what are the properties of the quantization (measurement) noise. For simplicity we assume that the quantization noise is white, and that the noise variance is directly proportional to the excitation variance,

$$\hat{\sigma}_{nk} = \alpha \hat{\sigma}_{wk}, \tag{7.1}$$

with some fixed proportionality constant, α.

A value is thus required for the constant α, and a simulation tuning approach was undertaken to obtain this value. As discussed in the previous chapter, objective performance measures such as standard SNR figures do not provide any significant information on the subjective quality of the system with perceptual weighting and Kalman filtering. Hence a table of objective measures for different values of the parameter α is not worthwhile. However, by performing informal subjective comparisons, we were able to determine the 'best' value for the constant α.

From listening comparisons of Kalman filter use, without the LD-CELP adaptive postfilter, the parameter range

$$\frac{1}{12} \leq \alpha \leq \frac{1}{8}, \tag{7.2}$$

was observed to result in some subjective improvement in output speech quality.

With the addition of the adaptive postfilter after the Kalman filter, the higher values of α were observed to produce some minor output degradation in the form of slight muffling. The revised parameter range:

$$\frac{1}{25} \leq \alpha \leq \frac{1}{17}, \tag{7.3}$$

was observed to provide some performance improvement. The resultant improvement over the standard LD-CELP output was, however, only relatively minor.

We believe that the failure to observe substantial performance gains with the use of the Kalman filter is related to the fact that LD-CELP already exhibits high output speech quality. Of course, another possibility that can not be ignored is the fact that the above method of modelling the quantization noise introduced by the LD-CELP codebook is not satisfactory.

As the codebook is trained based on a perceptually weighted distortion criterion, it is possible that some element of colouring of the quantization noise exists. In addition, the gain-shape format of the codebook suggests that the quantization noise may be proportional to the gain magnitude.

These arguments indicate that in order to obtain any significant performance benefit from the use of the Kalman filter in LD-CELP, a closer integration of the Kalman filter and the LD-CELP system must be considered. Integration even to the extent of using the Kalman filter as a replacement for the standard linear predictor in the LD-CELP encoder should be considered, and is left as a topic for possible future research. However, due to the additional complexity and modifications to the existing LD-CELP standard that would appear to be required to get any significant performance benefit from Kalman filtering, the practical use of such an approach would appear to be minimal.

7.3 Kalman Filtering and LD-CELP Transmission Errors

As transmission bit errors are simply another form of measurement noise, it is useful to consider the Kalman filter as a tool for providing performance improvement where bit errors are present. In the previous section we have observed very little performance gain with the Kalman filter applied to LD-CELP to filter quantization noise. One of the reasons indicated for this minimal performance improvement was the low level of quantization noise present. From Chapter 5 it is known that the Kalman filter gives larger relative performance improvements for higher noise situations.

For LD-CELP transmission bit error rates of up to 10^{-2}, the excitation measurement noise (consisting of quantization noise and transmission errors) is certainly significant. This is readily observed via the noisy quality of the reconstructed output speech. It might thus seem logical that the Kalman filter would provide significant performance benefit.

Unfortunately, for the Kalman filter to operate efficiently requires a reasonably accurate estimate for the measurement noise variance. For a bit error rate equivalent to one bit error every 100 bits, we have one in every 10 codevectors in error (on average). The measurement noise (quantization noise) present in the 9 codevectors without bit error is quite small, while the one codevector with the bit error can be assumed to have a relatively large measurement error variance.

Normally we assume that the bit errors are randomly distributed, and without error detection to determine which codevectors are in error, we can only assume some 'average' value for the measurement error variance. Simulations show that a large estimate for the measurement error variance would appear to provide reasonable performance for the vectors containing errors, but results in severely degraded speech output by being far too conservative for the 'good' vectors. A much smaller measurement error variance is not conservative enough for the vectors with errors, while intermediate values do allow some trade-off, but in terms of net performance Kalman filtering does not appear to be effective. (Again, negligible overall performance improvement was observed.)

Assuming some form of error detection, such as a parity bit added to the 10 code bits from each vector, some more substantial performance gain might be expected from the use of the Kalman filter, as we are able to obtain a much more accurate measurement noise parameter. Some elementary simulations did indeed display SNR and segmental SNR improvements of the order of 0.5 to 0.6 dB. This was accompanied by some slight perceptual improvement in the form of less severe errors. However, a decision was made at this stage to terminate the line of research due to the observation of only small performance improvements, and the perhaps impractical requirement of error detection on each codevector.

7.4 FS 1016 4.8 kbps CELP

Another CELP system introduced in Chapter 2 is the US Federal Standard 4.8 kbps CELP coder FS 1016[25, 26]. This coder is a forwards adaptive system incorporating a 10th order predictor, adaptive codebook (for pitch redundancy), perceptually weighted search of the stochastic codebook, and postfiltering. The bit rate and reconstructed speech quality is significantly lower than that of LD-CELP, and as the Kalman filter gives more relative advantage for larger levels of measurement noise, we might expect significant gain from the use of the Kalman filter.

The fact that the predictor is forwards adaptive allows consideration of the replacement of the standard synthesis filter in the decoder with a Kalman filter. This approach is not really possible for a backwards adaptive system, as the different output obtained from the Kalman filter would result in different predictor coefficients at encoder and decoder.

As in the previous chapter the effect of perceptual weighting is accounted for by the Kalman filter. However, the parameters of the perceptual weighting filter differ in FS 1016 from those in LD-CELP. With the notation from Chapter 6, the perceptual weighting filter has the form

$$W(z^{-1}) = \frac{1 - A(z^{-1}/\gamma_1)}{1 - A(z^{-1}/\gamma_2)}, \tag{7.4}$$

where for FS 1016 the parameters γ_1 and γ_2 are 1.0 and 0.8 respectively. The impulse response used for the Kalman filter measurement vector is thus:

$$\begin{aligned} W(z^{-1}) &= h_0 + h_1 z^{-1} + h_2 z^{-2} + h_3 z^{-3} + \ldots, \qquad (7.5)\\ h_0 &= 1, \\ h_1 &= -0.2a_1, \\ h_2 &= -0.16a_1^2 - 0.36a_2, \\ h_3 &= -0.128a_1^3 - 0.416a_1a_2 - 0.488a_2, \\ h_4 &= \ldots, \end{aligned}$$

which is a slight departure from the values used in Chapter 6.

With the use of the Kalman filter, estimates are of course required for the process noise and measurement noise parameters. Ignoring any possible effect of the adaptive codebook, we can assume that the process noise and measurement noise are related to each other by a simple constant factor. This is clear by observing that gain-scaling the codevector to account for changes in the process noise results in an equivalent gain-scaling of the quantization or measurement noise. As the Riccati equation is only sensitive to relative values of the process and measurement noises in steady state, the proportionality factor is the only parameter we need to choose. (We can assume that the process and measurement noises do not change over the 60 sample FS 1016 subframe, and hence the steady state assumption is valid.)

Unfortunately, current results indicate only minor improvements in performance through the use of the Kalman filter in FS 1016 CELP. Hence the overall utility of Kalman filter application to CELP would still appear to be in doubt.

7.5 Deterministic Codebook FS 1016

Recent work by Kao and Baras[109] has considered the use of a deterministic codebook structure in association with FS 1016 CELP. This approach is observed to greatly reduce the computational complexity of the FS 1016 coder, while maintaining performance.

The ternary-valued stochastic codebook is replaced with a sparse ternary valued deterministic codebook. For the deterministic codebook (of length 60), there are nine non-zero elements, with each element either a +1 or a -1 value on the basis of one of the nine 'codeindex' bits. The non-zero elements are spaced evenly over the 60 sample codevector, with 6 or 7 sample periods between each one. As most of the energy of the impulse response of the predictor can be assumed to lie within 6 samples, each non-zero element can be selected without any concern of what values the other 8 non-zero elements take.

This deterministic codebook approach can be viewed as a form of down-sampling, as we have one measurement, and then no measurements for another 5 or 6 samples. Strictly there are measurements for the other samples. The adaptive codebook, which is used for pitch prediction means that some account is taken of the other samples, and perceptual weighting means that the sparse measurement is performed in a domain where it is a measurement of several speech samples, rather than one. However, ignoring the effect of the adaptive codebook, and working in the perceptually weighted speech domain, a form of downsampling can be assumed to exist.

As far as the Kalman filter is concerned, the regular structure of the deterministic codebook from [109] implies that more accurate parameters for the measurement noise should be obtainable. Also, the work by Ramabadran and Sinha[147, 163] suggests that an alternative measurement vector should be utilized within the sparse measurement system.

7.6 Chapter Conclusion

The Kalman filter has not been observed to provide any significant benefit for filtering of quantization noise in CCITT Recommendation G.728 LD-CELP. Although our simulations did not involve full integration of the Kalman filter with the LD-CELP system, we might expect the postfiltering Kalman filter approach to obtain significant performance improvement if any substantial potential for Kalman filtering gain did exist.

For attempting to provide improved LD-CELP performance in the presence of bit errors the Kalman filter was again only observed to provide minimal performance improvement. The task of obtaining an accurate value for the measurement noise was observed to be important, and the use of a parity error detection approach was proposed. However, the practical application of such an approach is questionable, and further research on the topic was not performed.

Further application of the Kalman filter to FS 1016 4.8 kbps CELP has also been inconclusive. Although some minimal subjective performance gains appear to have been observed by our (biased) informal tests, the absence of any significant improvement or any objective improvement does not allow us to claim any benefit.

Chapter 8

Frame Erasures in LD-CELP[1]

8.1 Chapter Motivation

Personal Communications Systems (PCS) and the Future Public Land Mobile Telecommunications Systems (FPLMTS) imply the need for high quality (Toll quality), low delay speech coding. As with all speech coding, we ideally would also like a low bit rate and low computational complexity.

The CCITT Recommendation G.728 LD-CELP[34] is a 16 kbps toll-quality speech coder, designed for universal applications including wire-based networks. As it is both high quality and low delay, LD-CELP's suitability for wireless applications, such as PCS, is worth consideration. The topic of improving the performance of LD-CELP in the presence of frame erasures is considered within this chapter, and is something that is a vital requirement for the PCS type application. It may also seem desirable to reduce the computational complexity of the coder. We do not pursue this topic, but attempt to ensure that we do not increase the complexity with any of our proposed modifications.

The work reported in this chapter displays the inherent high robustness of the LD-CELP system, and shows that with only minimal modifications the original CCIR[2] requirement of less than 0.5 MOS (Mean Opinion Score) degradation (compared with clear-channel G.728) at 3% frame erasure rate, can be easily met. In fact, the AT&T

[1]This work was undertaken with supervision by Juin-Hwey Chen, whilst a consultant at AT&T Bell Laboratories, Murray Hill, New Jersey. In March 1994 an AT&T contribution to the ITU-T (International Telecommunication Union – Telecommunications Standardization Sector (formerly CCITT)) entitled "G.728 Decoder Modifications for Frame Erasure Concealment" was made. This ITU-T contribution incorporates much of the work in this chapter[12].

[2]The CCIR (International Radio Consultative Committee) is now known as ITU-R (ITU – Radiocommunication Sector) after recent restructuring of the ITU (International Telecommunication Union).

March 1994 ITU-T contribution[12] demonstrates a MOS degradation of only 0.13 to 0.2. This has been achieved with only decoder changes, and no net computation increase.

8.2 Introduction

16 kbps LD-CELP (Low-Delay Code Excited Linear Prediction) speech coding has been adopted by the CCITT as the G.728 speech coding standard. As part of the standardization procedure, the coder was required to display a high level of robustness to random bit errors. Both Pseudo Gray coding of the 7 bit shape codebook[194], and Bandwidth Expansion of the backwards adaptive predictor coefficients, were found to be useful tools for increasing the robustness to single bit errors. G.728 maintains high quality speech at bit error rates of 10^{-3}, and produces intelligible speech at a 10^{-2} error rate.

Wireless communications, such as in systems proposed for PCS (Personal Communication Systems), has substantially different characteristics to the wire based telecommunications network. Whole frames of bits are often corrupted, resulting in highly unreliable data bits being received at the decoder, and the frame being effectively lost. The mechanism for this frame erasure phenomenon in mobile communications is often that of multipath fading, however high levels of radio interference will have the same effect. Other applications such as packet network transmission will also result in similar frame loss effects, and the work presented in this chapter should be equally relevant to those applications. Regardless of the particular application, it is obvious that the frame erasure scenario is substantially different from a random bit error scenario, and a different error recovery approach is required.

Improvements in wireless networking, such as the implementation of CDMA systems, beamforming at the base station, and smaller cell sizes, have the ability to greatly increase the reliability of the radio channel. Many other techniques of introducing space, time, and frequency diversity could also provide benefits. However, any practical engineering solution to the wireless communication problem is still likely to involve a significant probability of frame erasure errors. It is thus vital that any source coding system proposed for wireless communications is able to withstand frame erasures.

The application of the toll quality G.728 coding system to a transmission network system with radio channel links may be desirable for some situations, such as PCS. Our study is aimed at determining the performance that can be obtained. For simplicity and compatibility reasons, we attempt to make as little changes as possible to the existing G.728 LD-CELP system, and do so only when significant performance improvement potential is displayed.

The significance of the LD-CELP frame erasure problem has resulted in it being independently investigated by more than one speech coding research group. Husain and Cuperman from Simon Fraser University have undertaken a somewhat similar project[90, 91], that has been briefly discussed in [47]. Due to only recent awareness of this work, we provide no detailed comparison with the approach presented in this chapter. However, as normally occurs with independent work, an element of similarity is observed, in addition to some significant differences in methodology. An in depth analysis may even reveal potential for performance improvement through a combination of the complementary approaches.

In the next section we review the basic structure of LD-CELP, emphasizing the main features that are important when considering frame erasure errors. A brief coverage of frame erasures is then given, and the following section discusses the 'do nothing' option for the use of G.728 standard LD-CELP directly with a mobile radio channel. This forms the base-line, or 'worst case' situation. We then attempt to exploit the redundancy in the excitation signal to obtain a better estimate for the missing excitation at the decoder. Following this, the use of the Kalman Filter is discussed, and finally we look at modifying the encoder to obtain better frame erasure performance. We then present the results of some simulations and MOS test results from AT&T's contribution [12] to ITU-T, before finishing with some possibilities for future work, and conclusions.

8.3 LD-CELP Overview

LD-CELP, described in [34], is a backward adaptive analysis-by-synthesis coder. The trained codebook is in a gain-shape format, consisting of 8 gain levels, and 128 shape vectors, of five samples in length. Thus, ten codebook index bits are transmitted every five samples, resulting in a bit rate of 16 kbps. The coefficients for the gain predictor, and the short-term LPC filter are updated every 20 samples, via analysis on windowed previous reconstructed data (backwards adaptation).

A simplified block schematic of the LD-CELP decoder is shown in Figure 8.1. The encoder effectively consists of a copy of the decoder minus the postfilter, and an addi-

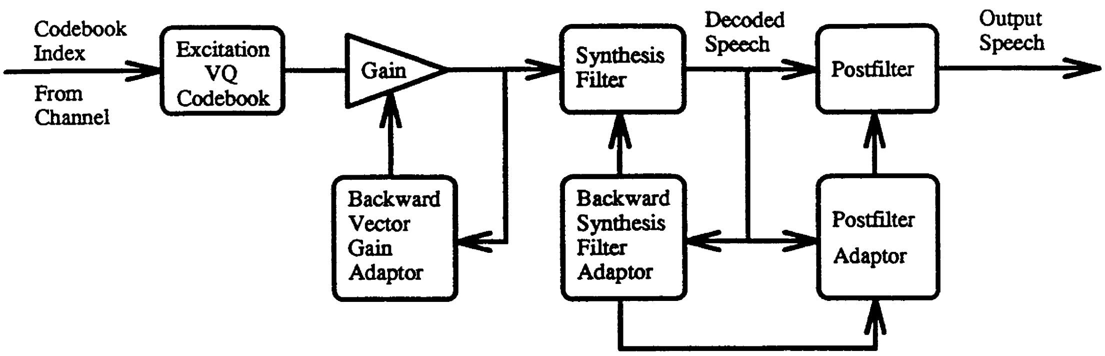

Figure 8.1: LD-CELP Decoder Block Schematic

tional outer loop structure to perform the codebook search. (The encoder has already been presented in block schematic form in Figure 2.3, and is not reproduced in this chapter.) A perceptually weighted mean-square error (WMSE) criterion is used to choose the best codebook entry. The perceptual weighting filter has an ARMA (Auto Regressive Moving Average) structure, for which the coefficients are obtained by bandwidth expansion of a version of the predictor coefficients. A 10th order perceptual weighting filter is used, and the coefficients are calculated from the unquantized (clean) speech. This is possible since the perceptual weighting filter only appears in the encoder.

Postfiltering also involves the use of an ARMA filter structure with coefficients obtained via bandwidth expansion of predictor coefficients. As the postfilter is 10th order, and the predictor is 50th order, the Levinson-Durbin recursion for the backwards adaptive predictor coefficients is halted at an intermediate stage to allow the 10th order coefficients to be extracted for use in the postfilter[34].

A 10th order AR (Auto Regressive, or all-pole) predictor is used for the gain. This gain predictor operates in the logarithmic gain (log-gain) domain, and computes a gain prediction for each five sample vector. The synthesis filter consists of a 50th order LPC predictor for the speech. For the LPC predictor, a prediction of the following form is produced:

$$\hat{S}_{k|k-1} = a_1\hat{S}_{k-1|k-1} + a_2\hat{S}_{k-2|k-2} + \ldots + a_{50}\hat{S}_{k-50|k-50}, \tag{8.1}$$

where the notation $\hat{S}_{k|k-1}$ refers to the prediction of the sample at time k given information up to time $(k-1)$, $\hat{S}_{k-i|k-i}$ is the reconstructed sample at time $(k-i)$, and $a_i, i = 1..50$ are the predictor coefficients. The synthesis filter simply obtains the reconstructed samples by adding the excitation to the prediction:

$$\hat{S}_{k|k} = \hat{S}_{k|k-1} + \hat{e}_k, \tag{8.2}$$

where $\hat{e}_k$ is the corresponding excitation sample obtained from the codebook. The log-gain predictor operates similarly. However, there is no gain excitation (innovations process) explicitly transmitted. Instead, the reconstructed gain for use by the log-gain predictor is calculated directly from the gain scaled excitation.

A pitch predictor is not used in Recommendation G.728 LD-CELP. Chen *et al.*[34] discuss the reasons for this as being due to the delay constraint restricting the pitch predictor to be backwards adaptive, and the lack of robustness to transmission errors of the backwards adaptive pitch predictor. However, it was found that the elimination of the pitch predictor significantly degraded female speech (the effect on male speech is noted as being less significant). To compensate for this quality loss, the linear predictor order was increased from 10 to 50. Hence some pitch redundancy is able to be extracted via the 50th order linear predictor, and the high order linear predictor was found to be fairly robust to channel errors.

The non-stationarity of the speech signal implies that predictor coefficients can be only confidently used over the small window of speech from which they are obtained. In forwards adaptive coding schemes, the windows used for predictor coefficient analysis are often symmetric, and centred around the coder frame. For a backwards adaptive coding system such as LD-CELP, only previous decoded speech can be used for the predictor coefficient analysis procedure. It is thus logical that the windows used for the analysis concentrate the majority of their energy in the most recent past, and the update procedure is performed frequently. For a 50th order predictor with updates every 20 samples, there is a high computation cost involved. LD-CELP attempts to minimise this computation cost through the use of a novel 'hybrid window' approach.

The hybrid window consists of a cosine shaped component meshed with an exponential tail. The auto-correlation parameters used by the Levinson-Durbin recursion are obtained by the addition of contributions from the cosine window part and the exponential tail. As the window is shifted the new contribution from the exponential tail is obtained by scaling the previous contribution, and adding the effect of the samples newly covered by the tail. In this way the extra state information for the separate exponential tail contribution to the auto-correlation saves a significant amount of computation.

Frame erasure disturbances mean errors are present in the excitation sequence.

Due to the backward adaptive nature of LD-CELP, these errors will result in some mismatch of the predictor coefficients during, and for a period of time after, the frame erasure. The backward adaptation approach is necessary in order to meet the low delay constraint, but it means that there are complicated feedback dependencies present in the LD-CELP system. These feedback effects mean any frame erasure recovery strategy must be particularly careful in the type of errors introduced in the output speech. The effective amplification over time of any undesired structure in the output to which the filters are able to adapt must be avoided.

Unlike forward adaptive coders, after a period of frame erasure we do not have a correct set of predictor parameters at the decoder. Hence, we need to ensure that not only does the filter memory of the decoder converge to that of the encoder, but also that the filter parameters converge. For a stable predictor, errors in internal state are decayed over time. This convergence is something that is easy to display theoretically, as it is really nothing more than the common definition of stability. Theoretical analysis of the stability and convergence properties of the adaptive system is more difficult. However, this is a common problem in adaptive control, and we refer the reader to books on the subject such as that by Anderson *et al.*[3].

8.4 Frame Erasures in Mobile Communications

Wireless communication generally involves source and channel coding, modulation, and RF transmission. This is shown in a simplified block schematic form in Figure 8.2. The demodulator and channel decoding attempt to recover the encoded source signal from the noisy transmitted signal. For 'well behaved' transmission noise, we can assume that the output of the channel decoder is a faithful representation of the input to the channel encoder.

Mobile wireless transmission channels exhibit tremendous variability in properties. Techniques exist to accommodate some of these channel variations through the use of methods such as channel tracking, adaptive equalization, and convolutional coding. However, it is practically impossible to completely account for these variations at the channel coding and modulation levels. Thus, in realistic conditions, the output of the channel decoder cannot be assumed to be a perfect representation of the channel encoder input.

A significant difficulty with mobile communications is the problem of multipath

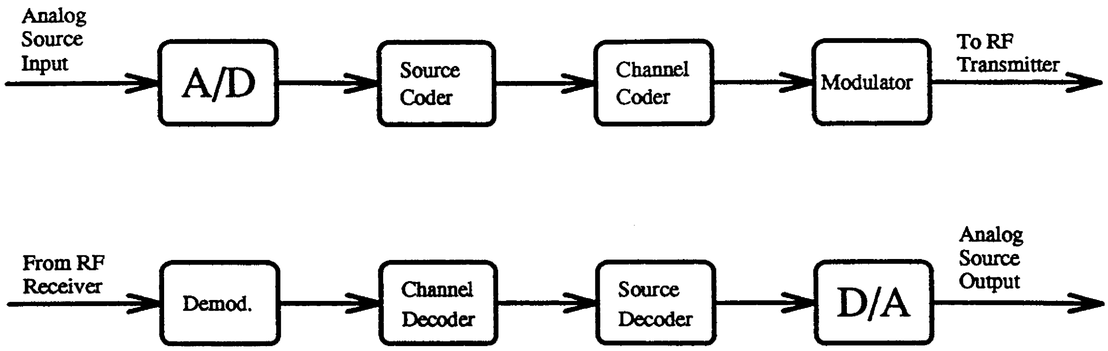

Figure 8.2: RF Channel Digital Transmission Schematic

fading. In this situation, due to destructive addition of multipath components, the received signal power is very low compared to the received noise power. The result is large probabilities of transmission errors during the fade. Fades can last for periods of time long enough to mean that several adjacent frames of data are effectively lost. For speech coding in mobile communications (and in general), there is no time to request retransmission, due to the delay constraint imposed by real-time two way communication. Thus the source decoder must be able to account for the lost data.

Frame interleaving and channel coding are often used to flatten out the effects of fading. However, practical constraints such as interleaving depth or delay, and channel coding complexity mean that frame erasures exist. This must be accepted as a property of the channel, and dealt with at source coding level. Channel coding issues for mobile communications channels that are assumed to be either Rayleigh or Rician fading in nature are covered in many texts, such as those by Proakis[143] and Steele[169].

In performing simulations of the effect of frame erasures on the speech coding system, an obvious requirement is the construction of a framing methodology, and the adoption of a frame erasure model. Both frame construction and erasure statistics are to a large extent (but not completely) at the control of the communications system designer. The method of frame construction is quite directly controllable. For the application of LD-CELP to a mobile communications system it makes sense to use a frame size that is a multiple of the basic LD-CELP adaptation cycle of 20 samples (2.5 ms). Frame sizes of 5, 10, 15, and 20 ms might thus appear logical candidates. As the original CCIR requirement was for a frame size of 10 ms, this is the frame size our work targets.

Remark 8.1 It is important to note that the choice of frame size is not straightforward, with some mobile communications systems using 20 ms frame sizes and even longer.[3] To further confuse the issue, for PCS applications, frame sizes at the air-interface as low as 2 ms appear to be receiving serious consideration, and a 20 ms frame might seem an appropriate size for an ATM (Asynchronous Transfer Mode) transmission scenario.

Frame erasure statistics are less directly controllable from the source coding perspective. Erasures are dependent on environmental conditions (often the man-made environment) within the area of the mobile user and the base station. The speed and path of the mobile unit are also important, as are the other components of the communications system, such as the RF transmitter, antenna, channel coding, and equalizer. Hence, it is clear that frame erasures are difficult to characterize and model accurately.

The CCITT Users' Group on Software Tools produced a library of software tools in May 1992. One of these tools allows simulation of both bit transmission and frame erasure errors (following the N state Markov model approach from Varma[180]). These software tools were used for the production of the frame erasure patterns for the simulations reported in this chapter.

8.5 G.728 and Frame Errors

Direct application of G.728 to wireless communications is possible, since as far as the LD-CELP decoder is concerned, it is simply processing an input bit stream. If we assume that no channel coding is present for error protection, then a frame erasure can be assumed to correspond to random bits being passed on by the demodulator of the radio channel. In order to simulate this situation we use a frame erasure pattern that dictates when to pass the G.728 decoder reliable bits, and when to corrupt the frame by passing the decoder random bits.

As could be reasonably expected, the output obtained from such a simulation has perceptually extremely severe distortions. These appear as many large magnitude, but short duration, 'explosions'. Although these errors make the speech almost painful to listen to, it is still mostly intelligible, even for frame erasure rates of up to 20%. For error rates of 1% or less, the performance may be tolerable for some applications, but

[3]The longer frame sizes would appear to suggest the consideration of alternative speech coders to LD-CELP, as the relaxation of the delay requirement could be used to obtain performance or computation advantages.

is certainly not pleasant. The fact that the output is intelligible for high frame erasure rates, is tribute the inherent robustness (and residual redundancy) of LD-CELP, and its ability to recover from large encoder/decoder discrepancies.

