Advanced techniques for Error Robust Audio and Speech Communications

Huseyin Oztoprak

Submitted for the Degree of
Doctor of Philosophy
from the
University of Surrey

UNIVERSITY OF SURREY

I-Lab Multimedia Communications Research Centre for Vision, Speech and Signal Processing Faculty of Engineering and Physical Sciences University of Surrey Guildford, Surrey GU2 7XH, UK

October 2011

© H. Oztoprak 2011
Abstract

The past decades have seen a very fast growth of the telecommunications industry. Mobile telephony has evolved from a specialist application to being commonplace and affordable, and is now a mass-market industry. Like mobile telephony, multimedia communications has also evolved, where voice, video and data are all to be integrated into one device. Today's audio and speech communication systems are characterised by heterogeneous networks, and varying natural environment conditions. The resilience of employed coding paradigms against network related problems is one of the principal factors in determining the satisfaction of end user. The aim of the research presented here is to improve the error resilience of audio and speech codecs using the dedicated redundancy in a source-aware way.

Firstly, Index Assignment based Channel Coding (IACC), a joint source channel codec designed for alleviating the effects of bit errors on the speech and audio codecs is introduced. Although IACC is a type of joint source channel coding, it does not intervene with the source codec design. The proposed scheme takes into account source characteristics and adjusts the amount of coding according to the sensitivity of the different values of the source parameters. It is shown that source characteristics play an important role in the performance of IACC. A scheme which concatenates IACC and convolutional coding is also presented. The performance of IACC based schemes has been evaluated by applying them to the parameters generated by AMR-WB+ audio codec. A method for perceptual training of IACC codes is also proposed. Subjective tests comparing the performance of IACC based schemes and established convolutional coding have also been performed.

Next, various new techniques for improving the performance of multiple description coding techniques in protecting audio in networks with packet losses are presented. AAC is chosen as the underlying audio codec. Firstly, two methods for improving the performance of multiple description transform coding in application to spectral coefficients are proposed. Secondly, multiple description vector quantisation is adapted to AAC spectral coefficients and a method for improving its performance is presented. Thirdly, a coding scheme which lowers the side information burden in multiple description coding is proposed. Lastly, the performance of techniques and single description coding are compared in networks with various packet loss rates. Useful operating points for all these schemes are obtained.

A scalable multiple description scheme is introduced as the last contribution in the thesis. The proposed system provides multiple description for the hierarchical two layers. The trade-off between the first and second layers and the trade-off between the central and side distortions are controlled parametrically.
Acknowledgments

I owe my deepest gratitude to my co-supervisor Dr. Stephane Villette and my principal supervisor Professor Ahmet Kondoz whose guidance, support, suggestions, and advices during my PhD study were invaluable. Without them, this thesis would not have been in its current form.

I would like to take this opportunity to express my gratitude to my colleagues and the staff in the I-Lab, CCSR, and CVSSP of the University of Surrey; I would like to especially to thank Dr. Erhan Ekmekcioğlu, Dr. Murat Badem, Dr. Emre Unver, Dr. Ozhan Oztug and Cagri Ozcinar in helping me in the review process of this thesis.

Last but not the least; my heart-felt thanks go to my parents and sister for their endless love, great encouragement, and support not only throughout my PhD study, but also throughout my life.
To my grandfather, Niyazi
Table of Contents

List of Figures...............................................................................................................................................iii
List of Tables................................................................................................................................................v
Glossary of Terms.......................................................................................................................................vi

1 Introduction .............................................................................................................................1
  1.1 Background ............................................................................................................................1
  1.2 Thesis Outline ..........................................................................................................................3
  1.3 Original Contributions .............................................................................................................5

2 Audio and Speech Coding ............................................................................................................6
  2.1 Introduction .............................................................................................................................6
  2.2 Design Criteria ..........................................................................................................................7
    2.2.1 Bit-Rate and Quality ............................................................................................................7
    2.2.2 Delay ..................................................................................................................................8
    2.2.3 Implementation Complexity and Cost ....................................................................................8
    2.2.4 Robustness to Input Signal Variations ...............................................................................8
    2.2.5 Robustness to Channel Errors ............................................................................................9
  2.3 Overview of Audio and Speech Codecs ....................................................................................9
    2.3.1 Audio Codecs ......................................................................................................................9
    2.3.2 Speech Codecs ....................................................................................................................10
  2.4 Quantisation .............................................................................................................................11
    2.4.1 Sampling .............................................................................................................................11
    2.4.2 Quantisation .........................................................................................................................11
  2.5 Review of Audio Coding Algorithms and Standards .............................................................14
    2.5.1 Building Blocks of an Audio Codec ....................................................................................14
    2.5.2 Important Audio Coding Algorithms and Standards ..........................................................17
  2.6 Review of Speech Coding Algorithms and Standards .............................................................24
    2.6.1 LP Modelling of Speech .........................................................................................................24
    2.6.2 General Speech Coding Paradigms .......................................................................................26
  2.7 Conclusion ................................................................................................................................31

3 Error Robustness Techniques in Audio and Speech Communications ............................................32
  3.1 Introduction .............................................................................................................................32
  3.2 Robustness against bit errors ..................................................................................................34
    3.2.1 Error detection ......................................................................................................................34
    3.2.2 Error Concealment ..............................................................................................................35
    3.2.3 Robust source coding ..........................................................................................................35
    3.2.4 Error Correction Coding ....................................................................................................36
    3.2.5 Unequal error protection .....................................................................................................38
    3.2.6 Adaptive source-channel coding allocation .........................................................................39
    3.2.7 Joint Source Channel Coding .............................................................................................39
  3.3 Robustness against packet losses ...........................................................................................41
    3.3.1 ECC ....................................................................................................................................41
    3.3.2 Multiple Description Coding ...............................................................................................42
  3.4 Conclusion ................................................................................................................................46

4 Index Assignment based Channel Coding and its application to AMR-WB+ .............................47
  4.1 Introduction .............................................................................................................................47
  4.2 Index Assignment Based Channel coding ...............................................................................48
    4.2.1 Theory ................................................................................................................................48
**Table of Contents**

4.2.2 Application to codec parameters ................................................................. 49
4.3 Application of IACC to AMR-WB+ ................................................................. 52
4.3.1 System Preferences .................................................................................. 52
4.3.2 Application of IACC ................................................................................ 52
4.3.3 Application of IACC-CC ....................................................................... 55
4.3.4 Selection of the distortion measure ......................................................... 57
4.3.5 Results ..................................................................................................... 58
4.4 Conclusion ..................................................................................................... 61

5 Multiple Description Coding for AAC ............................................................... 62
5.1 Introduction .................................................................................................... 62
5.2 Application of Multiple Description Correlating Transform to AAC .............. 63
  5.2.1 Correlating Transform .......................................................................... 64
  5.2.2 Application to AAC ................................................................................ 66
  5.2.3 Parameter Updating ................................................................................ 67
  5.2.4 Inter-band pairing ................................................................................. 71
5.3 Application of Multiple Description Vector Quantisation to AAC ................. 72
5.4 Unconstrained MDVQ .................................................................................. 73
  5.4.1 Design of unconstrained MDVQ ............................................................ 74
  5.4.2 Application to AAC .............................................................................. 75
  5.4.3 Parameter updating ............................................................................... 77
5.5 Side Information Compression ........................................................................ 79
5.6 Comparison in lossy channel conditions ...................................................... 82
  5.6.1 Experiment setup .................................................................................. 82
  5.6.2 Experiments at the base rate of 64 kb/s ................................................... 83
  5.6.3 Experiments at the base rate of 32 kb/s ................................................... 84
  5.6.4 Discussion ............................................................................................ 85
5.7 Conclusion ..................................................................................................... 86

6 Scalable multiple description coding ................................................................. 88
6.1 Introduction .................................................................................................... 88
6.2 Quantisation for multiple description coding ............................................... 90
  6.2.1 Theory .................................................................................................. 90
  6.2.2 Experiments ........................................................................................ 91
  6.2.3 Adaptive multiple description coding ................................................... 94
6.3 Quantisation for scalable coding .................................................................... 95
  6.3.1 Theory .................................................................................................. 95
  6.3.2 Experiments ........................................................................................ 96
  6.3.3 Adaptive scalability ............................................................................. 97
  6.3.4 Experiments ........................................................................................ 102
  Side distortion .............................................................................................. 104
  6.3.5 Adaptive scalable multiple description coding ...................................... 105
6.4 Conclusion ..................................................................................................... 105

7 Conclusions ..................................................................................................... 107
7.1 Preamble ...................................................................................................... 107
7.2 Concluding Overview ................................................................................... 109
7.3 Future work .................................................................................................. 110

List of Publications & Presentations ................................................................... 112

Bibliography ...................................................................................................... 113
List of Figures

Figure 2.1 An example of the partitioning of a two dimensional space ...................................................13
Figure 2.2 Generic diagram of an audio encoder...................................................................................  14
Figure 2.3 Simultaneous masking and absolute hearing threshold..........................................................16
Figure 2.5 MPEG-1 Layer III encoder ......................................................................................................19
Figure 2.6: Basic building blocks of AAC encoder .................................................................................22
Figure 2.7 Source-filter model of speech production ..............................................................................25
Figure 2.8: Generic CELP diagram ........................................................................................................29
Figure 3.1 Generic diagram of audio or speech communication model ; ....................................34
Figure 3.2 Generic diagram of concatenated coding .............................................................................38
Figure 3.4 Speech coding for channel splitting ....................................................................................43
Figure 4.4: Application of IACC to AMR-WB+ parameters ..................................................................53
Figure 4.5 Application of IACC-CC to AMR-WB+ parameters ...........................................  ..56
Figure 4.6 Comparing the performance of IACC codes optimized with SNR and PESQ...58
Figure 4.7 Comparing the PESQ performances of IACC, IACC-CC-1, IACC-CC-2 and convolutional coding .................................................................................................................................59
Figure 5.1 Modified AAC Encoder with MDTC .....................................................................................64
Figure 5.2 Performance of AAC-MDTC-UPDT depending on updating interval .................................68
Figure 5.3 Updating process in AAC-MDTC .......................................................................................70
Figure 5.4 Effect of VQ rate on the performance ........................................................................  70
Figure 5.5 Inter and Intra-band pairing in AAC-MDTC ........................................................................72
Figure 5.6 Modified AAC Encoder with unconstrained MDVQ .............................................................73
Figure 5.7 Index assignment matrixes: (a) moderate redundancy, (b) duplication .................................75
Figure 5.8 Performance of AAC-MDVC-UPDT depending on updating interval .................................78
Figure 5.9: Performance of SDC and MDC schemes at the base rate of 64 kb/s as a function of redundancy.................................................................................................................................83
Figure 6.1 The effect of \( \gamma \) on the performance .......................................................................................93
Figure 6.2 The effect of bit allocation on the performance .....................................................................94
Figure 6.3 The effect of adaptation on the performance .....................................................................95
Figure 6.4 Proposed encoder output structure .....................................................................................98
Figure 6.5 Scenario 1 .................................................................................................................................99
Figure 6.6 Scenario 2.................................................................................................................................99
Figure 6.7 Scenario 3 .................................................................................................................................99
List of Figures

Figure 6.9 Scenario 5 ................................................................. 100
Figure 6.10 Scenario 6 ................................................................. 100
List of Tables

Table 2.1 The Bit Allocation Table for the MELP at 2.4 kbps ................................................................. 28
Table 2.2 AMR-WB bit allocation, 12.65 kb/s ....................................................................................... 29
Table 2.3 AMR-WB+ ACELP mode, 13.6 kb/s ...................................................................................... 42
Table 2.4 AMR-WB+ TCX mode, 13.6 kb/s ......................................................................................... 30
Table 4.3 ACELP Parameters and IACC protection in AMR-WB+ ......................................................... 54
Table 4.4 TCX parameters and IACC protection in AMR-WB+ ............................................................... 55
Table 5.1 Percentage of side and main information in AAC ................................................................. 79
Table 5.2 Occurrence probabilities of Huffman codebooks given the lowest scalefactor range .......... 81
Table 5.3 Effect of side information compression on AAC performance ............................................. 81
Table 5.4 Effect of side information compression on AAC side decoder performance ..................... 82
Table 6.1 Side and central distortions as a function of $\gamma$ ................................................................. 92
Table 6.2 The effect of $\beta$ on the layer 1 and 2 performances .............................................................. 97
Table 6.3 The effect of adaptation on the layer 1 and 2 performances ................................................ 98
Table 6.4 Performance, $\beta = \gamma = 1$ .............................................................................................. 102
Table 6.5 Performance, $\beta = 1$, $\gamma = 0.996$ .............................................................................. 102
Table 6.6 Performance, $\beta = 1$, $\gamma = 0.99$ ................................................................................ 103
Table 6.7 Performance, $\beta = 0.99$, $\gamma = 1$ ............................................................................. 103
Table 6.8 Performance, $\beta = 0.9$, $\gamma = 1$ ................................................................................ 104
Table 6.9 Performance, $\beta = 0.99$, $\gamma = 0.996$ ....................................................................... 104
Table 6.10 Performance $\beta = 0.9$, $\gamma = 0.99$ .......................................................................... 105
Table 6.11 Performance of adaptive scheme ............................................................................ 105
Glossary of Terms

3GPP 3rd Generation Partnership Project
AAC Advanced Audio Coder
ACELP Algebraic CELP
ADPCM Adaptive Differential Pulse Coded Modulation
AMR Adaptive Multi-Rate
AMR-WB Adaptive Multi-Rate Wideband
AMR-WB+ Extended Adaptive Multi-Rate Wideband
APC Adaptive Predictive Coder
ATSC Advanced Television Systems Committee
BCH Binary Coded Hexadecimal
BER Bit Error Rate
BSA Binary Switching Algorithm
BSAC Bitsliced Arithmetic Coding
BWE Bandwidth Extension
CCVQ Channel Constrained Vector Quantization
CD Compact Disk
CRC Cyclic Redundant Check
CELP Code Excited linear Prediction
COVQ Channel Optimized Vector Quantization
DAB Digital Audio Broadcasting
DAB+ Digital Audio Broadcasting plus
DAT Digital Audio Tape
DCC Digital Compact Cassette
DSL Digital Subscriber Line
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSP</td>
<td>Digital Signal Processing</td>
</tr>
<tr>
<td>DVB</td>
<td>Digital Video Broadcasting</td>
</tr>
<tr>
<td>ECC</td>
<td>Error Correction Coding</td>
</tr>
<tr>
<td>EFR</td>
<td>Enhanced Full Rate</td>
</tr>
<tr>
<td>ETSI</td>
<td>European Telecommunication Standards Institute</td>
</tr>
<tr>
<td>FEC</td>
<td>Forward Error Correction</td>
</tr>
<tr>
<td>FLC</td>
<td>Fixed Length Coding</td>
</tr>
<tr>
<td>FPC</td>
<td>Factorial Pulse Coding</td>
</tr>
<tr>
<td>FR</td>
<td>Full Rate</td>
</tr>
<tr>
<td>GAPES</td>
<td>Gapped-data Amplitude and Phase Estimation</td>
</tr>
<tr>
<td>GMDTC</td>
<td>Generalised Multiple Description Transform Coding</td>
</tr>
<tr>
<td>GSM</td>
<td>Global System for Mobile Communications</td>
</tr>
<tr>
<td>HE-AAC</td>
<td>High Efficiency -Advanced Audio Coder</td>
</tr>
<tr>
<td>HF</td>
<td>High Frequency</td>
</tr>
<tr>
<td>HILN</td>
<td>Harmonic and Individual Lines plus Noise</td>
</tr>
<tr>
<td>HR-GSM</td>
<td>Half Rate-GSM</td>
</tr>
<tr>
<td>IA</td>
<td>Index Assignment</td>
</tr>
<tr>
<td>IACC</td>
<td>Index Assignment Based Channel Coding</td>
</tr>
<tr>
<td>IACC-CC</td>
<td>Index Assignment Based Channel Coding - Convolutional Coding</td>
</tr>
<tr>
<td>IETF</td>
<td>Internet Engineering Task Force</td>
</tr>
<tr>
<td>ISO</td>
<td>International Organisation for Standardisation</td>
</tr>
<tr>
<td>ISP</td>
<td>Immittance Spectral Pairs</td>
</tr>
<tr>
<td>ISDN</td>
<td>Integrated Services Digital Network</td>
</tr>
<tr>
<td>ITU</td>
<td>International Telecommunication Union</td>
</tr>
<tr>
<td>ITU-R</td>
<td>ITU Radiocommunication Sector</td>
</tr>
<tr>
<td>ITU-T</td>
<td>ITU Telecommunication Standardisation Sector</td>
</tr>
<tr>
<td>JND</td>
<td>Just Noticeable Distortion</td>
</tr>
<tr>
<td>Term</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>JSCC</td>
<td>Joint Source Channel Coding</td>
</tr>
<tr>
<td>KLT</td>
<td>Karhunen-Loève Transform</td>
</tr>
<tr>
<td>LBG</td>
<td>Linde-Buzo-Gray</td>
</tr>
<tr>
<td>LD-CELP</td>
<td>Low Delay-Code Excited Linear Prediction</td>
</tr>
<tr>
<td>LDPC</td>
<td>Low-Density Parity-Check codes</td>
</tr>
<tr>
<td>LF</td>
<td>Low Frequency</td>
</tr>
<tr>
<td>LP</td>
<td>Linear Prediction</td>
</tr>
<tr>
<td>LSF</td>
<td>Low Sampling Frequency</td>
</tr>
<tr>
<td>LSF</td>
<td>Line Spectral Frequencies</td>
</tr>
<tr>
<td>LTP</td>
<td>Long Term Prediction</td>
</tr>
<tr>
<td>MD</td>
<td>Multiple Description</td>
</tr>
<tr>
<td>MDC</td>
<td>Multiple Description Coding</td>
</tr>
<tr>
<td>MDCT</td>
<td>Modified Discrete Cosine Transform</td>
</tr>
<tr>
<td>MDLVQ</td>
<td>MD Lattice Vector Quantization</td>
</tr>
<tr>
<td>MDSQ</td>
<td>Multiple Description Scalar Quantization</td>
</tr>
<tr>
<td>MDC</td>
<td>Multiple Description Coding</td>
</tr>
<tr>
<td>MDVQ</td>
<td>Multiple Description Vector Quantization</td>
</tr>
<tr>
<td>MELP</td>
<td>Mixed Excitation Linear Prediction</td>
</tr>
<tr>
<td>MIRS</td>
<td>Modified Intermediate Reference System</td>
</tr>
<tr>
<td>MPEG</td>
<td>Moving Picture Experts Group</td>
</tr>
<tr>
<td>MOS</td>
<td>Mean Opinion Score</td>
</tr>
<tr>
<td>MP2</td>
<td>MPEG-2 Audio Layer II</td>
</tr>
<tr>
<td>MP3</td>
<td>MPEG-2 Audio Layer III</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>MSVQ</td>
<td>Multi-Stage Vector quantisation</td>
</tr>
<tr>
<td>PAC</td>
<td>Perceptual Audio Coder</td>
</tr>
<tr>
<td>PCM</td>
<td>Pulse Code Modulation</td>
</tr>
<tr>
<td>Term</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Density Function</td>
</tr>
<tr>
<td>PEAQ</td>
<td>Perceptual Evaluation of Audio quality</td>
</tr>
<tr>
<td>PESQ</td>
<td>Perceptual Evaluation of Speech Quality</td>
</tr>
<tr>
<td>PNS</td>
<td>Perceptual Noise Substitution</td>
</tr>
<tr>
<td>PS</td>
<td>Parametric Stereo</td>
</tr>
<tr>
<td>PSTN</td>
<td>Public Switched Telephone Network</td>
</tr>
<tr>
<td>RAT</td>
<td>Robust Audio Tool</td>
</tr>
<tr>
<td>RELP</td>
<td>Residual Excited Linear Prediction</td>
</tr>
<tr>
<td>PQMF</td>
<td>Pseudo-Quadrature Mirror Filter</td>
</tr>
<tr>
<td>RCPC</td>
<td>Rate Compatible Punctured Convolutional</td>
</tr>
<tr>
<td>RS</td>
<td>Reed Solomon</td>
</tr>
<tr>
<td>RVLC</td>
<td>Reversible Variable Length Coding</td>
</tr>
<tr>
<td>SB-LPC</td>
<td>Split-Band LPC</td>
</tr>
<tr>
<td>SBR</td>
<td>Spectral Band Replication</td>
</tr>
<tr>
<td>SCDD</td>
<td>Source Constrained Channel Decoding</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal to Noise Ratio</td>
</tr>
<tr>
<td>SMR</td>
<td>Signal to Mask Ratio</td>
</tr>
<tr>
<td>SPL</td>
<td>Sound Pressure Level</td>
</tr>
<tr>
<td>STP</td>
<td>Short Term Prediction</td>
</tr>
<tr>
<td>SVQ</td>
<td>Split Vector Quantisation</td>
</tr>
<tr>
<td>TCP</td>
<td>Transport Layer Protocol</td>
</tr>
<tr>
<td>TDBWE</td>
<td>Time-Domain Bandwidth Extension</td>
</tr>
<tr>
<td>TCX</td>
<td>Transform Coded Excitation</td>
</tr>
<tr>
<td>TwinVQ</td>
<td>Transform-domain Weighted Interleave Vector Quantization</td>
</tr>
<tr>
<td>TNS</td>
<td>Temporal Noise Shaping</td>
</tr>
<tr>
<td>UEP</td>
<td>Unequal Error Protection</td>
</tr>
<tr>
<td>VLC</td>
<td>Variable Length Coding</td>
</tr>
</tbody>
</table>
## Glossary of Terms

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLSI</td>
<td>Very-large-scale integration</td>
</tr>
<tr>
<td>VQ</td>
<td>Vector Quantization</td>
</tr>
<tr>
<td>WIMAX</td>
<td>Worldwide Interoperability for Microwave Access</td>
</tr>
</tbody>
</table>
Chapter 1

1 Introduction

1.1 Background
In the past few decades, there has been fascinating development in multimedia representation and communications. First of all, it has become very clear that all aspects of media are digitized from representation to transmission. Secondly, there have been significant advances in digital multimedia compression and communication algorithms, which make it possible to deliver high-quality audio and speech at relatively low bit rates in today's networks. Thirdly, the advancement in VLSI technologies has enabled sophisticated software to be implemented in a cost-effective manner. Last but not least, the establishment of many international standards by ISO/MPEG and ITU-T laid the common groundwork for different vendors and content providers. At the same time, the explosive growth in wireless and networking technology has profoundly changed the global communications infrastructure.

Speech is the most important form of communication between humans. However it was limited to very near distances until the invention of the telephone by Alexander Graham Bell in 1876. It has since been accepted as a primary means of communications worldwide. Although image and data
communications are continually growing, telephony still constitutes a major part of telecommunications. Since its invention, telephony has seen many improvements, the most significant being the move from an analogue system to a digital system. Analogue systems have a major disadvantage in that the signal is prone to noise at every stage in the transmission. This noise accumulates over the transmission, and effectively limits the maximum communication distance as well as the quality of the received signal. On the other hand, digital systems sample and quantise the signal into a binary bit stream. It can then be regenerated at regular intervals, thereby removing the limitation on the maximum possible transmission distance. Moreover the digital signal can be easily manipulated: multiplexing, forward error correction, encryption and storage are all available. Moreover digital hardware is cheap, reliable, and much less sensitive to external factors than analogue systems. Meanwhile audio processing had a similar transformation from analogue to digital. Laser compact disk technology was introduced in 1982 and by the late 1980s became the preferred format for Hi-Fi stereo recording. Analog compact cassette players and all analog recording devices started fading away by the late 1980s. The launch of the digital CD audio format in the 1980s coincided with the advent of personal computers, and took over in all aspects of music recording and distribution.

The original Pulse Code Modulation (PCM) of speech required a sampling frequency of 8 kHz and 8 bits per sample when using logarithmic companding. This set the bit rate of digital speech at 64 kb/s, which generally translates to a significant bandwidth allocation in wireless communications.

On the other side, general audio required a much higher bandwidth when coded with PCM. Conventional Compact Disk (CD) and Digital Audio Tape (DAT) systems are typically sampled at either 44.1 or 48 kHz using PCM with a 16-bit sample resolution. This results in uncompressed data rates of 705.6/768 kb/s for a monaural channel, or 1.41/1.54 Mb/s for a stereo-pair. Both digital speech and audio have gone through a “compression period”, where the bit rate required for the same perceptual quality got lowered with new algorithms and standards. However, this accomplishment was obtained by following different paths in speech and audio coding. Audio coding algorithms approached the compression problem from a psychoacoustical perspective where the speech coding algorithms used a source based approach exploiting the production model of speech.

When transmitted within an error-prone environment, both speech and general audio are perceptually sensitive to the losses in the communication channel/network being used. The compression techniques employed in source coding can further increase the effect of erasures on the perceived quality. Virtually all speech communication and many of the audio applications are delay sensitive and thus
retransmission is not a prominent solution. Mostly, Forward Error Correction (FEC) which allocates explicit bandwidth to alleviate the effects of bad channel conditions is needed. These schemes can be pre-designed for minimising the decoded bit error rate and applied to the output of the source codec which is pre-designed to minimise coding-oriented distortion. Alternatively, the source and channel codecs can be designed together, jointly minimising the end distortion of the rendered media. Assuming infinite delay and complexity, first approach is completely sufficient. However, in practical audio streaming and two-way communication scenarios, delay and complexity are important constraints. Therefore, Joint Source Channel Coding (JSCC) is an important phenomenon to be investigated. The main aim in this research is to introduce novel or improve existing JSCC techniques to provide error resilient audio and speech communications.

1.2 Thesis Outline

The research work described in this thesis focuses mainly on new JSCC strategies or the extension or combination of existing ones. The strategies that have been developed for combating bit errors have then been implemented using AMR-WB+ speech and audio codec and the strategies that have been developed for combatting packet losses have been implemented using MPEG-2 AAC codec.

Chapter 2 gives a brief overview of speech and audio coding as well as some fundamental principles. The main criteria for the design of speech and coding algorithms, such as bit-rate, quality, delay are discussed. The three main speech coding paradigms and prominent audio coding techniques are also presented along with brief discussions. Quantisation is a key concept in speech and audio coding. After the introduction of basic quantisation techniques, the basic building blocks of a perceptual audio coder are given. These include the analysis filterbank, psychoacoustical model and quantisation and coding blocks. Analysis filterbank separates the input signal into frequency components. Psychoacoustical analysis gives the perceptually irrelevant parts of the signal. Finally, quantisation and coding block removes the statistical dependencies of the signal. Important audio coding algorithms and standards are also presented. For speech coding, firstly, speech production model is presented. This is a powerful model used by many speech coders. The modulation part of this model is modelled using a Linear Prediction (LP) filter which effectively removes the short term correlations existent between speech samples. The representation of the LP filter coefficients by Line Spectral Frequencies (LSF) is also discussed.

Chapter 3 covers the error robustness techniques against bit errors and packet losses. Beginning with non-redundant schemes like error concealment and robust source coding, main error robustness
techniques against bit errors are presented. Error correcting codes play an important role in alleviating the problems against bit errors. Main types of error correcting codes such as block codes and convolutional codes are briefly explained. Another prominent scheme, joint source channel coding, is reviewed as well. Unequal error protection and adaptive bit allocation between channel and source coding is also discussed. Error correcting coding and multiple description coding are the two main methods discussed for combating packet losses.

Chapter 4 introduces IACC, a joint source channel codec designed for alleviating the effects of bit errors on the speech and audio codecs. Although IACC is a type of joint source channel coding, it does not interfere with the source codec design. The proposed scheme takes into account source characteristics and adjusts the amount of coding according to the sensitivity of the different values of the source parameters. It is shown that source characteristics play an important role in the performance of IACC. The performance of IACC has been evaluated by applying it to the parameters generated by AMR-WB+ audio codec. A method for perceptual training of IACC codes is also proposed.

Chapter 5 presents various new techniques for improving the performance of multiple description techniques for protecting audio in networks with packet errors. Firstly, two methods for improving the performance of MDTC in application to spectral coefficients are proposed. Secondly, unconstrained MDVQ is adapted to MDCT coefficients and a method for improving its performance is presented. Thirdly, a coding scheme which lowers the side information burden in MDC is proposed. Lastly, the performance of MDVQ, MDTC and single description coding are compared in networks with various packet loss rates. Useful operating points for all these schemes are obtained.

Chapter 6 introduces a scalable multiple description scheme. The proposed system provides multiple description for the hierarchical two layers. The trade-off between the first and second layers and the trade-off between the central and side distortions is controlled parametrically. Scalability and multiple description are delivered by manipulating the classical one-stage VQ codebook. Modified Linde-Buzo-Gray (LBG) algorithm is used to train the system.

Chapter 7 gives a summary of the discussions in the previous chapters. Most significant achievements are highlighted. In addition to highlighting the most significant achievements of this project, possible areas for future research are suggested.
1.3 Original Contributions

The original contributions included in this thesis can be summarized as follows:

- Development of a JSCC scheme, IACC, for protection of speech and audio parameters against bit errors in wireless channels
- Perceptual enhancement of IACC by employing PESQ based training
- Improvement of the performance of MDTC in the context of AAC codec by development of two extensions
- Increasing the performance of MDC in high redundancy region by adapting unconstrained MDVQ to AAC
- Improvement of the performance of unconstrained MDVQ by development of an extension.
- Development of a side information compression scheme that increases the performance of MDC schemes especially at low bit rates.
- Investigation of the optimum redundancy and packet loss ranges for the application of unconstrained MDVQ, MDTC and the conventional single description coding by comparing them in AAC context.
- Development of a scalable multiple description scheme that provides multiple description coding for available two layers.
2 Audio and Speech Coding

2.1 Introduction

Digital signals encompass many benefits over analogue signals, such as ease of regeneration, security and flexibility. Therefore representing the audio and speech signals in digital format is very advantageous. In digital speech coding, the digital speech signals are processed using sophisticated signal processing techniques to achieve efficient compression in order to be used for transmission or storage. The invention of Pulse Code Modulation (PCM) in 1938 was the first example of digital speech communication systems. PCM became very popular later with the availability of the necessary hardware and was applied to private and public switched telephone networks. Today almost all of the Public Switched Telephone Network (PSTN) are based upon PCM and its spin-off technologies.

However, the use of PCM, depending on the modulation, may require more bandwidth than the original analogue signal does. This poses a problem especially in communication links where the bandwidth is limited, such as satellite or cellular mobile radio systems. As the user demand on such systems has grown, there has been extensive research interest in order to develop signal processing algorithms aiming at efficient compression of the source speech and data.

Especially with the advancements in Very Large Scale Integration (VLSI) technologies, new Digital Signal Processing (DSP) hardware has been produced, fuelling rapid developments in the speech and
audio compression area, which then has allowed the widespread acceptance and use of these technologies by the end user. There are different approaches to efficiently code the speech and audio signals. In speech coding, in order to produce high quality yet reasonably bandwidth efficient speech, some kind of a parametric model that mimics the human speech production mechanism is required instead of sample by sample coding of the waveform. Therefore speech coding algorithms can be generally classified as source-specific [1]. The parametric model makes use of the repetitions and correlations between the consecutive speech samples. The short term correlations between speech samples can be removed by using LP coding, which is a powerful tool and used extensively in speech coding. In audio coding, the model of human hearing system is exploited to remove the perceptually irrelevant components to obtain good coding efficiency. Therefore, audio coding algorithms can be generally classified as hearing specific [1].

2.2 Design Criteria

There are several criteria which need to be considered when designing a speech or audio coding algorithm. These criteria are often conflicting, and improving the algorithm with respect to one criterion may result in degradation with respect to another. Therefore, a balance must be sought during the design process with an optimal trade-off between the criteria depending on the needs of the application.

2.2.1 Bit-Rate and Quality

Bit rate and quality are probably the most important design criteria, which are usually conflicting, since a drop in bit-rate is usually accompanied by degradation in quality. Generally, there is an optimum operating range in terms of the bit-rate for each coding algorithm; exceeding the upper limit brings little or no benefit, while operating below the lower limit causes severe degradation. It is very difficult to find an objective assessment of the speech and audio quality. Especially at low bit-rates, where matching the input and synthetic speech or audio waveform is usually not possible, human subjects are usually used for determining the suitability of a speech or audio coder to a specific application. Often subjective assessment techniques are used. One of these technique is the Mean Opinion Score (MOS) [2]. Another method of measuring output quality is using objective techniques which take psychoacoustics into account. Perceptual Evaluation of Speech Quality (PESQ) [3] and
Perceptual Evaluation of Audio Coding (PEAQ) [4] are the commonly used techniques for measuring output speech and audio quality respectively.

### 2.2.2 Delay

Delay is an important criterion for designing real-time speech communication applications. Most modern speech and audio coders operate on blocks of data called frames. This introduces some delay, since at least one speech frame is buffered for analysis. Using more frames can lead to more efficient coding by removing the redundancies between them. In some coding algorithms, future values of the input signal are used for redundancy removal, and as a result look-ahead delay occurs. In addition to the buffering delay, the processing time required by the encoder and the decoder also incur delays on the system. Delays may also be introduced in the transmission channel. When the end-to-end delay of the system becomes too much, typically over 250 ms, two-way conversation may become uncomfortable because of the echo effect. Delay over mobile and satellite communication systems is very large and therefore there is a requirement for echo cancellation. Streaming applications, on the other hand can tolerate delay more than two-way communications.

### 2.2.3 Implementation Complexity and Cost

The complexity of an algorithm determines whether the algorithm can practically be implemented or not. With the recent advances in the DSP technology, it has been possible to implement more complex algorithms. However, the cost and the power consumption issues are still important for mass market applications, especially where speech coding is extensively used, such as mobile telephony. Similarly, the amount of memory required for an algorithm is also an issue when it comes to implementation. For buffering and internal processing, fast memory is needed, which is usually expensive and can become a problem with mass market applications. In summary, sometimes the complexity and implementation costs become the major criteria rather than the quality for speech coders.

### 2.2.4 Robustness to Input Signal Variations

The coders may be required to handle various types of input signal effectively. General audio coders which do not rely on specific source models can encode different types of sound signals effectively. However some speech codecs, especially vocoders, which are specialised in very low bit rate speech coding may have difficulties in adapting to other types of audio signal.
2.2.5 Robustness to Channel Errors

In most cases, the bitstream encoded by speech and audio coders is transmitted over a communication channel. In case of errors in the channel, depending on the bits affected, severe degradations might occur, such as annoying blasts in the output due to distorted filter coefficients. At low Bit Error Rates (BER), such as $10^{-6}$ to $10^{-5}$ present in PSTN, designing inherently robust algorithms may solve the problem. However, with higher BER, such has 1% to 5% in mobile and satellite communication channels, it is necessary to include FEC techniques. This is achieved by introducing high degrees of redundancy into the bitstream. If complete loss of frames is probable, Joint Source Channel Coding (JSCC) techniques such as Multiple Description Coding (MDC) should be considered.

2.3 Overview of Audio and Speech Codecs

2.3.1 Audio Codecs

Audio coders can be categorised as lossless or lossy. Lossless codecs produce representations of audio which can be decoded to exactly the bit stream before encoding and generally achieve compression rates of 40-60 depending on the input audio. The main tool used in lossless coding is the exploitation of statistical redundancies. Lossless audio codecs are generally used in archiving, editing and high fidelity playback. An important drawback of lossless audio coding algorithms is the variability in the output bit rate. Important lossless formats include Apple Lossless [5], MPEG-4 Audio Lossless Coding [6], Windows Media Audio Lossless [7], Free Lossless Audio Codec [8] and Monkey's Audio [9].

Lossy audio codecs also exploit the perceptual irrelevancies of the input audio signal in order to further reduce the output bit rate which can be as low as less than 5% of the original 1.41 Mb/s. Lossy audio codecs generally provide various bit rate options. The need for high quality audio and video in CD format led to establishment of a new group which was named as Moving Picture Experts Group (MPEG). MPEG-1 Audio Layer I [10] was an important milestone in audio coding. Layer I supports bit rates from 32 kb/s to 448 kb/s where the near-transparent quality is provided at bit rates above 192 kb/s per channel. It was used in Digital Compact Cassette (DCC). MPEG-1 Audio Layer I was succeeded by MPEG-2 Audio Layer II (MP2) [10] which provides near-transparent quality at around 128 kb/s per channel. MP2 has been extensively used in Digital Video Broadcasting (DVB) and
Digital Audio Broadcasting (DAB). Layer III [10] which lowered the bit rate required for the near-transparent quality to 96 kb/s has been extensively used in internet applications and ISDN. MPEG-1 Layer II and III have been selected by ITU-R task group for broadcasting applications in recommendation BS.1115. MPEG-2 [11] introduced Low Sampling Frequency (LSF) for better precision at the lower frequencies at the expense of omitting higher frequencies. The popular so-called “MP3” format corresponds to typically the Layer III with the improvements over MPEG-2. MP3 has become the de facto standard for Internet audio. MPEG-2 also embraced Advanced Audio Coder (AAC) [12] which provides near-transparent quality at 64 kb/s and accordingly provides five full bandwidth channels at 64 kb/s. Performance of AAC is further improved with the emergence of HE-AAC v1 and HE-AAC v2 [13]. HE-AAC v2 has been used in Digital Audio Broadcasting plus (DAB+) and standardised by 3GPP. MPEG 4 also comprises a list of audio codecs operating at very low bit rates. Other important audio codecs include Dolby Digital [14], Perceptual Audio Coder (PAC) [15], Vorbis [16] and Windows Media Audio [17].

2.3.2 Speech Codecs

The first generation Public Switched Telephone Networks (PSTN) employed the ITU G.711 companded PCM at 64 kbps [18]. Later it was replaced by a more efficient version, the ITU G.721 Adaptive Differential Pulse Coded Modulation (ADPCM) at 32 kbps [19]. As the number of subscribers grew and more bandwidth efficient coding was required, the ITU G.728 Low Delay Code Excited Linear Prediction (LD-CELP) at 16 kbps [20] was introduced. Then the ITU G.729 at 8 kbps [21] was developed, producing near toll quality with a higher delay due to the 10 ms frame sizes.