Remark 8.2 The severity of the errors where random code indices are received, and in particular the fact that the reconstructed signal is not even close to speech like in nature, can be viewed as an indication of the high level of unexploited compression potential in the transmitted signal. This is obvious from an Information Theory viewpoint, as the particular code index sequences received during the frame erasures do not result in output signals in the speech category. Hence some compression benefit theoretically can be obtained by not allowing these sequences any transmission codespace. Of course, obtaining any practical compression advantage would appear unlikely, due to considerations such as computation and delay constraints.

Remark 8.3 The recovery of the decoder after a frame erasure period indicates that some residual redundancy is present in the transmitted bit stream. Our task is not to improve the coder to remove this redundancy, but to attempt to exploit it to improve the frame erasure performance.

In considering source coding for wireless communications, it is common to assume we have a reliability measure of the data frame. We are not concerned here by how this is provided, although it generally involves a significant quantity of error detection bits added to the data frame, and may also involve some channel state information passed on by the demodulator. We simply assume that we have a reliable indication of whether a frame has been erased. For no frame erasure, we assume faithful reproduction of the encoder output bit stream at the decoder input, and where we have an erasure, we assume there is no information available in the received bit stream, and indeed we assume the received bits are random.

Remark 8.4 In practical situations, there are often residual bit error rates to be accounted for in addition to the frame erasures. However, we note that the standard LD-CELP bit error robustness is likely to be sufficient for this problem, as it is known that G.728 gives little degradation up to a bit error rate of 10^{-3}. Actually, for a received frame with one or two isolated (random) bit errors, it is possible that the performance obtained via the use of the 'corrupted' frame will be superior to that obtained by raising the frame erasure flag, and implementing a frame erasure strategy. For larger error bursts, frame erasure detection and a recovery strategy becomes vital.

Under the reliability measure assumption, and even without interfering with the encoder or decoder, and thus not affecting G.728 compatibility, we can exploit information about frame erasures that is available from the channel decoder. One assumption that we can make is that the excitation from the codebook is a zero-mean 'random' signal. Thus, the use of an all-zero excitation signal during the frame erasure period is a logical thing to use as a first approximation to the frame erasure problem. Unfortunately, there does not exist an all-zero codeword in the G.728 codebook, so this approach can not be implemented without losing standard compatibility (in its strictest sense).

Remark 8.5 Note that the strict sense compatibility with G.728 is important when dealing with an implementation of the coder where there is no ability to modify internal behaviour. This would be the case for an ASIC (Application Specific IC) implementation, or a DSP object code implementation purchased from a third party. Within this chapter we also consider a 'looser' form of compatibility with G.728, which is bit-stream compatibility. This implies that no modifications are made to the encoder, but that internal modifications are permissible for the decoder.

We are able to perform some pre-processing on the bit stream before it enters the decoder that will allow an approximation to the all-zero codevector. This may have some practical use, especially if G.728 becomes cheaply available on application specific IC's (ASIC's). One approximation to the all-zero approach is to use the smallest magnitude codevector, but this risks the introduction of periodicity on the 5 sample scale. This periodicity could be especially troublesome with amplification via backwards adaptation of the predictor.

Another all-zero approximation is to use a random shape code index, but to only use the smallest gain values. Yet another approach is to use only a subset of the 128 shape codevectors that correspond to the lowest energy entries. Simulations show that the pre-processing approach gives reasonable performance, with a significant improvement over the situation where totally random code index bits are fed to the decoder. Further discussions of the performance obtained via the approach are relegated to the results section (Section 8.9).

Remark 8.6 The use of the pre-processing concept need not be restricted to the assumption of zero mean excitation. Indeed, it is possible to envisage a system that uses excitation redundancy, in a similar manner to that discussed in the next section. This would be expected to give substantial improvement over the zero excitation approach, but would require the duplication of some computations that are already executed in the decoder. Thus, the approach would only make sense with a G.728 decoder chip being available extremely cheaply. For this reason and the fact that other modifications to the decoder also give improved performance, we do not consider this approach any further.

In order to consider the zero excitation option, we do need to make a change to the LD-CELP decoder, since there is no all-zero codevector. Certainly this implies that our decoder is already different from the standard G.728 decoder, but as we are dealing with such a minor change, we include it in this section. The performance obtained via this zero excitation approach is very similar to that obtained via the use of the low-level random excitation, and there exist some periods of large decoder errors. Again, more detailed discussions are included in Section 8.9.

8.6 Waveform Substitution using Excitation Redundancy

Observation of the gain-scaled excitation waveform in LD-CELP reveals that there is a large amount of excitation redundancy. One reason for this can be related to the lack of a pitch predictor in the system. However, even with a pitch predictor we would still expect a significant amount of residual redundancy, related to predictor optimality. We consider making some minor modifications to the LD-CELP decoder, such that when faced with a frame erasure, it exploits the excitation redundancy to the advantage of obtaining better reconstructed output speech quality.

It is important to note that we are performing waveform extrapolation in the excitation domain, unlike other approaches such as that contained in the paper by Goodman *et al.*[78] which perform the extrapolation directly on the output speech. One of the reasons for this is to avoid the possibility of large output waveform discontinuities at frame boundaries, that would require some additional transition smoothing approach. The discontinuity problem in the excitation is not so serious, as the synthesis filter smoothes the excitation to some extent.

Another motivation for extrapolation in the excitation domain is that by doing this we are able to ensure that the many internal states of the predictors and backwards adaptation routines have reasonable values at the end of the frame erasure period. This is important in order to achieve faster reconvergence of the decoder states.

The possibility of using an interpolation approach for the excitation, rather than one-sided extrapolation, is also important. Performing interpolation in the output speech domain would be dangerous, since due to the backward adaptive nature of the system, the output speech takes a significant amount of time to converge to its correct value. On the other hand, the excitation reconverges significantly faster, and interpolation is possible.[4]

The reason for employing the 50th order predictor in LD-CELP was to attempt to extract some pitch periodicity, especially to improve female speech. On this basis the question needs to be asked if the high order predictor is removing too much periodicity from the excitation sequence for our excitation extrapolation approach to operate adequately. The answer is that even with the 50th order predictor, there is still a significant amount of long term, or pitch, redundancy left in the excitation. Of course, the real test is whether the approach provides suitable performance.

8.6.1 Voicing Classification

We observe that during voiced speech there is a significant amount of periodic type redundancy in the excitation signal, which is not present during unvoiced sections. In order to exploit this fact, it is obvious that we need to be able to classify the speech into voiced and unvoiced. Fortunately we are able to do this without any significant computation, as a measure of the level of periodicity is available from the postfilter. In fact the postfilter uses this measure to perform its own classification.

The selection of the classification threshold is quite significant. If we mistakenly classify a frame as unvoiced rather than voiced, our excitation reconstruction strategy will not produce the periodicity required, and the output will be degraded. On the other hand, we can't classify everything as voiced, since the result is insertion of too much periodicity into unvoiced sections. Obviously a trade-off exists between the two classes, and a simulation approach to tuning of the threshold was taken.

A number of simulations were performed in order to determine what the best threshold level was according to perceptual performance criteria. The LD-CELP postfilter calculates the optimal tap weight of a single-tap pitch predictor for the decoded speech. The postfilter declares the frame as voiced if the tap coefficient is above the threshold constant of 0.6. Through simulation, we found that a lower threshold of the postfilter

[4] Since we are dealing with the gain-scaled excitation, we might expect some mismatch of the excitation while the gain predictor reconverges, but this is likely to cause significantly less disturbance than the combined effects of both the predictors, and the backwards adaptation blocks reconverging.

threshold divided by 1.4, (0.6/1.4), gave better performance, as it appears perceptually better to err on the side of voiced speech.

8.6.2 Voiced Speech

Faced with a frame erasure, it makes sense to utilize some of the excitation redundancy to attempt to reconstruct the missing excitation. For voiced speech, the procedure we apply is quite straight-forward. We work in the gain-scaled excitation domain, and utilize the pitch period available from the decoder postfilter to obtain a pitch estimate at the start of the frame erasure. This pitch estimate is used as a delay value for replication of the previous excitation at the decoder. In this way, we deal with a one-sided extrapolation technique. For production of the extrapolated excitation at time k, we thus use

$$\hat{e}_k = \hat{e}_{k-kp}, \tag{8.3}$$

where kp is the pitch period obtained from the postfilter.

Goodman *et al.* [78], also discuss two sided approaches to waveform substitution, and note that some important improvements are obtained for packet voice communications. The problem of frame erasure is almost identical to that of packet loss from the source coding perspective, and it was mentioned above that a two-sided interpolation approach is possible. However, additional delay is significant with the interpolation approach, and for the case of multiple frame erasure bursts it is unlikely to be of any use. Another point is the question of the added complexity for the interpolation approach. We do not provide any further consideration of this issue, and it is left as a possible topic for future research.

An attempt to account for the high level of uncertainty in the extrapolated excitation was also made by the conservative approach of reducing the magnitude of the excitation by scaling over time. In the case of multiple frame erasures, the effect of this approach is to continually scale down the excitation, reflecting the fact that the uncertainty in the extrapolated excitation is increasing, and for maximum uncertainty, zero excitation would appear to be the best reconstructed excitation we can obtain. However, this approach was not found to give any substantial perceptual benefit, and was not included in the core recommendations of the AT&T ITU-T contribution[12].

8.6.3 Unvoiced Speech

During unvoiced speech, the excitation is more like white gaussian noise. However, in order to maintain similar characteristics to the previous excitation, we proposed the use of a pattern matching technique for excitation extrapolation. The 30 excitation samples immediately prior to the erased frame are compared with a window of previous excitation to find the correlation peak. The corresponding lag of this correlation peak is then used for the extrapolation.

Unfortunately, unvoiced speech needs a slightly different approach, as the approach just mentioned has the effect of introducing too much periodicity into the excitation waveform (problem compounded by backwards adaptation), and causes annoying glitches in the decoder output. We proposed an *ad hoc* approach to the avoidance of introducing this periodicity, by altering the delay used for waveform substitution at regular intervals. The perceptual improvement obtained through the use of this *ad hoc* approach is quite significant, and the overall improvement through the use of excitation redundancy results in 'acceptable' frame erasure performance at an error rate of, say, 3%.

The pattern matching procedure does have the disadvantage of requiring an amount of computation that is non-negligible. Based on the observation that the excitation during unvoiced sections is somewhat random, the use of a complicated extrapolation technique is questionable. Subsequent work at AT&T considered the use of a purely random delay for extrapolation purposes. The performance was found to be similar to the pattern matching approach.

8.6.4 Predictor Updates

During frame erasure periods, the internal states of the encoder and decoder differ, even with excitation extrapolation, and the backwards adaptive predictor coefficients start to diverge. It has been found by simulation that freezing the backwards adaptive predictor updates can have a significant effect on the overall system performance. Although simply freezing the coefficients means that we guarantee a difference in predictor coefficients between the encoder and decoder, it appears that this difference is perceptually less noticeable than the differences that occur if the coefficients are updated normally (on corrupted data). This type of approach has been proposed previously for conventional CELP coders where bit errors are detected in the LPC parameters[41].

We have observed that a better approach is to be even more conservative in the choice of predictor coefficients, and not just freeze, but 'soften' the coefficients through the use of Bandwidth Expansion (BWE). Bandwidth expansion is already used for the backwards adaptive predictor coefficients and has been noted in [33] as improving filter robustness to numerical and channel errors. For the frame erasure problem, we apply additional bandwidth expansion with a BWE factor of 0.97, and use it in a compound fashion for multiple frame erasures. Hence at the beginning of the first erased frame, where we would normally perform Levinson-Durbin recursion to obtain the coefficients, we simply bandwidth expand the previous coefficients. These coefficients are used for the whole erased frame, but for a burst of frame erasures, we perform further bandwidth expansion at the start of each frame. This has the effect of continually softening the predictor coefficients over time to account for the increase in uncertainty during a frame erasure burst.

As we are extrapolating in the gain-scaled excitation domain, such an approach for the gain predictor is not necessary since the gain predictor is not used during the frame erasure period.[5] However, the use of the bandwidth expansion approach for the LPC predictor does lead to an important subjective improvement.

Other more involved methods of predictor coefficient extrapolation may give further improvements, but this approach is computationally inexpensive, and obtains good performance.

By the avoidance of Levinson-Durbin recursion during the erased frames we are able to obtain a substantial computation saving, which can be used to perform the other aspects of the frame erasure recovery, such as excitation extrapolation. Of course, there are a number of operations that do still need to be performed during the frame erasure to avoid large discrepancies in internal states at the end of the frame erasure period. The hybrid window approach involves a significant amount of internal state information, and although the output auto-correlation values are not being used during the frame erasure (although a small number of values are used for the postfilter), the internal states must be updated. Other internal states that must be updated include the logarithmic gain values for use in the log-gain predictor.

Remark 8.7 The above sections complete the description of the decoder modifications proposed in AT&T's contribution to ITU-T[12]. The next two sections discuss other

[5] The fact that the gain predictor is not used during frame erasure periods allows a computation saving important in off-setting the additional computation required by the frame erasure recovery strategy.

approaches investigated by the author while at AT&T, and although mentioned in [12], are not recommended modifications for the PCS type frame erasure problem.

8.7 Kalman Filter Application

We attempt to account for some of the uncertainty in the excitation sequence by applying the Kalman Filter to the two backwards adaptive linear predictors (gain and LPC predictors). Of course, we concentrate on the LPC predictor, as we expect this to have the largest effect on the output quality.

The Kalman Filter has received widespread use in speech enhancement applications, but has been used very little in speech coding. However, as presented in previous chapters and the paper by Crisafulli *et al.*[44], the Kalman Filter is effectively a simple extension of the standard linear predictor, and similar to dealing with uncertainty in the input signal for speech enhancement, it can assist with excitation uncertainty for speech coding.

The Kalman Filter efficiently accommodates input signals with additive white Gaussian noise. When performing waveform substitution, it is true that there is a large amount of uncertainty in the excitation sequences for the two predictors. We make the assumption that this uncertainty can be modelled (rather coarsely) as an additive white noise. Under this assumption, the use of the Kalman Filter is a worthwhile consideration. In speech coding and speech enhancement applications, this white noise assumption is often violated, and a coloured noise assumption must be used. However, the white noise assumption is considered here, and any extension to a coloured noise Kalman filter is left as a possible topic for future research.

The Kalman predictor exploits smoothing to obtain a prediction of the form

$$\hat{S}_{k|k-1} = a_1\hat{S}_{k-1|k-1} + a_2\hat{S}_{k-2|k-1} + \ldots + a_{50}\hat{S}_{k-50|k-1}, \tag{8.4}$$

which can be seen to be an extension of the standard linear prediction of equation 8.1.

The use of the Kalman Filter in LD-CELP, with a 50th order predictor would normally be considered to be too expensive in terms of computation to be of any practical use. We use a type of 'hybrid' Kalman Filter approach, where we use only a low order KF (Kalman Filter) (typically 4th order), with the full order predictor. This has been described in [184], and presented in Chapter 5. Hence we use the following prediction:

$$\hat{S}_{k|k-1} = a_1\hat{S}_{k-1|k-1} + a_2\hat{S}_{k-2|k-1} + \ldots + a_4\hat{S}_{k-4|k-1} + a_5\hat{S}_{k-5|k-2} + \ldots + a_{50}\hat{S}_{k-50|k-47}, \tag{8.5}$$

where smoothing is performed to four lags.

The use of the Kalman filter for log-gain prediction is analogous, except that the absence of an explicitly transmitted innovations process for the log-gain implies that it must be calculated. This is a simple matter of subtracting the gain prediction from the calculated value which was previously used to update the predictor memory.

Freezing the predictor updates during the period of frame erasure provides a computation saving that can be offset against the additional computation required by the Kalman filter. The computation required by the Kalman filter is of course dependant on the order of the Riccati equation required, and the ability to use steady-state solutions. These issues have already been covered in Chapter 5.

Noting that after a frame erasure period, there is uncertainty in the decoder's filter coefficients, we can extend the use of the KF beyond the frame erasure, in an attempt to accelerate convergence. Unfortunately this implies an increase in computation, as off-setting the computation with the computation saved by bandwidth expanding the predictor coefficients is not applicable after the period of frame erasure.[6] For the low-order hybrid approach, the additional cost of the KF can be kept fairly small by the use of techniques such as steady state KF approximations as discussed briefly in Chapter 5. However, the issue of using the Kalman filter after the frame erasure period is not pursued further within this chapter.

Performance improvements through the use of the KF in this way are inconclusive, judged both by informal listening tests, and by the MOS scores from [12], although some small improvement in SNR values was observed. Due to the increased computation requirements, and the fact that no subjective performance improvement has been shown, we can currently only recommend that the KF is not used as part of the frame erasure recovery solution. However, this is not to say that the Kalman filtering approach should be dismissed. Our study was quite brief, and many questions still need to be answered before a decision of that sort can be made.

Obvious topics for future research on the use of the Kalman filter include investigation of the white noise assumption. Work by Gibson *et al.*[76] has shown how a simple

[6] After the period of frame erasure, the decoder again operates at full complexity, and any additional processing implies an overall increase in complexity.

coloured noise assumption can provide significant performance improvements, and this should be considered. The need for an estimate of the uncertainty in the extrapolated excitation could also be a cause for poor KF performance, as could be the use of the reduced order approach. It may be the case that the high level of uncertainty in the extrapolated excitation means the reduced order approach to the Kalman filter needs to be of a high order, and this has not been accounted for. Of course, if this is a problem, a side effect of solving the problem may be to make the computation inhibitive.

8.8 Relaxation of Compatibility Constraints

Although the above improvements result in good performance for most frame erasures, there are still significant distortions observed for some frame errors. We observed a significant correlation between large perceptual degradations and frame erasures at the start of talk spurts and at transition regions. This is due to the fact that at the start of a talk spurt, there are rapid changes in predictor states, and coefficients, through the backward adaptation process. Any frame erasures during these periods of large change result in large discrepancies between encoder and decoder states, taking a long time to recover. By making some changes to the encoder, we are able to accelerate the convergence of the decoder states, and thus reduce the effect of the frame errors. The requirement is also present not to decrease the clean channel speech quality by a perceptually significant amount, although some degradation may be permissible, especially if it gives rise to a significant increase in the error performance.

We propose the use of spectral smoothing and further bandwidth expansion to assist in the reconvergence following severe error hits, and thus reduce the perceptual distortions resulting from these frame erasures. The spectral smoothing technique of Itakura[176] has previously been applied to the problem of numerical sensitivity in fixed-point LD-CELP[32]. There it was observed that due to the sharp spectral peaks in the all-pole predictor without spectral smoothing, small differences in pole position can translate to large differences in filter output. With the spectral smoothing technique, the spectral peaks are flattened, and consequently also reduced in amplitude. Small differences in pole positions thus give rise to very similar filter performance. This approach was also found to assist in overcoming divergence problems with signals that have all poles very close to the unit circle (the P50 problem)[11]. We expect the approach to be useful in assisting with frame erasure handling, since after an erased frame, we would like small differences in pole positions to result in similar filter performance.

The level of spectral smoothing chosen was 60 Hz for the LPC predictor and 45 Hz for the log-gain predictor, which is a significant increase over the values of 35 Hz and 45 Hz used in [32].

Bandwidth expansion is designed to move the poles away from the unit circle, and thus reduce stability[7] problems. It seems logical that the increased stability of the coder, and the reduced prediction gain afforded by bandwidth expansion, would imply increased reconvergence speed for the system. This is indeed the case. The G.728 LD-CELP standard dictates a bandwidth expansion factor of $253/256 \approx 0.9883$ for the LPC predictor coefficients, and $29/32 \approx 0.90625$ for the gain predictor coefficients. We decreased these factors to 0.96 and 0.87, respectively.

The use of increased white noise correction[32] was also considered as another approach to increasing reconvergence by decreasing predictor performance. Indeed, this was seen to significantly increase frame erasure performance, but unfortunately the degradation in clean channel performance was also quite significant. Clean channel performance was only minimally degraded with spectral smoothing and bandwidth expansion, so these techniques were used, and white noise correction was not.

The parameters for spectral smoothing and bandwidth expansion were tuned based on simulations at a 10% frame erasure rate. In the next section the reason for working at this rate will become clear. The subjective performance was significantly improved through the use of the encoder changes at the 10% rate, but was not substantially altered at the 3% rate, and for reasons of G.728 compatibility the encoder changes are not part of the core recommendations in [12].

Further system changes such as the use of different window philosophies for the backwards adaptation were also considered, but found to provide no significant benefit. The approaches trialed included lengthening the hybrid window in an attempt to both increase predictor stability (decrease performance), and smooth out the effects of previous errors in the reconstructed output. A shorter window length was also considered in an attempt to obtain better convergence by quickly ignoring previous reconstruction errors.

[7] Note that stability is closely related to recovery from transmission errors, as the effect of the errors is decayed over time.

8.9 Simulation Results

As mentioned previously, for simulations of frame erasure conditions we made use of the CCITT error insertion software tool. This tool is designed to simulate transmitting a data frame through a channel that introduces bit errors, and denotes a frame erasure by raising a flag in the frame header. However, we 'abused' the tool by simply using it to generate very long frame erasure patterns at different erasure rates. In order to ensure the use of different erasures for our simulations, the CCITT tool was used to generate an erasure pattern at least an order of magnitude greater in length than that needed for the length of the input files. A random offset into this file was then used to obtain different erasure patterns for the simulations.

The modelling of bursty frame erasures is obtained through the use of a Markov structure within the CCITT error insertion software tool. The choice of the burst factor really needs careful consideration to ensure that the output frame erasures are of a nature that would be typically expected in practice. Unfortunately, our ability to assess what is a typical frame erasure pattern is limited, with no small part of this limitation being the fact that the coder modifications are being performed without a specific system application. We thus performed simulations with a variety of burst factors, but concentrating on burst factor parameters of 0.01, 0.03, 0.1, and 0.3.

The lower burst factors result in effectively random frame erasures, and the higher factors give erasure patterns that might be expected as more typical. Subjective characteristics of the output were observed to vary slightly with the use of different burst factors. However, effects on comparisons between different frame erasure recovery strategies were not observed to be significant. Hence, for the simulation results presented and discussed in this section, we give no further consideration to the burst factor.

Most of our development was performed with simulations using a single input speech file of about 22 seconds duration. This file consisted of eight sentences, four of which were male speech, and four of which were female speech. In order to confirm that our results were not biased to this particular input file, a large number of simulations were run with other input. No significant differences in the nature of the output were observed across the range of input.

8.9.1 SNR Results and Discussions

Tables 8.1 and 8.2 present SNR and segmental SNR values for simulations of all the major approaches mentioned above, at various frame erasure rates. For segmental SNR measures the segment size was chosen to be 128 samples, which is not a simple multiple of the 10 ms, or 80 sample, frame size used for most of the simulations. The tables include SNR values both with and without the use of the LD-CELP postfilter. As a general rule the postfilter improves the subjective quality of the output, but decreases SNR measures.

All comparisons of subjective performance made within this subsection are based mainly on listening tests performed by the author. However, we are careful to ensure that no bias effects are introduced in the following discussions, and we believe that the conclusions drawn here correlate extremely well with the results of the MOS tests performed by AT&T and presented shortly.

Speech Coder Variant:	Frame Erasure Rate	Without Postfilter SNR	Without Postfilter Segmental SNR	With Postfilter SNR	With Postfilter Segmental SNR
Random Code Index	10%	0.15	-1.76	0.21	-1.87
Random Code Index	3%	3.00	6.72	2.97	5.82
Random Code Index	1%	6.52	12.68	6.51	11.16
Low Level Random	10%	4.95	6.84	4.79	6.47
Low Level Random	3%	6.61	11.86	6.50	10.56
Low Level Random	1%	8.89	14.86	8.23	12.94
Zero Excitation	10%	2.85	6.63	2.78	6.25
Zero Excitation	3%	5.92	11.86	5.90	10.70
Zero Excitation	1%	7.22	14.62	6.79	12.80
G.728 (16 kbps LD-CELP)	0%	15.93	18.19	14.69	15.46

Table 8.1: SNR and Segmental SNR Measures for Minimal Change Options

Table 8.1 presents the results of simulations where minimal changes are made to the LD-CELP decoder to overcome frame erasures. The random code index version refers to the situation where a frame erasure occurs and effectively results in random bits being fed to the G.728 LD-CELP decoder. From the SNR figures, it is clear that the approach gives very poor performance. Even at the 1% frame erasure rate perceptually severe output 'explosions' occur. At higher error rates listening to the speech becomes painful. However, the intelligibility is largely intact. This is an indication of the inherent robustness of G.728 LD-CELP.

Again using the assumption that a frame erasure produces random bits, but adding the assumption of frame erasure detection, we are able to preprocess the bit stream fed to the G.728 decoder. By masking the two gain bits to correspond to the lowest gain values, we are able to produce low level random excitation during the frame erasures. A significant SNR improvement is observed through the use of this approach, and the corresponding perceptual improvement is substantial. The elimination of the harsh error 'explosions' observed with the random code index imply that for small frame erasure rates this approach may have some practical use. Certainly it does not affect the decoder compatibility with the G.728 LD-CELP standard, and requires only a negligible amount of preprocessing.

From Table 8.1 we observe that the zero excitation approach results in similar, but slightly worse, SNR performance to that of the low level random excitation approach. The subjective performance of the output speech is judged to correlate with these SNR observations, but the difference in character of the residual distortions in the output speech makes a judgement of the 'better' approach very difficult. The fact that the SNR values are better for the low level random approach, and it appears to be slightly better subjectively, is an interesting observation in the light of the motivation for consideration of the approach. We chose to consider the low level random excitation as an approximation to the zero excitation approach, since no zero vector exists within the excitation codebook of G.728 LD-CELP. We note that the decoder has a large amount of internal state information, and the low level random excitation may slow down the decay of these internal states compared to the zero excitation approach. Hence after a frame erasure period the internal states are at more realistic values, allowing faster reconvergence.

Table 8.2 presents the results of our proposed system with decoder changes, the use of Kalman filtering, and the system with encoder changes. For comparison the SNR figures for the G.728 LD-CELP system in error free transmission conditions are included in the table.

Comparing the results for the proposed system with decoder modifications to those obtained for the minimal modifications shown in Table 8.1, we observe significant increases in SNR measures. These are accompanied by substantial subjective performance improvements. The subjective performance is perhaps significantly better than that indicated by the SNR measures, and at 1% frame erasure rate there is practically no observed degradation between the proposed system and G.728 for the zero error condition.