Cellular telephony is a major application area for speech coding with a very large number of subscribers on a limited-bandwidth radio link. The speech coders to be used must provide good speech quality under a high BER, with delay kept reasonably low. The GSM standard which was set up by the European Telecommunication Standards Institute (ETSI) in 1988 employs the GSM Full Rate (FR) [22] operating at 22.8 kbps gross rate with 13 kbps used for speech coding and the rest for channel coding. There is also the ETSI Enhanced Full Rate (EFR) [23] coder with better speech quality than the GSM FR, and operates at 22.8 kbps gross rate with 12.2 kbps used for speech. The Half-Rate GSM (HR-GSM) [24] operates at 11.4 kbps with 5.6 kbps used for speech, although FR and EFR are used more frequently. Finally, the Adaptive Multi-Rate (AMR) standard [25] has been introduced operating at a gross rate of 22.8 kbps or 11.4 kbps.
2.4 Quantisation

2.4.1 Sampling

A sound signal on a certain time interval has an infinite number of values with infinite precision. In order to digitally store or transmit a signal, it needs to be sampled first. The rate which the samples are taken is called the sampling rate. If the highest frequency in the signal is \( F \), the lowest sampling frequency which provides perfect reconstruction is \( 2F \) called the “Nyquist Rate”. If the sampling rate is \( 2F \) and the signal has frequency components higher than \( F \), the signal should be low pass filtered to avoid aliasing.

2.4.2 Quantisation

Quantisation is the process in which the outputs of audio or speech are converted into a suitable form for transmission or storage. During quantisation, a continuous or a discrete value signal with an infinite range is mapped to a set of levels with a finite range. Quantisation can be applied to the time domain sampled values or to the output parameters of a specific codec. The difference between the initial value and the mapped value is known as the quantisation error or noise. The main aim of a quantisation scheme is to represent the parameters in such a way that the quantisation noise is perceptually imperceptible. Quantisation can be performed using scalar or vector quantisation.

2.4.2.1 Scalar Quantisation

In scalar quantisation, a single continuous value is mapped to the nearest level from a possible number of levels. If \( l \) is the number of possible levels, and \( D \) is the finite range, then the number of bits for representing the selected level is given as

\[
B = \log_2(l)
\]  

(2.1)

The spacing between levels can be uniform (with \( D/l \) spacing) or non-uniform. Uniform quantisation suits the situations where the distribution of the values is even. However, this may not always be the case, and usually the parameter values are distributed unevenly. In that case, the quantisation levels should be determined by taking the statistical distribution of the values. For example, in places with a concentration of values more levels could be assigned for more accurate representation of the values. It is also possible that some regions of the input signal are perceptually more sensitive than the other regions. Non-uniform quantisation is more suitable for such cases as well. A non-uniform
quantisation can be designed to minimise Mean Squared Error (MSE) by analysing the expected input signal's PDF. Alternatively, companding functions such as $\mu$-law and A-law companding can be used.

### 2.4.2.2 Vector Quantisation

Scalar quantisers are simple and efficient in terms of memory usage. However, in terms of achieving more efficient quantisation, vector quantisation performs better than scalar quantisation. In vector quantisation, a group of values are combined in a vector and quantised jointly. Vector quantisation can exploit the correlations existing between the values, which leads to an improved quantisation efficiency.

If $x$ is an $N$ dimensional vector with real valued elements given by

$$x = [x^1, x^2, ..., x^N]$$  \hspace{1cm} (2.2)

It is then mapped onto another $N$ dimensional real valued vector, $y$. $y$ is the quantised version of $x$ typically chosen from a finite set of values, $y = [y_i, 1 \leq i \leq C]$, where $y_i = [y_{i1}, y_{i2}, ..., y_{iN}]$. The set of vectors $y$ is called a codebook with size $C$, which is usually chosen to be a power of 2.

Designing a codebook requires optimal partitioning of an $N$ dimensional space into $C$ regions. Each region is represented by a code vector $y_i$ which is generally the centroid of the region. Figure 2.1 illustrates partitioning of a two dimensional space.

One of the most popular methods for codebook design is the Linde Buzo Gray (LBG) algorithm [26] which is an iterative algorithm for obtaining optimum partitions and code vectors. When the number of bits used for a codebook becomes too large, codebooks for LSF quantisation for example, the complexity and storage requirements may render the system impractical for implementation. For a vector quantiser consisting of $L$ codebook entries with a size of $N$, the complexity of a full-search is given by:

$$C = NL = N2^B$$  \hspace{1cm} (2.3)

where $B$ is the number of bits allocated. The memory requirement of this system can be given by:

$$M = NL = N2^B$$  \hspace{1cm} (2.4)

in words.
For large codebooks, sub-optimal design and search techniques exist. One of the most common techniques is Split-Vector Quantisation (SVQ). In this scheme, the vector is divided into sub-vectors, which use their own codebooks. Due to the reduced vector size, $L$, and the number of entries, $N$, each codebook has significantly lower complexity and memory requirements. Hence the total complexity and memory requirements are less than the non-split case. There are, however, disadvantages of this system. Since the sub-vectors are treated separately, the intra-vector correlations cannot be exploited properly. Moreover, the splitting of the vector into sub-vectors may not optimal. For example, during LSF quantisation, sometimes an LSF pair corresponding to a formant can be split into different sub-codebooks. As a result, quantisation may not be very efficient. Finally, the bit allocation scheme for each sub-codebook is fixed and can only cater for the perceptual importance of each value in a limited fashion. For example, in a case where all the elements of a sub-vector are of low importance, the number of allocated bits remains the same, and as a result the quantisation efficiency is lowered.

Another technique which addresses the problems of the SVQ is the Multi-Stage Vector quantisation (MSVQ) technique, where the combination of smaller codebooks is used to quantise the input vector [27]. In MSVQ, the vectors in each codebook has the same length as the input vector, which makes it possible to exploit the intra-vector correlations as well as perceptual weighting of the values within each vector. However, testing each combination of vectors from each stage can be computationally very complex.
Instead, the stage codebooks can be searched sequentially, finding the best index for each stage and searching for the best one at the next stage, minimizing the residual error at each stage. The disadvantage of this method is the fact that the combination with the lowest intermediate distortion may not result in the lowest overall distortion. An optimum trade-off between complexity and quantization efficiency can be achieved using an M-Best tree search algorithm. In this search technique, $M$ indices of the first stage are kept. The residual for each of the $M$-indices are then searched at the next stage, and again $M$ indices are kept resulting in the lowest distortion. Therefore, at each stage the codebook is searched $M$-times, one for each previously kept $M$ indices. This results in $M$ candidate paths giving the lowest intermediate distortion. Finally, the candidate path resulting in the lowest overall distortion is chosen. When designing codebooks for MSVQ, LBG algorithm can be used on the input training set for the first stage and on the residual of the previous stage for the next ones. Finally, these codebooks are jointly-optimized using several iterations.

### 2.5 Review of Audio Coding Algorithms and Standards

#### 2.5.1 Building Blocks of an Audio Codec

Despite the fact that there are differences in the technical details of current audio coding schemes, most audio coders employ filterbanks[28]. Filter banks allow for signal decorrelation and therefore provide a framework for removing redundancy in an audio signal. In addition, irrelevant components of the signal can be separated from relevant ones based on models of human perception. By subdividing the signal into its frequency components, a great reduction in the amount of data needed to reproduce the signal can be achieved. Redundancy can be further reduced by entropy coding of the spectral components. Figure 2.2 shows the generic diagram of an audio encoder.

![Generic diagram of an audio encoder](image)
2.5.1.1 Analysis Filterbank

The input signal is mapped to subsampled spectral representation using various types of analysis filter banks. For reasons of coding efficiency, modern coding schemes typically employ filterbanks with critical sampling (i.e. same number of input samples and spectral coefficients) and overlapping analysis windows between subsequent analysis frames. Examples include Transform Coded Excitation (TCX) [29], modified discrete cosine transform (MDCT) [30], polyphase filterbanks [31] or hybrid structures [31].

2.5.1.2 Psychoacoustical model

The inner ear performs short term critical band analyses where frequency to place transformations occur along the basilar membrane. The power spectra are not represented on a linear frequency scale but on limited frequency bands called critical bands. The auditory system can roughly be described as a band-pass filter bank, consisting of strongly overlapping bandpass filters with bandwidths in the order of 100 Hz for signals below 500 Hz and up to 5000 Hz for signals at high frequencies. Twenty-five critical bands covering frequencies of up to 20 kHz have to be taken into account.

There are several psychoacoustical phenomena [32] which can be exploited for audio compression. Temporal masking is the characteristic of the auditory system where sounds are hidden due to the maskers which have just disappeared or even maskers which are to appear. Simultaneous masking is a frequency domain phenomenon where a low level signal (the maskee) can become inaudible (masked) by a simultaneously occurring stronger signal (the masker), if maskee and masker are close to each other in frequency. A masking threshold can be measured where the low level signal will not be audible below. The masking threshold, also known as threshold of Just Noticeable Distortion (JND), varies with time. It depends on the Sound Pressure Level (SPL), the frequency of the and the characteristics of masker and maskee. The slope of the masking threshold is steeper towards lower frequencies, i.e. higher frequencies are more easily masked. It should be noted that noise is a better masker than a tonal signal. A signal can also be unperceivable if its sound pressure is below the threshold in quiet which depends on frequency. This is called absolute hearing threshold. Simultaneous masking and absolute hearing threshold are illustrated in Figure 2.3. The distance between the level of the masker and the masking threshold is called signal to mask ratio (SMR). Within a band, quantisation noise will not be perceived as long as its SNR is lower than SMR. A perceptual audio coder's performance significantly depends on its ability to exploit the perceptual irrelevancies of the input signal. If the necessary bit rate for a complete masking of distortion is
available, the coding scheme will be perceptually transparent, i.e. the decoded signal is then subjectively indistinguishable from the source signal.

![Simultaneous masking and absolute hearing threshold](image)

**Figure 2.3 Simultaneous masking and absolute hearing threshold**

### 2.5.1.3 Quantisation and Coding

Quantisation can be uniform or non-uniform where the latter increases the perceptual efficiency of the coder. VQ is also used especially in speech coding and very low bit rate general audio coding. After the quantisation of the parameters, the quantised values need to be encoded for transmission or storage. Encoding can be done in two ways: Fixed Length Coding (FLC) and Variable Length Coding (VLC) or Entropy Coding [33]. In FLC, all quantisation values are treated equally and encoded with equal length codewords. On the other hand, in entropy coding, quantisation values are mapped into variable length codewords depending on the frequency of each quantisation value. There are entropy-coding schemes have been proposed including Huffman coding, Rice Coding, Golomb Coding, arithmetic coding and Lempel-Ziv coding [34]. Among these most commonly used entropy coding algorithm in audio coding is Huffman coding. Unlike the quantisation, entropy coding schemes are typically noiseless. Noiseless systems can reconstruct the input signal perfectly from its coded representation. In contrast, a coding scheme incapable of perfect reconstruction is called lossy. It should be noted that using VLC codewords in error prone channels can result in loss of synchronisation and may consequently necessitate employment of strong channel coding schemes.
which may offset the bandwidth saved with VLC. In addition, employment of VLC complicates the
design of the encoder if fixed frame rate is desired.

### 2.5.2 Important Audio Coding Algorithms and Standards

#### 2.5.2.1 MPEG 1

MPEG-1 is a standard that defines the compression of combined audio and video. It includes video,
audio and system specifications [10]. Although reference software is provided, there is no imposed
encoder structure in the specification. MPEG-1 supports the sampling rates of 32, 44.1 and 48 kHz.
One or two channels are supported. The rates supported vary between 32 and 324kb/s which
correspond to compression ratios of 24:1 and 2.7:1 respectively. MPEG 1 comprises 3 layers. As the
layer number increases, the compression efficiency increases at the expense of complexity.

#### 2.5.2.1.1 Layer I and II

Layer I and II are derived from MUSICAM [35]. Application areas of Layer I includes Digital
Compact Cassette, (DCC) at 192 kb/s. Layer II is used by DAB and DVB at bit rate around 128 kb/s.
Layer II was also selected by ITU-R task group, TG, 10/2 for emission at the rate of 128 k/s per
channel and for distribution and contribution at data rates above 180 kb/s.

The block diagram of Layer I and II is shown in Figure 2.4. Both layers utilise a 32-band (Pseudo
Pseudo-Quadrature Mirror Filter (PQMF) bank which decomposes the input signal into 32 critically
subsampled subbands. The channels are equally spaced such that a 48 kHz input signal is split into 750
Hz subbands, with the subbands decimated 32:1. The subband coefficients are then scaled and
quantized with a uniform midtread quantiser. The precision of the quantiser is determined by the
output of the psychoacoustic model. The psychoacoustic model uses 512 sample frame size in Layer I
and 1024 sample frame size in Layer II. In each subband, 6 bits are used for quantizing scale factors
and 4 bits are used for representing the quantiser selections. In Layer II, the maximum subband
quantiser resolution is increased by utilising temporal masking such that temporal masking is
exploited considering the properties of the adjacent 12-sample block and optionally transmitting one, two or three scale factors.

Audio Encoded

- PCM Uniform Midtread Quantizer
- Filter Bank (32 Sub-bands)
- DFT 512/1024 Hann Window
- Psychoacoustic Model
- Coding of Side Information

Figure 2.4 MPEG-1 Layer I and II encoder

2.5.2.1.2 Layer III

While the design of PQMF provides very good time resolution with a relatively simple structure, there are some disadvantages. Firstly, adjacent bands have significant frequency overlap, i.e. a signal at a single frequency can affect two adjacent frequency bands. Secondly, the width of the frequency band is much larger than the critical bandwidth values for frequencies below 2000 Hz, therefore the signal representation does not provide sufficient separation for the optimal bit distribution implied by the psychoacoustic model. Consequently, in Layer III a hybrid model which cascades 32 channel PQMF and MDCT is used. This model increases frequency resolution while maintaining the backward compatibility with layers I and II.

Layer III specifies two different MDCT block lengths: a long block of 18 samples or a short block of 6. There is a 50 per cent overlap between successive transform windows so the window size is 36 and 12, respectively. The long block length allows greater frequency resolution for audio signals with stationary characteristics while the short block length provides better time resolution for transients [33]. Note the short block length is one third that of a long block. In the short block mode, three short blocks replace a long block so that the number of MDCT samples for a frame of audio samples is unchanged regardless of the block size selection. For a given frame of audio samples, the MDCT’s can all have same block length (long or short) or have a mixed-block mode. In the mixed block mode the
MDCT's for the 2 lower frequency subbands have long blocks and the MDCT's for the 30 upper subbands have short blocks. This mode provides better frequency resolution for the lower frequencies, where it is needed the most, without sacrificing time resolution for the higher frequencies. Figure 2.5 illustrates the basic building blocks of MPEG-1 Layer III encoder.

In Layer III, quantization is done with a non-uniform function which shapes the error depending on coefficient amplitude as shown in Equation (2.5). The SNR remains relatively constant for a wider range of energy values when compared to a uniform quantiser. The quantisation function is given by [33]:

\[ iX_i = \text{sign}(x_i) \left( \frac{|x_i|}{\sqrt{2 \times \text{global\_gain\_scale\_factor}}} - 0.0946 \right)^{0.75} \]  

(2.5)

2.5.2.1.2.1 Bit allocation

A system of two nested iteration loops is the common solution for quantization and coding in a Layer III encoder. Quantization is done via a power-law quantiser. In this way, larger values are automatically coded with less accuracy and some noise shaping is already built into the quantization process. The quantized values are coded by Huffman coding. To adapt the coding process to different local statistics of the music signals the optimum Huffman table is selected from a number of choices. The Huffman coding works on pairs and, only in the case of very small numbers to be coded,
quadruples. To get even better adaption to signal statistics, different Huffman code tables can be selected for different parts of the spectrum. Since Huffman coding is basically a variable code length method and noise shaping has to be done to keep the quantization noise below the masking threshold, a global gain value (determining the quantization step size) and scalefactors (determining noise shaping factors for each scalefactor band) are applied before actual quantization. The process to find the optimum gain and scalefactors for a given block, bit-rate and output from the perceptual model is usually done by two nested iteration loops in an analysis-by-synthesis way:

**Inner iteration loop:** The Huffman code tables assign shorter code words to (more frequent) smaller quantized values. If the number of bits resulting from the coding operation exceeds the number of bits available to code a given block of data, this can be corrected by adjusting the global gain to result in a larger quantization step size, leading to smaller quantized values. This operation is repeated with different quantization step sizes until the resulting bit demand for Huffman coding is small enough. The loop is called rate loop because it modifies the overall coder rate until it is small enough.

**Outer iteration loop:** To shape the quantization noise according to the masking threshold, scalefactors are applied to each scalefactor band. The system starts with a default factor of 1.0 for each band. If the quantization noise in a given band is found to exceed the masking threshold (allowed noise) as supplied by the perceptual model, the scalefactor for this band is adjusted to reduce the quantization noise. Since achieving a smaller quantization noise requires a larger number of quantization steps and thus a higher bit-rate, the rate adjustment loop has to be repeated every time new scalefactors are used. In other words, the rate loop is nested within the noise control loop. The outer (noise control) loop is executed until the actual noise (computed from the difference of the original spectral values minus the quantized spectral values) is below the masking threshold for every scalefactor band (i.e. critical band). While the inner iteration loop always converges, this is not true for the combination of both iteration loops. If the perceptual model requires quantization step sizes so small that the rate loop always has to increase them to enable coding at the required bit-rate, both can go on forever. To avoid this situation, several conditions to stop the iterations early can be checked.

### 2.5.2.2 MPEG-2

Some extensions were added to MPEG-1 layers defined in MPEG-1. **Low Sampling Rates (LSR)** [33] is simply the halving of the sampling rates provided by MPEG-1 Audio (i.e. 32kHz, 44.1 kHz and 48 kHz) into (i.e. 16 kHz, 22.05 kHz and 24 kHz). Although this results in sharp limitations of the
available bandwidth, it provides better coding efficiency in low bitrates and decreased computational demands. MPEG-2 specification is also an important milestone in multichannel audio coding since it provides a representation for “5:1” format which indicates the reproduction of audio by five full range loudspeakers and a low frequency effects channel.

### 2.5.2.2.1 AAC

Although it shares the basic coding paradigm of Layer III (like two nested iteration loops), AAC improves on Layer III in a lot of details and uses new coding tools for improved quality at low bit rates [36]. Following are the differences between AAC and MP3.

- The number of frequency lines in AAC is up to 1024 compared to 576 for Layer III (e.g. better frequency resolution)

- Long term prediction (LTP) is used to remove the signal periodicity and decrease the number of bits needed for coding by reducing the energy of the input signal.

- Efficiency of Huffman coding is increased by applying quadruples of frequency lines. In addition, the assignment of Huffman codebooks to coder partitions can be much more flexible.

Beyond these basic blocks, an important coding tool was introduced with AAC: Prior to quantization a procedure called Temporal Noise Shaping (TNS) [37] is applied to the frequency domain samples. This technique, which is based on a linear predictor working on the spectral samples, allows some control of the temporal shape of the quantization noise within one transform block. Since the prediction in frequency domain translates to modelling of the envelope in the time domain, the coder’s performance is enhanced for transient signals, especially for speech signals. This technique is very effective in preventing the phenomenon known as “pre-echo”. Basic building blocks of AAC encoder are shown in Figure 2.6.
2.5.2.3 MPEG-4

2.5.2.3.1 Improvements over AAC

MPEG-4 introduced further improvements to the design of AAC:

- Perceptual Noise Substitution (PNS) [38]: The similarity of noise signals in human perception is enhanced. Noise-like bands are detected on encoder side, and only their energy level has to be transmitted to the decoder, where these bands are reconstructed by filling them with a signal constructed by a random noise generator.

- Spectral Band Replication (SBR) [39]: In traditional audio coding, a significant amount of information is spent in coding the high frequencies, although the psychoacoustic importance of the last one or two octaves is relatively low. This triggered the basic idea behind SBR.
Based on the cognition of a strong correlation between the high- and the low-frequency range of an audio signal, a good approximation of the original input signal high band can be achieved by a transposition from the low band. High Efficiency AAC version 1 profile (HE-AAC v1) combines SBR and low complexity AAC.