Speech Coder Variant:	Frame Erasure Rate	Without Postfilter SNR	Without Postfilter Segmental SNR	With Postfilter SNR	With Postfilter Segmental SNR
Proposed System	20%	3.04	3.59	2.83	3.37
Proposed System	10%	5.69	7.84	5.36	7.32
Proposed System	3%	7.59	12.79	7.32	11.35
Proposed System	1%	9.67	15.61	8.96	13.53
Kalman Filtering	20%	3.14	3.80	2.97	3.66
Kalman Filtering	10%	5.69	7.96	5.40	7.47
Kalman Filtering	3%	7.79	12.99	7.60	11.55
Kalman Filtering	1%	9.51	15.54	8.80	13.49
Encoder Changes (no KF)	20%	5.31	7.76	5.14	7.14
Encoder Changes (no KF)	10%	8.34	12.00	8.02	10.82
Encoder Changes (no KF)	3%	10.39	15.14	9.94	13.33
Encoder Changes (no KF)	1%	12.29	16.59	11.51	14.47
Encoder Changes (no KF)	0%	15.32	17.51	14.34	15.14
G.728 (16 kbps LD-CELP)	0%	15.93	18.19	14.69	15.46

Table 8.2: SNR and Segmental SNR Measures for Major Options Considered

Due to the high subjective performance, at frame erasure rates of 1% and 3%, of the system with decoder changes only, and with regard to the desire to remain G.728 standard compatibility, this is the system that we propose as a practical solution to the LD-CELP frame erasure problem. To perform these decoder modifications we have incurred no net computation increase.

Kalman filtering consumes a significant amount of computational resources, even with the reduced order approach previously mentioned, and presented in Chapter 5. The improvements we were able to obtain with the use of the Kalman filter were minimal, as shown by the SNR figures in Table 8.2, and echoed by subjective performance. Hence Kalman filtering is not currently part of the proposed frame erasure solution, although we again note the possibilities for further work on the topic.

A comparison of SNR figures for our system with encoder changes and the proposed system with decoder changes only, shows that in all cases except the clean channel situation we obtain significant SNR improvements. Due to the high level of performance of the proposed system at the lower error rates, it is impossible to claim that the SNR improvements observed with encoder changes at these rates translates to a subjective performance improvement. However, at the higher rates of 10% and 20% errors, substantial subjective performance improvements are observed.

The reason our encoder changes do not form part of the core proposal in [12] is the wish to retain as much compatibility with G.728 as possible, and that the performance at the ITU-T target frame erasure rate of 3% is extremely good without encoder changes. Obviously at very high frame erasure rates the encoder changes become worthwhile, and the similarity of the changes considered and those previously investigated for the problems of convergence and numerical sensitivity[32] also indicates that the approach is useful.

The lower SNR values at zero error rate for the system with encoder changes are not of great concern, as they only correspond to a slight degradation in subjective performance. Redesign of the codebook might be expected to recover most of this small performance loss.

8.9.2 Decoder Waveform Comparisons

The above discussions and SNR tables are able to give a general idea of the performance of the various frame erasure strategies considered. However, in order to better understand the effect of, and recovery from, a frame erasure, it is useful to consider the speech output on a scale that is closer to the scale of the frame erasure.

The following waveform diagrams in Figures 8.3 to 8.9 show 250 ms of output speech during which time there are three frame erasure periods. The frame size used is 10 ms, the erasure rate is 10%, and the frame erasure pattern is identical for all schemes. The plot in Figure 8.3 is the clear channel case for G.728 LD-CELP, included as a reference.

The outputs shown relate to the approaches already discussed. The three approaches with minimal decoder modifications of: (1) random codeindex excitation; (2) low level random excitation; and (3) zero excitation, are displayed in Figures 8.4 to 8.6. The next three Figures, 8.7 to 8.9, refer to the proposed system with decoder modifications, decoder modifications and Kalman filtering, and encoder modifications.

A quick comparison of Figures 8.3 to 8.5 show clearly that a frame erasure simply resulting in random bits fed into the G.728 decoder leads to large waveform distortion, while the simple procedure of masking the gain bits to produce a low level random excitation during the frame erasure provides far improved performance. The waveform in Figure 8.5 is seen to be able to recover a significant amount of the input characteristics after a frame erasure, certainly obtaining better performance than for the random excitation approach in Figure 8.4.

Figure 8.6 concerns the zero excitation approach, and is useful to compare with that

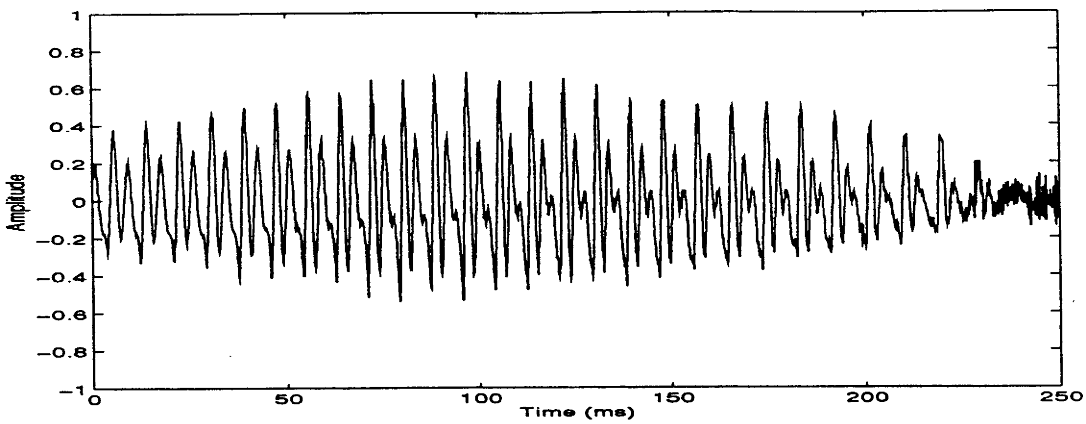

Figure 8.3: LD-CELP Clear Channel Decoder Waveform

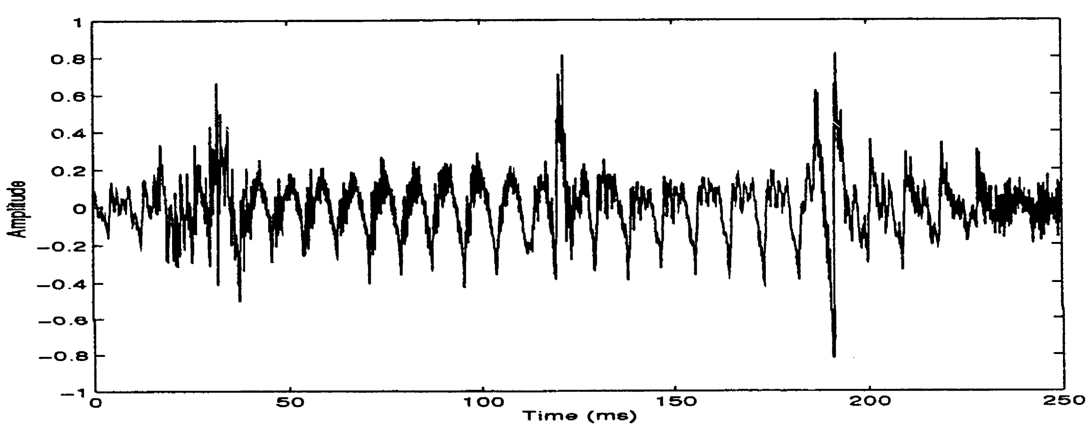

Figure 8.4: LD-CELP Random Excitation Waveform

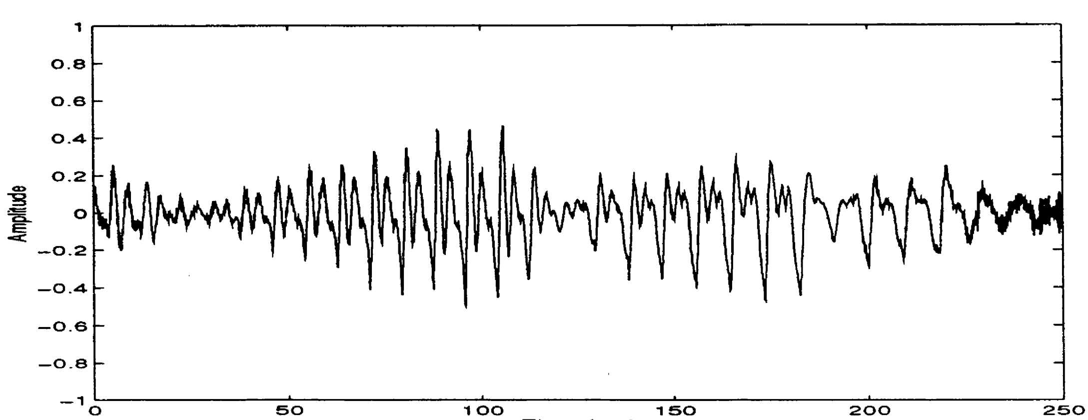

Figure 8.5: LD-CELP Low Level Random Excitation Waveform

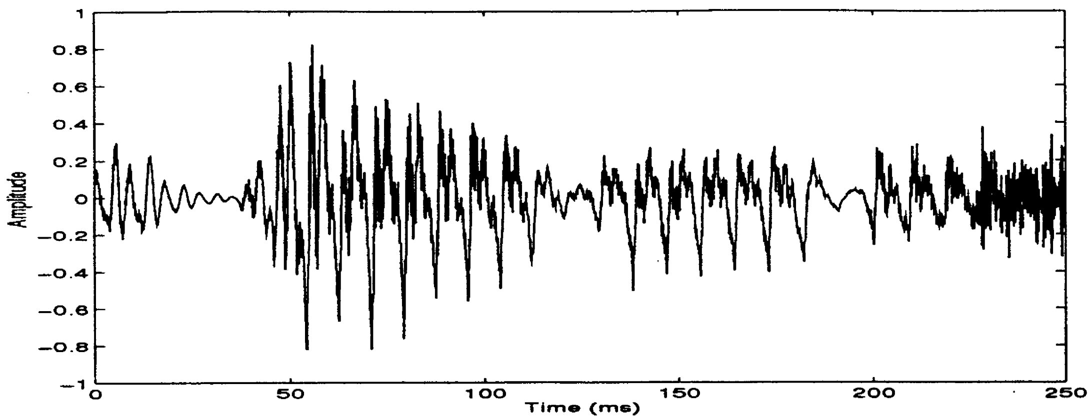

Figure 8.6: LD-CELP Zero Excitation Decoder Waveform

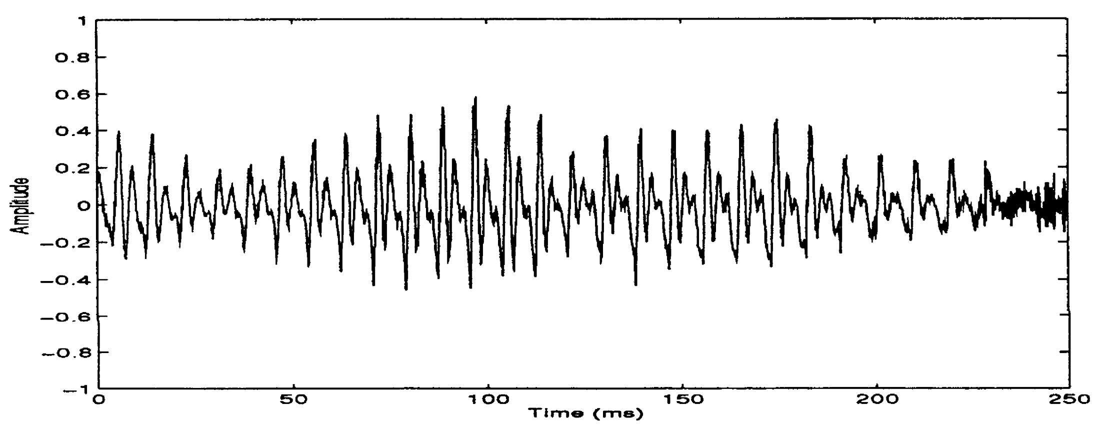

Figure 8.7: LD-CELP Proposed Decoder Modifications Waveform

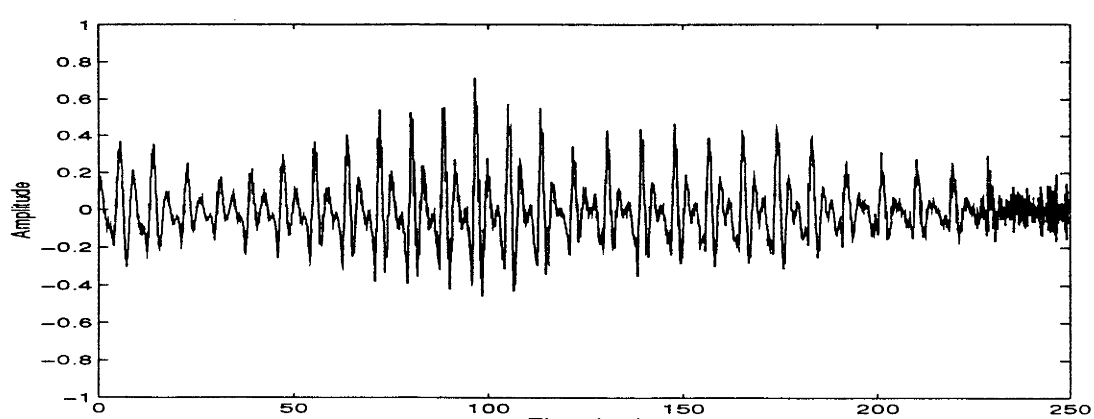

Figure 8.8: LD-CELP Kalman Filter Decoder Waveform

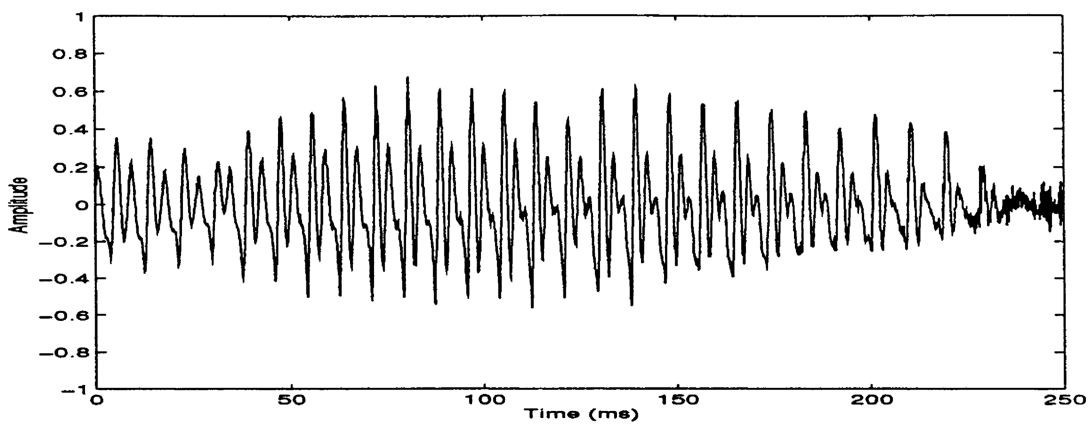

Figure 8.9: LD-CELP Encoder Modifications Waveform

of the low level random approach in 8.5. Again we observe that the zero excitation performance is inferior to that of the low level random excitation. We can form a conjecture that this is the result of the system placing too much emphasis on the excitation immediately after a frame erasure, since the zero excitation has allowed the internal states to decay too fast. Assuming zero internal states at the end of the period of frame erasure, the output reconstructed speech signal just after the frame erasure is effectively simply the excitation. The backwards adaptive predictor adjusts to this output signal, and results in undesired structure in the output for a large amount of time after the period of frame erasure.

Remark 8.8 We might expect to be able to improve performance for the zero excitation approach by modifying the backwards adaptation procedure after a frame erasure to better condition the predictor. However, the original motivation for the zero excitation approach was for minimal decoder modifications. Hence we do not pursue this option.

The output for the proposed decoder modifications is shown in Figure 8.7. Here we see that a frame erasure is accommodated with a resultant down-scaling of the output waveform. The scaling effect is suggested intuitively by the fact that the excitation uncertainty increases during the frame erasure period. The characteristics of the decoder waveform are thus seen to be very similar to the original, with no large discontinuities.

As with previous discussions of Kalman filter performance, the decoder waveform shown in Figure 8.8, and compared to Figure 8.7, leave us unable to distinguish any significant differences.

The decoder waveform obtained from the system with encoder changes (shown in Figure 8.9) is significantly better than that obtained from the proposed system in Figure 8.7. Of course, for low frame erasure rates, this difference was not found to be perceptually significant.

8.9.3 Mean Opinion Score Results

As with most speech coding, SNR measures do not give a good indication of the performance of the system. Formal subjective testing is substantially more useful, but usually more difficult to perform. Fortunately in preparing the ITU-T contribution [12], AT&T obtained a number of MOS (Mean Opinion Score) test results which we can refer to. The AT&T MOS testing for the G.728 frame erasure modifications was carried out in October and November of 1993. Both IRS (Intermediate Reference System) weighted speech and non-IRS (flat) speech were used. For the convenience of the reader the AT&T MOS results presented in [12] are summarized in Table 8.3.

Speech Coder Variant:	Frame Erasure Rate	October MOS		November MOS	
		IRS Speech	non-IRS Speech	IRS Speech	non-IRS Speech
Proposed System	10%	2.27	2.55	–	–
Encoder Changes (and KF)	10%	2.79	2.99	–	–
Zero Excitation	3%	2.26	2.65	–	–
Proposed System	**3%**	–	–	**3.77**	**3.59**
Encoder Changes (no KF)	3%	–	–	3.75	3.61
Proposed System	**1%**	–	–	**3.95**	**3.82**
G.721 (32 kbps ADPCM)	0%	3.77	3.70	3.90	3.78
G.728 (16 kbps LD-CELP)	**0%**	**3.88**	**3.77**	**3.90**	**3.79**

Table 8.3: AT&T MOS Results from October and November 1993 Tests

From [12] and Table 8.3 we see that the MOS degradation for the case of the zero excitation approach was quite large, at up to 1.6 points for 3% frame erasures. A similar type of MOS degradation was experienced for the proposed system with decoder changes and a 10% frame erasure rate. Encoder changes resulted in recovery of 0.5 points in MOS at the 10% erasure rate.

At a 1% frame erasure rate the proposed system with decoder changes resulted in no MOS degradation, and in fact, a slight increase in MOS value was obtained, but not enough to be statistically significant. For 3% frame erasure, the MOS degradation was

0.2 points or less. This is quite a small degradation, and is within the original CCIR target of 0.5 points or less. For the 3% frame erasure rate, the encoder changes did not improve the MOS scores over that obtained with only decoder changes.

8.9.4 Frame Size Effects

Although most of the investigations reported in this chapter were carried out with a 10 ms frame size, we did perform a small number of simulations with other frame sizes. We considered a shorter frame size of 5 ms, and a longer frame size of 20 ms. Unfortunately we believe our approach to using the same error patterns as for the 10 ms frame size is not realistic. As frame erasures are a property of the mobile communications channel, the choice of frame size should have very little effect on the net error pattern. In particular, if for a 10 ms frame size we have a maximum burst length of six frame erasures, we would expect that for a 5 ms frame this translates to twelve frames, and three frames for the 20 ms frame size.

Speech Coder Variant:	Frame Size (ms)	Frame Erasure Rate	Without Postfilter		With Postfilter	
			SNR	Segmental SNR	SNR	Segmental SNR
Proposed System	5	3%	8.04	10.96	7.68	9.94
Proposed System	10	3%	7.59	12.79	7.32	11.35
Proposed System	20	3%	8.87	14.21	8.62	12.54

Table 8.4: Frame Size Effects at 3% Frame Erasure Rate

Table 8.4 presents SNR figures for the three different frame sizes considered. Of course these results are questionable due to problem of the frame erasure patterns as discussed above. However, a number of points can be made. A significant increase in SNR is displayed for the 20 ms frame size, and although generally the subjective performance was probably better than for the 10 ms frame size, some isolated large error bursts were observed. These effects would result in the 10 ms frame size being preferred over the 20 ms frame, although tuning for the 20 ms frame size might improve matters.

At the 3% rate, the subjective performance of the 10 ms and 5 ms frame sizes was observed to be similar. The segmental SNR decrease for the 5 ms frame possibly corresponds to a small subjective degradation, but it is also likely that tuning for the 5 ms frame size, rather than 10 ms, will negate this effect.

In summary, the issue of frame size does appear to be highly significant, and different frame sizes really dictate different tunings of the basic frame erasure strategy presented in this chapter. However, possible changes to the basic approach may need to be considered for radically different frame sizes, such as a 2 ms frame size that appears to be receiving consideration for some PCS applications.

8.10 Further Work

The results of the studies carried out at AT&T were promising enough to effectively solve the problem of frame erasures within LD-CELP. At least the MOS results mentioned in the previous section appear to confirm this for the 10 ms frame size and a 3% frame erasure rate. However, as with most research, and especially with most short duration research projects, many opportunities for future work were identified during the course of the project. Some of these are mentioned in this section.

Codebook redesign for the system with encoder changes is an obvious option, as previous experience at AT&T has shown that there is potential to recover a significant proportion of the lost clean channel performance. Of course the prospects for practical use of the system with encoder changes is limited by the high performance of the system with decoder changes only, and the desire to retain standard compatibility.

Based on the observation that perceptually severe distortions are highly correlated with frame errors at the start of talk spurts, or where there are significant changes in filter characteristics, it makes sense to further consider 'levelling-out' the frame information content. Ideally we wish to reduce large changes significantly, but not touch small changes, in order to attempt to maintain the clean channel performance. This suggests a non-linear type of approach to predictor updates, and unfortunately any modification such as this brings up questions of yet another parameter that must be synchronized after errors.

Further investigations of more practical aspects of PCS like systems should be made. Original assumptions on direct G.728 usage in wireless networks appear a little dubious from the technical perspective, especially when also considering the problem of computational complexity. However, the political considerations relevant indicate that LD-CELP is favoured for PCS and FPLMTS type applications, related to the fact that G.728 is a well established standard. On the technical side, the use of a low delay coder with an air-interface frame size of 10 ms would need to be considered carefully. Some groups appear to be looking at frame sizes for low-tier PCS applications of as low as 2 ms. The effect of these considerations should be taken into account.

Again ignoring the political benefits of standard compatibility, and focusing on technical (and perhaps somewhat academic) research issues, concepts from combined source and channel coding should be considered for any practical system. An elementary form of combined source/channel coding is the standard process of channel coding error detection/error correction, taking account of different coder output bit sensitivities. Much more advanced (and perhaps more academic) issues involve consideration of spread-spectrum communications, and 'embedded' coding approaches. An example would be the possible redesign of the LD-CELP codebook to reflect coarse and fine detail, which might be useful for both a form of robust multi-resolution transmission over the wireless channel, and simplified codebook search to reduce complexity in the mobile unit. Of course a major challenge here would be to also maintain the coder performance.

One very important consideration from the results of this research is that very high frame erasure rates are able to be accommodated. This fact together with the fact that some frames lead to very small performance degradations, would imply that frame dropping in the encoder should be considered, as a possible means of increasing system capacity in a CDMA system. This would appear to be a fairly logical extension to usual approaches of VAD (Voice Activity Detection) and DTX (Discontinuous Transmission)[8]. The fact that the encoder performs the frame dropping allows it to choose which frames to drop, and allows it to adjust for the missing frame in a similar way to the decoder, hopefully resulting in very little output degradation.

Remark 8.9 Based on the high level of robustness of LD-CELP shown by the frame erasure work contained in the AT&T March 1994 ITU-T contribution[12], novel 12 kbps and 8 kbps variants of LD-CELP are proposed within Appendix F. Within this appendix, a frame dropping type of strategy is considered as a means of decreasing the bit rate.[9] Normally it would be unusual to include this work in an appendix rather than in the body of the thesis, but it is done here mainly to avoid confusion with the rest of the research in this chapter which was undertaken whilst at AT&T.

Another possible consideration for further work is the exploitation of the different frame sensitivities for variable error protection. Again, this might be something that

[8] Techniques of VAD and DTX are also used within TDMA systems, mainly to reduce battery power drain, as spectrum efficiency is not really improved as it is in CDMA systems.

[9] The work in Appendix F is noted as being only one simple embodiment of the general frame dropping approach discussed in the paragraph above.

fits well with a variable rate CDMA system. Also, the most obvious application of the scheme for PCS, would appear to be in a CDMA system. Residual bit errors can be made very low, and as the bit sensitivities are fairly close (almost equal), mostly error detection would be required. Thus a system with a gross data bit rate of not much more than 16 kbps may be possible. With VAD and DTX, a low rate place-holder would conceivably be used.

8.11 Chapter Conclusion

Perceptually severe output speech degradations occur when G.728 LD-CELP is used directly in wireless communications, with no frame erasure detection or recovery strategy. However, intelligible speech output is produced, at least for small frame erasure rates, demonstrating the inherent robustness of LD-CELP.

With frame erasure detection, and either pre-processing of the decoder input bit stream, or minor modifications to the G.728 decoder to allow zeroing of the excitation signal during frame erasures, the performance for small frame erasure rates is (surprisingly) good. This, again, is related to the inherent robustness of G.728.

The use of some minimal modifications to the LD-CELP decoder, such as the use of residual redundancy to perform excitation extrapolation, significantly improves performance, and makes LD-CELP feasible at a frame erasure rate of as high as 10%. Importantly, all the perceptually severe (painful) output disturbances are eliminated, even for the high frame erasure rates, and the modifications come at no net computation increase.

Kalman filtering to account for uncertainties in the excitation sequence during frame erasure, appears to reduce some distortions, and has been observed to provide some SNR improvements. However, the increase in complexity and the fact that it has not yet been shown to provide a significant subjective performance improvement, means that it does not currently form part of the proposed LD-CELP frame erasure solution. There are many possibilities for further research aimed at improving the Kalman filter performance for this application, and the Kalman filter may yet earn its place as an integral component of similar systems in the future.

Simple changes to the encoder assist in increasing the reconvergence rate of the encoder and decoder after a frame erasure, and further improve performance. The system with encoder changes was found to provide significant subjective and MOS improvements at a frame erasure rate of 10%, but at the ITU-T target rate of 3% it was not found to result in any substantial benefit. Hence encoder changes are not part of the AT&T core recommendations to the ITU-T[12].

From AT&T[12], with the proposed decoder changes the MOS degradation at a three percent frame erasure rate was 0.13 to 0.2. This is well within the ITU-T target MOS degradation of less than 0.5. At a 10% frame erasure rate the MOS degradation was up to 1.6 without encoder changes, but with encoder changes this was reduced considerably to around 0.9, which is acceptable considering the high error rate.

LD-CELP is thus seen to be possible for wireless communications, by giving reasonable output speech quality even for high frame erasure rates. Practical use of the scheme in wireless communications systems such as PCS or FPLMTS may be possible, but depends on many other issues, both technical and political.