- Parametric Stereo (PS) [40]: Whereas SBR exploits the possibilities of a parameterised representation of the high band, the basic idea behind PS is to parameterise the stereo image of an audio signal such as “panorama”, “ambience”, or “time/phase differences” of the stereo channels to enhance the coding efficiency of the codec. In the encoder, only a monaural downmix of the original stereo signal is coded after extraction of the Parametric Stereo data. Just like SBR data, these parameters are then embedded as PS side information in the ancillary part of the bit-stream. In the decoder, the monaural signal is decoded first. After that, the stereo signal is reconstructed, based on the stereo parameters embedded by the encoder. High Efficiency AAC version 2 profile (HE-AAC v2) combines PS and low complexity HE-AAC v1.

2.5.2.3.2 Twin VQ

The Transform-domain Weighted Interleave Vector Quantization (TwinVQ) [41] is an alternative VQ-based coding kernel which is designed to provide improved coding performance at low bitrates (at or below 16 kbit/s). The coder kernel has been adapted to operate within the spectral representation provided by the AAC coder filterbank. The TwinVQ kernel performs a quantization of the spectral coefficients in two steps: In the first step the spectral coefficients are normalized to a specified target range and are then quantized by means of a weighted vector quantization (VQ) process. The spectral normalization process includes a LPC spectral estimation scheme, a periodic component extraction scheme, a Bark-scale spectral estimation scheme, and a power estimation scheme which are carried out sequentially. As a result, the spectral coefficients are flattened and normalized across the frequency axis. In the weighted vector quantization process, the flattened spectral coefficients are interleaved for complexity reasons and divided into sub-vectors for vector quantization. For each sub-vector, a weighted distortion measure is applied. In this way, perceptual control of the quantization distortion is achieved.
2.5.2.3.3 HILN

MPEG-4 specification contains two very low bit rate parametric coding schemes called Harmonic and Individual Lines plus Noise (HILN) for audio coding [42]. In HILN, the input signal is decomposed into 3 types of atomic signal objects which are described by appropriated source models and represented by model parameters. Object models for individual sinusoids, harmonic series of sinusoids, and noise are utilized in the HILN coder. Linear Prediction Coding (LPC) is used for representing the spectral envelope of the input signal. The efficiency of the codec is further increased by using a perceptual model and discarding the irrelevant components and by applying entropy coding. One interesting property of HILN is that it both combines a parametric model of source and a perceptual model of hearing in the same coder.

2.6 Review of Speech Coding Algorithms and Standards

2.6.1 LP Modelling of Speech

Linear Predictive Coding (LPC) [43] is one of the most powerful analysis techniques and used in virtually all modern speech specific codecs. There is usually a significant amount of correlation between successive speech samples, known as short term correlations. The amount of short term correlations depend on the characteristics of the speech signal. LPC analysis tries to model these correlations using a short order filter. Due to the modelling of the sample-to-sample correlations (formants), the LPC is also called Short Term Prediction (STP).

2.6.1.1 The Source-Filter Model

The majority of low bit-rate speech coders employ a speech production model mimicking the human speech production mechanism, and is formed of two parts: the excitation and the modulation. Excitation can either be voiced or unvoiced. During voiced excitation, the vocal folds open and close at regular intervals, breaking the air forced from the lungs into quasi-periodic pulses whose frequency is controlled by pitch. Unvoiced excitation, on the other hand, is caused by turbulent air from the lungs. The excitation signal then passes through the vocal tract which acts as the modulation filter. The shape of the modulation filter depends on the positions of the tongue, velum, lips and the nasal cavity. Figure 2.7 shows the popular Source-Filter model which is widely used in speech coding [44].
model is assumed to be linear with independent excitation and modulation parts. This way a simple and practical implementation is possible.

![Source-filter model of speech production](image)

**Figure 2.7 Source-filter model of speech production**

### 2.6.1.2 Linear Prediction

The accuracy of the modelling of the modulation filter in the source-filter model is critical for good performance. While there are many different techniques developed for the modelling of this modulation filter, Linear Prediction (LP) is the most widely used one [44]. In LP, the combined effects of the vocal tract, glottal flow and the lips represented by the time varying filter is modelled as a pole-zero filter. Finding the optimal coefficients for this filter is a challenging task. However, this filter can be simplified to an all-pole model with a high-enough order due to the human speech production mechanism. Since there are no more than 4 or 5 formants in human speech limited to 4 kHz in bandwidth, a 10th order filter is usually sufficient, representing each formant with two poles. The main objective here is to determine the optimal coefficients which minimize the MSE.

### 2.6.1.3 LSF Representation of the LP Coefficients

When using the LP filter coefficients in speech coders, they have to be quantised and often interpolated. However, this causes some problems. Since these are the coefficients of an IIR filter, their sensitivity to small changes is very large. This is especially a problem during quantisation where some addition of quantisation noise is unavoidable. As a result, the resulting filter may be completely different or even unstable. Lack of a stability check for the LP coefficients is a serious shortcoming in this case. Moreover, interpolation between two sets of LP coefficients is very difficult due to the unpredictable relation between the filter coefficients and the associated frequency response. The result
of an interpolation between two sets of LP coefficients may have no resemblance to either set. In order to overcome the difficulties discussed above, an alternative form of representation for the LP coefficients is required. The most common form of representation are the Line Spectral Frequencies (LSF) which are easier to manipulate, robust to small distortions, open to interpolation, and have a stability check. The conversion between the LP coefficients and LSF is a lossless transformation. LSF have strong relations to the speech spectrum. For speech sampled at 8kHz, the LSF are limited to the range (0, 4000) Hz. Two consecutive LSF come closer near the formant frequencies where their distance depends on the strength of the formant. This fact causes the LSF sets to contain the redundancies in the speech spectrum as a result of the correlations between successive speech frames. Another important property of the LSF is the convenient stability check. The corresponding filter is guaranteed to be stable when the LSF are in an increasing order, i.e.:

\[ 0 < \text{LSF}_1 < \text{LSF}_2 < ... < \text{LSF}_{10} < 4000 \] (2.6)

### 2.6.2 General Speech Coding Paradigms

Speech coders are generally classified into three categories depending on the exploitation of the speech signal:

- Waveform Coding
- Parametric Coding
- Hybrid Coding

#### 2.6.2.1 Waveform Coding

Waveform Coders attempt to match the original speech waveform on a sample-per-sample basis. It is not speech specific, and can also encode other types of signals. They can provide high quality at the expense of high bit rates. For example toll quality speech requires the speech to be sampled at 8 kHz with a 13-bit accuracy, resulting in around 100 kb/s. Logarithmic companding techniques such as A-law and \( \mu \)-Law [45] can be used to reduce the number of bits allocated to each sample. This has led to the adoption of the well-known ITU G.711 64 kb/s PCM standard [18], which is now widely used in the telecommunications industry.

The high correlation existing between consecutive speech samples can be exploited by using some form of prediction, giving further reductions in bit rate. This has led to the adoption of the ITU G.721
Chapter 2, Audio and Speech Coding

32 kb/s ADPCM standard [19]. However the speech quality produced by waveform coders rapidly decreases when the bit rate is lowered.

2.6.2.2 Parametric Coding

Parametric coders do not attempt to match the original speech waveform. Instead they characterise the speech using a certain number of parameters, which are measured on the speech input at regular intervals. These parameters are then quantised and transmitted to the decoder, where synthetic speech having the same properties as the original speech is generated. The parameters are chosen so that the output speech is perceptually identical to the input speech, by way of a speech production model. By reducing the speech to its fundamental characteristics, very low bit rates can be achieved. However the speech quality is limited by the speech production model used, and the accuracy of the parameter extraction and quantisation. This usually limits the quality that can be achieved, although recent improvements have allowed vocoders to produce natural sounding speech at around 2.4 kb/s [46][47]. Moreover such low bit rate coders require a speech production model optimised for human speech only, and as a result parametric coders usually do not cope well with non-human sounds, as is the case when speech is contaminated with acoustic background noise. Parametric coders usually operate in the region of 800 bit/s for low quality speech to 4.8 kb/s for quite natural sounding speech.

2.6.2.2.1 Mixed Excitation Linear Prediction (MELP)

In this subsection, MELP which will be used as a base codec for the developed channel coding scheme is discussed. The MELP coder [46] aims a robust LPC synthesis by using mixed excitation. It also incorporates aperiodic pulses, adaptive spectral enhancement for matching formant waveforms and a pulse dispersion filter for matching the natural excitation characteristics more efficiently. MELP uses a mixture of pulses and noise as the excitation signal in each frequency band. The pulse train and the noise sequence are passed through separate spectral shaping filters and then added together to give the final full-band excitation. Additionally, the use of aperiodic pulses is introduced in order to remove the buzziness caused by the erratic glottal pulses. This is done by adding uniformly distributed jitter to pulse positions. Moreover, an adaptive spectral enhancement filter is used in order to prevent the premature decaying of the synthetic waveform in formant regions so that a better match to the natural speech waveform can be achieved. The pulse dispersion filter, on the other hand, improves the match between the synthetic and natural waveforms in the frequency bands without a formant resonance.
The MELP Coder, extracts parameters using frames of 22.5 ms, and quantises them using 54 bits in total, thus operating at 2.4 kbps. The bit allocation scheme of the MELP coder at 2.4 kbps can be found in Table 2.1.

Table 2.1 The Bit Allocation Table for the MELP at 2.4 kbps

<table>
<thead>
<tr>
<th>LPC Coefficients (10 LSPs)</th>
<th>34</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gain (2 per frame)</td>
<td>8</td>
</tr>
<tr>
<td>Pitch and Overall Voicing</td>
<td>7</td>
</tr>
<tr>
<td>Bandpass Voicing</td>
<td>6</td>
</tr>
<tr>
<td>Aperiodic flag</td>
<td>1</td>
</tr>
<tr>
<td>Total</td>
<td>54 bits/22.5 msec</td>
</tr>
</tbody>
</table>

2.6.2.3 Hybrid Coding

Hybrid coders combine the advantages of both parametric coding and waveform coding. As a result they are able to offer near toll quality at medium bit rates, and are the most popular speech coders at these rates. They basically use a speech production model, as for parametric coders, to reduce the correlation between neighbouring speech samples. The resulting residual signal is then waveform coded. The transmitted signal therefore comprises of the coefficients of the predictors used to remove redundancy, and the waveform coded residual signal. Early schemes used Analysis-and-Synthesis (AaS) models, where the input speech is inverse filtered using the speech production model, and the residual is then quantised, using one of many techniques. Examples of such coders are the Adaptive Predictive Coder (APC) [48], and the Residual Excited Linear Prediction (RELP) coder [49], which mainly differ in the technique used for quantisation of the residual. These offer high quality speech at 16 kb/s and good quality at 9.6 kb/s respectively. Increases in the amount of computational power available in modern DSP has allowed the emergence of new coders called Analysis-by-Synthesis (AbS) coders. In these coders the waveform matching process is performed on the actual output speech rather than the residual signal from the speech production model. This involves applying the speech production model to every candidate block of synthetic excitation, and selecting the excitation according to the synthetic output produced. The most notable example of an AbS coder is the Code Excited Linear Predictive (CELP) algorithm [50], variants of which are used in many applications at low to medium bit rates. Such coders are capable of producing near toll quality at bit rates from 4.8 to 13 kb/s. Figure 2.8 illustrates the generic block diagram of CELP coder.
2.6.2.3.1 AMR-WB

The AMR-WB codec is based on the Algebraic CELP (ACELP) technology [51]. LPC information is represented by Immittance Spectral Pairs (ISP) which is similar to LSF representation. Two frequency bands, namely 50–6400 Hz and 6400–7000 Hz, are coded separately in order to decrease complexity and to focus the bit allocation into the subjectively most important frequency range. The bandwidth provided by AMR-WB goes far beyond the narrowband speech codecs which are limited to 200-3400 Hz. The wideband codec is an adaptive codec capable of operating with a multitude of speech coding bit-rates ranging from 6.6 to 23.85 kb/s. Table 2.2 shows the bit allocation between the parameters of AMR-WB codec per frame operating at 12.65 kb/s [52].

<table>
<thead>
<tr>
<th>VAD-flag</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISP</td>
<td>46</td>
</tr>
<tr>
<td>LTP-filtering</td>
<td>4</td>
</tr>
<tr>
<td>Pitch delay</td>
<td>30</td>
</tr>
<tr>
<td>Algebraic code</td>
<td>144</td>
</tr>
<tr>
<td>Gains</td>
<td>28</td>
</tr>
<tr>
<td>Total</td>
<td>253</td>
</tr>
</tbody>
</table>
1.1.1.1 Extended Adaptive Multi-Rate Wideband (AMR-WB+)

The AMR-WB+ coder is based on a hybrid ACELP/TCX model, this allows switching between LP-based and transform-based coding depending on the signal characteristics. The input signal can be mono or stereo with sampling frequencies ranging from 16 kHz up to 48 kHz. The bit rates supported range from 6 to 48 kb/s. The AMR-WB+ audio codec processes input frames equal to 2048 samples at a variable internal sampling frequency. The internal sampling frequency is limited to the range 12800-38400 Hz. The 2048-sample frames are split into two critically sampled equal frequency bands. This results in two superframes of a 1024 samples corresponding to the low frequency (LF) and high frequency (HF) band. Each superframe is divided into four 256-samples frames. The LF and HF signals are then encoded using two different approaches: the LF is encoded and decoded using the "core" encoder/decoder, based on switched ACELP and TCX. In ACELP mode, the standard AMR-WB codec is used. The HF signal is encoded with relatively few bits (16 bits/frame) using a bandwidth extension (BWE) method. Table 1.3 and 1.4 [54] show the bit allocation of AMR-WB+ in ACELP and TCX modes respectively for 13.6 kb/s, operating at nominal sampling frequency 25.6 kHz. The ACELP mode approximately corresponds to AMR-WB 12.65 kb/s plus HF parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode bits</td>
<td>2</td>
</tr>
<tr>
<td>ISF</td>
<td>46</td>
</tr>
<tr>
<td>Mean Energy</td>
<td>2</td>
</tr>
<tr>
<td>Pitch delay</td>
<td>30</td>
</tr>
<tr>
<td>Pitch Filter</td>
<td>4</td>
</tr>
<tr>
<td>Algebraic code</td>
<td>144</td>
</tr>
<tr>
<td>Codebook Gains</td>
<td>28</td>
</tr>
<tr>
<td>HF ISF</td>
<td>9</td>
</tr>
<tr>
<td>HF gain</td>
<td>7</td>
</tr>
<tr>
<td>Total in bits</td>
<td>272</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode bits</td>
<td>2</td>
</tr>
<tr>
<td>ISF</td>
<td>46</td>
</tr>
<tr>
<td>Noise factor</td>
<td>3</td>
</tr>
<tr>
<td>Global Gain</td>
<td>7</td>
</tr>
<tr>
<td>Algebraic VQ</td>
<td>198</td>
</tr>
<tr>
<td>HF ISF</td>
<td>9</td>
</tr>
<tr>
<td>HF gain</td>
<td>7</td>
</tr>
<tr>
<td>Total in bits</td>
<td>272</td>
</tr>
</tbody>
</table>

Table 2.3 AMR-WB+ ACELP mode, 13.6 kb/s

Table 2.4 AMR-WB+ TCX mode, 13.6 kb/s
2.7 Conclusion

In this chapter a brief overview of speech and audio coding as well as some fundamental principles such as quantisation. The main criteria for the design of speech and coding algorithms, such as bitrate, quality, delay and error robustness were discussed. Quantisation is a key concept in speech and audio coding. After the introduction of basic quantisation techniques, the basic building blocks of a perceptual audio coder were given. These include the analysis filterbank, psychoacoustical model and quantisation and coding blocks. Analysis filterbank separates the input signal into frequency components. Psychoacoustical analysis gives the perceptually irrelevant parts of the signal. Finally, quantisation and coding block removes the statistical dependencies of the signal. For speech coding, the popular source-filter model was presented. In this model, the excitation is modelled by periodic pulses or random noise, and the vocal tract is modelled by an all-pole LP filter. Important audio algorithms and speech coding algorithms and standards were also presented.
3 Error Robustness Techniques in Audio and Speech Communications

3.1 Introduction

In an audio transmission scenario, the audio is first compressed by an audio encoder to reduce the data rate and the compressed bit stream is then segmented into fixed or variable length frames (packets). The packets might be sent directly over the network, if the network guarantees bit error free transmission. Otherwise, they usually undergo a channel encoding stage, typically using FEC, to protect them from transmission errors. At the receiver end, the received packets are FEC decoded and unpacked, and the resulting bitstream is then input to the audio decoder to reconstruct the original audio.

Transmission errors can be roughly classified into two categories: random bit errors and erasure errors. Random bit errors are caused by the imperfections of physical channels which result in bit inversion. Especially in wireless channels, the characteristics of bit errors can be the principal factor in determining the end quality of the audio transmission system. Depending on the coding methods and the affected information content, the impact of random bit errors can range from negligible to intolerable. When fixed length coding is used, a random bit error will only affect one codeword, and
the caused damage may be generally acceptable. But if VLC is used, random bit errors can desynchronize the coded information such that many following bits are undecodable until the next synchronization codeword appears. Erasure errors on the other hand, can be caused by packet loss in packet networks such as the Internet, burst errors in storage media due to physical defects or system failures for a short time. Packet losses can be due to the congestion in a packet network.

Random bit errors in VLC coded streams can also cause effective erasure errors since a single bit error can lead to many following bits being undecodable. The effect of erasure errors (including those due to random bit errors) is much more destructive than random bit errors due to the loss or damage of a contiguous segment of bits.

Error control in audio communications is very challenging for several reasons. First, compressed audio streams are extremely vulnerable to transmission errors because of the use of predictive coding and VLC by the source coder. Second, even the FLC codewords can carry perceptually very important information and be significantly affected by a single bit error.

To make the compressed bit stream resilient to transmission errors, one must add redundancy into the stream, so that it is possible to detect and correct errors. Typically, this is done by employing FEC, which operates over coded bit streams generated by a source coder. The classical Shannon information theory states that one can separately design the source and channel coders, to achieve error-free delivery of a compressed bit stream, as long as the source is represented by a rate below the channel capacity. Therefore, the source coder should compress a source as much as possible (to below the channel capacity) for a specified distortion, and then the channel coder can add redundancy through FEC to the compressed stream to enable the correction of transmission errors. Figure 3.1 shows the generic diagram of an audio communication scenario. An ideal error-free delivery can be achieved only with infinite delays in implementing FEC and are not acceptable in practice. Therefore, joint source and channel coding is often a more viable scheme, which allocates a total amount of redundancy between the source and channel coding. Many error-resilient encoding methods essentially work under this premise, and intentionally make the source coder less efficient than it can be, so that the erroneous or missing bits in a compressed stream will not have a disastrous effect in the reconstructed audio quality.

When an audio or speech frame is missing due to transmission errors, the decoder can estimate them based on surrounding received samples by making use of inherent correlation among temporally
adjacent samples. Such techniques are known as error concealment. This is possible because practical source coders do not completely eliminate the redundancy in a signal in the encoding process. Error concealment has the advantage of not employing any additional bitrate, but adds computational complexity at the decoder.

3.2 Robustness against bit errors

3.2.1 Error detection

Error detection can be done by inserting redundant bits systematically into the encoded bit stream. The simplest way of detecting errors is to simply add up the sum of a certain number of words and store the sum with the data. If the replayed data does not add up to the so-called check sum, the data has an error. This is called Cyclic Redundancy Check (CRC) [55]. Error detection can also be done in the source level by prohibiting the usage of some codewords [56]. If the source decoder detects a prohibited codeword, it returns a flag that the received data is corrupted. Depending on the part of the data which is corrupted, the decoder can omit corrupted part and may apply pre-defined error concealment technique.
3.2.2 Error Concealment

Error concealment offers a mechanism to replace or fill in corrupted speech and audio parameters once detected by the decoder. In order to achieve the best performance with error concealment, the algorithm must be capable of accurately predicting the speech or audio signal and making a smooth transition between the previous decoded signal and inserted fragment. These techniques can generally be used for concealing the packet losses as well. There are various error concealment techniques:

- **Silence substitution**: The corrupted frame is substituted by silence.
- **Noise substitution**: The lost packet is substituted by noise. Compared to silence it has better performance as intelligibility improves.
- **Repetition**: The lost packet is substituted by a copy of the last correctly received frame before the loss.
- **Interpolation**: These schemes synthesise speech and audio signals which are used to replace the corrupted frame or corrupted parameter. It is possible to interpolate coding parameters such as linear prediction coefficients (LPC) or gain, instead of getting an estimation of the speech waveform. One prominent interpolation technique for corrupted audio frames is Gapped-data Amplitude and Phase Estimation (GAPES) algorithm for replacing the missing data, using interpolation in the spectral domain [57].

3.2.3 Robust source coding

Error robustness of an audio or speech codec can be increased by carefully designing the source codec with little or no redundancy. If vector quantisation is used, a reference vector recalled by a corrupted index can degrade the perceived audio quality significantly depending on the difference between the reference and the original vector. The Index Assignment (IA) [58] is a non-redundant process which reorders the codebook in order to minimise the effects of the errors in the transmission vectors by usually assigning indices with similar binary patterns to similar reference vectors. This process can take expected source and channel characteristics into account.

In MPEG-4, the performance of AAC codec in a wireless channel can be improved by replacing VLCs with Reversible Variable Length Coding (RVLC) [56]. The RVLC uses symmetric codewords to enable both forward and backward decoding of scale factor data. MPEG-4 also defines Virtual Codebooks tool [56] which limits the largest absolute value possible within a certain scale factor band.
where escape values are allowed and Huffman Codeword Reordering tool [56] which avoids the error propagation into the priority codewords by placing them into the known positions.

3.2.4 Error Correction Coding

Also known as channel coding or FEC, Error Correction Coding (ECC) [59] is a system of error control for data transmission, whereby the sender adds systematically generated redundant data to its messages. The carefully designed redundancy allows the receiver to correct a limited number of errors occurring anywhere in the message without the need to ask for additional data. This advantage is at the cost of a fixed channel bandwidth. ECC is therefore applied in situations where retransmissions are relatively costly or impossible such as when broadcasting to the receivers. Error Correcting Codes can be divided into two main groups: block coding and convolutional coding.

3.2.4.1 Block Codes

A block code transforms a message \( m \) consisting of a sequence of information symbols over an alphabet \( \Sigma \) into a fixed length sequence \( c \) of \( n \) encoding symbols, called a codeword [55]. In a linear block code each input message has a fixed length of \( k < n \) input symbols. The redundancy added to a message by transforming it into a larger codeword enables a receiver to detect and correct errors in a transmitted codeword. The redundancy is described in terms of its rate \( k/n \). One particular type of linear block codes is cyclic codes based on an algebraic structure that leads to strong error correcting capabilities and to computationally efficient encoding and decoding algorithms. Binary Coded Hexadecimal (BCH) codes form a class of cyclic block codes which can be decoded with the algebraic method known as syndrome decoding.

3.2.4.1.1 Reed-Solomon Codes

Reed-Solomon (RS) [55] codes are cyclic BCH codes which can be decoded by the computationally effective efficient decoding algorithm called Berlekamp-Massey algorithm. This leads to the implementation of long RS codes with strong error correcting capability. RS coders can be particularly designed for channels with burst bit errors. They are widely used in consumer electronics such as DVDs, in data transmission such as Digital Subscriber Line (DSL) and Worldwide Interoperability for
Microwave Access (WIMAX) and in broadcast system such as DVB and Advanced Television Systems Committee (ATSC).

### 3.2.4.1.2 Low Density Parity Check Codes

Low-density parity check codes (LDPC) [55] are a class of recently re-discovered highly efficient linear block codes using an iterated soft-decision decoding approach. This is contrary to other types of block codes which are restricted to hard decision decoding. Their complexity is linear to the block length used. Their performance approaches to the theoretical maximum. Because of these reasons they are widely used in recent high speed communication standards such as DVB-S2 and 10GBase-T Ethernet.

### 3.2.4.2 Convolutional codes

Convolutional coding [60] extends the concept of a block code to allow memory from block to block. Each encoded symbol is therefore a linear combination of information symbols in the current block and a selected number of preceding blocks. In convolutional coding, the transformation is a function of the last $k$ information symbols, where $k$ is the constraint length of the code.

A binary convolutional encoder can be represented as a shift register. The outputs of the encoder are modulo 2 sums of the values in the certain register's cells. The input to the encoder is either the uncoded sequence (for non-recursive codes) or the uncoded sequence added with the values of some register's cells (for recursive codes).

Convolutional codes can be systematic and non-systematic [61]. Systematic codes are those where an uncoded sequence is a part of the output sequence. Systematic codes are almost always recursive, conversely, non-recursive codes are almost always non-systematic.

Convolutional codes can be punctured from a mother code to obtain higher coding rates by discarding some output bits of the convolutional encoder.

Decoding of a convolutional coding is usually done with a Viterbi decoder which does soft decoding by computing maximum likelihood of a sequence. That is an important advantage of convolutional codes over block codes except LDPC codes.
3.2.4.3 Concatenated codes

The probability of decoding error can be made to decrease exponentially as the block length $N$ of the coding scheme goes to infinity. However, the complexity of a naive optimum decoding scheme that simply computes the likelihood of every possible transmitted codeword increases exponentially with $N$, so such an optimum decoder rapidly becomes infeasible.

Concatenated coding schemes were first proposed by Forney [63] as a method for achieving large coding gains by combining two or more relatively simple building block or component codes. The resulting codes had the error-correction capability of much longer codes, and they had a structure that permitted relatively easy to moderately complex decoding. Serial concatenation of codes is mostly used for power-limited systems such as transmitters on deep-space probes. The most popular of these schemes consists of a Reed-Solomon outer (applied first, removed last) code followed by a convolutional inner (applied last, removed first) code. The generic block diagram of such a system is shown in Figure 3.2.

![Figure 3.2 Generic diagram of concatenated coding](image)

3.2.4.4 Turbo codes

The description in the previous subsection was given for what is called a serially concatenated code. Turbo codes as described first in 1993, implemented a parallel concatenation of two convolutional codes, with an interleaver between the two codes and an iterative decoder that would pass information forth and back between the codes [63]. This construction had much higher performance than all previously conceived concatenated codes.

3.2.5 Unequal error protection

Unequal Error Protection (UEP) is an efficient method to improve the error robustness of channel coding schemes. It is used by various speech and audio coding systems operating over error-prone
channels such as mobile telephone networks or DAB. The bits of the coded signal representation are first grouped into different classes according to their error sensitivity. Then error protection is individually applied to the different classes, giving better protection to more sensitive bits. In order to ensure an optimal quality, one must conduct a bit error sensitivity analysis to evaluate the relative importance of the bits. For example, in MPEG-4 5 sensitivity classes are defined [64]. In speech codecs like AMR-WB, there are also defined hierarchies for bit error sensitivities [65]. It was shown that even a small amount of redundancy can increase the error resilience of AMR-WB+ significantly provided that resource allocation is done carefully [66].

3.2.6 Adaptive source-channel coding allocation

Error correction by means of channel coding provides powerful mechanisms to guarantee a low bit error rate. Nevertheless, under very poor transmission conditions the allocated bandwidth may not be enough for recovering the corrupted frames. On the other hand, at very good channel qualities the bandwidth allocated for channel coding can exceed the needed threshold. These drawbacks of fixed channel coding schemes motivated the standardization effort of an AMR codec for GSM. AMR aims at an adaptive bit rate assignment to source and channel coding within the gross bit rate of the half-rate (TCH/HS) and full-rate (TCH/FS) speech channel [67].

3.2.7 Joint Source Channel Coding

Channel codecs are generally designed and applied separately from the source codecs and are interested to minimise the error in the channel domain rather than the perceptual domain. Their aim is to fully recover the transmitted bits. However, there are some drawbacks with this approach. First of all, we have to allow infinite complexity and delay in the coders in order to reach optimality. Evidently this is problematic for real-time communication. Secondly, the theorem is no longer valid for multi-path channels, and in those cases we no longer have an optimal system. Thirdly, such systems tend to break down completely when the channel quality falls under a certain threshold, and the channel code is no longer capable of correcting the errors. This phenomenon is often referred to as the threshold effect. In summary, perfect transmission of the source coder output is an artificial aim, the real goal should be a low-distortion reconstruction of the original source as the source coding introduces some distortion even before channel transmission. These drawbacks of separate source and channel coding led to research in JSCC. There is noticeable amount of literature in JSCC. Farvardin et
al. [68] proposed a joint optimization method, which does not require any explicit channel coding. Instead, the source encoder is directly optimized with respect to the conditions on the disturbed transmission channel. The error-protecting capability of this so-called channel-optimized vector quantization (COVQ) is a result of leaving some of the possible quantizer output symbols unused, which implicitly increases the redundancy. Goldsmith et al. [69] combine COVQ and UEP in terms of rate-compatible punctured convolutional (RCPC) codes.

Skoglund proposed channel-constrained vector quantization (CCVQ) [70], which adjusts the quantizer regions such that the a priori distribution of the codevectors results in an increased redundancy. To achieve higher channel robustness, the code is trained on a subspace of the available code space. Optimization of redundancy-increasing index assignments for discrete channels is also considered.

Bozantzis et al. combined vector quantization with a redundancy-increasing index assignment and trains both components alternately [71]. A drawback of this approach is that the number of both codevectors and codewords is constrained to powers of two, which disables a fine-granular adjustment of error-protecting redundancy.

Goodman et al. applied simulated annealing to train quantization and channel coding jointly [72]. In the training process, several contiguous cells of an initial quantizer are comprised and assigned the same codeword. However, this makes it difficult to extend the method to vector quantization.

A joint source/channel coding method, which does not require any training was proposed by Kim [73]. He uses a fixed class of distance-preserving codes, so-called snake-in-the-box codes to perform a robust, redundancy-increasing index assignment. However, it is difficult to extend this concept to vector quantization. Another drawback is that an adaptation of the code to known source and channel statistics is not possible.

Joint optimization is not limited to the coding scheme, but can be extended to the detection/estimation taking place in the receiver. A jointly optimized receiver exploits, e.g., residual redundancy, which results from the imperfections of practical source encoding. So a channel decoder can improve its decisions by exploiting a non-uniform distribution of source bits as a priori information (e.g., source-controlled channel decoding (SCCD) [74]).
Alternatively, residual redundancy can be used as a priori information in an estimation-based source decoder. The advantage of the second approach is that a perception-related quality measurement can be taken into account.

3.3 Robustness against packet losses

The main problems that arise and result in degraded quality of speech at the receiver's end when transmitting speech over a packet-switched network are i) packet loss, ii) delay, iii) jitter and iv) echo [75]. The main focus with respect to MDC is given to packet loss which is also the major problem faced while transmitting packets (assuming its random nature) and the main factor contributing to the quality degradation of the received signal. Most packet losses occur in the routers, either due to high router load or to high link load. In both cases packets in the queues might be dropped. Packet loss also occurs when there is a breakdown in a transmission link. In wireless channels, corruption of VLCs with bit errors can also make packets non-usable. In either case, the result is the incomplete packet is dropped. In non-real-time applications, the problem of packet loss is solved with retransmission through the use of a transport layer protocol, TCP. For telephony applications, retransmission is not a viable solution since it incurs overlong delays that could result in annoying echo or talker overlap. The ITU-T recommendation for standard G.114 states that the one-way delay should be kept lower than 150 ms for acceptable speech quality [75].

3.3.1 ECC

ECC is an option to decrease the packet loss probability experienced by delay sensitive traffic, such as real-time multimedia, when retransmission schemes can not be used to recover losses due to strict delay constraints. There are two main directions of FEC design to recover from packet losses. One solution, proposed by the Internet Engineering Task Force (IETF) and implemented in Internet audio tools like RAT [76] and Freephone [77] is to add a redundant copy of the original packet to one of the subsequent packets [78]. In the case of packet loss the information is regained from the redundant copy. The other set of solutions conventional block coding schemes based on algebraic coding, e.g. Reed-Solomon coding [79]. The error correcting capability of RS codes with \( k \) data packets and \( c \) redundant packets is \( c \) if data is lost. While Reed-Solomon codes are typically used to correct bit errors, they can be used to recover lost packets via block interleaving as described in [80]. Given a
block of $k$ packets, the packets are prefixed by their lengths in bytes, and packets shorter than the longest one in the block are padded by zeros. Reed-Solomon coding is applied to the $i^{th}$ symbol (typically byte) of each packet (in total $k$ symbols) to form the $i^{th}$ symbols of the $c$ redundant packets. Packets are then transmitted one-by-one, without the padding zeros. The loss of a packet appears as the loss of a symbol in a block of $c+k$ symbols at the receiver and can be corrected as long as the number of lost packets is no more than $c$. The performance of both FEC schemes depends on the burstiness of the loss process; the performance of the media-dependent FEC scheme depends on the probability of consecutive packet losses; the capability of the second, media-independent FEC scheme to recover from losses depends on the distribution of the number of packets lost in a block.

### 3.3.2 Multiple Description Coding

Multiple Description Coding (MDC) addresses the problem of joint source and channel coding for overcoming packet losses. Originally it was designed for the transmission of multiple descriptions of a single source over independent channels. If only one of the descriptions is received, it is used for reconstruction with a certain accuracy. If more than one descriptions are received, then the information from the other descriptions can be used to enhance the accuracy. MDC has been rediscovered recently for use in packet switched networks [81]. Instead of using separate channels, one can time-shift the different descriptions. In the general case the amount of information sent over the separate channels (packets) can be different. The generic diagram of MDC is shown in Figure 3.3.

![Figure 3.3 The generic MDC model](image-url)
3.3.2.1 MDC employing prediction

The earliest MDC system that incorporated prediction was based on the concept of alternating the PCM samples of a signal and transmitting them through two separate channels. When the signal arrived at the destination, the two side decoders could construct the missing description and in consequence the missing samples by interpolating the samples of the received description. In the case where both descriptions arrived at the receiver, the decoder combined both of them by interleaving the samples [82]. A block diagram of this concept is presented in Figure 3.4. The problem, as presented in [83] consists of finding the optimal prediction coefficient in the encoder and finding the optimum transfer function in the decoder.

![Figure 3.4 Speech coding for channel splitting](image)

3.3.2.2 MDC employing repetition with optimised bit allocations

Here, the more general area where repetition of the input signal is incorporated into the approach while the interest focuses on finding an optimal bit allocation technique is going to be looked at. The basic idea of a context-based coding technique is to make use of the knowledge of the neighbourhood statistics for the data to be encoded [84]. In particular, in their system, each polyphase component of the input signal is coded independently using a fine quantiser and packed into one packet. For error protection, each packet also carries a coarsely quantised version of neighbouring polyphase components. In case of channel failures, this coarsely quantised data can be used to recover the lost packets. Between the polyphase components of a given signal, there usually exists strong correlation, that can be exploited in the system for better coding efficiency.
3.3.2.3 MDC employing perceptual models

This technique provides an implicit forward error concealment mechanism to handle random erasures of the channel. The main concept behind this approach is to group the individual acoustic subchannels of their auditory model into different transport subchannels or packets [85]. Due to the strongly overlapping, redundant filterbank structure of the model, reconstruction of the speech without audible degradation would become possible even if a significant percentage of the channels was erased (for example, up to 40% in a 50-channel auditory model for narrow-band speech). This kind of error concealment exploits the tolerance of human perception to irrelevant detail. If the erasure probabilities for specific transport subchannels are not known to the transmitter, a redundant joint source-channel coding scheme is called for that, that spreads the source information over multiple description which are carried over \( M \) independent subchannels. The grouping of the \( N \) auditory channels into \( M \) transport channels should follow the equal-importance principle, \( N \) should be chosen as an integer multiple of \( M \) such that a constant number of \( N/M \) auditory channels are packaged together into a transport channel. In this respect, each transport channel is obtained by frequency-domain subsampling of an oversampled signal representation.

3.3.2.4 MD quantization

A prominent method for MD coding of a memoryless, continuous-valued source is MD scalar quantization (MDSQ) [81]. MDSQ is the use of two separate scalar quantizers to give two descriptions of a scalar source sample, with an additional central decoder that makes use of both descriptions. Each quantizer outputs an index that can be used by itself to estimate the source sample. To improve the performance of a MDSQ scheme, the quantization cells should be disconnected. The performance of fixed rate MDSQ can be enhanced with employing entropy-constrained MDSQ [86]. Extending the formulation of MD scalar quantization to vectors is easy. However, the index assignment problem becomes more difficult because the code vectors cannot be naturally ordered. In addition, the encoding complexity increases with dimension. An effective technique that avoids these difficulties is the MD lattice vector quantization (MDLVQ) of Servetto et al. [87]. The index assignment problem is simplified by lattice symmetries, and the lattice structure also reduces encoding complexity. In [88], a different approach which does not put any constraints to the quantiser design and solely optimises the index assignments of multiple description vector quantiser by applying Binary Switching Algorithm.
Chapter 3. Error Robustness Techniques in Audio and Speech Communications

(BSA). Since it doesn’t lead to any changes in the source coder, this approach is advantageous to MDLVQ.

3.3.2.5 MD Correlating Transforms

The transform in this technique explicitly adds redundancy contrary to the odd/even separation which uses similar inherent redundancy. Suppose \( X_1 \) and \( X_2 \) are independent, zero-mean Gaussian random variables with variances \( \sigma_1^2 \) and \( \sigma_2^2 \). \( X_1 \) and \( X_2 \) are then transformed to \( Y_1 \) and \( Y_2 \) using a linear correlating transform \( T \). The amount of the correlation depends on the transform. When one description is lost, the received description gives information about both \( X_1 \) and \( X_2 \). Assuming no quantisation, \( X_1 \) and \( X_2 \) can be recomputed by the inverse transform \( T^{-1} \). In [89], it was shown that quantising first and then applying a transform performs better than transforming first and then applying quantisation.

Although it approaches to the theoretical limit in the low redundancy region, one particular problem with MDTC is that adding redundancy beyond a certain point does not improve the side decoder performance. This is because two dimensional information can be represented by a one dimensional variable only in limitation. To overcome this problem, a multiple description method called Generalised Multiple Description Transform Coding (GMDTC) [90]. With GMDTC, in each description, not only one of the transformed variables but some information about the estimation error for the other variable are included as well. Performance of GMDTC approaches to theoretical limit in high redundancy region.