Chapter 9

AC-ADPCM Resynchronization and Combined Source/Channel Coding

9.1 Chapter Motivation

This chapter considers the issues of resynchronization and combined source/channel coding from both the perspective of the AC-ADPCM system, and a more general or philosophical perspective. No concrete research results are presented, and most of the chapter is significantly 'blue sky' in nature.

As the AC-ADPCM system was introduced (in Chapter 4) it was noted that the problem of resynchronization from transmission errors is significant. Obviously for any communications or storage purpose the occurrence of a single bit error resulting in complete loss of synchronization between the encoder and decoder is unacceptable. With Arithmetic Coding this situation is entirely possible, and steps must be taken to ensure resynchronization.

This chapter discusses both frame based resynchronization techniques, and more sample oriented approaches. In general resynchronization is guaranteed by the insertion of redundancy (extra bits) in the transmitted bit stream, however, this is not always true. It is seen that in some cases resynchronization performance can be dramatically improved without the insertion of any additional redundancy. A simple example of this is the use of Gray coding.

9.2 Introduction

Variable bit rate ADPCM using Arithmetic Coding has been shown to provide high quality speech for low computational complexity at low bit rates (Chapter 6). One problem that must be tackled if we are to use the coder in any practical applications is the issue of robustness to bit errors.

Bit errors can be assumed to occur in all digital storage or transmission media. Of course the probabilities of these bit errors varies dramatically from one situation to the next. For magnetic disk storage media, bit errors can generally be assumed to occur with probabilities of 10^{-12} or less, while with some mobile radio channels, the probabilities could be higher than 10^{-2}. To cover such a range of scenarios is obviously very difficult, however within this chapter we discuss general concepts that should provide reasonable performance across the channel bit error spectrum.

We consider two significantly different approaches to the resynchronization problem. The first is the introduction of frame based resynchronization. This approach is suitable for use in systems with reasonably low probabilities of bit errors, or systems where whole frames may be lost. The next approach we consider is more applicable to higher bit error rates, where the redundancy must realistically be distributed throughout the bit stream in some systematic fashion.

9.3 Separation Principle

Due to stability analyses (such as that contained in Chapter 3), we know that standard ADPCM systems subjected to some random error disturbance will resynchronize to track the encoder after some finite amount of time. Backwards adaptation and adaptive quantization would appear to complicate this process, and make theoretical analysis extremely difficult. However, considerable practical experience suggests that the disturbances are filtered out over time, and leakage factors are often incorporated to aid reconvergence.

Both standard fixed rate ADPCM such as CCITT Recommendation G.721 32 kbps ADPCM, and CCITT G.728 LD-CELP are backwards adaptive coding systems that have reasonable error recovery performance. Largely this is due to the fact that after a period of bit errors, a long enough period of correctly transmitted coder excitation will result in the decoder 'forgetting' previous discrepancies with the encoder.

Due to the similarity between the backwards adaptive AC-ADPCM system and LD-CELP, we expect rapid recovery from initial error conditions after the excitation has recovered from any effects of bit transmission errors.

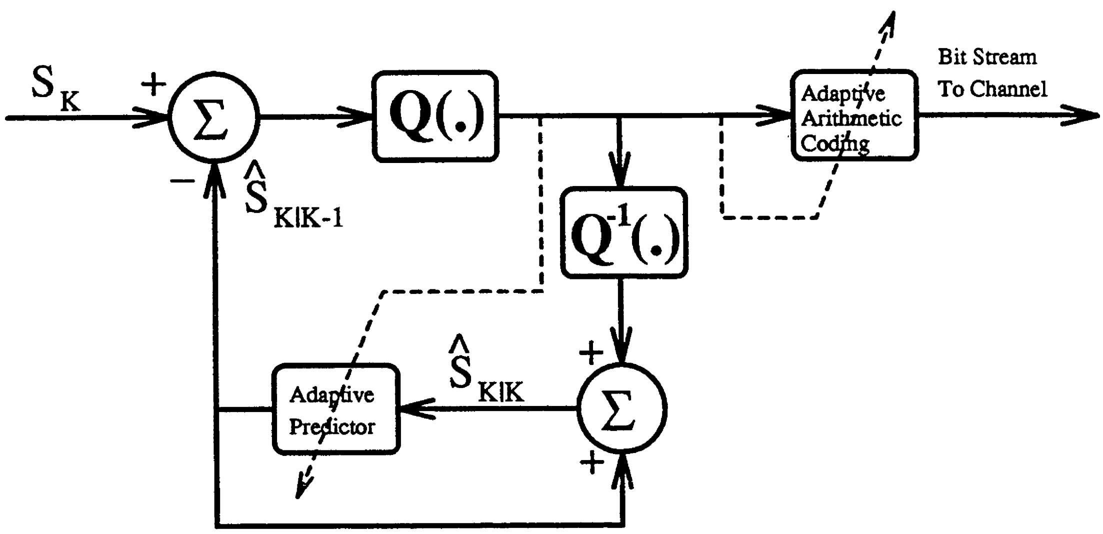

Figure 9.1: Arithmetic Coding ADPCM Encoder Block Diagram

The AC-ADPCM encoder is shown in Figure 9.1. Ignoring the Arithmetic Coding block, it is clear that the rest of the system is simply a standard fixed rate (high rate) ADPCM system. Reasonably well known techniques exist to improve the resynchronization properties of such a coding system. The Kalman filter has been found useful for dealing with measurement noises (channel errors) that are nicely behaved, such as white noise (Chapter 5). In Chapter 7 an attempt was made to apply the Kalman filter to assist in recovery from transmission errors in LD-CELP. One of the reasons for the failure of the Kalman filter can be assumed to be the fact that the measurement noise introduced by a bit error in LD-CELP is not an easily modelled noise.

Other techniques that have been found to assist with resynchronization are Bandwidth Expansion and Spectral Smoothing[33]. For recovery from frame transmission errors, frame erasure strategies such as that considered in Chapter 8 become important.

For coders such as ADPCM or LD-CELP, it is possible to say that the excitation immediately after a period of bit errors is correct, while this is not the case with AC-ADPCM. A transmission bit error may imply that the Arithmetic Coding decoder produces incorrect excitation for a significant amount of time after the bit error (possibly infinite). This incorrect excitation will result in seriously degraded output speech quality for a long period.

Remark 9.1 Note that a transmission error for standard ADPCM or LD-CELP that results in loss of bit synchronization between encoder and decoder will also result in seriously degraded output speech quality for a possibly infinite time period.

We observe that the ADPCM part of the AC-ADPCM system shown in Figure 9.1 should provide reasonable performance in the presence of isolated errors in the excitation sequence. Hence we can limit immediate attention to ensuring that bit errors input to the Arithmetic Coding block only produce isolated excitation errors, at the output of the Arithmetic Coding decoder. Although the whole system will need to be considered at a later stage, this separation of concerns should greatly simplify our analysis.

The separation principle discussed above can be loosely associated with the usual separation principle of source and channel coding. In the strictest sense, it can be assumed that the channel coding deals completely with recovery from channel impairments, such that the channel coding output fed to the source decoder is exactly the source encoder output. (A number of well known channel coding approaches are discussed in Appendix D.)

In practice, due to limits on delays and implementation cost, channel coding is never perfect. Hence the source decoder must take account of the fact that it is fed erroneous source encoder output. Likewise we might imagine that even once we have ensured reconvergence of the Arithmetic Coding block, some additional account may need to be taken of the nature of the excitation errors fed to the ADPCM part of the system. Thus this separation principle would appear useful, but must be used with caution.

9.4 Frame Based Resynchronization

For low probabilities of bit errors, we usually do not wish to introduce too much redundancy into the bit stream. For very low error rates, we may also be able to tolerate fairly long output speech disturbances. For example, a bit error rate of around 10^{-12} implies that we will obtain one bit error for about every 4 years of recorded or transmitted speech at an 8 kbps rate. For this situation, we may not be overly concerned if we loose several seconds of speech.

For higher frame error rates, a frame recovery strategy such as that discussed in

Chapter 8 may be of significant use with the AC-ADPCM system. Hence in order to ensure adequate reconvergence of AC-ADPCM, at the start of each frame the internal states of the Arithmetic Coding block would need to be reset, and the variance value used for probability calculation transmitted at the start of the frame. For the tabular Arithmetic Coding approach, 16 tables or less would be required, and thus the overhead in transmitting the variance would be relatively minor.

On a first inspection it might then seem that the problem of frame based resynchronization is solved. However, the variable bit rate output from the AC-ADPCM system complicates the process. For standard fixed rate systems, the fact that a certain number of speech samples is represented by a frame of a fixed number of bits means that the number of samples missing during a frame erasure is known precisely. For efficient frame reconstruction approaches, it is quite important to know the number of samples lost in a frame.

For the application of AC-ADPCM to a situation where a frame consisting of a fixed number of bits is used, the loss of a frame implies that the decoder has no knowledge of how many samples were transmitted in that frame. Hence a time stamping approach would be required. Depending on the application, it may not be important to know exactly how many samples were lost, allowing some saving in overhead. It would appear that 16 bits would be completely adequate for transmission of this time stamp, and for most applications it could be expected that many of these bits would not be required.

Remark 9.2 Unfortunately the above consideration lacks any specific detail, but in the absence of a particular application the general discussion suffices. Chapter 10 covers some possible applications for the AC-ADPCM scheme, but even here the details of the particular applications are not sufficiently well formed to allow a practical frame based resynchronization approach to be integrated with the basic AC-ADPCM system. The purpose of this chapter is thus to outline some possibilities, and show that the use of Arithmetic Coding does not imply that the AC-ADPCM system can only be used for error-free channels.

A problem with the use of variable rate entropy coding for transmission is that recovery from frame errors is compounded by the fact that equal length frames of output bits correspond to different length strings of input symbols. The 'time stamp' discussed above is designed to overcome this problem.

For a channel where variable bit length frames are possible, the resynchronization problem is simplified by retaining frames corresponding to an equal number of source symbols. Spread spectrum communications systems such as the Qualcomm CDMA system mentioned in Appendix D have the capability of providing a variable rate channel that can be used to transmit efficiently variable rate frames.

9.5 Uniform Redundancy Distribution

Arithmetic Coding is a practically optimal entropy coding scheme, and hence suffers from the fact that there is no redundancy present in the output bit stream that can be used to assist with resynchronization. Of course this is a significant advantage of Arithmetic Coding, but is also the reason why a single bit error has the potential to completely destroy the rest of the decoded symbol stream.

Fixed rate codes such as a simple binary code do not allow any propagation of bit errors. However, for non-uniform symbol distributions, fixed rate codes imply large overhead costs in terms of transmission of redundancy. Hence, for efficiency reasons, we need to consider the use of variable rate codes. (Without the use of variable rate coding, the AC-ADPCM system would consist of just the ADPCM section shown in Figure 9.1 without adaptive quantization, and result in an output bit rate of 40 or 48 kbps for toll-quality speech coding.)

Arithmetic Coding is often viewed as not being particularly useful when bit errors are present. We believe that Arithmetic Coding should not be ignored in these situations, as the bit error problem can be tackled through the insertion of redundancy to the bit stream, in a controlled manner. The use of Huffman coding is preferred by many over Arithmetic Coding due to its resynchronization properties. However, we note that there is *a priori* no reason to believe that the redundancy left in the Huffman code stream is optimal for resynchronization purposes, or that it is the right amount of redundancy considering the trade-off with the cost and probability of a bit error.

For this reason we consider the reinsertion of redundancy into the Arithmetic Coding bit stream. We further note that Huffman coding is actually subsumed by Arithmetic Coding. This can be seen by simply performing Arithmetic Coding on the probabilities, that have been modified to restrict them to all be inverse powers of two.

Some recent work deals with the topic of Block Arithmetic Coding[21]. This approach appears to actually reduce the computational resources required for the entropy coding, and is capable of improving the transmission error performance by limiting errors to the block in question. The use of adaptive Arithmetic Coding complicates this, but it may still be possible to provide good error performance through the use of the block approach.

Other work on the topic of obtaining robust entropy codes includes the paper by Fowler and Ahalt[65], where an entropy biased vector quantization scheme is used to avoid the entropy coding stage, and the consequent resynchronization problems. This work is based on DPCM coding of images, but would appear to provide a link between entropy coded DPCM systems and CELP systems. Of course part of the motivation in dealing with an entropy coded ADPCM system was to avoid the computational complexity of CELP. Hence an approach to overcoming resynchronization problems that results in a CELP system, rather than an ADPCM system is not desirable from a computational perspective.

9.6 Resynchronization without Additional Redundancy

Error recovery in the form of obtaining perfect information even in the presence of channel bit errors requires additional redundancy to recover from these errors. Convolutional coding is often used in conjunction with Viterbi decoding to insert and exploit this additional redundancy (Appendix D).

For speech coding applications, perfect reconstruction of the encoder output bit stream at the decoder is often not a critical requirement. In cases where the encoder output is received with bit errors at the decoder, significant improvements to resynchronization can be achieved with the use of some basic approaches, such as Gray coding. CCITT Recommendation G.728 LD-CELP incorporates pseudo Gray coding of the 7 bit shape vector indices. The effect of this can alternately be viewed as limiting the divergence due to a bit transmission error, or improving the resynchronization performance (in the broad context).

This resynchronization improvement without redundancy can be examined in conjunction with variable rate entropy codes. Ferguson and Rabinowitz[58] examine the requirements for a Huffman code to self-synchronize without the addition of redundancy. However, to the best knowledge of the author, this type of work has not been extended to Arithmetic Coding, and hence exists as a possible area for future research.

9.7 Chapter Conclusion

This chapter has not presented the results of any research. However, it has served to outline a number of important issues that may need to be considered in conjunction with the application of the AC-ADPCM system.

Frame based resynchronization appears relatively straight-forward in general, yet is quite applications dependent.

Increasing the robustness of the AC-ADPCM system in the presence of bit errors is also potentially a very important problem. The most obvious way to tackle this problem is to separate the Arithmetic Coding resynchronization problem from the rest of the system, and deal with this first.

Due to the highly adaptive and non-linear nature of the Arithmetic Coding process, an effective solution to the resynchronization problem does not appear simple. However, the flexibility afforded by the speech coding application may allow a practical approach to be developed through significant further research.

Chapter 10

AC-ADPCM Applications

10.1 Chapter Overview

A significant part of this thesis has involved the introduction of a variable rate ADPCM speech coding system using Arithmetic Coding and Kalman Filtering techniques (Chapters 4 and 6). The original motivation for this work was somewhat academic in nature. However, it has become apparent that the proposed AC-ADPCM system may have some significant advantages over other fixed rate speech coding systems.

In this chapter we consider a number of possibilities for application of the AC-ADPCM system. The chapter is discussion oriented in nature, and probably leaves more questions open than it answers. No new research results are presented here, and some of the applications discussed could perhaps easily be labelled as 'blue sky'.

A number of the applications are reasonably direct, and some immediate simulations should be performed to assess potential. This work is far beyond the scope of this thesis and currently simply exists as an avenue for future research. One of the aims of this chapter is to utilise the insight gained from previous investigations of the AC-ADPCM approach to attempt to highlight key aspects of the system in relation to the applications.

In general there are a number of important advantages a variable rate system can provide over fixed rate coding systems. Of course there is the advantage of exploiting the bursty or variable information rate nature of the source, but an advantage that is possibly even more important is the flexibility afforded by the variable rate system. Examples of the sort of flexibility that might be important include the ability to provide adequate performance across a range of hostile coding environments (such as high levels of background noise), or the ability to adapt to user requirements on bit rate/cost considerations (quality of service choices).

With the AC-ADPCM system these gains come at the expense of the variable bit rate[1], and resynchronization problems. The scheme thus has significant advantages and disadvantages, which may make it an ideal proposition for some applications, while not being worthy of much consideration for others. The applications discussed below are some of the more noteworthy ones as far as AC-ADPCM is concerned. However, the list is not claimed to be exhaustive in any sense.

10.2 Speech Storage

One application where variable bit rates are not a significant disadvantage, and bit error rates are extremely low, is that of speech storage. Speech storage may be required for many purposes, from use in computer games to data logging of calls at banks or airports. The common requirements of such systems can usually be assumed to be minimising the average bit rate, whilst keeping computational complexity, and hence the system cost, down. These requirements fit well with the AC-ADPCM system capabilities.

Other possible requirements include the capability to vary quality of the output speech on demand from either the user or the system, and being able to scan through the stored speech at much quicker than real time. These issues are also capabilities of AC-ADPCM.

Pseudo random access to the entropy coded data is easily able to be accommodated through the use of an indexing procedure, similar to that discussed in [100].

With the speech storage application, coding delay is not usually considered to be a significant issue. Hence the use of a forwards adaptive system would appear to make sense. This is easily incorporated into the basic AC-ADPCM structure, turning it into an APC (Adaptive Predictive Coder) system. All forwards adaptive parameters may be easily accommodated via the arithmetic coding approach. With the use of pitch prediction we could also expect a substantial performance gain for the forwards adaptive system.

It is important to note that many practical speech coding systems perform voice classification, and code differently according to the current voice class. Again this can be accommodated within the AC-ADPCM (or AC-APC) system. In Chapter 4 it was

[1]The use of a variable bit rate often makes the task of other components of the communications system more complicated.

noted that the basic AC-ADPCM system performed poorly with unvoiced speech at rates of below 8 kbps. It is thus natural to investigate these options, but is not central to this thesis, and hence has been left as a possible topic for future work.

10.3 Digital Answering Machines

A specific application of speech storage technology is in the domestic answering machine market. Digital answering machines have no tapes, hence are more reliable, and are easily able to provide advanced features, such as time-stamping, and message selection. Digital machines have begun to appear on the overseas markets over the last two years. In these systems, the important considerations are the trade-offs between speech quality, output bit rate, and the computational requirements. It appears that as far as these three parameters are concerned, the AC-ADPCM (or AC-APC) system may provide a significant advantage over other systems.

In order to assess potential for this application, studies should be performed with speech sampled at 6.4 kHz, as there is unlikely to be a need to sample at 8 kHz. The above three parameters will then need to be compared to other systems on the market. Other important considerations for this application are input noise filtering, which is able to be performed through the use of the Kalman Filter, and input level tracking, which is a fairly common practical requirement.

Very closely aligned to the above domestic application is the office answering machine market. There are two major differences within this corner of the market. The first is the fact that the systems are quite often multi-user systems, and the second is that in almost all modern offices, networks of computers are present. Functions of coding and distributing messages to various employees can easily be carried out with the existing computer resources, with very little additional interface hardware.

Another closely related application is that of office dictation equipment. Here it is conceivable that an IC RAM card with the compressed data could be transferred from the recording machine to the secretary's playback machine. Apart from the obvious advantage of eliminating tapes, a consequent increase in battery life is experienced for portable dictation devices, and for the dictaphone application there are a number of functions that the variable bit rate ADPCM approach may be able to provide. The obvious requirement with dictaphones is the possibility of providing rapid playback for scanning. It would appear that to avoid the cost of a much faster chip to perform the playback, the ability to sacrifice some amount of performance for playback speed would be a distinct advantage.

For all the digital speech storage applications mentioned in this section, another useful function which the AC-ADPCM coder can provide is that of an embedded coding approach. This could conceivably be useful to provide 'graceful' performance degradation, or a 'soft capacity', when the memory capacity of the storage media has been reached. Thus, more speech can be stored at a reduced resolution, by throwing away some of the fine detail structure of the previously stored speech.

Again we note that simulations should be performed for the purpose of further assessment, but that this is a side issue to the central thesis thrust.

10.4 Voice Mail/Packet Voice

Voice mail is a closely related application to that of office answering machines. However, with the current market penetration of office computer networks, and trends for the future including the 'Information Super-Highway' infrastructure, a closer link between existing telephone and computer network systems appears to be of substantial interest.

A voice mail system that extends email, rather than being centered solely around the telephone system would appear to be an obvious application. If some predictions of computer network proliferation are correct there may be substantial demand for this type of system within only a small number of years.[2] Current network penetration would appear to imply that several niche application areas would immediately exist, such as within the university and research environments.

Again, the large amount of flexibility present with the AC-ADPCM system suggests that consideration should be given to this application. It appears that to get a skeleton system operational on a small network test-bed, minimal development effort would be required.

A logical extension to voice mail would be that of two-way packet voice communication over a computer network. The market here would appear to be significant especially if considered as an extension to current UNIX 'talk' sessions. However, more development would be required, especially for network protocols that introduce large transmission delays.[3]

[2] A more conservative approach is that substantial changes to the telecommunications industry will occur, but most likely over a period of many years, due to technical, social, and legal/political issues.

[3] Due to the latency that exists, such a system would be unlikely to compete in any way with the

10.5 ATM Voice Compression

ATM networks provide a variable rate capability, and hence it would appear that a system such as AC-ADPCM would be a natural choice for speech coding over such a network. A large amount of consideration has been given to the topic of ATM speech compression by many authors[117, 133], and variable rate coding, and the ability to utilise packet priorities efficiently are important issues. AC-ADPCM has the ability to utilise packet priorities, so that more important information is placed in high priority, and hopefully high reliability, packets.

Careful consideration needs to be given to resynchronization after packet loss within such a network. Other important issues are whether speech coding within ATM networks is possible or desirable. ATM is designed to facilitate broadband communications, and hence it is possible that speech may only be an insignificant component of the total. Delay requirements may also mean that it is not possible to fill an entire packet with speech at 64 kbps, so there is little point to compression.[4]

Chapter 9 discussed the possibility of overcoming the resynchronization problem of the AC-ADPCM system through the use of a frame based strategy. This is particularly applicable to the ATM application, where frame losses are probably the most common form of error, with frame loss rates of 10^{-7} for high priority frames, and 10^{-4} for low priority frames being assumed.

10.6 Security Applications

There are several applications where audio monitoring is required for security purposes. Compact representation of the audio data is often a requirement, whether to save storage space, or to aid remote monitoring.

Again the flexibility afforded via the AC-ADPCM approach is of substantial significance. Difficult monitoring environments of multiple talkers or background noise can be accommodated simply, with a corresponding increase in the average bit rate. Another possible requirement in security monitoring is the use of stereo pickup. To efficiently exploit the redundancy in the stereo signal, some modifications would need to be considered for AC-ADPCM.

Other security applications require the coded speech to be secure in the sense of

telephone system, and would mainly be viewed as an augmentation of the UNIX 'talk' protocol.

[4] Issues such as this are cause for debate well outside the scope of this thesis.

only being understandable to those parties authorised to receive. AC-ADPCM may have a number of advantages for this application, closely related to the transmission error problems previously noted.

Again embedded coding approaches and the ability to scan quickly through recorded speech are important issues.

10.7 Wideband Speech and Audio

Wideband speech and audio storage are other areas with many possible applications. Although the AC-ADPCM system has initially been developed for narrow band speech, a well used approach to wideband audio is separate coding of signal sub-bands. The AC-ADPCM system could be applied in this way, as it is a simple matter to integrate the separate Arithmetic Coding streams into the one output bit stream. This approach may provide some significant advantages over more traditional wideband audio coding approaches, but this must be determined by future research on the topic.

One specific application to consider for this is the requirement of producing half an hour of delay in broadcasts common to the states of NSW/Victoria, and South Australia. Another is the requirement of a broadcast group such as the ABC (Australian Broadcasting Corporation) to store and edit large amounts of audio. Currently information available to the author about these applications is fairly limited, and hence an assessment of potential is difficult, but it would appear that both applications are worth future consideration.

Another possibility with wideband audio is even more 'blue sky' than those above. Techniques already in use in some audio CD systems that provide increased robustness to decoding errors induced by such things as scratches and jolts, also have the capability of providing a variable bit rate storage medium. Hence variable rate CD audio coding is theoretically possible. Of course, practical feasibility is heavily dependent on the cost–performance compromise, and the reluctance to modify standards. Also, it is not necessarily in the best interests of the music industry to include more music information on a CD.

10.8 IC Music Storage

An application which appears realistic with the availability of cheap IC storage elements, such as RAM memories, is that of IC storage of music. The Sony 'Walkman' type of market would benefit from this application, as the portable unit could be extremely small and robust to jolting, as there would be no moving parts. The absence of moving parts may also increase battery life significantly. A number of system configuration possibilities exist with either a hard programmed music IC being purchased, or a rewritable IC card being programmed by a CD player or similar. From the music industry perspective a big consideration is the potential for copyright violations.

As with tape players, it is possible that the rewritable IC cards could be of the same format for personal dictaphones, as for the music application.

Obviously major modifications to the basic AC-ADPCM speech coding system would be required for this music application. However, the simplicity and flexibility of the Arithmetic Coding approach may prove useful.

10.9 Mobile Communications

Due to its low complexity, low average output bit rate, and high flexibility, AC-ADPCM would appear to be a worthwhile consideration for mobile communications. Unfortunately, the variable bit rate is a problem with most current mobile communications networks. However, future networks, such as CDMA spread spectrum systems, are likely to offer a variable rate capability. This makes AC-ADPCM attractive for further investigation.

Existing variable rate speech coding systems, such as QCELP[51, 66], utilize a distinct number of output frame rates. AC-ADPCM produces an instantaneously variable rate output, and as such does not produce a fixed output frame rate. However, there is some potential for the 'continuously' variable output bit rate to be handled efficiently within a CDMA system via the use of a number of basic transmission frame rates. The next highest rate above the coder output bit rate would be used for transmission, and spectral efficiency (or at least battery life) may be increased by discontinuing transmission after the data bits have been sent. Of course, other system components such as convolutional encoding and interleaving would complicate matters as far as this 'discontinuous' transmission is concerned.

The resynchronization problem will need to be addressed before the mobile application is possible. The frame synchronization techniques mentioned for other applications above, are unlikely to be adequate for use with the severe variations in the mobile radio channel. Further research will hopefully solve this problem (Chapter 9).

With the mobile communications application, the flexibility of AC-ADPCM could be utilised to advantage for handling difficult situations such as multiple talkers and conference calls, music coding, and background noise. Also, both user and network controls over the bit rate versus quality trade-off may provide significant advantages, such as that of 'soft capacity' in a mobile cellular system.

Another worthwhile consideration with mobile communications may be the potential for variation in computing power over a speech utterance. Modern chips are able to save power by operating at slower speeds when they are not required to operate at full rate. Predictor updates consume large amounts of computation resources, but only need to be performed frequently for some parts of speech where there are large changes in signal statistics. By performing updates only when required, it may be possible to save significant power.[5] This type of approach is not easily attainable with systems that don't have the large flexibility of the instantaneously variable rate that is present in AC-ADPCM.