3.3.2.6 Multiple-Description Coding by Dithered Delta-Sigma Quantization

In [91], a two-channel MD scheme based on two times oversampled dithered Delta-Sigma quantization, which is inherently symmetric in the description rate and as such there is no need for source splitting was presented. The rate loss when employing finite-dimensional quantisers (in parallel) is therefore given by that of two quantisers. Asymptotically as the dimension of the vector quantiser and order of the noise shaping filter approach infinity, it is shown that the symmetric two-channel MD rate-distortion function for a memoryless Gaussian source and MSE fidelity criterion can be achieved at any resolution. An arbitrary number of descriptions can be created by increasing the oversampling ratio.
3.4 Conclusion

In this chapter the ways for error resilient audio and speech communications have been reviewed. The problems in the communication systems can roughly divided into two groups: Bit errors and packet losses. When an error or loss is detected, the receiver can request for the retransmission of the missing data if feedback channel is provided. This solution is viable unless there are delay constraints as in two way communications. The feedback implosion problem in multicast scenarios also make the deployment of retransmission schemes difficult. For bit errors and packet losses, error detection followed by an appropriate error concealment technique can be a perceptually effective and resourcefully efficient technique as long as the loss rate is modest.

For bad channel and network conditions, adding a significant amount of redundancy for the receiver to correctly decode the intended signal seems a necessity. There are advanced channel coding systems for the practical applications. On the other hand, delay and complexity constraints limit the performance of separated design of channel and source codecs. It can be argued that perfect transmission of the source coder output is an artificial aim, the real goal should be a low-distortion reconstruction of the original source as the source coding introduces some distortion even before channel transmission. JSCC addresses this problem.

MDC, which is a type of JSCC specialised for packet networks, can be designed with concern for every combination of received descriptions with defined probability. If so desired, this can give performance that varies gracefully with the number of received descriptions where a technique based on conventional channel coding would exhibit the threshold effect.
Chapter 4

4 Index Assignment based Channel Coding and its application to AMR-WB+

4.1 Introduction

As discussed in chapter 3, joint source-channel coding (JSCC) has been popular as it generally provides better performance than separable schemes. One of the limitations of the many proposed JSCC techniques is that the source codec is integrated with the channel codec and therefore can not be separated from the channel codec to be employed in different communication scenarios than the one considered in the design procedure. In the present research, a scheme named Index Assignment based Channel Coding (IACC) that enables unequal protection among different quantization vectors of the same encoder parameter is presented. IACC is block coder, which takes source and channel characteristics into the design procedure. IACC does not intervene in the internal quantizer design of the source coder and therefore enables the design of a JSCC system even if a pre-designed source coder has to be used. In this research, IACC has also been concatenated with convolutional coding (IACC-CC) as an outer code. Theoretical explanation and examples of IACC are presented in Section 2. Performance of IACC in some important source distributions is also elaborated. Section 3
demonstrates the application of IACC and IACC-CC to AMR-WB+ illustrating how source characteristics of a particular codec can be incorporated into the IACC design. Both objective and subjective simulation results in different channel conditions are given in Section 4.

4.2 Index Assignment Based Channel coding

4.2.1 Theory

In a communication system, the encoder employed for compressing speech generally produces parameters of varying importance. Thus, corruption of the coded parameters due to transmission errors have different impact on the quality of the decoded speech. Unequal Error Protection is widely used to adjust the level of protection for different parameters of the encoded speech according to their sensitivities to bit errors. It is also possible that transitions between some codevectors of a parameter lead to more perceptual distortion than the other transitions. Assigning channel codewords to codevectors such that “close” codevectors are mapped to close codewords in Hamming distance can diminish the effect of the channel errors on the received source quality. Reordering of channel codebook is also exploited for error detection [92]. This is done by isolating the codewords representing sensitive codevectors with more invalid codewords than the codewords representing less sensitive codevectors. For the energy parameter of SB-LPC coder, this technique leads to a better decoded quality than the classic error detection techniques.

Now, let us consider a vector source which is represented by a fixed-rate source codebook $S = \{s_0, s_1, \ldots, s_{2^l-1}\}$ with $l$ bits. For protection against channel errors, additional $K$ bits are allocated and each codevector $s_i$ is uniquely mapped to a codeword within the channel codebook $C = \{c_0, c_1, \ldots, c_{2^l-1}\}$ by the function

$$M(s_i) = c_{s_i} \quad (4.1)$$

Assuming a noisy channel $N$, the receiver side will get $c_j$, the channel-transformed version of $c_{s_i}$ such that

$$N(c_{s_i}) = c_j \quad (4.2)$$

At the receiver side, the channel codeword $c_j$ will be inverse-mapped to the source codebook, $S$ by
where the inverse-mapping can be intuitively done by searching the closest valid codeword to \( c_j \) and then reading the corresponding quantization level from the channel codebook.

Considering all possible channel-dependent deviations from the transmitted codeword \( c_{s_i} \), total distortion function for \( s_i \) will be,

\[
D_{s_i} = \sum_{j=0}^{K-1} \left( d(s_i, M^{-1}(c_j)) \times p_i(c_{s_i}, c_j) \right) 
\]

*where* \( d(s_i, M^{-1}(c_j)) \) is the distortion between the transmitted codevector and decoded codevector and \( p_i(c_{s_i}, c_j) \) is the transition probability from the valid channel codeword \( c_{s_i} \) to \( c_j \). Assuming that occurrence probability of every codevector is equal, the total distortion function will be,

\[
D_T = \sum_{i=0}^{I-1} D_{s_i} 
\]

### 4.2.2 Application to codec parameters

Let us consider a simple example of IACC with \( I=2 \) and \( K=2 \) where \( s = \{0.2, 0.3, 0.6, 5.0\} \) and respective channel codebook is \( c = \{0000, 0001, 0010, 1111\} \). It is immediately noticed that the source value \( s_3 \) is significantly larger than the other discrete values. Its corresponding channel codeword is also far from the other ones in the Hamming space. Figure 4.1 shows two different codewords which are modified by the same error vector. Quantization value 5.0 is decoded correctly where 0.2 is decoded as 0.3 by finding the closest valid codeword to the corrupted codeword.

![Figure 4.1 An example of decoding corrupted codewords with IACC.](image)

As seen in the previous example, IACC is able to protect distant source values more than the closely concentrated ones. It can be therefore argued that sources whose variations are the results of extreme but rare deviations are more likely to benefit from IACC than the sources whose variations are the results of modest and frequent deviations, i.e. sources with larger kurtosis will benefit more from
Chapter 4. Index Assignment based Channel Coding and its application to AMR-WB+

IACC. To illustrate this, IACC is applied to several well known distributions: Uniform (kurtosis = 1.2), Gaussian (kurtosis = 3.0) and Laplacian (kurtosis = 6.0) distributions. All of the distributions have zero mean and unit variance and coded with K=7 bits optimized with Lloyd-Max training. Another N=8 bits is used for IACC. Two types of information are needed to optimize IACC codes for the minimization of total error given in Equation 4.4. Firstly, a distortion criterion for the codevectors is needed. In this experiment we have used MSE. Secondly, transition probabilities between the codewords must be known. Assuming a memoryless binary symmetric channel, the transition probability is a function of hamming distance and BER. In order to analyse the results, a (15,7) BCH code, which fully utilizes its correcting capacity by correcting all 1-bit, 2-bit errors and some 3-bit errors is used as a benchmark reference. The MSE and variance of MSE associated with IACC and BCH code are illustrated in Table 4.1 and Table 4.2 respectively.

Table 5.1 MSE

<table>
<thead>
<tr>
<th></th>
<th>Uniform</th>
<th>Gaussian</th>
<th>Laplacian</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCH</td>
<td>0.0008</td>
<td>0.005</td>
<td>0.015</td>
</tr>
<tr>
<td>IACC</td>
<td>0.0003</td>
<td>0.001</td>
<td>0.002</td>
</tr>
</tbody>
</table>

Table 5.2 Variance of MSE

<table>
<thead>
<tr>
<th></th>
<th>Uniform</th>
<th>Gaussian</th>
<th>Laplacian</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCH</td>
<td>0.0027</td>
<td>0.06</td>
<td>0.46</td>
</tr>
<tr>
<td>IACC</td>
<td>0.0011</td>
<td>0.006</td>
<td>0.014</td>
</tr>
</tbody>
</table>

Results show that IACC performs better than BCH at all chosen distributions and BERs but the difference is more noticeable as kurtosis of the given source increases. The difference is also more noticeable as the BER gets higher since multiple errors become more probable and therefore the prioritized protection of extreme deviations becomes more essential. This is supported by the dramatic drop in the variance of distortion when IACC is employed in Laplacian distribution.
To see the performance of IACC on a real speech codec, an experiment is set for the 1st LSF vector and pitch gain parameters obtained by MELP speech codec. 10000 LSFs and pitch gain parameters extracted from 10000 MELP frames are fed to IACC. IACC codes are trained at half rate. The performance is then tested by modifying the IACC coded parameters with sufficiently random error vectors at various BERs. MSE is employed as a distortion measure in both training and testing. The reference scheme is a half rate convolutional coder with constraint length 5. This scheme takes 100 parameters (700 bits) as data chunk and maps it to a single codeword. The results for pitch gain and 1st LSF are shown in Figure 4.2 and Figure 4.3 respectively.

Convolutional coder performs better than IACC at low BERs. Although, IACC coder performs better than the convolutional coder at high BER for both parameters, the crossover point is more biased towards IACC in Figure 4.2. This is because 1st LSF has a higher kurtosis than pitch gain. The crossover point for the variance of distortion is at lower BERs, especially in LSF case showing the importance of the kurtosis of the parameter in the relative performance of IACC to conventional coding. Despite being nonlinear, IACC shares certain restrictions of short length block coding and is similarly inferior to the convolutional coding at low BERs.

It should be noted the "effective" (perceptual) kurtosis of a parameter may be related to the perceptual distortion of the parameter. For example, taking the kurtosis of the average energy extracted from a speech frame directly, can be misleading since the distortion in the logarithmic domain is perceptually more meaningful. On the other hand, setting distortion criterion as the logarithm of energy is likely to underestimate the perceptual distortion caused by outliers which are frequently experienced in noisy
channels but only rarely seen in noise free conditions. By employing a perceptually meaningful and globally fair distortion criterion between all codevectors, IACC can be adapted to perceptual characteristics of any parameter. These will be further elaborated in the next section.

4.3 Application of IACC to AMR-WB+

4.3.1 System Preferences

Some simplifications and preferences are made to realize IACC and IACC-CC in AMR-WB+ context:

1) Sampling rate is set to 25.6 kHz and bits per frame to 272, which leads to the speech coding rate of 13.6 kb/s.
2) Stereo extension is turned off.
3) Either 4*256-sample TCX or 4*256-sample ACELP is used in a super-frame.
4) Mode bits are assumed to be transmitted correctly.
5) No CRC checking is done.
6) 208 bits/frame (10.4 kb/s) is used for error correction in both developed and reference schemes.

4.3.2 Application of IACC

4.3.2.1 Generic Model

Application of IACC to AMR-WB+ needs the optimization of a specific codebook for each parameter produced by the AMR-WB+ encoder. This is illustrated in Figure 4.4. For each frame AMR-WB+ processes, every parameter generated is channel coded by its dedicated IACC encoder \(\text{En}(\text{IACC})\). The outputs of the IACC codes are then fed into the transmission channel together. At the receiver side, the dedicated IACC decoder, \(\text{De}(\text{IACC})\), decodes its corresponding channel codeword for every parameter and the reconstructed parameters are fed into the AMR-WB+ decoder. UEP among codec parameters can be obtained by setting IACC rates of parameters differently.
4.3.2.2 Technical Configuration

In order to realize the IACC based system, global coding allocation among IACC has to be determined first. In the present research this is done by looking at the error sensitivities of the parameters and arranging the allocation accordingly as it has been done in [66][93]. This subsection illustrates bit allocation among IACC codes of different parameters in ACELP and TCX modes. The total number of allocated IACC bits are 208 in both cases.

Table 4.3 shows the bit allocation among parameters in ACELP mode as well as the coding allocation among their corresponding IACCs. Some parameters, for example 5th and 6th ISF subvectors, jointly form a single IACC to benefit from the efficiency of larger codes. However maximum size of source rate plus IACC addition can not exceed a certain number (18 bits in this work), due to the massive training and running complexity. For example, first ISF vector could not be allocated more than 10 bits, although this would significantly increase the error robustness of the codec.
Table 4.1 ACELP Parameters and IACC protection in AMR-WB+

<table>
<thead>
<tr>
<th>Parameters</th>
<th>ISF vectors</th>
<th>Mean Energy</th>
<th>Pitch Lag</th>
<th>Codebook Indices</th>
<th>Codebook Gains</th>
<th>High-band ISF</th>
<th>High-band Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Bits</td>
<td>46</td>
<td>2</td>
<td>30</td>
<td>144</td>
<td>28</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>Number of parameters</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>32 track positions and 8 sign bits</td>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Bit Allocation</td>
<td>8,8,7,6,5,5</td>
<td>2</td>
<td>9,6,9,6</td>
<td>4 bits for positions and 1 bit per sign</td>
<td>7,7,7</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>IACC Allocation</td>
<td>10,8,2,2,8,1,1</td>
<td>6</td>
<td>9,6,9,6</td>
<td>(2 track positions and 1 sign bit are jointly allocated 6 bits) * 16</td>
<td>8,8,6,6</td>
<td>9</td>
<td>7</td>
</tr>
</tbody>
</table>

Application of IACC to TCX mode (Table 4.4) has to be flexible because of the unary Codebook numbers. TCX employ 32 * 8 dimensional vectors are quantized by (0, 4, ..., 20) bit quantizers. For each vector, the length (codebook number) of the chosen quantizer is sent to the receiver by a unary codeword. We employ a convolutional code with a rate of 1/3 [94] to protect the first half of codebook numbers (starting from the end of the stream) and 2/3 for the second half. Starting from the lowest frequency, first half of the codebook indices are encoded with 1/2 IACC codes and the second half is encoded with 2/3 as long as there is remaining bit reservoir. If a codebook index in first half is encoded with more than 8 bits, it is channel encoded with convolutional code of rate 1/2. Similarly, if a codebook index in the last half is encoded with more than 12 bits, it is channel encoded with convolutional code of rate 2/3.
Table 4.2 TCX parameters and IACC protection in AMR-WB+

<table>
<thead>
<tr>
<th>Parameters</th>
<th>ISF vectors</th>
<th>Noise factor</th>
<th>Global gain</th>
<th>Algebraic codebook</th>
<th>Algebraic codebook</th>
<th>Highband ISF</th>
<th>High-Band Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Bits</td>
<td>46</td>
<td>3</td>
<td>7</td>
<td>Around 50 (Variable)</td>
<td>Around 148 (Variable)</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>Number of parameters</td>
<td>7</td>
<td>1</td>
<td>1</td>
<td>32</td>
<td>32</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Bit Allocation</td>
<td>8,8,7,7,6,5,5</td>
<td>3</td>
<td>7</td>
<td>Variable</td>
<td>Variable</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>IACC Allocation</td>
<td>10,8,2,2,8,1,1</td>
<td>3</td>
<td>11</td>
<td>Around 75 (Variable) (Convo)</td>
<td>Around 90 (Variable)</td>
<td>9</td>
<td>7</td>
</tr>
</tbody>
</table>

4.3.2.3 Training

It would be desired to map all quantization levels to channel codebook so that $D_r$ (4.5) is minimized. However this will require a search of $(2^k)/(2^e - 2')!$ combinations and can therefore be computationally intractable. Instead, BSA can be used iteratively to make sure that $D_r$ is reduced to a local minimum. In each iteration, $s_i$ with the highest $D_r$ is searched and $M(s_i)$ is switched to a new channel codeword such that $D_r$ is minimized. Extensive information about BSA can be found in [58]. The main difference with [58] is that $M(s_i)$ can be switched to an invalid codeword which then becomes valid.

4.3.3 Application of IACC-CC

4.3.3.1 Generic Model

As seen in section 4.2.2, despite being nonlinear IACC suffers from the restrictions of block coding in low BERs. Another disadvantage of the IACC is the limit on maximum length of the code due to the
training and running complexity. Consequently, an alternative scheme named IACC-CC has been developed where IACC and convolutional coder are concatenated. The scheme works as follows: Firstly the total channel coding rate (for example $\frac{1}{2}$) for a codec parameter with length (for example, 8) is decided. Then the parameter is encoded with (12,8) IACC code. Then the codeword and the IACC codewords of other codec parameters fed into a convolutional coder of rate $En(Con)$, to obtain a total coding rate of $\frac{1}{2}$. Coding rates for different parameters can also be adjusted either by concatenating a fixed rate convolutional coder to various rate IACC codes, or concatenating a variable rate recursive convolutional rate code to IACC codes which have the same rate. The generic diagram of IACC-CC is given in Figure 4.5.

4.3.3.2 Technical Configuration

Two fixed rate convolutional coders with rates 4/5 and 3/4 are considered [94]. These are named as IACC–CC-1 and IACC–CC-2 respectively. In both cases, the rates of IACCs are arranged such that concatenated coding rate of IACC and convolutional code for every individual parameter is as similar as possible with its pure IACC counterpart.

4.3.3.3 Training

Training is similar to the pure IACC case except that transition probabilities between codewords is no longer a simple function of hamming distance as the additional inner layer of convolutional code affects the residual bit error patterns IACCs face. The transition probabilities between the input sequences of convolutional encoder and output sequences of convolutional decoder are obtained by simulating a sufficient number of input sequences and random error vectors. IACCs are then trained according to these transition probabilities.
4.3.4 Selection of the distortion measure

For each of the parameters AMR-WB+ codec produces, IACC code is optimized by BSA which needs a distortion criterion. The distortion criterion can be simply set to parameter MSE or more complex measures such as the perceptual effect of the parameter distortion on the speech/audio quality could be used. We have compared the performances of MSE and PESQ based distortion measure. In [95], it was stated that PESQ achieves a correlation rate of 0.9 with subjective MOS test. PESQ calculates a value called frame disturbance for every 32 ms frames then uses these values for calculating the final PESQ score for approximately 10 s speech segments. We used the frame disturbance values as the distortion criterion such that for example a pitch lag codevector obtained from a real speech segment is switched to another vector and the resulting frame disturbance is considered as the distortion between these codevectors. However, since distortion can change from frame to frame, measurements on a sufficiently large number of frames have been done and mean and variance of the frame disturbance have been calculated. Sum of the mean and a suitable factor of variance then have been declared as the distortion between codevectors. Variance factors have been included in order to ensure that large frame disturbances between two codevectors in some occasions are not underestimated due to averaging.

Figure 4.6 shows the performance comparison between IACC coders optimized with SNR and PESQ. PESQ optimized IACC outperforms parameter MSE optimized IACC at all BERs. The difference increases with the BER. This is most probably due to the fact that PESQ penalizes the large distortions more and these distortions are better compensated with PESQ based optimization scheme. Since PESQ doesn’t measure general audio quality, only the ACELP part of the AMR-WB+ is used in this experiment. Although the testing procedure is obviously in favor of PESQ training, this experiment indicates that selection of distortion criterion can play an important role in the performance of IACC and the flexibility in distortion criterion can be an important advantage of IACC against conventional channel coding schemes.
Figure 4.6 Comparing the performance of IACC codes optimized with SNR and PESQ

4.3.5 Results

4.3.5.1 Objective Results

In this subsection, the performance of pure IACC, IACC-CCs and pure convolutional coding are compared objectively. The convolutional coders used are punctured from the recursive systematic convolutional codes (RSC) with constraint length 5 which are defined in [94] and used in [93] for AMR-WB+ previously. The convolutional coder is punctured such that it provides various coding rates like 1/4, 1/2, 3/4 etc. Bits are then placed on the stream according to their segmented SNR sensitivity. Some less important bits are not protected. In this way, UEP can be applied to AMR-WB+ stream according to the error sensitivities of the bits and a powerful reference against IACC schemes is provided. IACC based schemes are also designed in effort to maximize PESQ. The input audio consists equally of music and speech.

In the experiments, each channel coder is fed with 1,000,000 mono audio frames and then each channel coder output is artificially corrupted with 10000 memoryless error vectors with specified BER. Channel is binary symmetric without memory.
Chapter 4. Index Assignment based Channel Coding and its application to AMR-WB+

Figure 4.7 Comparing the PESQ performances of IACC, IACC-CC-1, IACC-CC-2 and convolutional coding

Figure 4.7 shows the average segmented PESQ results for all schemes. It is seen that at low BERs, convolutional coder outperforms the developed schemes. In the 0.006-0.01 region, IACC-CC-1 outperforms all other schemes, in the 0.02-0.03 region IACC-CC-2 outperforms the others and IACC performs best in the high BER region since the convolutional coders collapses at these BERs.

Despite the fact that it is operating on blocks and therefore comprising important disadvantages of block coding, IACC has given good results at BERs higher than 0.03. Various allocations of redundancy between IACC and convolutional coding give better results than convolutional coding at the BERs between 0.006 and 0.2.

4.3.5.2 Subjective Results

The performance of IACC based schemes in the previous experiment is promising however it is known that high SNR may not correspond to high speech quality. Furthermore, segmented SNR does not punish the high variance in distortion and may therefore underestimate the potential of IACC. Accuracy of PESQ in noisy channel conditions is also debated. Consequently, a listening test using the Comparative Category Rating method as described in [96] has been carried out in order to compare the performance of the developed schemes with the conventional convolutional coding. All schemes are
designed in effort to maximize resulting PESQ. Other differences with the previous experiment are that only speech segments are used in training and AMR-WB+ is forced to run in ACELP mode only. Higher band information is also neglected since PESQ is able to measure the quality of speech restricted to below 8 kHz.

In a special listening room, 20 listeners have been asked to listen and compare the quality of two sentences of male and two sentences of female speech processed with different configurations. The sentences have been taken from the NTT database, downsampled to 8 kHz, MIRS filtered and encoded with AMR-WB+ in ACELP mode.

The following configurations have been included in the listening test:

- IACC trained and tested at BER=0.04
- IACC-CC-1 trained and tested at BER=0.01
- IACC-CC-2 trained and tested at BER=0.002
- Convolutional coder with UEP at BER=0.05, 0.01, 0.002

For each sentence pair being tested, the first sentence is identified as being much better, slightly better, slightly worse or much worse than, or about the same quality as the second sentence. These opinion scores are then mapped to numerical values using 2, 1, -1, 2 or 0 respectively. The combined results are given in Table 4.5 with their 95% confidence intervals. A positive score means that the first configuration has been found to be somewhat better than the reference configuration and a negative score means vice versa. As it becomes difficult to distinguish the two codecs being compared, the score gets closer to 0.

<table>
<thead>
<tr>
<th>Tested Configuration</th>
<th>Reference</th>
<th>Score</th>
<th>Confidence Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>IACC @ 0.04</td>
<td>Con, UEP @ 0.04</td>
<td>+1.124</td>
<td>±0.012</td>
</tr>
<tr>
<td>IACC-CC-1 @ 0.02</td>
<td>Con, UEP @ 0.02</td>
<td>+0.89</td>
<td>±0.027</td>
</tr>
<tr>
<td>IACC-CC-1 @ 0.01</td>
<td>Con, UEP @ 0.01</td>
<td>+0.56</td>
<td>±0.063</td>
</tr>
<tr>
<td>IACC-CC-2 @ 0.005</td>
<td>Con, UEP @ 0.005</td>
<td>+0.29</td>
<td>±0.078</td>
</tr>
<tr>
<td>IACC @ 0.04</td>
<td>Con, UEP @ 0.02</td>
<td>-0.12</td>
<td>±0.014</td>
</tr>
</tbody>
</table>
In the first four rows of Table 5, the best developed scheme at that BER is compared against the convolutional coder. It is seen that developed schemes clearly outperform the convolutional coder at BERs tested. In accordance with the objective tests, performance gap increases as the BER increases. In the fifth row, it is interesting to see that IACC tested at 0.04 subjectively performs very similarly to the convolutional coder tested at 0.02. This is due to the fact that even though UEP is applied to the convolutional coder, occasional errors on some very important bits deteriorate the perceptual performance significantly. IACC, on the other hand, implicitly 'spreads' this importance to a larger number of bits and thus prevents sharp deteriorations in the speech quality. This is at the expense of a more persistent but less impulsive and consequently less disturbing noise in the speech. Subjective results indicate that IACC can adapt to different distortion criteria effectively and the implicit gain over conventional channel coding in objective tests becomes more prominent.

4.4 Conclusion

In this paper, a source optimized channel coding scheme called IACC is presented and its application to AMR-WB+ is evaluated. The main advantage of this scheme over other joint source channel coding techniques is its compatibility with a given source codec without significant drop in the performance.

It is shown that source distribution is an important factor in the relative performance of IACC against conventional channel coding. As the kurtosis of the source parameter increases, employing IACC becomes more meaningful. Choice of the perceptual distortion criterion is also important since the perceptual kurtosis of the signal might be different than the mathematical one. Especially, in the cases where outliers are perceptually unacceptable, IACC becomes a powerful error protection scheme.

Although pure IACC is only promising at high BERs, the proposed IACC-CC schemes give better performance than the benchmark convolutional coder at medium BERs.

In the experiments, transmission channel is assumed to be memoryless as the interleaving in the system configuration would be sufficient to randomize channel errors. However, preliminary experiments results show that IACC performs even better against convolutional coding in channels with memory because of it can be trained with respect to those channels.
Chapter 5

5 Multiple Description Coding for AAC

5.1 Introduction

Advance Audio Coder (AAC), which was briefly discussed in Chapter 2, is accepted as state of the art codec in the audio coding literature. It is logical to think that a multiple description audio coding scheme can benefit significantly from the established audio compression algorithm of AAC, which has proved to be very efficient especially in medium to low bit rates. In this chapter, application of multiple description coding to AAC is discussed.

Although there are several publications about application of different multiple description techniques to audio coding, different multiple description techniques have not been applied to the same audio coder at similar bit rates, packet loss rates etc. It is necessary to find the possible optimum range for each multiple description coding technique in application to audio transmission. Two approaches, namely Multiple Description Transform Coding (MDTC) and Multiple Description Vector Quantisation (MDVQ), have been considered as possible ways of applying multiple description coding to AAC. Previously, MDTC has been applied to PAC [97] and MDSQ has been applied to MPEG Layer 2 Audio Coder [98]. However, it has been realised that there is still room for further
improvement of the application of these schemes to audio coding. Challenges in the adaptation of these methods to audio coding and specifically to AAC have been addressed. Novel methods to increase the performance of these schemes have are suggested.

Since AAC does not provide much extra coding gain above 64 kb/s, experimental setup in this chapter assumes a base rate of 64 kb/s or below. However, in other codecs with less efficient coding algorithms such as MP3, the experimental range would exceed 64 kb/s. On the contrary, for more efficient coders like HE-AAC v1 upper limit in the experimental range would be below 64 kb/s.

As the side information is duplicated in two-channel multiple description coding, the reduction of the side information can be very beneficial especially at low bit rates, where side information takes an important part of the overall bit rate. A novel scheme to decrease the overhead of side information is going to be presented. This scheme is then used in conjunction with the proposed multiple description coding techniques to show its contribution to the coding performance in low rate multiple description audio coding.

Performances of all schemes have been compared in error free and error prone packet network conditions. Different base rates and different amounts of redundancy for the suggested schemes have been considered and the optimal operation ranges of the different methods are discussed.

Section 5.2 discusses the application of MDTC to AAC and suggests techniques to improve its performance. Section 5.3 discusses the application of MDVQ to AAC. Section 5.4 discusses the proposed side information compression scheme and shows its benefits at low bit rate multiple description coding. Section 5.5 compares single description AAC, MDTC-AAC and MDVQ-AAC in lossy channel conditions. Section 5.6 gives the conclusion of the chapter.

5.2 Application of Multiple Description Correlating Transform to AAC

In subsection 5.2.1, correlating transform which will be used in subsequent experiments is reviewed. In subsection 5.2.2, adaptation of the transform to AAC and a novel method which takes perceptual information in redundancy allocation is introduced. Then, two novel methods for improving the performance of MDTC in AAC context are presented in subsections 5.2.3 and 5.2.4. White boxes in Figure 5.1 indicate the points in AAC algorithm where the novel work is done.
5.2.1 Correlating Transform

Correlating linear transform matrix, \( T_\alpha \) is given by [89],

\[
T_\alpha = \begin{bmatrix}
\alpha & (2\alpha)^{-1} \\
-\alpha & (2\alpha)^{-1}
\end{bmatrix}
\]  

(5.1)
This matrix is in the optimal set of correlating matrices with the extra advantage of giving two outputs with equal variance and thus equal rate. The parameter $\alpha$ controls the trade-off between redundancy and quality when one component is lost.

Since this transform is in general non-orthogonal, quantisation in the transform domain leads to performance loss. Therefore a "lifting" transform is applied. Quasilinear integer to integer equivalent of $T_\alpha$ is given by,

$$T_\alpha(x_q) = \left( \begin{array}{cc} 1 & 0 \\ 1-2\alpha & 1 \end{array} \right) \left( \begin{array}{c} 1 \\ 0 \end{array} \right) \left( \begin{array}{c} 1 \\ 2\alpha(\alpha-1) \end{array} \right) \left( \begin{array}{c} x_q \\ 1 \end{array} \right) \right)$$

(5.2)

where $x_q$ is the quantised version of original two dimensional vector, $x$. If both components are received, the original coefficients can be reconstructed with the inverse transform:

$$T_\alpha^{-1} = \left( \begin{array}{cc} (2\alpha)^{-1} & -(2\alpha)^{-1} \\ \alpha & \alpha \end{array} \right)$$

(5.3)

If one of the components is missing, the original coefficients can be estimated with

$$\hat{x}_q = \frac{2\alpha}{4\alpha^4\sigma_1^2 + \sigma_2^2} \left[ 2\alpha^2 \sigma_1^2 \right] y_1$$

(5.4)

or

$$\hat{x}_q = \frac{2\alpha}{4\alpha^4\sigma_1^2 + \sigma_2^2} \left[ -2\alpha^2 \sigma_1^2 \right] y_2$$

(5.5)

where $\sigma_1^2$ is the variance of the coefficient with larger variance and $\sigma_2^2$ is the variance of the coefficient with smaller variance. Given the allocated redundancy for the pair and the variances of input coefficients, $\alpha$ can be set as [89]

$$\alpha = \frac{\sigma_2}{\sqrt{2\sigma_1 (2^p - \sqrt{2^{2p}} - 1)}}$$

(5.6)

where $p$ is the redundancy allocated for the pair. The average one sided distortion for the pair is then given by [89],

$$D = \frac{1}{2} \sigma_2^2 + \frac{\sigma_1^2 - \sigma_2^2}{4.2^p (2^p + \sqrt{2^{2p}} - 1)}$$

(5.7)
5.2.2 Application to AAC

When there are more than two coefficients to code as in audio coding, the pairing should be arranged to minimize the average distortion per coefficient. Considering the best pairing configuration, the average distortion per coefficient is given by [89]

\[ D = \frac{1}{2M} \sum_{m} \sigma_{\alpha_{m}}^{2} + \frac{1}{2} \left( \prod_{m} \gamma_{m}^{2} \right)^{1/M} 2^{-4p} \]  

(5.8)

where \( m \) represents the \( m^{th} \) pair, \( M \) represents the total number of pairs and

\[ \gamma_{m}^{2} = (\alpha_{1,m} - \alpha_{2,m})/4 \]

(5.9)

Best pairing configuration is such that [89] the coefficient with largest variance is paired with the coefficient with the smallest variance and etc. In [89], the pairings are done only within scalefactor bands to avoid the complications due to the inter-band pairing. For the time being, pairing is done in the same manner with [89]. However, in Section 5.2.4, inter-band pairing is applied leading to important coding gains. The redundancy allocation between pairs within a scalefactor band is given by [89],

\[ p_{m} = p + \frac{1}{4} \log_{2} \frac{\gamma_{m}^{2}}{\left( \prod_{i} \gamma_{i}^{2} \right)^{1/M}} \]

(5.10)

where \( p \) is the average redundancy per pair in the band and \( p_{m} \) is the redundancy for the \( m^{th} \) pair. In practice, psychoacoustics can be taken into account in the redundancy allocation procedure. The perceptual variances of the coefficients are different from the mathematical ones. AAC already employs a scalefactor mechanism to differentiate the quantisation precision among different bands in the quantisation process (Equation 2.5). This mechanism can also be used in calculating perceptual variances. Every sample is firstly normalised by using the scalefactor and global gain chosen for that frame by AAC algorithm:

\[ x_{p_{i}} = \text{sign}(x_{i}) \left( \frac{|x_{i}|}{4\sqrt{2} \text{global\_gain\_scale\_factor}} \right)^{0.75} \]

(5.11)

where \( x_{i} \) is the \( i^{th} \) MDCT coefficient from an arbitrary frame in the audio database, and \( x_{p_{i}} \) is its perceptual counterpart. Perceptual variances are calculated by using perceptually weighted
coefficients. Redundancy allocation is then done based on the perceptual variances and estimation formulas also use the perceptual variances.

5.2.3 Parameter Updating

In the theoretical design of MDTC, it is assumed that the coded signal is stationary, which is not valid for real audio signals. Since the estimation is based on the variances of MDCT coefficients, changes in those can affect the accuracy of the coefficients estimated by side decoders. Furthermore, changing variances imply that the optimal pairing of MDCT coefficients and the redundancy allocation among pairs might have changed as well. The decoder can be able to approximately update the variance of the coefficients without receiving explicit update. However, if there are losses in the transmission of the audio packets, the decoder would update the variances wrongly. Furthermore, as the pairing of the coefficients follows the updated variances, pairing can be wrongly configured as well.

In [97], variance information is not mentioned to be transmitted or updated in the decoder. This implicitly means that it is either deduced from the transmitted audio segment and transmitted beforehand, or deduced from a large collection of audio signals and wired to the decoder as a constant. The set of \( \alpha \) and pairing information are mentioned to be transmitted before the streaming of audio starts.

In this work, a new approach is introduced: Transmitting the variance at regular intervals and updating the variance, pairing and redundancy information both in the encoder and the decoder accordingly. This process can be considered similar to the transmission of variances and covariances of a multivariate Gaussian distribution periodically, when Karhunen–Loève Transform (KLT) is performed for coding efficiency. To understand the effect of the frequency of updating on the audio quality, an experiment is performed, where the variance information is not quantised but assumed to be transmitted to the decoder correctly. The side encoders are implemented by adding 20 kb/s to the original encoder bit-rate, where 13 kb/s out of 20 kb/s is for the duplication of AAC side information and 7 kb/s is for the actual MDC redundancy. The original encoder has a rate of 64 kb/s. As seen from the Figure 5.2, the performance of the decoder drops, as the updating interval increases. At low updating intervals, the dropping rate of the performance is rather small and it starts increasing after 0.3 s. After 0.7 seconds, the dropping rate gets lower and the performance converges. Since there isn’t significant gain in lowering the frequency interval beyond 0.3 s, the updating interval will be taken as 0.3 s in the following experiments.
Although updating seems very beneficial in terms of quality, in a real time streaming scenario, the extra bit rate required for finer quantisation and transmission of the side information can overwhelm the benefits. Furthermore, variance information needs reliable transmission, which can increase the effective bandwidth needed.

Consequently, a scheme, where variances of MDCT coefficients in each scalefactor band are vector quantised, is developed to decrease the overhead of MDTC side information. To further increase the coding efficiency, a moving average predictor with order 2 is combined with the vector quantisation. In the encoder, together with scalefactors, these vectors are then used to calculate perceptual variances, to arrange pairings and redundancy allocation and to calculate transform parameters. The vectors' indices are transmitted to the decoder and the same steps are followed in the decoder as well. Updating process is illustrated in Figure 5.3.

In the initial experiments, fine quantisation of the variances of higher bands proved to be unnecessary in terms of the performance gain it brings. Furthermore, the large dimension of the vectors in higher bands complicates the training. Therefore, only gain information about these bands is transmitted and the shape of the variances is set to the mean of the global variances. The optimal cut-off point separating high and low frequency regions depends on the audio content. However, experiments in this research show that the end of the last 20-coefficient long scalefactor band (corresponds approximately to 4.6 kHz) is generally appropriate as a cut-off point. The number of bits used for representing the gain of higher bands is kept fixed (3 bits for each band). This coarse representation of higher bands' variances only costs 0.09 on average in PEAQ scale. In informal listening tests, it has been realised that except for the audio signals with high frequency tonal content, coarse representation of higher
bands variances does not degrade the audio quality. Figure 5.4 shows the performance of one sided decoder with respect to the number of bits allocated for quantising variances of each scalefactor band. The redundancy configuration is the same as with the previous experiment. It is seen that the performance of the vector quantisation scheme is very bad at low rates, even worse than no updating scheme. The performance increases with the bit rate and converges 0.09 below the performance of AAC-MDTC-UPDT without quantisation. Considering the overhead due to reliable transmission of variance vectors, setting the quantisation rate at 220 bits/update seems reasonable, as there isn’t much performance gain beyond that bit rate. The combination of 0.3s updating interval – 220 bits/update is observed to achieve a better performance than the combinations which lead to the same bit rate.
Chapter 5 Multiple Description Coding for AAC

Encoder

AAC Algorithm

Coefficient variances

Quantize variances

Calculate perceptual variances

Arrange pairings

Calculate redundancies

Calculate transform parameters

Scalefactors

Decoder

Quantized variances

Calculate perceptual variances

Arrange pairings

Calculate redundancies

Calculate inverse transform parameters

Scalefactors

AAC Algorithm

Figure 5.3 Updating process in AAC-MDTC

Figure 5.4 Effect of VQ rate on the performance
5.2.4 Inter-band pairing

In the previous works [97] and previous experiments in this thesis the pairing of the coefficients has been done in the same band, since it is favourable in terms of system simplicity. However, as it is given by (5.7), while the redundancy gets higher, the average distortion per pair converges to the variance of the coefficient with the smaller variance. This means that when the pairing is done within a scalefactor band, the distortion of a band converges to the sum of variances of the coefficients whose variances are in the lower half of the variances in that band. The overall distortion is then the sum of all bands’ distortion. On the other hand, if inter-band pairing is allowed, the overall distortion will converge to the sum of the variances of the coefficients whose variances are in the lower half of all variances. This is either equal or lower than the intra band-pairing.

Let us have a spectrum, which is simply represented by 4 coefficients and a 2 band system as an example to illustrate the difference between intra-band and inter-band pairing. Each band has 2 coefficients. The coefficients in the first band have variances as \( \sigma_1 = 1.0, \sigma_2 = 0.7 \) and the coefficients in the second band have variances \( \sigma_3 = 0.4, \sigma_4 = 0.2 \). In the intra-band pairing scheme, 1st and 2nd coefficients form a pair and 3rd and 4th coefficients form the second one. As the redundancy increases, according to Equation 5.8, the total distortion per coefficient converges to 0.4. Similarly, if inter-band pairing is allowed, 1st and 4th coefficients will form one pair and 2nd and 3rd coefficients will form the second one. According to Equation 5.8, total distortion will then converge to 0.2. As seen from this simple example, inter-band pairing shows important potential over intra-band pairing.