10.10 Chapter Summary

A number of possible applications for the Arithmetic Coding ADPCM (AC-ADPCM) system have been discussed briefly. Some of these appear well suited to the basic coding approach, with possibly significant advantages arising from the inherent flexibility. Other applications appear less direct, but further research could find that at least the basic Arithmetic Coding concept is useful.

The flexibility of the AC-ADPCM system and the relevant computation considerations are extremely important with most applications. This flexibility includes the possibility of a forwards adaptive AC-APC system.

Resynchronization after bit errors is perhaps the largest current problem that needs to be overcome in order for the system to find wide application. Some considerations on this topic have already been mentioned in Chapter 9.

[5] In a very elementary form, it should be clear that filter parameter updates during silence periods are not particularly useful, and can safely be traded-off for battery power savings.

Finally we note that the existence of high quality speech coding systems operating at around the 4 kbps rate does not mean that the AC-ADPCM system is unable to find application. The basic system provides reasonable performance at 8 kbps, with minimal computation requirements, and the potential for a more sophisticated system to provide better performance, whilst still obtaining high computation efficiency, is very good. The AC-ADPCM system flexibility also must not be forgotten.

Chapter 11

Conclusions and Further Research

11.1 Summary of Contributions

This thesis has covered a number of different topics in signal coding. The techniques discussed range from the very theoretical to the more practical in nature. The thesis also contains a significant element of 'blue-sky', or more philosophical type discussion. In point form, the major contributions contained in this thesis are:

- **ADPCM Stability Analysis:** A theoretical analysis of the effect of adaptive quantization on stability of ADPCM was undertaken. The results of this analysis confirm that less adaptive quantization leads to greater stability, and that the maximum rate of quantizer step-size decrease is the important factor for stability considerations. This fact can be viewed as a theoretical justification for the shape of the multiplier curve for Jayant 'One-Word Memory' adaptive quantization. As adaptive quantization is generally required to maintain performance, a stability versus performance trade-off exists.

- **Arithmetic Coding ADPCM:** A novel ADPCM approach to speech coding was proposed, using variable rate entropy coding. The Arithmetic Coding ADPCM system (AC-ADPCM) was found to obtain good speech quality performance for average output bit rates in the range from 16 kbps to 8 kbps. Significant flexibility is obtained with the basic AC-ADPCM approach, which can be exploited for user, system or designer control over parameters such as output speech quality, output bit rate, coder computational complexity, and even coder delay.

- **Kalman Filter Computation Reduction:** The Kalman Filter was discussed in terms of its application to signal coding. Techniques were then presented for computation reduction via a reduced order Riccati equation, and periodic updates of the Riccati equation. It has been observed that significant Kalman filter performance gains are available for some speech coding applications with manageable increase in computational complexity over systems utilizing standard linear predictors.

- **AC-ADPCM with Kalman Filtering:** The AC-ADPCM system introduced in Chapter 4 has been shown to obtain substantial advantages through the use of Kalman filtering techniques. The resulting system with 15th order predictor updated every 80 samples (10 ms) at an average bit rate of approximately 12 kbps has been judged by informal listening tests to be of comparable subjective quality to 16 kbps LD-CELP. For closer comparison with LD-CELP we used a 50th order predictor updated every 20 samples, and in this case the AC-ADPCM performance at an 8 kbps average rate was judged to be equivalent to that of LD-CELP. For the higher complexity AC-ADPCM version the Bark Spectral Distortion (BSD) objective measure also indicated equivalent performance to LD-CELP at the 8 kbps average rate.

- **Kalman Filtering in CELP:** The Kalman filter was used as a tool in an attempt to provide performance improvement in CELP systems by filtering quantization noise. Unfortunately the results of the simulations performed for this thesis were inconclusive, indicating that further work would have to be performed before any significant gain is observed.

- **Frame Erasures in LD-CELP:** The problem of frame erasures in LD-CELP has been studied for applications such as PCS or FPLMTS. The frame erasure approach presented in Chapter 8 (work performed during time spent at AT&T) was shown by AT&T MOS testing to produce between 0.13 and 0.20 points MOS degradation from LD-CELP clean channel performance at a 3% frame erasure rate. This is substantially better than the original CCIR target of less than 0.5 points MOS degradation for 3% frame erasure. Bypassing bit-stream compatibility with the existing CCITT G.728 LD-CELP standard, the use of some minimal encoder modifications allowed very good performance at frame erasure rates of up to 10%. At even higher frame erasure rates the system with encoder changes was found to provide largely intelligible speech, with no perceptually severe output error bursting.

- **12 and 8 kbps LD-CELP with Frame Dropping:** Novel 12 kbps and 8 kbps variants of LD-CELP were proposed where every 4th and every 2nd vector (respectively) was not transmitted. The performance appears to compare poorly to other coders operating at these rates, but may have application due to the minimal alterations to LD-CELP. One possible scenario is for use to provide 'soft-capacity' in a communications network.

- **AC-ADPCM Resynchronization:** The issue of transmission error robustness of the AC-ADPCM system was discussed in Chapter 9. Robustness to frame or packet errors was observed to be possible via the use of a moderate level of redundancy inserted on a frame basis.

11.2 Further Research

Throughout this thesis, many issues have been raised which are potential topics for further research work. The philosophical nature of some parts of this thesis has meant that a significant number of 'blue-sky' possibilities for future work have been discussed in addition to more direct follow-up research. The most important of these identified topics are described below.

- **Stability Analysis:** The stability analysis performed in Chapter 3 provides some useful insight into the problem of adaptive quantization within ADPCM. However, the theorem we presented is a sufficient, but not necessary, condition for stability, due to the assumptions and simplifications required for the theory. Potential thus exists for a more 'elegant' stability theorem.

- **Arithmetic Coding ADPCM:** The AC-ADPCM system presented in Chapter 4 and extended in Chapter 6 with the addition of the Kalman filter, needs more development to suit applications, as discussed in Chapter 10. General development issues would include trying to improve performance at around the 4 kbps rate by the use of techniques such as voice classification and separate treatment of voiced/unvoiced; the use of pitch prediction; delayed decision coding; quantizer step size scaling; and the possible use of non-uniform quantization.

- **AC-ADPCM Resynchronization:** The problem of recovery from transmission bit errors is generic to the pursuit of speech coding. Resynchronization issues for

AC-ADPCM have been discussed in Chapter 9. However, it appears that although recovery from frame or packet errors is relatively simple, adequate recovery from random bit error needs significant further research attention. This is an important general problem in entropy coding, and a practical engineering solution could have far reaching consequences.

- **Kalman Filtering in CELP:** The Kalman filter has been observed to provide significant subjective (and some objective) performance improvement in a number of signal coding applications. For low bit rate systems, quantization noise is relatively high, and it is here that we expect significant performance improvements with the use of the Kalman filter. A worthwhile topic for more research attention is the application of the Kalman filter as an integral part of CELP coding systems. Results of the application of the Kalman filter to CELP within this thesis are inconclusive, possibly due to the elementary stage of the work performed. Intuitively the high levels of quantization noise that exist in low rate CELP systems should provide possibilities for Kalman filter application. However, it may also be true that the CELP approach is able to inherently obtain a proportion of the Kalman filter advantage. Whatever the case, only further research is capable of determining the cause of the apparent failure of the Kalman filter to obtain performance improvements in CELP.

11.3 Conclusions

This thesis has presented a number of signal coding concepts that are particularly useful in the speech coding area. The research has been quite successful in terms of practically oriented engineering goals. In addition, it is believed that some element of academic merit has been exhibited by sections of the work.

Major discussions and conclusions arising from the research undertaken for this thesis are included in the relevant chapters, and a dot-point summary of research contributions is presented above. However, some general conclusions can be drawn here:

- Speech coding, in particular for mobile communications applications, is a fertile area for practical engineering research. Speech coding schemes discussed in this thesis and other existing schemes still appear to leave significant amounts of redundancy in the transmitted signal. We can attempt to further remove this redundancy to improve coding performance by lowering the transmission bit rate, or exploit the transmitted redundancy for error recovery purposes (such as frame erasure or even smoothing of quantization errors – Kalman filtering).

- Although standard fixed rate ADPCM systems tend to provide poor performance at rates of 16 kbps and below, variable rate systems such as the AC-ADPCM system presented in this thesis, are capable of providing very good performance for the same average bit rates. The AC-ADPCM system maintains a low average bit rate by effectively only using a very small bit rate in highly predictable sections of the speech signal, and using a larger instantaneous bit rate when required.

- With the use of Kalman filtering techniques, perceptual weighting, and adaptive postfiltering, the AC-ADPCM performance has been judged by informal listening tests to be equivalent to that of 16 kbps LD-CELP at an average rate of as low as 8.0 kbps (for the high complexity AC-ADPCM system). For the lower complexity AC-ADPCM system, we claim the subjective performance at a 12 kbps average output bit rate is equivalent to that of LD-CELP.

- The performance of the (low complexity version) AC-ADPCM system at an average rate of 8 kbps is (surprisingly) good. The performance of the AC-ADPCM system at 4 kbps is unfortunately fairly poor. However, the use of standard techniques such as pitch prediction, and delayed decision coding, etc., may assist in significantly improving performance at the lower rates.

- Exploiting residual redundancy in LD-CELP to extrapolate in the gain-scaled excitation domain results in impressive performance for frame erasure rates of up to 3%. For frame erasure rates of 10%, some minor modifications to the LD-CELP encoder, that only slightly degrade the clean channel performance, result in high quality output speech. The fact that very high frame erasures are able to be tolerated, and the observation that some frame erasures lead to very small perceptual degradations, suggests the possible use of an encoder frame dropping strategy as a possible extension to Voice Activity Detection (VAD) and Discontinuous Transmission (DTX) techniques.

11.4 Coda

It should be clear by the number of areas indicated for possible future research that to some extent more questions are left open, than are answered in this thesis. Mobile communications brings together many concepts in electrical engineering: from DSP and VLSI techniques, to predictive algorithms, to radio communications, and network topologies. As such, it is a difficult area in which to perform practically useful research without at least some ideas about the broader communications perspective. As this broad perspective can only be obtained from many years of experience in telecommunications, I am extremely grateful to all the researchers I have conversed with over the last three years. I certainly believe I have asked my share of stupid questions and come up with many crazy ideas. Thanks are deserved by all who were willing to listen and respond helpfully.

On a final philosophical note, it would appear that telecommunications is a rapidly moving field of endeavour. Recent developments indicate the increasing globalization of telecommunications. Two examples of this are the Information Super-Highway, and Low Earth Orbit satellite systems. For such projects, politics and engineering are both extremely important ingredients. Projects such as the Information Super-Highway also cast doubts on the adequacy of current legal systems, especially when considering the multi-national nature of the projects. Some engineers appear to have identified the need for upheaval of the legal system, although lawyers seem to be more naturally conservative on the issue. A stable legal system is, to some extent, at the very heart of our society structure. However, the situation where single court actions take ten years or more to reach resolution is alarming when considered in conjunction with the projected rapid changes in telecommunications over the coming decades.

The next ten or twenty years should see telecommunications engineers taking more active roles in both political lobbying and legal issues. It is also set to be a fruitful period for telecommunications research across the entire research spectrum.

Appendix A

Information Theory and Entropy Coding

Arithmetic Coding is a practically optimal entropy coding technique, that has received popular research attention and use since the publication of papers such as that by Rissanen and Langdon, in 1979[150]. Bell, Cleary and Witten[15] describe briefly the origins of Arithmetic Coding, tracing it back to work of Elias in 1963 (unpublished). In this appendix, we provide an overview of Arithmetic Coding, presenting it in terms of a logical progression from elementary fixed rate coding, through the well known variable rate Huffman coding, to Arithmetic Coding, and Quasi-Arithmetic Coding (which will be described below).

Arithmetic Coding is one type of entropy coding, and as such it is important to understand the concept of entropy. Entropy in this context stems from Shannon's Information Theory[160]. Information Theory states that associated with each message in a communications system, there is a measure called the information content of that message, which is inversely related to the probability of occurrence of the message. In units of 'bits', which are not to be confused with bits in other digital senses, the measure is:

$$Information = \log_2 (1/Probability) . \tag{A.1}$$

Intuitively, the information content in units of bits is equal to the number of binary (yes/no) questions that must be answered to transmit the message. For example, a message with probability 0.5 can be transmitted by answering one (yes/no) question, and hence it's information content is 1 bit. Similarly, message probability 0.25 translates to an information of 2 bits.

It is important to note that the above says nothing about the use of coding in the transmission system. For instance, a system with two messages, one of probability 0.75, and the other 0.25, could utilise a coding system such that a single bit is used to transmit each message. However, the information contained in the first message is 0.415 bits, and that of the second message is 2 bits.

Entropy is associated with a message stream, and is the expected information content of a source symbol. Hence it is the probability weighted sum of the message information over all message symbols:

$$Entropy = \sum_{i=1}^{N} P_i \log_2 (1/P_i), \tag{A.2}$$

where P_i is the probability of the ith message symbol, from the symbol alphabet of N message symbols.

For a message stream, the entropy represents a theoretical minimum bound on the average transmission bit rate. The aim of many source coding systems is to approach this entropy bound as closely as possible. However, there are other associated concerns, such as delays, complexity, and transmission error robustness.

Remark A.1 The entropy is defined in terms of the actual message probabilities of the source. In many engineering applications, since the source is usually not fully understood, or due to complexity constraints, we do not have these message probabilities available. However, we are usually able to model the source, at least crudely. Using probabilities obtained from the model, it is often possible, and beneficial, to use 'entropy' calculations. Although, in this case, there is no guarantee that the entropy represents a lower bound.

A.1 Fixed Rate Codes

Fixed rate codes are perhaps the simplest form of binary source coding, with each source symbol being allocated a distinct sequence of a fixed number of binary digits. The number of binary digits, or bits, R, is determined by the total number of source symbols, N, such that $R \geq \log_2 N$, with the obvious requirement that R is an integer.

For many applications, fixed rate codes produce average bit rates that are far above the lower bound given by the entropy. For some situations, it is possible that the fixed rate code will perform at an average rate equal to the entropy. The requirements for this are that the source symbols have uniform probability distribution, and that the number of symbols is such that $N = 2^R$.

Desirable properties of fixed rate codes are related to simplicity. Transmission error performance is usually very good, with bit-flip errors giving rise to only single symbol errors. Even the effect of these single symbol errors can be minimised, through the use of techniques such as Gray coding[19]. For symbol distributions where $N < 2^R$, the redundancy that exists in the transmitted bit stream due to the fact that not all of the 2^R bit sequences are utilised, can be exploited to provide some degree of error detection or error correction. This is closely associated with the concept of Hamming codes and Hamming distances[80]. The issue of error detection and error correction is generally assumed to be the domain of channel coding, and will be considered in a little more detail in Appendix D.

A.2 Variable Rate Codes – Huffman Coding

Variable rate codes allow different numbers of bits to be used for different symbols. In this way, it may be possible to obtain an average code rate that is much closer to the entropy than that given by a fixed rate code. We would expect a variable rate code to give a significant advantage over the fixed rate where there is a substantial departure in source statistics from those that are required for the fixed rate code to obtain the entropy bound. Hence, where the number of source symbols, N, is appreciably less than 2^R, or where the source symbol probability distribution departs substantially from a uniform distribution.

Huffman coding[89] is an example of a variable rate code that uses a binary code generation tree on the symbol alphabet to assign bit codes to each source symbol. Thus source symbols can have different integral numbers of bits. A property of Huffman codes is that the output bit stream is uniquely decodable, as no valid code is a prefix to any other valid code. This is certainly useful for communication systems.

Bit transmission errors in a Huffman coding system have the potential to result in a different number of source symbols being decoded than that which were encoded. This 'disadvantage' can be considered to be the price that must be paid for the bit rate advantage that is afforded through the variable rate. Some Huffman codes automatically resynchronize[58], and Huffman coding can thus be practically useful in some applications where bit transmission errors are present. For many applications, the 'jitter' that occurs due to differing numbers of source symbols being transmitted and received is not of significant consequence.

The expected average bit rate of a Huffman coded source is equal to the entropy of the source if all source symbol probabilities are negative powers of 2. Where this is not the case, there still may be significant advantages in average bit rate with Huffman coding over a fixed rate code. The fact that each source symbol must be coded with an integral number of bits can be a significant disadvantage, especially in situations where one symbol has probability very close to one, such that the information content of that symbol is only a small fraction of a bit.

A.3 Block Huffman Coding

Since it is initially difficult to see how it is possible to code a symbol directly using fractions of a bit, one approach is to block a number of symbols together to form a 'super-symbol'. The alphabet of super-symbols formed from all possible combinations of the symbols is then treated as the new alphabet, used for Huffman coding.

It is possible to show that increasing the block size leads to a monotonic decrease in the average bit rate, towards the lower entropy bound. An infinite block size corresponds to an average rate equal to the entropy, and dependent on the symbol probability distributions, it may be possible to practically reach the entropy bound with a finite block size. For more on this topic, refer to a text such as Carlson[27].

The disadvantage to the use of large block sizes is that the computation required for code generation increases, and transmission delays increase due to the necessity of buffering the input symbol stream. Also, problems with bit transmission errors are generally increased. However, since we are removing more redundancy from the transmitted bit stream, problems of this sort are, to a certain extent, a natural consequence.

Although transmission error effects are not to be ignored, an important fundamental concept in Information Theory is that if a communications system is formed using separately optimised source coding and channel coding units, the entire system is optimal[160]. Unfortunately, the assumptions for this concept to be true are strictly violated in almost every practical situation, due largely to restrictions on complexity and delay, and the variable nature of the channel impairments. Even having said this, we note that separation of source and channel coding is an extremely useful approach, found to varying degrees in almost all communications systems. Channel coding will be discussed in Appendix D.

With block Huffman coding, it is therefore theoretically possible, although practically restrictive, to reach the entropy bound for general probability distributions. Unfortunately, in addition to the other problems mentioned above, adaptive symbol probability distributions also cause significant trouble with block Huffman coding. In this situation, adaptation would usually only be permissible between blocks, requiring the block size to be small, in direct conflict with the requirement for a large block size to achieve performance close to the entropy bound.

A.4 Practical Entropy Coding – Arithmetic Coding

Arithmetic coding (AC) can be thought of as using an infinite length block strategy on the input symbol stream, similar to block Huffman coding, and hence is able to achieve the entropy bound. However, AC differs from block Huffman coding in a way that can be explained intuitively by considering setting the first bits of the 'codeword' by using information contained in only the first few source symbols. Hence these bits of the 'codeword' can be transmitted before the end of the (infinite) source stream is reached, and the first few symbols can be decoded without first waiting for the whole stream to be encoded. For a good general description of Arithmetic Coding, the paper by Bell et. al.[15], should be useful.

Performing the coding in this way ensures the delay from a symbol being seen at the encoder, and the same symbol being received at the decoder is finite (except in certain pathological situations). For typical situations, the delay incurred by the encoding and decoding process is a few samples at most. Also, the coding complexity is greatly reduced over that of the infinite block length Huffman code. AC is thus seen to be a form of entropy coding, that is optimal in the average bit rate (entropy) sense, and is also practically useful.

As each symbol can be decoded in sequence, without depending in any way on the following symbols, full backwards adaptation approaches can be used for estimation of the source symbol probabilities. It is this fact that allows AC to be used so successfully for adaptive text compression[14, 60, 131, 139, 146, 188], and image compression[86, 87], where source symbol statistics change very rapidly. The use of adaptive source distributions with Huffman coding is also possible (although it may consume significant computation resources), but the adaptation must be limited for block Huffman coding.

Unfortunately AC is still not 'practical' enough for all applications. The variable output bit rate can cause some concern, as almost all digital communications links are based on fixed bit rates. The encoding/decoding delays are effectively unbounded, and could be extremely long for some source distributions. Bit transmission errors also cause substantial concern, as a single bit error has the potential to render the rest of the message stream meaningless.

Remark A.2 Huffman coding is also not guaranteed of resynchronization, and a bit transmission error in Arithmetic Coding is a slight improvement on infinite block-length Huffman coding, where the entire message stream can be rendered meaningless. However, it is generally believed that the error recovery performance of Arithmetic Coding is inferior to that of Huffman coding.

It is important to understand that all of the above 'disadvantages' of AC have resulted from the fact that we have effectively removed all redundancy from the bit stream, and are hence really natural consequences, and not side-effects of the approach. Indeed, the above factors can be managed, most simply through the re-insertion of redundancy into the bit stream. (Improvements may also be possible even without the insertion of additional redundancy, loosely similar to the benefits gained from Gray coding.) This comes back to the separate consideration of source and channel coding, as mentioned in the previous section, and discussed in Appendix D. We do not discuss these issues any further in this appendix, other than to say that AC is often dismissed as being impractical due to a lack of understanding of the approach.

The use of limited precision arithmetic to perform Arithmetic Coding has received widespread research interest[36, 151]. Piece-wise Arithmetic Coding has been investigated[174], and techniques of minimising delays have also received some interest[157, 158]. These and other approaches are all directed at making the general AC methodology more practical for classes of applications often found in communications systems. It appears that there is substantial potential for valuable research contributions in this area in the future.

A.5 Quasi-Arithmetic Coding

Quasi-Arithmetic Coding is a modification to AC, introduced by Howard and Vitter[87, 88]. Through a limitation on possible states between symbols, Quasi-AC is able to reduce computational complexity with the use of tables for the encoding/decoding process.

Arithmetic Coding forms a narrow interval on the real line, symbol by symbol, and transmits (identifies) the same interval, bit by bit. Since the symbol boundaries do not in general correspond to the bit boundaries, there is an element of state information contained within the Arithmetic Coding process between input symbols. For infinite precision Arithmetic Coding, the possible states are generally infinite. However, for finite precision AC, the number of states are finite, and with small trade-offs in performance, the number of states can often be reduced dramatically to a level where the tabular approach is ideal.

Quasi-AC can also be viewed as an extension of Huffman coding. For one intermediate state, the tabular approach is equivalent to Huffman coding. The use of only a very small number of states is often capable of providing significant benefit over the performance of Huffman coding. Howard and Vitter note that although Quasi-AC is not an instantaneous coding procedure, the codes are uniquely decodable, with bounded coding delay.

When viewed this way, it is clear that Quasi-Arithmetic Coding is able to logically link Arithmetic Coding and Huffman coding. In fact, closer examination may reveal Quasi-AC being very closely linked to block Huffman coding. It is thus clear that reconsideration of decisions to utilise Huffman coding, but not Arithmetic Coding, may be needed in the light of Quasi-Arithmetic Coding.

Quasi-Arithmetic Coding may also prove extremely useful for controlling transmission error resynchronization, and tackling the encoding/decoding delay problem of Arithmetic Coding. Indeed, it is not difficult to see that the delay issue is closely related to the number of states used in the Quasi-AC approach.

Appendix B

Kalman Prediction for Zero Measurement Noise

For the situation of zero measurement noise (both zero input or background and zero coding or quantization noises) the standard linear predictor provides us with the optimal linear prediction, according to the all-pole signal model. As presented in Chapter 5, the all-pole signal model is:

$$S_k = a_1 S_{k-1} + a_2 S_{k-2} + a_3 S_{k-3} + \cdots + a_N S_{k-N} + w_k. \tag{B.1}$$

For the zero noise case, the standard linear predictor, from equation 5.8, forms the prediction:

$$\hat{S}^{LP}_{k|k-1} = a_1 S_{k-1} + a_2 S_{k-2} + a_3 S_{k-3} + \cdots + a_N S_{k-N}. \tag{B.2}$$

This is the optimal linear prediction based on the above model, as given in equation 5.6.

We would like to show that the Kalman Filter approach results in the same prediction for the zero noise case. We do this by considering the steady-state solution to the Riccati Difference Equation, 5.19, with the measurement noise set to zero:

$$P_{k+1} = F P_k F^T - F P_k H^T \left[H P_k H^T \right]^{-1} H P_k F^T + Q_k. \tag{B.3}$$

The above equation can be rewritten to reflect measurement and time updates:

$$P_{k+1} = F \left[P_k - P_k H^T \left[H P_k H^T \right]^{-1} H P_k \right] F^T + Q_k. \tag{B.4}$$

Here the part of the equation inside the outer square brackets reflects the measurement update, and the rest of the equation represents the effect of the time update. We can further illustrate this point by splitting the equation to form the measurement update equation

$$P_{k|k} = P_{k|k-1} - P_{k|k-1} H^T \left[H P_{k|k-1} H^T \right]^{-1} H P_{k|k-1}, \tag{B.5}$$

and the time update equation

$$P_{k+1|k} = FP_{k|k}F^T + Q_k. \tag{B.6}$$

In the process of splitting the equation, we have defined $P_{k|k}$, which we can refer to as an *a posteriori* error covariance matrix. $P_{k+1|k}$ is defined as the previously defined P_{k+1}, and is an *a priori* error covariance matrix.

The reason for splitting the Riccati equation should now become clear. After a measurement update, and before a time update, we would expect the *a posteriori* error covariance matrix, $P_{k|k}$, to be zero (for the case of zero measurement noise). Indeed, this is effectively the case with the standard linear predictor, as the reconstructed values are exactly equal to the actual sample values. Hence, the *a posteriori* error covariance matrix, $P_{k|k}$ equals zero. We wish to show that this is a steady state solution for the above Riccati equation, B.3.

Setting $P_{k|k} = 0$ implies $P_{k+1|k} = Q_k$ (from B.6), which when substituted into equation B.5 (with H from Chapter 5), indeed gives $P_{k+1|k+1} = 0$. Hence, $P = Q$ is a steady-state solution to equation B.3. Since the Kalman gain vector is found from:

$$K_k = P_k H^T \left[H P_k H^T \right]^{-1}, \tag{B.7}$$

the corresponding steady state solution is $K = [1, 0, \ldots, 0]^T$, which is identical to the standard linear predictor. This is clear by comparison of the equations 5.23 with 5.8 and 5.9. A somewhat similar derivation of the steady state solution is given in [42].

We have not yet shown that the Kalman filter does collapse to the standard linear predictor, since although we have found the steady-state solution, we have not displayed convergence from an arbitrary initial condition. There exist standard mathematical control theory requirements on stability and observability of the system that guarantee convergence to the steady state solution. However this approach encounters some difficulty with zero measurement noise, and readers not completely familiar with mathematical control theory techniques may find the approach does not really assist with an intuitive understanding of the Kalman filtering method. Also, due to the generality of the control theory approach, we do not easily obtain an understanding of how quickly convergence to the steady state solution occurs for the system under consideration. For those readers interested in pursuing the general mathematical approach to convergence, please refer to [18, 29, 167].

In order to explain convergence in a more intuitive manner, we consider the system with arbitrary initial error covariance matrix (symmetric):

$$P_{k|k-1} = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,N} \\ p_{1,2} & p_{2,2} & \cdots & p_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ p_{1,N} & p_{2,N} & \cdots & p_{N,N} \end{pmatrix}, \tag{B.8}$$

and examine how this error covariance matrix evolves over time.

Substituting the above $P_{k|k-1}$ into the measurement update equation, B.5, we obtain

$$P_{k|k} = \left(\begin{array}{c|ccc} 0 & 0 & \cdots & 0 \\ \hline 0 & & & \\ \vdots & & X & \\ 0 & & & \end{array}\right), \tag{B.9}$$

where X is some arbitrary symmetric matrix block. This form of error covariance matrix reflects the fact that although starting with an arbitrary error covariance matrix, after having received the first (zero noise) measurement, we are completely certain of the value of the transmitted sample, and hence the corresponding entries in the error covariance matrix are zero.

Further substitution of $P_{k|k}$ into the time update equation, B.6, leaves

$$P_{k+1|k} = \left(\begin{array}{cc|ccc} x_{1,1} & 0 & x_{1,3} & \cdots & x_{1,N} \\ 0 & 0 & 0 & \cdots & 0 \\ \hline x_{1,3} & 0 & & & \\ \vdots & \vdots & & X & \\ x_{1,N} & 0 & & & \end{array}\right), \tag{B.10}$$

where X is again some arbitrary symmetric matrix block (not necessarily the same as above). Once more, the zeros are related to the certainty in the previously transmitted sample. To assist with verifying the above, please feel free to refer to equations C.5, C.6, and C.7 in Appendix C.

Another measurement update results in

$$P_{k+1|k+1} = \left(\begin{array}{cc|ccc} 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & 0 & & & \\ \vdots & \vdots & & X & \\ 0 & 0 & & & \end{array}\right), \tag{B.11}$$

where X is once more some arbitrary symmetric matrix block. It is thus clear that two zero error measurements are reflected in the all-zero section of the matrix. We see then that after only N measurements (system order), we arrive at our steady state solution, where $P_{k|k} = 0$, and $P = Q$ is the solution to equation B.3.

In summary, we have shown that the steady-state solution of the Kalman filtering approach to linear prediction is optimal for the case of zero measurement noise (both background and quantization noise). In this situation, both the Kalman filter approach and the standard linear predictor approach are effectively identical. Also, the particular Kalman filter converges to the steady state solution after N measurements for arbitrary initial conditions, and although not explicitly discussed, after reaching steady state, model parameter updates do not result in any perturbations in the solution.

Appendix C

Reduced Order Kalman Filter Justification

In Chapter 5 a reduced order Kalman filtering approach was introduced. An error covariance matrix was used of the form:

$$P_k = \left(\begin{array}{c|c} P_k^{KFn} & 0 \\ \hline 0 & 0 \end{array}\right), \tag{C.1}$$

where P_k^{KFn} is found from the reduced order Riccati Difference Equation 5.45. It was claimed that through this approach we could obtain much of the advantage of Kalman filtering for only a fraction of the computational cost.

In this appendix, we examine the form of the error covariance matrix that arises from the full order Riccati equation 5.19, and attempt to justify our approximations for P_k.

As in Appendix B, we consider splitting the Riccati equation into a measurement update:

$$P_{k|k} = P_{k|k-1} - P_{k|k-1}H^T \left[HP_{k|k-1}H^T + R_k\right]^{-1} HP_{k|k-1}, \tag{C.2}$$

and a time update equation

$$P_{k+1|k} = FP_{k|k}F^T + Q_k. \tag{C.3}$$

The matrices F, H, Q, and R, are all as defined in Chapter 5. In particular, we note that the system matrix, F, has the form:

$$F = \begin{pmatrix} a_1 & a_2 & \cdots & a_{N-1} & a_N \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}. \tag{C.4}$$

After a measurement update, we have an *a posteriori* error covariance matrix, $P_{k|k}$, of the form:

$$P_{k|k} = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,N} \\ p_{1,2} & p_{2,2} & \cdots & p_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ p_{1,N} & p_{2,N} & \cdots & p_{N,N} \end{pmatrix}. \tag{C.5}$$

It is important to note that $P_{k|k}$ and $P_{k+1|k}$ are non-negative definite matrices, as they are error covariance matrices[18]. Hence $P_{k|k} \geq 0$ and $P_{k+1|k} \geq 0$, and as a consequence, all the diagonal terms are non-negative, $p_{i,i} \geq 0, \forall i \in 1..N$.

We would like to examine some of the properties of the above error covariance matrix, $P_{k|k}$. As in Appendix B, we attempt to minimise the use of the generalized mathematics, and deal more specifically with the system of concern.

The first property that we would like to display is that the diagonal terms $p_{1,1}$ to $p_{N,N}$ are monotonically decreasing, in steady state. We do this by considering the effect of a single time update, followed by a single measurement update, on the error covariance matrix given in C.5.

After the time update we obtain (with the use of notation from Chapter 5)

$$P_{k+1|k} = \begin{pmatrix} \sum_{i=1}^{N} a_i \sum_{j=1}^{N} a_j p_{i,j} + \sigma_{wk}^2 & \sum_{i=1}^{N} a_i p_{1,i} & \cdots & \sum_{i=1}^{N} a_i p_{N-1,i} \\ \sum_{i=1}^{N} a_i p_{1,i} & p_{1,1} & \cdots & p_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{N} a_i p_{N-1,i} & p_{1,N-1} & \cdots & p_{N-1,N-1} \end{pmatrix}. \tag{C.6}$$

We substitute for $P_{k+1|k}$ in the measurement update equation to obtain a solution of the form:

$$P_{k+1|k+1} = \tag{C.7}$$

$$\begin{pmatrix} \beta\gamma & \beta \sum a_i p_{1,i} & \cdots & \beta \sum a_i p_{N-1,i} \\ \beta \sum a_i p_{1,i} & p_{1,1} - \frac{(\sum a_i p_{1,i})^2}{\gamma + \sigma_{nk}^2} & \cdots & p_{1,N-1} - \frac{(\sum a_i p_{N-1,i})(\sum a_i p_{1,i})}{\gamma + \sigma_{nk}^2} \\ \vdots & \vdots & \ddots & \vdots \\ \beta \sum a_i p_{N-1,i} & p_{1,N-1} - \frac{(\sum a_i p_{N-1,i})(\sum a_i p_{1,i})}{\gamma + \sigma_{nk}^2} & \cdots & p_{N-1,N-1} - \frac{(\sum a_i p_{N-1,i})^2}{\gamma + \sigma_{nk}^2} \end{pmatrix},$$

where for convenience the summation ranges have been omitted, and we have introduced β and γ, defined by

$$\beta = 1 - \frac{\sum a_i \sum a_j p_{i,j} + \sigma_{wk}^2}{\sum a_i \sum a_j p_{i,j} + \sigma_{wk}^2 + \sigma_{nk}^2}, \tag{C.8}$$

$$\gamma = \sum a_i \sum a_j p_{i,j} + \sigma_{wk}^2. \tag{C.9}$$

Since $P_{k+1|k} \geq 0$, $\gamma \geq 0$, as it is the first diagonal term in $P_{k+1|k}$ (from equation C.6). Hence the second diagonal term in $P_{k+1|k+1}$ is

$$p_{2,2}^{k+1} = p_{1,1}^{k} - \frac{(\sum a_i p_{1,i}^{k})^2}{\gamma + \sigma_{nk}^2} \leq p_{1,1}^{k}, \tag{C.10}$$

where the introduced superscript notation refers to the error covariance matrix at time $(k+1)$ or k respectively. Likewise for all the diagonal terms, we have

$$p_{l,l}^{k+1} - \frac{(\sum a_i p_{l,i}^{k})^2}{\gamma + \sigma_{nk}^2} \leq p_{l,l}^{k}. \tag{C.11}$$

Here we observe that the covariance term relating to a particular sample is non-increasing over time:

$$p_{1,1}^{k} \geq p_{2,2}^{k+1} \geq p_{3,3}^{k+2} \geq \cdots \geq p_{N,N}^{k+N-1}. \tag{C.12}$$

Hence, in steady state,

$$p_{1,1} \geq p_{2,2} \geq p_{3,3} \geq \cdots \geq p_{N,N}, \tag{C.13}$$

reflecting the effect of smoothing resulting in better sample estimates for the use of more lags.

We assert that the rate of decrease in magnitude of the $p_{i,i}$ terms is closely linked to the relationship between the excitation covariance, as given by Q, and the measurement noise covariance, as given by R. (We note there is also a relation to the system matrix, F, but do not discuss this.) It is obvious that for infinite measurement noise we obtain the equality in C.13, with $p_{i,i} = \gamma = \sum a_i \sum a_j p_{i,j} + \sigma_{wk}^2$, $\forall i \in 1..N$. For values of measurement noise less than infinity, the error covariance decreases for increasing lag. For small measurement noise, we expect that this decrease is rapid. Anderson and Moore[4] discuss the smoothing benefit, and note that most gain is achieved within only a few lags.

This observation leads to the assumption that after only a small number of lags we can assume that the error covariance is constant. Further assuming that this constant final error covariance value is much smaller than the process noise, σ_{wk}^2, allows the assumption that after only a small number of lags, n, the error covariance is close to zero.

$$p_{l,l} \approx 0, \qquad \forall l \in (n+1)..N \tag{C.14}$$

We note from appendix B that for the zero measurement noise case, $p_{l,l} = 0$ for all l.

If the assumption in equation C.14 is valid, then it is easy to show that $p_{l,m} \approx 0$, $\forall l, m \in (n+1)..N$. This fact comes directly from the non-negative definite property of the matrix $P_{k|k}$. Since $P_{k|k} \geq 0$, $x^T P_{k|k} x \geq 0$ for all column vectors, x. Taking $x = e_l + e_m$, where e_i is the unit basis vector, zero except for a one in the ith place, we obtain

$$p_{l,l} + p_{m,m} + 2p_{l,m} \geq 0, \tag{C.15}$$

which translates to:

$$p_{l,m} \geq -\frac{1}{2}\left(p_{l,l} + p_{m,m}\right), \tag{C.16}$$

giving a lower bound for $p_{l,m}$. The upper bound:

$$p_{l,m} \leq \frac{1}{2}\left(p_{l,l} + p_{m,m}\right), \tag{C.17}$$

is found by taking the mth place of x to be negative one. This leaves us with the absolute value bound

$$|p_{l,m}| \leq \frac{1}{2}\left(p_{l,l} + p_{m,m}\right), \tag{C.18}$$

implying that the assumption $p_{n+1,n+1} \approx 0$ is equivalent to C.14.

So far we have provided a justification for the use of a zero bottom right-hand block in C.1. From equation C.18, we are able to obtain an upper bound of $\frac{1}{2}p_{1,1}$ for those elements in the two off-diagonal blocks in C.1. However, this is not tight enough to allow us to claim immediately that the zero assumption is justifiable for the off-diagonal blocks. A slight modification to the above procedure is capable of giving us the result we desire. Taking $x = e_l - \alpha e_m$, where we $\alpha > 1$, and such that $\alpha^2 p_{m,m} = p_{l,l}$, $(m > l)$, gives

$$p_{l,l} + \alpha^2 p_{m,m} - 2\alpha p_{l,m} \geq 0. \tag{C.19}$$

Hence we obtain

$$p_{l,m} \leq \frac{p_{l,l}}{\alpha}, \tag{C.20}$$

and since the assumption that $p_{m,m} \approx 0$ is equivalent to α being large, it is safe to additionally assume $p_{l,m} \approx 0$. This shows that our reduced complexity choice for the error covariance, C.1, is a natural choice, based on low enough levels of measurement noise for C.14 to hold.

As noted in Chapter 5, using an error covariance matrix of the form (with notation defined in the chapter):

$$P_k = \left(\begin{array}{c|c} P_k^{KFn*} & 0 \\ \hline 0 & \sigma_{SR}^2 I \end{array}\right), \tag{C.21}$$

is equivalent to recognising the residual error covariance terms are non-zero, but ignoring their smoothing effect. We do not provide any further justification for this here.

One further point to note is that taking into account any terms off the main diagonal and outside of the top left hand block means a significant increase in computation, with very little additional increase in computation required to consider an enlarged top left hand block that engulfs the new term.

In summary, this appendix has displayed a number of properties of the error covariance matrix that allow us to place a certain amount of trust in our assumption that the top left hand block of the matrix is the most significant. Intuitively this correlates very well with the concept of smoothing gain being achieved primarily within the first few lags. However, care must still be taken before using an error covariance matrix of the form given in C.1.

Appendix D

Information Theory and Channel Coding

D.1 Information Theory

Information Theory from Shannon[160] consists of three major components:

- Source Coding
- Channel Coding
- Rate Distortion Theory

Source Coding has been the major concern of this thesis. In terms of Information Theory, source coding concerns studies of information measurement, and its conversion into different forms. A recent survey paper on source coding is the one by Kieffer[116]. This paper also discusses briefly the issue of combined source/channel coding, and provides an extensive reference list.

Channel Coding relates to the amount of information a channel can transfer without error. Channel capacity is a measure of how much information a channel can transmit, and dictates that rates below the channel capacity are theoretically possible, while rates above the capacity will result in unavoidable errors. As with source coding, channel coding is concerned with information measurement and conversion. Obviously source and channel coding are closely related, with the former taking a user perspective, and the latter more a channel perspective.

Rate Distortion Theory deals with where not enough information is available to ensure perfect reconstruction. Hence, the issue is to provide the best reconstruction based on the available symbols. A good coverage of the topic is provided in the book by Berger[16].

For a channel with 'nice' properties, such as some fixed, and constant, level of additive white noise, Information Theory gives us a simple calculation for the channel capacity, or the amount of information that we can transmit over the channel. In order to transmit this amount of information, channel signalling schemes must be used that are well adapted to the channel characteristics. This is the domain of channel coding, which is often further subdivided into error control coding, and modulation.

Unfortunately, most practical channels of concern to electrical engineers do not have such nice properties. Many channels exhibit time-varying transmission characteristics, such as fading, crosstalk, impulsive noise bursts, and Doppler frequency shifts. Information theory is sufficiently general to still determine the channel capacity, after considering the effects of the various disturbances. However, the practical realization of obtaining a channel transmission rate near that of the channel capacity becomes much harder. Slow fading phenomena may be countered through the use of automatic gain control on the transmitter power. However, fast fading, and the other distortions just mentioned are generally assumed to be the realm of channel coding.

The mobile communications application is one that is extremely important as far as speech coding systems are concerned. In this application, bit error rates are usually quite high, and bursty in nature. For this type of channel, the concept of separation of source and channel coding that we get from Shannon's Information Theory does not necessarily hold. For applications such as this, it is increasingly obvious that we must provide a uniform approach to the communications system by using combined source and channel coding.

D.2 Channel Coding

For efficiency reasons, we wish to code at rates approaching the channel capacity bound given by Shannon. However, in order to achieve this, channel coding is necessary, as simple baseband modulation techniques are not sufficient. Channel coding can simply be viewed as the introduction of diversity into the transmitted stream to combat the effects of channel distortion. Primarily the diversity introduced is time diversity, although schemes introducing frequency diversity are also included in channel coding. Other approaches involving such methods as antenna diversity are often considered more to be part of the channel, rather than of the coding process.

For applications such as mobile communications, channel coding often takes up an extremely large amount of the spectrum over and above that from the data bits. However, it must be noted that there is a fine line between the channel coding implying the need for more spectrum, and simply resulting in more efficient spectrum use. This fact is simple to see with a CDMA spread spectrum system, where the cost of channel coding is only really in terms of computation, rather than spectrum.

Mainly for simplicity and lack of better approaches, the channel coding is performed in the digital domain, and the resulting bit stream is modulated in exactly the same way as the original data stream would have been for a 'nicely' behaved transmission channel.

Remark D.1 The separation of concerns into digital error control coding and modulation is akin to the separation of source and channel coding. It is utilised mainly for simplicity, even though the conditions for optimality of the separation are often violated in practice. An illustration of this can be obtained by considering the use of soft-decision decoding[122, 143].

A typical channel coding approach is the use of a convolutional code, and the use of Viterbi decoding. However, for mobile communications, approaches more attuned to the properties of the transmission channel are really required. This implies immediately that consideration of bit error distributions is required, and perhaps even a more integrated approach to channel coding and modulation is needed. It is beyond the scope of this appendix to become too involved in the specifics of the mobile communications channel. However, below we review briefly a number of the more well known channel coding approaches, and the following sections point out some issues especially important to mobile communications.

D.2.1 BCH Coding

Bose-Chaudhuri-Hocquenghem (BCH) codes are among the most important cyclic codes. They are a large class of multiple error correcting codes published by Hocquenghem in 1959, and jointly by Bose and Ray-Chaudhuri in 1960[143]. They include both binary and non-binary alphabets, and are commonly used with hard decision decoding. The Reed-Solomon codes, described in the next subsection form an important subset of non-binary BCH codes.

Cyclic codes[5] are a specific subgroup of linear codes, obtained by imposing the structure of Galois fields. The polynomial description of finite fields leads to simple algorithms for encoding and decoding. For a code to be a cyclic code, it must satisfy:

- The sum of any two codewords is itself a codeword (linearity), and

- Cyclically shifting a codeword results in a codeword.

Cyclic codes thus have the property that their code generation matrix consists of rows that are cyclic shifts of each other. Hence the code generation matrix can be expressed in terms of a single row, and the cyclic code is expressed in terms of a generator polynomial.

The encoder for a binary cyclic code can be implemented simply through the use of shift registers as delay elements, and binary adders, wired between the new input bit and the shift register state, as dictated by the generator polynomial. The decoder for a binary cyclic code can also be constructed through the use of shift registers and binary logic.

The generator polynomial for a BCH code is defined in terms of the least common multiple of minimal polynomials of powers of a primitive element in the Galois field[5], and tables of generator polynomials exist for many BCH codes.

D.2.2 Reed-Solomon Coding

Reed-Solomon (RS) codes are non-binary BCH codes that are *maximum-distance separable*[143]. The distance property is one reason why Reed-Solomon codes are so important in practice. The existence of efficient hard-decision decoding algorithms allow the use of relatively long codes, which is an extremely desirable feature. The use of long codes means the ability to accommodate large bursts of errors, such as may commonly occur due to surface scratching of an audio CD.

D.2.3 The Viterbi Algorithm

The Viterbi algorithm is a general maximum-likelihood algorithm for a recursive optimal solution to state sequence estimation of discrete-time finite-state Markov processes. In many ways it is analogous to the Kalman filter, with the difference that the Kalman filter is based on gaussian processes and continuous states. An excellent tutorial style paper on the Viterbi algorithm is that by Forney[64].

One of the most common applications of the Viterbi algorithm is in decoding of convolutional channel codes. The algorithm was first developed for this application, and VLSI implementations are common, alternately utilising hard and soft decision decoding. As expected, the performance is improved with the use of soft decisions.

Convolutional codes are conceptually simple, and obtain performance that often exceeds that of block codes. Ease of practical implementation, and the existence of practical soft decision decoders help to explain their widespread use. As the name suggests, the added bits are calculated via the use of modulo-two convolutions, and can be implemented with shift registers and binary logic.

Convolutional codes can easily be represented with a Markov state transition diagram, which is often expanded over time to obtain a trellis diagram[122]. The Viterbi algorithm is basically a forward dynamic programming approach to finding the most likely path through the trellis. For each state transition in the trellis, there is an associated cost, and for time instant k, there is a minimum cost to reach each state. The minimum costs to reach the states at time $(k+1)$ is calculated from the state transitions from state at time k to $(k+1)$, and the original minimum costs for the states at time step k.

Convolutional codes are widely used in digital mobile communications systems. It is usual that in these applications the convolutional codes are terminated at frame boundaries, and effectively become block codes. The simplicity and performance of the block code produced in this way is usually superior to alternative block codes. The Viterbi algorithm is also able to combat some forms of intersymbol interference[64].

D.3 Bit Errors and Distributions

Bit errors in communications systems are the result of the addition of noise to the transmitted signal in such a way that the decoded signal does not agree with that transmitted.

Noise in communication systems comes from many sources. Thermal noise is always present, and crosstalk and echo are problems in networks such as the telephone system. Carrier jitter, dispersion and distortion, and inter-symbol interference also are common problems. Other problems more related to the telephony network are impulse noise (of the order of 10 ms in length), and TDM multiplexing errors. In mobile communications, fading and multipath are problems.

In many practical applications, error bursts are much more likely than random bit errors. However, approaches such as convolutional encoding and Viterbi decoding are only really useful for recovery from random bit errors. For this reason, techniques such as interleaving of data blocks are commonly used.

Frame interleaving spreads error bursts over many frames, with the aim of translating the bursts into effectively random errors. Unfortunately for long error bursts, to effectively randomise the errors would require high levels of frame interleaving, and long delays. Delay is one of the four basic considerations for communications systems, and it is often the case that engineering trade-offs are made that keep delay to reasonable levels. In practice, this means that some bursts of errors will be present.

The most common approach to handling these residual uncorrected error bursts is the use of error detection, and notification of the source coder that the frame is corrupted. In the GSM digital mobile system the most sensitive bits in a frame are protected such that uncorrected errors flag a corrupted frame[169].

D.4 Combined Source and Channel Coding

As previously mentioned, complete separation of source and channel coding is practically impossible, due mainly to delay and computation restrictions. Hence residual bit errors will be present in the data input to the source decoder. It is important that the source coding system is able to adequately filter these residual effects of channel impairments.

A source coding system that completely removes the redundancy from the input implies that bit transmission errors will result in loss of information at the decoder. To minimise the effects of this loss, it is often sufficient to localise the errors to a short interval in time. Where residual source redundancy exists in the transmitted data, more advanced attempts can often be made at error minimisation, through the exploitation of this redundancy such as discussed in [159]. Fortunately most practical source coding systems for signals such as speech leave a large amount of redundancy, which is often used implicitly by the decoder to filter effects of single bit errors, such as with pseudo Gray coded VQ (Vector Quantization), or explicitly by frame erasure recovery strategies.

The ability of the source coder to recover from residual channel errors can be viewed as one form of combined source/channel coding. In the broadest sense, there are many forms of source/channel coding. Another example that is a slight extension of the above is the use of soft-decision channel demodulation information to improve the decoder. Soft-decision vector quantization[49] is a further example of source/channel coding.

For mobile communications, techniques of adaptively altering the balance between source and channel coding, in response to time varying channel conditions have been proposed[171]. Embedded coding approaches, where a proportion of the transmitted bits are systematically sacrificed to satisfy channel demands, have received widespread use in telecommunications systems. More recently, issues of multiresolution transmission have been raised for broadcasting HDTV[148].

The concept that bits should have equal importance does not often apply to practical systems, and hence uneven bit protection is common. Speech coding systems used in conjunction with mobile radio channels usually have significant levels of error detection and correction redundancy added. In most practical coders, these error recovery bits are distributed according to how important it is to detect or correct errors in certain bits of the output. For example, in the GSM system, there are three levels of error protection within a frame[169]. In the lowest level, the less sensitive bits are left unprotected (apart from interleaving to avoid burst errors). The next level involves convolutional coding, with Viterbi decoding, and the highest level involves error detection of residual errors in the most sensitive bits.

D.5 Spread Spectrum Communications

Spread spectrum communications are basically defined as those where the information data rate is much less than the transmission bandwidth. General introductions to spread spectrum techniques can be found in many communications texts, with an example being the book by Cooper and McGillem[39].

There are many advantages to spread spectrum systems for both military and civilian applications. Some of the advantages from a military perspective are the resistance to jamming, and the possibility for covert communications, through the use of low power signals that become 'lost' in background noise. Other spread spectrum applications areas involve radar type systems, where the advantages of the broad bandwidth signal is used to obtain better range and velocity information.

For civilian, and particularly mobile, communications, the major advantages of spread spectrum communications are in multiple access, and the inherent 'encryption' that is available. Related to this is the issue of selective calling of mobile units from the base station. Also extremely important is the resistance to multipath fading, and the possibility of extremely low probabilities of undetected frame errors. These issues indicate a clear connection with many of the channel coding topics mentioned above, and system design often demands a global approach encompassing all aspects of telecommunications.

An issue with spread spectrum systems that has received somewhat less attention, but is presently gaining more popularity, is the potential for the production of a variable bit rate transmission system. Of course, it is possible to provide a variable bit rate system through TDMA, but currently the Qualcomm CDMA QCELP[51, 66] system is the only truly successful variable bit rate mobile communications system, and is certainly worthwhile monitoring for future movements. Future LEO (Low Earth Orbit) satellite communications systems also look set to utilize variable rate spread spectrum techniques.

One of the main disadvantages of CDMA spread spectrum systems is related to power control problems. The received power levels at the base stations must be kept within 1 dB, or significant performance degradation occurs. Other questions with CDMA is the existence of enough codes for high usage systems, and the issue of synchronization problems. However, the Qualcomm system appears to have overcome many of these issues, and proved that CDMA technology has some major advantages to TDMA.

Recently there has also been considerable interest in TDMA spread spectrum techniques such as slow frequency hopping[1]. This approach involves each user changing frequencies in a predetermined fashion between transmitted frames. Spectrum efficiency in such a system can be increased, as users in neighbouring cells with similar frequencies, only interfere randomly. Hence some frame loss is introduced, but it is random in nature, and spread across all users, allowing normal speech decoder frame loss recovery techniques to be used.

TDMA versus CDMA has been discussed, at times quite energetically, for many years. At present both approaches are producing considerable successes, and it is impossible to say that one system is better than the other, except for systems with specific constraints. As TDMA and CDMA are different approaches, it is perhaps ridiculous to expect to say that one is better than the other. They both have advantages and disadvantages. It is interesting to note that practical systems almost always involve a combination of techniques such as TDMA and FDMA. We are thus strictly dealing with a class of approaches. It is not difficult to imagine codes for a CDMA system that closely resemble TDMA, and a further stretch of the imagination applies to FDMA. In an extremely loose manner, CDMA can actually be viewed as a generalization of TDMA and FDMA.

Appendix E

Constant Step Size Uniform Quantizer Justification

In this appendix we wish to show that a constant step size uniform quantizer is 'optimal' with respect to minimising the MSE (Mean Square Error), or maximising the SNR (Signal to Noise Ratio) of an uncorrelated prediction difference signal.

Many authors have considered the problem of finding optimum quantizers for memoryless sources[17, 57, 142, 179]. The paper by Farvardin and Modestino[57] is a good example, with a number of references being provided to other work in the area. The work of Gish and Pierce[77] is noted as having established the high rate asymptotic optimality of entropy coded uniform quantization. The performance of the uniform quantizer when the high rate assumption is violated is discussed in [57], with the conclusion that the uniform quantizer performs effectively as well as the optimum quantizer.

In this appendix we assume the use of a uniform quantizer for quantizing each sample, but consider the effect of changes in input sample variance, and whether this implies a quantizer step-size adjustment.

We approach this problem by first considering the simplified case of a uniform distribution, and then attempt to extend this to the Laplacian distribution.

E.1 The Uniform Distribution

Consider a uniform probability distribution function:

$$\begin{aligned} f_k(x) &= \frac{1}{2\alpha_k}, & |x| \leq \alpha_k \\ &= 0, & |x| > \alpha_k, \end{aligned} \tag{E.1}$$

Where the k subscript denotes the fact that each sample is uniformly distributed, but with perhaps different variances resulting from different values for the α_k parameter.

In order to obtain maximum quantization efficiency for this distribution, with a standard (fixed rate) quantizer, we would evenly space our quantization cells along the non-zero portion of the distribution. For samples with different variances this implies changing the quantizer step size, Δ_k, in proportion to the α_k parameter. For situations where the α_k parameter only changes on a segmental basis, the tracking of the α_k parameter with the step size, Δ_k, results in effective maximisation of segmental SNR of the prediction difference signal.

The ability to use entropy coding allows more flexibility over the choice of quantizer step size, Δ_k, by effectively using more bits on some samples, and less for others. We thus attempt to select the quantizer step size to minimize the MSE.

With a small loss of generality (negligible for the high rate situation), we consider the selection of step sizes, Δ_1 and Δ_2, to minimise the MSE where $\alpha_1 = n_1\Delta_1$ and $\alpha_2 = n_2\Delta_2$, for some arbitrary integers n_1 and n_2.

For uniform quantization error probability and quantizer step size Δ, with Δ an integer fraction of the sample distribution range, the MSE of the quantizer is $\Delta^2/12$. Hence we must choose the step sizes to minimise $(\Delta_1^2 + \Delta_2^2)/24$, which is the MSE of the simple system under analysis.

Obviously an entropy constraint exists, otherwise the MSE could be minimised to zero by selecting zero quantizer step size. The relevant constraint is thus

$$R = \frac{1}{2}\left(\log_2\left(\frac{2\alpha_1}{\Delta_1}\right) + \log_2\left(\frac{2\alpha_2}{\Delta_2}\right)\right), \tag{E.2}$$

where R is the required bit rate (entropy). (Entropy has been defined in Appendix A.)

Alternatively, we can formulate the problem as one of minimising the entropy with a MSE constraint. The MSE constraint is thus

$$\frac{\Delta^2}{12} = \frac{(\Delta_1^2 + \Delta_2^2)}{24}. \tag{E.3}$$

Without loss of generality we introduce the parameter $\beta \geq 0$, such that

$$\Delta_1 = \frac{\Delta\sqrt{2}}{\sqrt{1+\beta^2}} \tag{E.4}$$

$$\Delta_2 = \frac{\beta\Delta\sqrt{2}}{\sqrt{1+\beta^2}}. \tag{E.5}$$

Thus equation E.2 becomes

$$R = \frac{1}{2}\left(\log_2\left(\frac{2\alpha_1\sqrt{1+\beta^2}}{\Delta\sqrt{2}}\right) + \log_2\left(\frac{2\alpha_2\sqrt{1+\beta^2}}{\beta\Delta\sqrt{2}}\right)\right). \tag{E.6}$$

Minimising equation E.6 is equivalent to minimising

$$\left(\frac{2\alpha_1\sqrt{1+\beta^2}}{\Delta\sqrt{2}}\right)\left(\frac{2\alpha_2\sqrt{1+\beta^2}}{\beta\Delta\sqrt{2}}\right) = \frac{4\alpha_1\alpha_2\left(1+\beta^2\right)}{2\Delta^2\beta}, \tag{E.7}$$

which is equivalent to the simple minimization of

$$\frac{1+\beta^2}{\beta}. \tag{E.8}$$

Hence the minimum entropy is obtained where $\beta = 1$, giving $\Delta_1 = \Delta_2 = \Delta$.

With some generalization to account for the assumptions made above, we have thus shown that for the case of a uniform probability distribution function, optimum MSE quantizer performance is obtained with the use of a fixed quantizer step size. The next section attempts to extend this result to the case of a Laplacian distribution.

E.2 The Laplacian Distribution

The Laplacian distribution is of the form:

$$f_k(x) = \frac{1}{\sqrt{2}\,\sigma_k}\exp^{-\sqrt{2}\,|x|/\sigma_k}, \tag{E.9}$$

where σ_k^2 is the sample variance. We can simplify notation by the use of the parameter $\lambda_k = \sqrt{2}/\sigma_k$, giving

$$f_k(x) = \frac{\lambda_k}{2}\exp^{-\lambda_k|x|}. \tag{E.10}$$

Before calculation of the entropy, we need to integrate the distribution to find the probability of each quantizer output level. For a uniform mid-rise quantizer with step size Δ_k, the probability in the ith cell from the centre of the distribution is

$$P_i = \int_{(i-1)\Delta_k}^{i\Delta_k}\frac{\lambda_k}{2}\exp^{-\lambda_k|x|}\,dx. \tag{E.11}$$

Performing the integration and simplifying we obtain

$$P_i = \frac{\exp^{\lambda_k\Delta_k} - 1}{2}\exp^{-\lambda_k\Delta_k i}, \tag{E.12}$$

which can now be used to obtain the entropy.

The entropy contribution from the sample at time k is

$$R_k = -\sum_{i=1}^{\infty}\left(\exp^{\lambda_k\Delta_k}-1\right)\exp^{-\lambda_k\Delta_k i}\log_2\left(\left(\frac{\exp^{\lambda_k\Delta_k}-1}{2}\right)\exp^{-\lambda_k\Delta_k i}\right), \tag{E.13}$$

which is expanded to

$$\begin{aligned} R_k &= \left(1-\exp^{\lambda_k\Delta_k}\right)\log_2\left(\frac{\exp^{\lambda_k\Delta_k}-1}{2}\right)\sum_{i=1}^{\infty}\exp^{-\lambda_k\Delta_k i} \qquad \text{(E.14)}\\ &\quad+\left(1-\exp^{\lambda_k\Delta_k}\right)\sum_{i=1}^{\infty}\exp^{-\lambda_k\Delta_k i}\log_2\left(\exp^{-\lambda_k\Delta_k i}\right)\\ &= \left(1-\exp^{\lambda_k\Delta_k}\right)\log_2\left(\frac{\exp^{\lambda_k\Delta_k}-1}{2}\right)\sum_{i=1}^{\infty}\exp^{-\lambda_k\Delta_k i} \qquad \text{(E.15)}\\ &\quad+\frac{\left(\exp^{\lambda_k\Delta_k}-1\right)}{\ln 2}\sum_{i=1}^{\infty}\lambda_k\Delta_k i\exp^{-\lambda_k\Delta_k i}. \end{aligned}$$

The first summation, $\sum_{i=1}^{\infty}\exp^{-\lambda_k\Delta_k i}$, can be replaced by the simple formula for the sum of an infinite series:

$$\sum_{i=1}^{\infty}\exp^{-\lambda_k\Delta_k i} = \frac{\exp^{-\lambda_k\Delta_k}}{1-\exp^{-\lambda_k\Delta_k}} = \frac{1}{\exp^{\lambda_k\Delta_k}-1}, \tag{E.16}$$

and the second summation can be simplified, since

$$\sum_{i=1}^{\infty} i\exp^{-\lambda_k\Delta_k i} = \frac{\exp^{\lambda_k\Delta_k}}{\left(\exp^{\lambda_k\Delta_k}-1\right)^2}. \tag{E.17}$$

Making these substitutions gives

$$R_k = \frac{1}{\ln 2}\left[-\ln\left(\frac{\exp^{\lambda_k\Delta_k}-1}{2}\right)+\frac{\lambda_k\Delta_k\exp^{\lambda_k\Delta_k}}{\exp^{\lambda_k\Delta_k}-1}\right], \tag{E.18}$$

which appears to be as simplified as we are able to obtain for the entropy expression.

Again we consider minimising the entropy with a constraint on the MSE. We wish to choose Δ_1 and Δ_2 to minimise

$$\begin{aligned} R &= \frac{1}{2\ln 2}\left[2\ln 2-\ln\left(\exp^{\lambda_1\Delta_1}-1\right)-\ln\left(\exp^{\lambda_2\Delta_2}-1\right)\right.\\ &\qquad\left.+\frac{\lambda_1\Delta_1}{1-\exp^{-\lambda_1\Delta_1}}+\frac{\lambda_2\Delta_2}{1-\exp^{-\lambda_2\Delta_2}}\right]. \qquad \text{(E.19)} \end{aligned}$$

Remark E.1 *The formulation for the MSE in terms of the quantizer step size, Δ, was based on the assumption of a uniform quantization error across the quantizer cell. Although it is strictly violated for the Laplacian distribution, we still use this assumption, and note that for high rate quantization the inaccuracy involved is negligible.*

We can incorporate the MSE constraint in a similar fashion to above by defining β as in equations E.4 and E.5. We thus obtain

$$\begin{aligned} R &= \frac{1}{2\ln 2}\left[2\ln 2 - \ln\left(\exp^{\lambda_1 \frac{\Delta\sqrt{2}}{\sqrt{1+\beta^2}}} - 1\right) - \ln\left(\exp^{\lambda_2 \frac{\beta\Delta\sqrt{2}}{\sqrt{1+\beta^2}}} - 1\right)\right. \\ &\qquad \left. + \frac{\lambda_1 \frac{\Delta\sqrt{2}}{\sqrt{1+\beta^2}}}{1 - \exp^{-\lambda_1 \frac{\Delta\sqrt{2}}{\sqrt{1+\beta^2}}}} + \frac{\lambda_2 \frac{\beta\Delta\sqrt{2}}{\sqrt{1+\beta^2}}}{1 - \exp^{-\lambda_2 \frac{\beta\Delta\sqrt{2}}{\sqrt{1+\beta^2}}}}\right]. \end{aligned} \tag{E.20}$$

It should be clear that minimising the entropy given in equation E.20 is not a simple task, as it was for the uniform distribution (equation E.8). We would like to be able to show that a constant quantizer step size, corresponding to $\beta = 1$, provides us with the minimum entropy. Recourse to numerical methods is able to confirm that this is indeed the case, for arbitrary choice of the parameters λ_1, λ_2, and Δ. Although this does not constitute a proof, it is an adequate illustration for our purposes, since the central question is the speech quality performance of the system constructed with the fixed quantizer step size in Chapter 4.

Appendix F

Frame Dropping in the LD-CELP Encoder for 12 and 8 kbps Coding

Analysis of the AT&T March 1994 ITU-T (ITU – Telecommunications Standardization Sector) contribution entitled "G.728 Decoder Modifications for Frame Erasure Concealment"[12], reveals a high level of robustness to frame erasures, with only minimal system modifications. We exploit this inherent robustness of LD-CELP to obtain variant coders operating at lower rates.

Within this appendix, we propose novel 12 kbps and 8 kbps variants of LD-CELP that obtain bit rate reduction by selectively dropping frames in the encoder. It is natural to expect that a 25 or 50 percent frame erasure rate would lead to substantial degradation in speech output quality. However, the fact that we are able to selectively erase frames in a deterministic fashion implies that we are able to obtain improved performance over the situation of frame erasure due to channel impairments.

The issue of which frames to discard is a significant one. Obviously we expect the loss of some frames to have a marked effect on the output, while the loss of others may have very little impact. As an example, consider the loss of a frame during a silence period. We would expect this to have very little overall effect. Unfortunately the luxury of being able to choose which frames to discard generally comes at the expense of additional delay, and some additional computation. Instead, we propose, somewhat arbitrarily, to simply discard one in four frames.

For frame erasure studies due to channel impairments, the frame size is effectively dictated by the air-interface. Hence, the proposal [12] considers mainly a 10 ms frame size. For our 12 kbps system, we have no air-interface restrictions, and thus the smallest possible frame size is seen as desirable from a delay consideration (assuming a fixed rate channel). We consider the five sample vector in LD-CELP, and for every four vectors, or 20 samples, we transmit three codevectors of 10 bits each, resulting in a total of 30 bits, or 12 kbps. Likewise for the 8 kbps rate we only transmit every second vector.

It is unlikely that this approach to designing 12 kbps and 8 kbps coders would be sensible for constructing a coder at those rates from scratch. However, our interest is mainly theoretical, and there may be reasons why a 16 kbps LD-CELP coder would want to switch to a lower rate for a short period of time.

We performed a number of simulations using an approach similar to the excitation extrapolation discussed in [12]. Of course, this implies the need for computation of pitch lags and the like in the encoder, which is additional computational complexity, but can be offset to some extent by the avoidance of 25 or 50 percent of the codebook searches. Interpolation, mentioned in [12] for possible use with frame erasures, is not considered here, due to the causality of the encoder.

An important observation that was made during our brief study was that the performance of the excitation extrapolation approach was almost identical to, and perhaps slightly worse than, that of using zero excitation for the missing vectors. This observation appears justifiable in the light of the very small vector of excitation we are trying to extrapolate, and the fact that zero mean excitation is about the best we can do on the small scale. It is also possible that the encoder changes are altering much of the periodic nature of the excitation that was observed for the frame erasure problem. Based on these observations and the computation issue, we simply use the zero excitation approach for the missing vectors. Another possible advantage to the zero excitation approach would be in bit error conditions, however simulations of this type are well beyond the scope of our immediate concern.

As with the frame erasure studies in [12], issues of standard compatibility must be considered. However, the dropping of frames at the encoder immediately implies that we have a non-standard encoder, and as such some other minor modifications seem attractive. For the frame erasure work, spectral smoothing was introduced, and the level of bandwidth expansion was increased. The white noise correction factor was also increased in simulations, but found to reduce clean channel performance, and hence left at the original value. Simulations were performed with different values for all three parameters at the 12 kbps rate, but the results showed very little effect on the perceived output quality. An explanation for this is that in the current studies there is no encoder/decoder divergence, while spectral smoothing and bandwidth expansion were found to significantly assist with convergence. Thus no changes are made to these parameters.

Unlike the AT&T frame erasure study, we do not propose to halt the filter coefficient update process. This is not practical where we are dealing with dropped vectors for every 20 sample adaptation cycle. Hence it is clear that the modifications to the LD-CELP system to obtain 12 kbps and 8 kbps variant coders are extremely simple.

Bit Rate:	Vectors Used:	SNR (dB):	segSNR (dB):
16 kbps	1,2,3,4	15.93	18.19
12 kbps	1,2,3	8.88	10.77
8 kbps	1,3	5.39	6.74
4 kbps	1	0.72	0.74

Table F.1: SNR Values for LD-CELP Variants

Table F.1 shows SNR and segmental SNR for the 12 and 8 kbps coders discussed above. For comparison reasons the figures for G.728 16 kbps LD-CELP are also shown. The final table entry is for a 4 kbps system where only one vector out of the four in each adaptation cycle is transmitted. As suggested by the low SNR values, the performance of the 4 kbps system is quite poor, and it does not constitute even a half-hearted approach to coding at that rate. The values shown in the table are obtained with the absence of postfiltering, although postfiltering was also applied to the coders.

As is often the case with speech coding, the SNR values for the systems at rates of 12 and 8 kbps do not give a clear indication of the subjective performance. This fact is clearly demonstrated via the use of the LD-CELP postfilter, which improves perceptual quality, yet decreases the SNR. With the inclusion of postfiltering, and judged purely by informal listening tests, we would rate the performance at 12 kbps to be equivalent to that of 'reasonable' mobile communications quality, and that at 8 kbps to be similar to fairly noisy mobile quality.

It might be reasonably expected that codebook redesign to take account of the altered requirements of the lower rate systems would result in further performance improvements. Again, this is beyond the scope of the current work.

The simplicity of the encoder modifications may mean that there are some potential applications for the approach discussed in this appendix. However, as already mentioned, it is natural to expect higher speech quality from a coder designed specifically for the bit rates required. Probably the most important conclusion that can be obtained from the results of this work is that the high level of robustness of the LD-CELP coder suggests that there is still plenty of scope for further useful research in speech coding.

Bibliography

[1] F. Adachi, D. Falconer, and B. Gudmundson. "TDMA for Personal Wireless Communications". *IEEE Communications Magazine*, Special Issue on Wireless Personal Communications 1994.

[2] B. D. O. Anderson. "Properties of Optimal Linear Smoothing". *IEEE Transactions on Automatic Control*, AC-14(1), February 1969.

[3] B. D. O. Anderson, R. R. Bitmead, C. R. Johnson, Jr., P. V. Kokotovic, R. L. Kosut, I. M. Y. Mareels, L. Praly, and B. D. Riedle. *"Stability of Adaptive Systems: Passivity and Averaging Analysis"*. MIT Press, 1986.

[4] B. D. O. Anderson and J. B. Moore. *"Optimal Filtering"*. Prentice-Hall, 1979.

[5] J. B. Anderson and S. Mohan. *"Source and Channel Coding: An Algorithmic Approach"*. Kluwer Academic Publishers, 1991.

[6] ANSI T1 Contribution T1A1/93-26. "Terms of Reference for "Toll-Quality" 4 kbit/s Speech Coding". *Boulder, Colorado*, April 26-30 1993.

[7] J. C. Arnbak. "The European (R)evolution of Wireless Digital Networks". *IEEE Communications Magazine*, 31(9), September 1993.

[8] B. S. Atal. "Predictive Coding of Speech at Low Bit Rates". *IEEE Trans. Communications*, COM-30(4), April 1982.

[9] B. S. Atal, V. Cuperman, and A. Gersho. *"Advances in Speech Coding"*. Kluwer Academic Publishers, 1991.

[10] B. S. Atal, V. Cuperman, and A. Gersho. *"Speech and Audio Coding for Wireless and Network Applications"*. Kluwer Academic Publishers, 1993.

[11] AT&T. "A Solution for the P50 Problem?". *Contribution to CCITT SG XV/Q.21*, May 1992.

[12] AT&T. "G.728 Decoder Modifications for Frame Erasure Concealment". *Contribution to ITU-T SG XV/Q.5*, March 1994.

[13] D. M. Balston and R. C. V. Macario. *"Cellular Radio Systems"*. Artech House, 1993.

[14] T. Bell, I. H. Witten, and J. G. Cleary. "Modeling for Text Compression". *ACM Computing Surveys*, 21(4), December 1989.

[15] T. C. Bell, J. G. Cleary, and I. H. Witten. *"Text Compression"*. Prentice-Hall, 1990.

[16] T. Berger. *"Rate Distortion Theory"*. Prentice-Hall, 1971.

[17] T. Berger. "Minimum Entropy Quantizers and Permutation Codes". *IEEE Trans. Information Theory*, IT-28(2), March 1982.

[18] S. Bittanti, A. J. Laub, and J. C. Willems. *"The Riccati Equation"*. Springer-Verlag, 1991.

[19] R. E. Blahut. *"Digital Transmission of Information"*. Addison-Wesley, 1990.

[20] J. Blanz, A. Klein, M. Nasshan, and A. Steil. "Performance of a Cellular Hybrid C/TDMA Mobile Radio System Applying Joint Detection and Coherent Receiver Antenna Diversity". *IEEE Journal Selected Areas in Communications*, 12(4), May 1994.

[21] C. G. Boncelet, Jr. "Block Arithmetic Coding for Source Compression". *IEEE Trans. Information Theory*, 39(5), September 1993.

[22] M. Bonnet, O. Macchi, and M. Jaidane-Saidane. "Mistracking in Successive PCM/ADPCM Transcoders". *IEEE Trans. Communications*, 37(8), August 1989.

[23] M. Bonnet, O. Macchi, and M. Jaidane-Saidane. "Theoretical Analysis of the ADPCM CCITT Algorithm". *IEEE Trans. Communications*, 38(6), June 1990.

[24] R. J. Brown and A. P. Sage. "New Algorithms for Estimation with Sequentially Correlated Measurement Noise". *International Journal of Systems Science*, 2(3):253–270, 1971.

[25] J. P. Campbell, Jr., T. E. Tremain, and V. C. Welch. "The Federal Standard 1016 4800 bps CELP Voice Coder". *Digital Signal Processing, A Review Journal*, 1 of 3:145–155, July 1991.

[26] J. P. Campbell, Jr., V. C. Welch, and T. E. Tremain. "An Expandable Error-Protected 4800 bps CELP Coder (U.S. Federal Standard 4800 bps Voice Coder". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Glasgow, Scotland, May 1989.

[27] A. B. Carlson. *"Communication Systems"*. McGraw-Hill, 1986.

[28] L. Cellario and D. Sereno. "Variable Rate Speech Coding for UMTS". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[29] S. W. Chan, G. C. Goodwin, and K. S. Sin. "Convergence Properties of the Riccati Difference Equation in Optimal Filtering of Nonstabilizable Systems". *IEEE Transactions on Automatic Control*, AC-29(2), February 1984.

[30] W.-W. Chang and J. D. Gibson. "Smoothed DPCM Codes". *IEEE Trans. Communications*, 39(9), September 1991.

[31] W.-W. Chang and C.-Y. Yang. "Kalman Filtering of Colored Quantization Noise in DPCM Systems". In *IEEE Workshop on Speech Coding for Telecommunications*, Canada, 1991.

[32] J.-H. Chen and R. V. Cox. "Convergence and Numerical Sensitivity of Backward-Adaptive LPC Predictor". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[33] J.-H. Chen and R. V. Cox. "The Creation and Evolution of 16 kbit/s LD-CELP: From Concept to Standard". *Speech Communication*, June 1993.

[34] J.-H. Chen, R. V. Cox, Y.-C. Lin, N. S. Jayant, and M. J. Melchner. "A Low-Delay CELP Coder for the CCITT 16 kb/s Speech Coding Standard". *IEEE Journal Selected Areas in Communications*, pages 830–849, June 1992.

[35] J.-H. Chen, Y.-C. Lin, and R. V. Cox. "A Fixed-Point 16 kb/s LD-CELP Algorithm". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Toronto, Canada, May 1991.

[36] D. Chevion, E. D. Karnin, and E. Walach. "High Efficiency, Multiplication Free Approximation of Arithmetic Coding". In *Proc. IEEE Data Compression Conference*, Snowbird, Utah, April 1991.

[37] D. L. Cohn and J. L. Melsa. "The Relationship Between an Adaptive Quantizer and a Variance Estimator". *IEEE Trans. Information Theory*, November 1975.

[38] A. Coleman, N. Gleiss, J. Sotscheck, P. Usai, and H. Scheuermann. "Subjective Performance Evaluation of the RPE-LTP Codec for the Pan-European Cellular Digital Mobile Radio System". In *Proc. IEEE Global Telecommunications Conference*, Dallas, Texas, November 1989.

[39] G. R. Cooper and C. D. McGillem. *"Modern Communications and Spread Spectrum"*. McGraw-Hill, 1986.

[40] D. C. Cox. "Universal Digital Portable Radio Communications". *Proceedings of the IEEE*, 75(4):436–477, April 1987.

[41] R. V. Cox, W. B. Kleijn, and P. Kroon. "Robust CELP Coders for Noisy Backgrounds and Noisy Channels". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Glasgow, Scotland, May 1989.

[42] S. Crisafulli. *"Adaptive Speech Coding via Feedback Techniques"*. PhD thesis, Australian National University, 1992.

[43] S. Crisafulli and R. R. Bitmead. "Adaptive Quantization: A Kalman Filtering Approach". *IEEE Trans. Communications*, 41(5), May 1993.

[44] S. Crisafulli, J. D. Mills, and R. R. Bitmead. "Kalman Filtering Techniques in Speech Coding". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, San Francisco, March 1992.

[45] S. Crisafulli, G. J. Rey, C. R. Johnson, and R. A. Kennedy. "A Coupled Approach to ADPCM Adaptation". *IEEE Trans. Speech and Audio Processing*, 2(1), 1994.

[46] P. Cummiskey, N. S. Jayant, and J. L. Flanagan. "Adaptive Quantization in Differential PCM Coding of Speech". *The Bell System Technical Journal*, 52(7), September 1973.

[47] V. Cuperman. "Advances in Speech Coding for Wireless Communications". In *Proc. Sixth International Conf. on Wireless Communications (Wireless 94)*, Calgary, Canada, July 1994.

[48] V. Cuperman, A. Gersho, R. Pettigrew, J. J. Shynk, and J.-H. Yao. "Backward Adaptation for Low Delay Vector Excitation Coding of Speech at 16 kbit/s". In *Proc. IEEE Global Telecommunications Conference*, Dallas, Texas, November 1989.

[49] V. Cuperman, F.-H. Liu, and P. Ho. "Soft Decision Vector Quantization for Noisy Channels". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[50] I. S. Dedes, D. R. Vaman, and C. V. Chakravarthy. "Variable Bit Rate Adaptive Predictive Coder". *IEEE Trans. Signal Processing*, 40(3), March 1992.

[51] A. DeJaco, W. Gardner, P. Jacobs, and C. Lee. "QCELP: The North American CDMA Digital Cellular Variable Rate Speech Coding Standard". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[52] C. A. Desoer and M. Vidyasagar. *"Feedback Systems: Input-Output Properties"*. Academic Press, 1975.

[53] S. Dimolitsas and C. Ravishankar. "Future Objectives in Low-Rate Speech Coding Technology Standardization". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[54] J. G. Dunham and A. Ghosh. "Improving DPCM System Performance". In *Proc. IEEE Global Telecommunications Conference*, Dallas, Texas, November 1989.

[55] W. W. Erdman. "Wireless Communications: A Decade of Progress". *IEEE Communications Magazine*, 31(12), December 1993.

[56] T. Eriksson and J. Sjoberg. "Evolution of Variable Rate Speech Coders". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[57] N. Farvardin and J. W. Modestino. "Optimum Quantizer Performance for a Class of Non-Gaussian Memoryless Sources". *IEEE Trans. Information Theory*, IT-30(3), May 1984.

[58] T. J. Ferguson and J. H. Rabinowitz. "Self-Synchronizing Huffman Codes". *IEEE Trans. Information Theory*, IT-30(4), July 1984.

[59] B. Fette, C. Bergstrom, S. You, and C. Jaskie. "High Quality 2400 bps Vocoder Research". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[60] G. Feygin, P. G. Gulak, and P. Chow. "Minimizing Error and VLSI Complexity in the Multiplication Free Approximation of Arithmetic Coding". In *Proc. IEEE Data Compression Conference*, Snowbird, Utah, March 1993.

[61] T. Fischer and M. Wang. "Entropy-Constrained Trellis-Coded Quantization". *IEEE Trans. Information Theory*, 38(2), March 1992.

[62] M. Fischetti. "The Cellular Phone Scare". *IEEE Spectrum*, June 1993.

[63] M. Foodeei and P. Kabal. "Low-Delay CELP and Tree Coders: Comparison and Performance Improvements". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Toronto, Canada, May 1991.

[64] G. D. Forney, Jr. "The Viterbi Algorithm". *Proceedings of the IEEE*, 61(3):268–278, March 1973.

[65] J. E. Fowler and S. C. Ahalt. "Robust, Variable Bit-Rate Coding Using Entropy-Biased Codebooks". In *Proc. IEEE Data Compression Conference*, Snowbird, Utah, March 1993.

[66] W. Gardner, P. Jacobs, and C. Lee. "QCELP: A Variable Rate Speech Coder for CDMA Digital Cellular". In *Speech and Audio Coding for Wireless and Network Applications.* Kluwer Academic Publishers, 1993.

[67] A. Gersho. "Advances in Speech and Audio Compression". *Proceedings of the IEEE*, 82(6), June 1994.

[68] A. Gersho and R. M. Gray. *"Vector Quantization and Signal Compression".* Kluwer Academic Publishers, 1992.

[69] A. Gersho and E. Paksoy. "Variable Rate Speech Coding for Cellular Networks". In *Speech and Audio Coding for Wireless and Network Applications.* Kluwer Academic Publishers, 1993.

[70] I. A. Gerson and M. A. Jasiuk. "Vector Sum Excited Linear Prediction (VSELP) Speech Coding at 8 kbps". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Albuquerque, New Mexico, April 1990.

[71] I. A. Gerson and M. A. Jasiuk. "Vector Sum Excited Linear Prediction (VSELP)". In *Advances in Speech Coding*. Kluwer Academic Publishers, 1991.

[72] J. D. Gibson. "Sequentially Adaptive Backward Prediction in ADPCM Speech Coders". *IEEE Trans. Communications*, COM-26(1), January 1978.

[73] J. D. Gibson. "Adaptive Prediction in Speech Differential Encoding Systems". *Proceedings of the IEEE*, 68(4):488–525, April 1980.

[74] J. D. Gibson. "Quantization Noise Filtering in ADPCM Systems". *IEEE Trans. Systems, Man, and Cybernetics*, SMC-10(8), August 1980.

[75] J. D. Gibson and W.-W. Chang. "Fractional Rate Multi-Tree Speech Coding". In *Proc. IEEE Global Telecommunications Conference*, Dallas, Texas, November 1989.

[76] J. D. Gibson, B. Koo, and S. D. Gray. "Filtering of Colored Noise for Speech Enhancement and Coding". *IEEE Trans. Signal Processing*, 39(8), August 1991.

[77] H. Gish and J. N. Pierce. "Asymptotically Efficient Quantizing". *IEEE Trans. Information Theory*, IT-14, September 1968.

[78] D. J. Goodman, G. B. Lockhart, O. J. Wasem, and W.-C. Wong. "Waveform Substitution Techniques for Recovering Missing Speech Segments in Packet Voice Communications". *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-34(6), December 1986.

[79] A. H. Gray and J. D. Markel. "Distance Measures for Speech Processing". *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-24(5), October 1976.

[80] R. S. Hamming. "Error Detecting and Correcting Codes". *The Bell System Technical Journal*, 29:147–160, 1950.

[81] E. Harborg, J. E. Knudsen, A. Fuldseth, and F. T. Johansen. "A Real-time Wideband CELP Coder for a Videophone Application". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Adelaide, Australia, April 1994.

[82] S. Haykin. *"Adaptive Filter Theory"*. Prentice-Hall, 1991.

[83] K. Hellwig, P. Vary, D. Massaloux, J. P. Petit, C. Garland, and M. Rosso. "Speech Codec for the European Mobile Radio System". In *Proc. IEEE Global Telecommunications Conference*, Dallas, Texas, November 1989.

[84] M. Honda and F. Itakura. "Bit Allocation in Time and Frequency Domains for Predictive Coding of Speech". *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-32(3), June 1984.

[85] M. L. Honig and D. G. Messerschmitt. "Comparison of Adaptive Linear Prediction Algorithms in ADPCM". *IEEE Trans. Communications*, COM-30(7), July 1982.

[86] P. G. Howard and J. S. Vitter. "New Methods for Lossless Image Compression Using Arithmetic Coding". In *Proc. IEEE Data Compression Conference*, Snowbird, Utah, April 1991.

[87] P. G. Howard and J. S. Vitter. "Parallel Lossless Image Compression Using Huffman and Arithmetic Coding". In *Proc. IEEE Data Compression Conference*, Snowbird, Utah, March 1992.

[88] P. G. Howard and J. S. Vitter. "Design and Analysis of Fast Text Compression Based on Quasi-Arithmetic Coding". In *Proc. IEEE Data Compression Conference*, Snowbird, Utah, March 1993.

[89] D. A. Huffman. "A Method for the Construction of Minimum-Redundancy Codes". *Proceedings of the IRE*, pages 1098–1101, September 1952.

[90] A. Husain and V. Cuperman. "Classification and Spectral Extrapolation Based Packet Reconstruction for Low-Delay Speech Coding". In *To appear Proc. IEEE Global Telecommunications Conference*, San Francisco, California, December 1994.

[91] A. Husain and V. Cuperman. "Techniques for Missing Packet Reconstruction in Speech Coding". In *Proc. 17th Bi-annual Queens Symposium on Communications*, 1994.

[92] A. N. Ince. *"Digital Speech Processing: Speech Coding, Synthesis and Recognition"*. Kluwer Academic Publishers, 1992.

[93] Special Issue. "Satellite and Terrestrial Systems and Services for Travellers". *IEEE Communications Magazine*, November 1991.

[94] Special Issue. "PCS: The Second Generation". *IEEE Communications Magazine*, December 1992.

[95] Special Issue. "Realizing Global Communications". *IEEE Communications Magazine*, October 1992.

[96] Special Issue. "Divestiture: Ten Year Later". *IEEE Communications Magazine*, December 1993.

[97] Special Issue. "Speech Processing". *IEEE Communications Magazine*, November 1993.

[98] Special Issue. "Standards: Their Global Impact". *IEEE Communications Magazine*, January 1994.

[99] V. Iyengar and P. Kabal. "A Low Delay 16 kb/s Speech Coder". *IEEE Trans. Signal Processing*, 39(5), May 1991.

[100] G. Jacobson. "Random Access in Huffman-Coded Files". In *Proc. IEEE Data Compression Conference*, Snowbird, Utah, March 1992.

[101] N. S. Jayant. "Adaptive Quantization with a One-Word Memory". *The Bell System Technical Journal*, 52:1119–1144, September 1973.

[102] N. S. Jayant. "Coding of Speech at Low Bit Rates". *IEEE Spectrum*, August 1986.

[103] N. S. Jayant. "High-Quality Coding of Telephone Speech and Wideband Audio". *IEEE Communications Magazine*, 28(1), January 1990.

[104] N. S. Jayant. "Signal Compression: Technology Targets and Research Directions". *IEEE Journal Selected Areas in Communications*, 10(5), June 1992.

[105] N. S. Jayant and J.-H. Chen. "Speech Coding with Time-Varying Bit Allocations to Excitation and LPC Parameters". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Glasgow, Scotland, May 1989.

[106] N. S. Jayant and P. Noll. *"Digital Coding of Waveforms: Principles and Applications to Speech and Video"*. Prentice-Hall, 1984.

[107] R. L. Joshi, V. J. Crump, and T. R. Fischer. "Arithmetic Coded (Uniform) Trellis Coded Quantization". In *IEEE Intelligent Signal Processing and Communication Systems International Workshop*, Sendai, Japan, October 1993.

[108] T. Kailath. "A View of Three Decades of Linear Filtering Theory". *IEEE Trans. Information Theory*, IT-20(2), March 1974.

[109] Y. Kao and J. S. Baras. "A New Deterministic Codebook Structure for CELP Speech Coding". In *IEEE International Symposium on Information Theory*, San Antonio, Texas, January 1993.

[110] A. Kataoka and T. Moriya. "A Backward Adaptive 8kbit/s Speech Coder Using Conditional Pitch Prediction". In *Proc. IEEE Global Telecommunications Conference*, December 1991.

[111] A. Kataoka, T. Moriya, and S. Hayashi. "An 8-kbit/s Speech Coder based on Conjugate Structure CELP". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Minneapolis, Minnesota, April 1993.

[112] A. Kataoka, T. Moriya, and S. Hayashi. "Conjugate Structure CELP Coder for the CCITT 8-kbit/s Standardization Candidate". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[113] A. Kataoka, T. Moriya, and S. Hayashi. "Implementation and Performance of An 8-kbit/s Conjugate Structure CELP Speech Coder". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Adelaide, Australia, April 1994.

[114] R. A. Kennedy, B. D. O. Anderson, and R. R. Bitmead. "Channels Leading to Rapid Error Recovery for Decision Feedback Equilizers". *IEEE Trans. Commun.*, 37(11):1126–1135, November 1989.

[115] R. A. Kennedy and C. R. Johnson, Jr. "Encoder Stability Study for the 32 kbit/s CCITT G.721 ADPCM Standard based on a Simplified Error Model". In *Proc. The Second International Symposium on Signal Processing and its Applications*, Gold Coast, Australia, August 1990.

[116] J. C. Kieffer. "A Survey of the Theory of Source Coding with a Fidelity Criterion". *IEEE Trans. Information Theory*, 39(5), September 1993.

[117] N. Kitawaki, H. Nagabuchi, M. Taka, and K. Takahashi. "Speech Coding Technology for ATM Networks". *IEEE Communications Magazine*, 28(1), January 1990.

[118] B. Z. Kobb. "Personal Wireless". *IEEE Spectrum*, June 1993.

[119] B. Koo, J. D. Gibson, and S. D. Gray. "Filtering of Colored Noise for Speech Enhancement and Coding". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Glasgow, Scotland, May 1989.

[120] P. Kroon, E. F. Deprettere, and R. J. Sluyter. "Regular-Pulse Excitation–A Novel Approach to Effective and Efficient Multipulse Coding of Speech". *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-34(5), October 1986.

[121] A. D. Kucar. "Mobile Radio: An Overview". *IEEE Communications Magazine*, 29(11), November 1991.

[122] E. A. Lee and D. G. Messerschmitt. *"Digital Communication"*. Kluwer Academic Publishers, 1988.

[123] O. Macchi and C. Uhl. "Stability of the DPCM Transmission System". *IEEE Trans. Circuits and Systems*, 39(10), October 1992.

[124] J. Makhoul and M. Berouti. "Adaptive Noise Spectral Shaping and Entropy Coding in Predictive Coding of Speech". *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-27, February 1979.

[125] K. Mano and T. Moriya. "4.8 kbit/s Delayed Decision CELP Coder using Tree Coding". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Albuquerque, New Mexico, April 1990.

[126] K. Mano, T. Moriya, S. Miki, and H. Ohmuro. "Design of a PSI-CELP Coder for Mobile Communications". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[127] M. W. Marcellin. "On Entropy-Constrained Trellis Coded Quantization". *IEEE Trans. Communications*, 42(1), January 1994.

[128] M. W. Marcellin and T. R. Fischer. "Trellis Coded Quantization of Memoryless and Gauss-Markov Sources". *IEEE Trans. Communications*, 38(1), January 1990.

[129] E. D. Martino. "Speech Quality Evaluation of the European, North-American and Japanese Speech Coding Standards for Digital Cellular Systems". In *Speech and Audio Coding for Wireless and Network Applications.* Kluwer Academic Publishers, 1993.

[130] S. Miki, K. Mano, T. Moriya, K. Oguchi, and H. Ohmuro. "A Pitch Synchronous Innovation CELP (PSI-CELP) Coder for 2-4 kbit/s". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Adelaide, Australia, April 1994.

[131] A. Moffat. "Implementing the PPM Data Compression Scheme". *IEEE Trans. Communications*, 38(11), November 1990.

[132] B. M. Mouy and P. E. de La Noue. "Voice Transmission at a Very Low Bit Rate On a Noisy Channel: 800 bps Vocoder with Error Protection to 1200 bps". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, San Francisco, March 1992.

[133] H. Nakada and K. Sato. "Variable Rate Speech Coding for Asynchronous Transfer Mode". *IEEE Trans. Communications*, 38(3), March 1990.

[134] S. H. Nam and J. D. Gibson. "Analysis of the Smoothed Residual-Driven Algorithm for Speech Coders". *IEEE Trans. Speech and Audio Proc.*, 2(3), July 1994.

[135] J. E. Natvig, S. Hansen, and J. de Brito. "Speech Processing in the Pan-European Digital Mobile Radio System (GSM) - System Overview". In *Proc. IEEE Global Telecommunications Conference*, Dallas, Texas, November 1989.

[136] H. Ohmuro, T. Moriya, K. Mano, and S. Miki. "Coding of LSP Parameters Using Interframe Moving Average Prediction and Multi-Stage Vector Quantization". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[137] E. Paksoy and A. Gersho. "A Variable Rate Speech Coding Algorithm for Cellular Networks". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[138] K. K. Paliwal and A. Basu. "A Speech Enhancement Method Based on Kalman Filtering". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1987.

[139] Y. Perl, V. Maram, and N. Kadakuntla. "The Cascading of the LZW Compression Algorithm with Arithmetic Coding". In *Proc. IEEE Data Compression Conference*, Snowbird, Utah, April 1991.

[140] R. Pettigrew and V. Cuperman. "Backward Pitch Prediction for Low-Delay Speech Coding". In *Proc. IEEE Global Telecommunications Conference*, Dallas, Texas, November 1989.

[141] G. Pirani and C. Scagliola. "Performance Analysis of DPCM Speech-Transmission Systems using Kalman Predictors". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, April 1976.

[142] K. Popat and K. Zeger. "Robust Quantization of Memoryless Sources Using Dispersive FIR Filters". *IEEE Trans. Communications*, 40(11), November 1992.

[143] J. G. Proakis. *"Digital Communications"*. McGraw-Hill, 1989.

[144] L. R. Rabiner, M. R. Sambur, and C. E. Schmidt. "Applications of a Nonlinear Smoothing Algorithm to Speech Processing". *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-23(6), December 1975.

[145] L. R. Rabiner and R. W. Schafer. *"Digital Processing of Speech Signals"*. Prentice-Hall, 1978.

[146] T. V. Ramabadran and D. L. Cohn. "An Adaptive Algorithm for the Compression of Computer Data". *IEEE Trans. Communications*, 37(4), April 1989.

[147] T. V. Ramabadran and D. Sinha. "Speech Coding using Least-Squares Estimation". In *Advances in Speech Coding*. Kluwer Academic Publishers, 1991.

[148] K. Ramachandran, A. Ortega, K. M. Uz, and M. Vetterli. "Multiresolution Broadcast for Digital HDTV Using Joint Source/Channel Coding". *IEEE Journal Selected Areas in Communications*, 11(1), January 1993.

[149] R. P. Ramachandran and P. Kabal. "Stability and Performance Analysis of Pitch Filters in Speech Coders". *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-35(7), July 1987.

[150] J. J. Rissanen and G. G. Langdon. "Arithmetic Coding". *IBM Journal of Research and Development*, 23(2):149–162, March 1979.

[151] J. J. Rissanen and K. M. Mohiuddin. "A Multiplication-Free Multialphabet Arithmetic Code". *IEEE Trans. Communications*, 37(2), February 1989.

[152] R. C. Rose and T. P. Barnwell. "Design and Performance of an Analysis-by-Synthesis Class of Predictive Speech Coders". *IEEE Trans. Acoust., Speech, Signal Processing*, 38(9), September 1990.

[153] D. Rowe and P. Secker. "A Robust 2400 bit/s MBE-LPC Speech Coder Incorporating Joint Source and Channel Coding". In *Proc. IEEE International*

Conference on Acoustics, Speech, and Signal Processing, San Francisco, March 1992.

[154] C. M. Rush. "How WARC '92 Will Affect Mobile Services". *IEEE Communications Magazine*, 30(10), October 1992.

[155] R. Salami, C. Laflamme, and J.-P. Adoul. "ACELP Speech Coding at 8 kbit/s with a 10 ms Frame: A Candidate for CCITT Standardization". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[156] R. Salami, C. Laflamme, and J.-P. Adoul. "8 kbit/s ACELP Coding of Speech with 10 ms Speech-Frame: A Candidate for CCITT Standardization". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Adelaide, Australia, April 1994.

[157] S. A. Savari. "Arithmetic Coding for Finite-State Noiseless Channels". *IEEE Trans. Information Theory*, 40(1), January 1994.

[158] S. A. Savari and R. G. Gallager. "Arithmetic Coding for Memoryless Cost Channels". In *Proc. IEEE Data Compression Conference*, Snowbird, Utah, March 1992.

[159] K. Sayood and J. C. Borkenhagen. "Use of Residual Redundancy in the Design of Joint Source/Channel Coders". *IEEE Trans. Communications*, 39(6), June 1991.

[160] C. E. Shannon. "A Mathematical Theory of Communication". *The Bell System Technical Journal*, 27:379–423;623–656, 1948.

[161] Y. Shoham. "High-Quality Speech Coding at 2.4 to 4.0 kbps based on Time-Frequency Interpolation". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Minneapolis, Minnesota, April 1993.

[162] Y. Shoham. "Speech Coding at 2.4 kbps and Below via Time-Frequency Interpolation". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[163] D. Sinha. "A New Approach to Speech Coding using Sparse Measurements and Least-Squares Estimation". Master's thesis, Department of Electrical Engineering and Computer Engineering, Iowa State University, 1989.

[164] R. Soheili, A. M. Kondoz, and B. G. Evans. "Techniques for Improving the Quality of LD-CELP Coders at 8 kb/s". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, San Francisco, March 1992.

[165] F. K. Soong and B.-H. Juang. "Optimal Quantization of LSP Parameters". *IEEE Trans. Speech and Audio Proc.*, 1(1), January 1993.

[166] C. B. Southcott, D. Freeman, G. Cosier, D. Sereno, A. Van der Krogt, A. Gilloire, and H. J. Braun. "Voice Control of the Pan-European Digital Mobile Radio System". In *Proc. IEEE Global Telecommunications Conference*, Dallas, Texas, November 1989.

[167] C. E. De Souza, M. R. Gevers, and G. C. Goodwin. "Riccati Equations in Optimal Filtering of Nonstabilizable Systems having Singular State Transition Matrices". *IEEE Transactions on Automatic Control*, AC-31(9), September 1986.

[168] A. S. Spanias. *"Speech Coding: A Tutorial Review"*. Arizona State University, 1994.

[169] Raymond Steele. *"Mobile Radio Communications"*. Pentech Press, 1992.

[170] H.-Y. Su and P. Mermelstein. "Improving the Speech Quality of Cellular Mobile Systems under Heavy Fading". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, San Francisco, March 1992.

[171] T. Taniguchi, F. Amamo, and S. Unagami. "Combined Source and Channel Coding Based on Multimode Coding". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Albuquerque, New Mexico, April 1990.

[172] T. Taniguchi, M. Johnson, and Y. Ohta. "Pitch Sharpening for Perceptually Improved CELP, and the Sparse-Delta Codebook for Reduced Computation". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Toronto, Canada, May 1991.

[173] T. Taniguchi, Y. Ohta, and J. Yao. "Structured Codebook for Variable-Rate CELP Coding". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[174] J. Teuhola and T. Raita. "Arithmetic Coding into Fixed-Length Codewords". *IEEE Trans. Information Theory*, 40(1), January 1994.

[175] T. Thorpe. "The Mean Squared Error Criterion: Its Effect on the Performance of Speech Coders". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Glasgow, Scotland, May 1989.

[176] Y. Tohkura, F. Itakura, and S. Hashimoto. "Spectral Smoothing Technique in PARCOR speech analysis-synthesis". *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-26:587–596, December 1978.

[177] I. Trancoso, C. Ribeiro, L. Almeida, L. Oliveira, and J. Marques. "CELP: A Candidate for GSM Half-Rate Coding". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Albuquerque, New Mexico, April 1990.

[178] T. E. Tremain. "The Government Standard Linear Predictive Coding Algorithm: LPC-10". In *Speech Technology*, pages 40–49, April 1982.

[179] V. A. Vaishampayan and J. Domaszewicz. "Design of Entropy-Constrained Multiple-Description Scalar Quantizers". *IEEE Trans. Information Theory*, 40(1), January 1994.

[180] V. K. Varma. "Testing Speech Coders for Usage in Wireless Communications Systems". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[181] M. Vidyasagar. *"Nonlinear Systems Analysis"*. Prentice Hall, 1993.

[182] S. Wang, A. Sekey, and A. Gersho. "An Objective Measure for Predicting Subjective Quality of Speech Coders". *IEEE Journal Selected Areas in Communications*, 10(5), June 1992.

[183] C. R. Watkins, R. R. Bitmead, and S. Crisafulli. "Destabilization Effects of Adaptive Quantization in ADPCM". *To appear IEEE Trans. Speech and Audio Proc.*, March 1995.

[184] C. R. Watkins, S. Crisafulli, and R. R. Bitmead. "Reduced Complexity Kalman Filtering for Signal Coding". In *IEEE Intelligent Signal Processing and Communication Systems International Workshop*, Sendai, Japan, October 1993.

[185] C. R. Watkins, S. Crisafulli, R. R. Bitmead, and R. J. Orsi. "Variable Bit Rate ADPCM via Arithmetic Coding". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Adelaide, Australia, April 1994.

[186] L. Watts and V. Cuperman. "A Vector ADPCM Analysis-by-Synthesis Configuration for 16 kbit/s Speech Coding". In *Proc. IEEE Global Telecommunications Conference*, Hollywood, Florida, November 1988.

[187] V. C. Welch and T. E. Tremain. "A New Government Standard 2400 bps Speech Coder". In *IEEE Workshop on Speech Coding for Telecommunications*, Quebec, Canada, October 1993.

[188] I. H. Witten, R. M. Neal, and J. G. Cleary. "Arithmetic Coding for Data Compression". *Communications of the ACM*, 30(6), June 1987.

[189] H. C. Woo and J. D. Gibson. "Low Delay Tree Coding of Speech at 8 kbit/s". *IEEE Trans. Speech and Audio Proc.*, 2(3), July 1994.

[190] Y. Wu, H. B. Hansen, K. J. Larsen, H. Nielsen, and J. A. Sorensen. "High Performance Coder: A Possible Candidate for the GSM Half-Rate System". In *Proc. International Conference on Acoustics, Speech, and Signal Processing*, Toronto, Canada, May 1991.

[191] J.-H. Yao, J. J. Shynk, and A. Gersho. "Low-Delay VXC at 8 kbit/s with Interframe Coding". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, San Francisco, March 1992.

[192] M. Yong. "Study of Voice Packet Reconstruction Methods Applied to CELP Speech Coding". In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, San Francisco, March 1992.

[193] M. Yong and A. Gersho. "Subband Vector Excitation Coding with Adaptive Bit-Allocation". In *Proc. IEEE Global Telecommunications Conference*, Dallas, Texas, November 1989.

[194] K. Zeger and A. Gersho. "Pseudo-Gray Coding". *IEEE Trans. Communications*, 38(12), December 1990.

Glossary

AC: Arithmetic Coding

AC-ADPCM: Arithmetic Coding ADPCM

AC-APC: Arithmetic Coding APC

ACELP: Algebraic Code Excited Linear Prediction

ADPCM: Adaptive Differential Pulse Code Modulation

AGC: Automatic Gain Control

AM: Amplitude Modulation

AMPS: Advanced Mobile Phone System (Analog)

ANSI: American National Standards Institute

APC: Adaptive Predictive Coder

AR: Auto Regressive

ARMA: Auto Regressive Moving Average

ASIC: Application Specific Integrated Circuit

AT&T: American Telephone and Telegraph Company (now AT&T Corp.)

ATM: Asynchronous Transfer Mode

BSD: Bark Spectral Distortion

BWE: Bandwidth Expansion

CCIR: International Radio Consultative Committee

CCITT: International Telegraph and Telephone Consultative Committee

CDMA: Code Division Multiple Access

CELP: Code Excited Linear Prediction

CEPT: European Conference of Post and Telecommunications

CRC: Cyclic Redundancy Check

CS-CELP: Conjugate Structure Code Excited Linear Prediction

CT: Cordless Telephone

CTIA: Cellular Telecommunications Industry Association

DRT: Diagnostic Rhyme Test

DSP: Digital Signal Processor

DTX: Discontinuous Transmission

EFRC: Enhanced Fixed Rate Codec

EVRC: Enhanced Variable Rate Codec

FDD: Frequency Domain Duplexing

FDMA: Frequency Division Multiple Access

FIR: Finite Impulse Response

FM: Frequency Modulation

FPLMTS: Future Public Land Mobile Telecommunications Systems

GSM: Groupe Speciale Mobile, OR Global System Mobile

G.711: 64 kbps CCITT Toll Quality PCM Speech Coding Standard

G.721: 32 kbps CCITT Toll Quality ADPCM Speech Coding Standard

G.728: 16 kbps CCITT Toll Quality LD-CELP Speech Coding Standard

HDTV: High Definition Television

IC: Integrated Circuit

IIR: Infinite Impulse Response

IRS: Intermediate Reference System

ISDN: Integrated Services Digital Network

ITU: International Telecommunication Union

ITU-R: ITU-Radiocommunication Sector (formerly CCIR)

ITU-T: ITU-Telecommunication Standardization Sector (formerly CCITT)

JDC: Japanese Digital Cellular

kbps: kilo bits per second

KF: Kalman Filter

LAR: Log Area Ratio

LD-CELP: Low Delay Code Excited Linear Prediction

LEO: Low Earth Orbit

LPC: Linear Predictive Coder

LSP: Line Spectral Pair

MA: Moving Average

Mips: Millions of instructions per second

MOS: Mean Opinion Score

MSE: Mean Square Error

NFC: Noise Feedback Coding

NTT: Nippon Telegraph and Telephone

PCM: Pulse Code Modulation

PCS: Personal Communications Systems

PM: Phase Modulation

PSI-CELP: Pitch Synchronous Innovation Code Excited Linear Prediction

QAM: Quadrature Amplitude Modulation

QCELP: Qualcomm Code Excited Linear Prediction

QPSK: Quadrature Phase Shift Keying

RAM: Random Access Memory

RF: Radio Frequency

RPE-LTP: Residual Pulse Excitation – Long Term Prediction

SNR: Signal to Noise Ratio

TDD: Time Domain Duplexing

TDMA: Time Division Multiple Access

VAD: Voice Activity Detection

VLSI: Very Large Scale Integration

VQ: Vector Quantization

VSELP: Vector Sum Excited Linear Prediction

WMSE: (Perceptually) Weighted Mean Square Error

WNC: White Noise Correction