There are several issues in actually implementing inter-band pairing in the context of AAC. Firstly, coefficients from different bands are likely to have different scalefactors. To conserve the perceptual importance of the coefficients assigned by AAC algorithm, the quantisation needs to be done as in original AAC. Secondly, arbitrary pairing will result in a complex system, such that assigning Huffman tables to the coefficients will be very difficult. Instead, coefficients in a band can be paired with the coefficients in another specific band. Bands having coefficients with the highest variances are paired with the bands having the lowest frequencies for the sake of improved coding efficiency. As the average audio spectrum tends to diminish with increasing the frequency, this generally means that bands in the lowest frequency range can be matched with the bands in the highest frequency bands and so on. Several bands in lower frequency range can be paired with the same band in higher frequency range, since higher frequency bands are comprised of a bigger number of coefficients. A lower frequency band is constantly matched with a particular location in a higher frequency band. However,
when parameter updating is applied, coefficients can change pairs within their associated band/location. Note that bands, which are higher than a certain frequency threshold, are omitted from pairing.

In the Huffman coding of the coefficients, a Huffman table is assigned to each group of pairs according to the standard AAC algorithm. Figure 5.5 shows the side decoder PEAQ performance of inter-band pairing with respect to intra-band pairing as a function of redundancy. Both cases employ parameter updating with 0.3s updating interval and 220 kb/s VQ rate. Configuration is the same with the last two experiments. Although indecisive at low redundancy rates, the performance gap between inter-band and intra-band pairing increases with the redundancy.

---

**Figure 5.5 Inter and Intra-band pairing in AAC-MDTC**

---

5.3 Application of Multiple Description Vector Quantisation to AAC

Section 5.3.1 explains why unconstrained MDVQ is chosen as the method of applying MDVQ to AAC. Section 5.3.2 reviews the theoretical design of unconstrained MDVQ. Section 5.3.3 describes the practical application of MDVQ to AAC, how the redundancy allocation between the AAC coefficients is done and the design methodology of the Huffman codebooks for side descriptions. Section 5.4.4 suggests a parameter updating method for MDVQ. White boxes in Figure 5.6 indicate the points in AAC algorithm where the novel work is done.
5.4 Unconstrained MDVQ

In order to apply multiple description quantisation to AAC, several methods can be considered. One of them is Multiple Description Scalar Quantisation (MDSQ). Although in AAC, quantisation of MDCT coefficients is done on a scalar basis, Huffman coding of coefficients with 2 and 4 tuples brings a vectoral aspect to the overall AAC coding. Therefore in order to apply MDSQ to AAC, one should compromise the advantage of joint Huffman coding of AAC. Another alternative is Multiple Description Lattice Vector Quantization (MDLVQ). However, MDLVQ comprises constraints on the reconstruction points which conflicts with the pre-existing reconstruction points of AAC. Lastly, unconstrained MDVQ can be applied to AAC. This technique has the advantage of leaving the original
Chapter 5 Multiple Description Coding for AAC

quantiser to be designed without any constraints. Next, the theory of unconstrained MDVQ is going to be explained.

5.4.1 Design of unconstrained MDVQ

Let a predesigned quantiser select the closest codevector, $y_I$, to the input vector $X$ where $y_I$ is from the codebook $Y = \{y_0, y_1, ..., y_{N-1}\}$ and $N$ is the size of the codebook. Let the index assignments be given by $I_i = a_i(I), i=1, ..., K$ where $K$ is the number of descriptions. $K$ is chosen as 2 in this thesis.

In [88], the framework of the optimal decoder is given when one description is lost, assuming a fixed codebook and a fixed index assignment. When the channel loss probabilities are equal, the conditional probability of codebook index, $\lambda$, to be received is given by [88]

$$P(I = \lambda | I_1 = \mu, I_2 = NA) = \begin{cases} P(I = \lambda) & \text{if } a_1(\lambda) = \mu \\ 0 & \text{if } a_1(\lambda) \neq \mu \end{cases}$$

(5.12)

(5.13)

where $\mu$ is the received index from channel 1 and NA stands for “not available”. The optimal decoder minimising MSE where the index from one of the descriptions is lost is given by

$$De(\mu, NA) = \sum_{\forall \xi \in S(a_1(\xi) = \mu)} P(I = \xi | I_1 = \mu, I_2 = NA) \cdot y_\xi$$

(5.14)

Assuming equal channel failure probabilities, the expected distortion resulting from one lost description is given by [88]:

$$D = \sum_{\xi=0}^{N-1} C(y_\xi)$$

(5.15)

where $C(y_\xi)$ is the expected distortion due to a single codebook index. $C(y_\xi)$ is given by

$$C(y_\xi) = P(I = \xi) \cdot (d(y_\xi, De(a_1(\xi), NA)) + d(y_\xi, De(NA, a_2(\xi)))$$

(5.16)

Letting $N_1$ and $N_2$ be the sizes of index-sets of description 1 and 2 respectively, placing the $N$ codebook indices to the $M = N_1 \cdot N_2$ locations with a brute-force approach is computationally intractable. In [88], BSA has been successfully employed to optimise the index assignment matrix. Considering this, BSA is employed to optimise the index assignment matrix in MDVQ coding.
5.4.2 Application to AAC

Unlike the seamless calculation of transform parameters in MDTC case, designing index assignment matrixes in AAC context is not straightforward since for each Huffman codebook and for each desired redundancy amount, a separate index assignment matrix has to be implemented.

As an illustrative example, in Figure 5.7 (a), it is shown that moderate redundancy is added to a 10 level scalar codebook whereas in (b), duplication of the information is shown. It should be noted that the decoder’s output is same in both cases when both descriptions are received.

![Index assignment matrixes](image)

Figure 5.7 Index assignment matrixes: (a) moderate redundancy, (b) duplication

The number of redundancy levels designed for each codebook varies depending on the average rate needed for using that codebook. More the average results in more the number of redundancy levels. Although BSA makes multi-levels of redundancy possible, excessive number of index assignment matrixes would make updating scheme (which will be discussed later) hard to realise. In addition to this, using excessive number of matrixes doesn’t seem to improve the performance significantly. Because of these reasons, the number of index assignment matrixes for each codebook is kept limited to 12.
The index assignment matrixes with the largest numbers represent the duplication of the original data. For example, index assignment matrix 5 associated with codebook 1 provides the exact replication of the original AAC data.

In [98], the redundancy allocation between subbands is done by exploiting underlying MPEG Layer II algorithm. However, AAC provides a much higher frequency resolution than MPEG Layer II and different redundancy allocations can be utilised between the coefficients in the same band. This is done as follows: In order to ensure rate stability, each coefficient is associated with an index assignment matrix number, which represents the amount of redundancy reserved for it. The coefficient is then encoded with the index assignment matrix (having the same number associated with the coefficient) of the selected codebook. Index assignment matrixes are designed such that a matrix is redundancy-equivalent with other matrixes that have the same number. For example, index assignment matrix 4 designed for codebook 1, on average, adds the same amount of redundancy with index assignment matrix 4 designed for codebook 7.

In some cases where a specific Huffman table is chosen to encode a coefficient, it is possible that pre-allocated redundancy exceeds the rate needed to duplicate the original rate. For example, a coefficient can be associated with the index assignment matrix number 8. However, for a particular frame, AAC can choose codebook 2 which does not have index assignment matrix 8 (which would exceed the duplication rate) designed for it. In this case, the coefficient is associated with the maximum index assignment matrix number available i.e. 5, for that frame. The residual redundancy is then assigned to another coefficient in the same scalefactor band where possible or a coefficient in another band.

Since there aren’t any analytical algorithms presented in the literature, redundancy allocation between coefficients in MDVQ is not as straightforward as in MDTC. Here is the algorithm proposed for the redundancy allocation to the coefficients:

1) Using a large audio database, calculate the initial no-redundancy one-side distortion for all coefficients.

2) Increase the redundancy (index assignment matrix number) of the coefficient such that the total distortion is decreased maximally.

3) If the total redundancy<allowed redundancy, go back to step 2.

4) Terminate.

Actual design of side description codebooks is done using a probabilistic approach exploiting the already known probabilities of the AAC Huffman codebooks. The occurrence probability of a
description entry can be calculated by summing up the probabilities of codebook indexes which are lying on the axis of that particular entry. The resulting side description codebooks have entries with uneven chances of occurrence and therefore need to be represented by variable length codebooks for the sake of coding efficiency. Once the probabilities of the entries of the side codebook entries are known, the corresponding Huffman codebooks can be designed. One side description for a particular index assignment matrix can have slightly higher rate than the other description, but since this difference is random and there are 1024 coefficients, rates of the two descriptions will be globally very similar.

5.4.3 Parameter updating

The fact that the redundancy is nearly constant for a particular coefficient leads to a suboptimal usage of the available redundancy since the audio signals are not stationary. This subsection illustrates the updating of MDVQ parameters. Updating can be done implicitly, i.e. the decoder configures the index assignment selection based on the ordinarily received data. Alternatively the encoder can explicitly send side information so that the decoder and encoder simultaneously configure the index assignment matrix selections. As in MDTC case, explicit transmission is selected since the performance of implicit synchronisation degrades quickly in the presence of packet erasures.

Since there aren't any analytical tools in the literature presented for optimising the redundancy allocation between the coefficients, a brute force approach similar to the one in the previous subsection is considered where the total allowed redundancy is allocated to the coefficients gradually minimising total distortion at each step.

To understand the effect of the frequency of updating on the audio quality, an experiment is performed, where the index assignment information is not quantised but assumed to be transmitted to the decoder correctly. The side encoders are implemented by adding 30 kb/s to the original encoder bit-rate, where 13 kb/s out of 30 kb/s is for the duplication of AAC side information and 17 kb/s is for the actual MDC redundancy. The original encoder has a rate of 64 kb/s. As seen from the Figure 5.8, the performance of the decoder drops as the updating interval increases. At low updating intervals, the dropping rate of the performance is rather small and it starts increasing after 0.3 s. After 0.8 seconds, the dropping rate gets lower and the performance converges. Since there isn't significant gain in decreasing the frequency interval beyond 0.4 s, the updating interval will be taken as 0.4 s in the following experiments. The results are very similar to the MDTC case except the detail that the optimum rate seems slightly higher. This might be due to the fact that precision of added redundancy
is much higher in the MDTC case so that more frequent updating is more meaningful in terms of the performance.

![PEAQ vs Updating Interval](image)

**Figure 5.8** Performance of AAC-MDVQ-UPDT depending on updating interval

Similar to the MDTC case, in a real time streaming scenario, the extra bit rate required for finer quantisation and transmission of the side information of MDVQ parameters can overwhelm the benefits. Consequently, as it is done with MDTC case, no updating is done for the coefficients higher than a cut-off point which is approximately 7 kHz.

Furthermore, a differential coding method which exploits the correlation in the audio signals is proposed. In this method, the index assignment matrix number chosen for a coefficient is encoded with respect to the previous index assignment matrix number. The range of the difference between the two consecutive numbers states the precision and the rate of the side information. If the algorithm chooses a number which is out of the range of differential coding, the differential coder is forced to choose the closest index available. This is the compromise paid for reducing the bit rate allocated for side information. Two coding precisions are considered for testing. First one is a three-tap \{-1,0,1\} which leads to an overhead of 346 bits/update and the second one is a five-tap \{-2,-1,0,1,2\} which leads to an overhead of 505 bits/update. The resulting PEAQ is -2.77 for three-tap coding and is -2.70 for five-tap coding. Both precisions lead to a very close performance to uncoded updating performance (-2.63) which indicates that differential coding is very effective in reducing the updating overhead. Since there isn't much significant gain in employing five-tap coding, three-tap coding is used in the further experiments.
5.5 Side Information Compression

Although promising results have been obtained in the previous subsections, the successful application of MDC to AAC has been restricted to the cases where the base rate is high. This is due to the fact that duplicated side information occupies a large proportion of the total bit-rate at lower bit-rates.

In basic AAC, there are two types of side information, which are allocated significant fraction of the total bit-rate. These are the scale factors and the choice of Huffman table information for each section. Scale factors are coded by taking the difference of neighbouring scale factors and then transmitted with variable length coding. For the Huffman codebook information; choice of the Huffman codebook and the length of section are coded with fixed rate and are allocated 9 bits. Out of these 9 bits, 4 bits are used for coding the table number and 5 bits are used for coding the length of the section. In Table 5.1, the ratio of the bits allocated for side information and main information at different rates per channel are shown. It is seen that the bit-rate allocated for side information does not decrease proportionally with the increased overall bit-rate. Thus, the precious bits at low bit-rates are allocated to side information rather than main information. This figure is more pronounced when multiple description coding is applied to AAC and the side information is duplicated. Assuming a case, where multiple description coding is applied to AAC operating at 16 kb/s, each description will need to be at least around 12 kb/s to preserve the original quality. When redundancy is applied, the resulting bit-rate will further increase. Unless the packet loss rate is very high, it would be wiser to use the extra bit-rate for increasing the base rate rather using it for multiple description coding. Therefore, side information compression is a necessity to apply multiple description coding to AAC at low bit rates.

<table>
<thead>
<tr>
<th>Rate (kb/s)</th>
<th>Scale Factors (%)</th>
<th>Huffman Codebooks (%)</th>
<th>Main Information (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>9</td>
<td>7</td>
<td>84</td>
</tr>
<tr>
<td>64</td>
<td>11</td>
<td>9</td>
<td>80</td>
</tr>
<tr>
<td>48</td>
<td>15</td>
<td>10</td>
<td>75</td>
</tr>
<tr>
<td>32</td>
<td>18</td>
<td>15</td>
<td>67</td>
</tr>
<tr>
<td>16</td>
<td>27</td>
<td>22</td>
<td>51</td>
</tr>
</tbody>
</table>
There have been several publications about side information compression for AAC. In [99], vector quantisation of scale factors is proposed. Since the coding of all scale factors with a single vector implies intractable training and real-time complexity, several methods are suggested to realise vector quantisation. Firstly, DCT is applied to scale factors, in order to decrease the dimensionality. Then a sub-vector approach is employed. It is shown that 50% bit reduction in scale factors can be obtained without introducing intolerable noise in the tested audio. In [100], Sreevinas et al further improved their work and employed the saved bits for coding the main information. It is shown that at all rates below 64 kb/s, VQ based scheme outperforms the original AAC scheme. As the bit-rate gets lower, the gain over the original scheme increases.

In AAC, a very small scalefactor will lead to a band containing many 0-valued coefficients. This can consequently mean the selection of the Huffman table 0, which means that all coefficients are quantised to 0. On the contrary, a very large scalefactor will lead to a band containing large valued coefficients. Some of the values may be larger than 16, which necessitates the usage of Huffman table 11. Between these extremes, small scalefactors generally lead to selection of Huffman tables with 4 tuples which encode small values and large scale factors lead to selection of Huffman tables with 2 tuples which encode large values. Considering this correlation between scalefactors and the choice of Huffman table, a scheme where the codebook number of a section is entropy encoded with respect to the average scalefactor in each section has been developed. By analysing large segments of audio data, probabilities of choosing Huffman tables according to 10 scalefactor intervals are investigated. Occurrence probabilities for the lowest scalefactor interval are given in Table 5.2. As expected, it is seen that Huffman tables used with lowest scalefactor interval are quite limited and entropy coding is likely to save an important number of bits used for coding Huffman Codebook numbers.
Table 5.2 Occurrence probabilities of Huffman codebooks given the lowest scalefactor range

<table>
<thead>
<tr>
<th>Codebook Number</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>% 78</td>
</tr>
<tr>
<td>1</td>
<td>% 12</td>
</tr>
<tr>
<td>2</td>
<td>% 8</td>
</tr>
<tr>
<td>3</td>
<td>% 1</td>
</tr>
<tr>
<td>4</td>
<td>% 1</td>
</tr>
<tr>
<td>5</td>
<td>% 0</td>
</tr>
<tr>
<td>6</td>
<td>% 0</td>
</tr>
<tr>
<td>7</td>
<td>% 0</td>
</tr>
<tr>
<td>8</td>
<td>% 0</td>
</tr>
<tr>
<td>9</td>
<td>% 0</td>
</tr>
<tr>
<td>10</td>
<td>% 0</td>
</tr>
<tr>
<td>11</td>
<td>% 0</td>
</tr>
</tbody>
</table>

Using a similar probability table, a Huffman table is designed for coding the Huffman tables for each scalefactor interval. On average, 4 bits used for coding the Huffman table decreases to 2 bits with the suggested method. Since 30 sections are coded per frame, this translates to a saving of 3 kb/s, which is useful at low bit rates.

To determine the effect of the saved bits in the quality of audio, these bits are used for main information and the results are compared with the original scheme. This new scheme is named as AAC-SIDE-COMPRESSION or AAC-SC. PEAQ results are shown in Table 5.3.

Table 5.3 Effect of side information compression on AAC performance

<table>
<thead>
<tr>
<th>Rate</th>
<th>Original AAC</th>
<th>AAC-SC</th>
</tr>
</thead>
<tbody>
<tr>
<td>64 kb/s</td>
<td>- 0.63</td>
<td>- 0.57</td>
</tr>
<tr>
<td>48 kb/s</td>
<td>- 1.56</td>
<td>- 1.51</td>
</tr>
<tr>
<td>32 kb/s</td>
<td>- 2.88</td>
<td>- 2.82</td>
</tr>
<tr>
<td>16 kb/s</td>
<td>- 3.47</td>
<td>- 3.41</td>
</tr>
</tbody>
</table>

It is seen that at all bit-rates the proposed scheme outperforms the original scheme. Although the numerical difference is steady among different bit-rates, informal listening tests show that the difference is perceived more significantly at lower bit-rates. This is expected, as the ratio of the saved bits to the overall bit-rate is higher in the low bit rates.
Although the 3 kb/s saving is meaningful in terms of quality, the application of multiple description to AAC at very low bit-rates is still a challenge. Therefore, in addition to the proposed technique, one of the techniques described above (VQ of scalefactors) is embedded to multiple description AAC configurations. Table 5.4 shows the side decoder performance of AAC-MDTC, AAC-MDTC with vector quantisation of scalefactors (AAC-MDTC-VQ), AAC with vector quantisation of scalefactors plus compression of Huffman codebook numbers (AAC-MDTC-VQ-SC) with respect to base rate. The redundancy is kept constant at 4 kb/s. The bit-rate required for the duplication of the side information is extracted from the base rate. The bit rate saved in AAC-MDTC-VQ and AAC-MDTC-VQ-SC are used for increasing the base rate.

<table>
<thead>
<tr>
<th>Rate</th>
<th>AAC-MDTC</th>
<th>AAC-MDTC-VQ</th>
<th>AAC-MDTC-VQ-SC</th>
</tr>
</thead>
<tbody>
<tr>
<td>64 kb/s</td>
<td>-3.00</td>
<td>-2.91</td>
<td>-2.87</td>
</tr>
<tr>
<td>48 kb/s</td>
<td>-3.21</td>
<td>-3.12</td>
<td>-3.05</td>
</tr>
<tr>
<td>32 kb/s</td>
<td>-3.49</td>
<td>-3.39</td>
<td>-3.32</td>
</tr>
<tr>
<td>16 kb/s</td>
<td>-3.86</td>
<td>-3.69</td>
<td>-3.62</td>
</tr>
</tbody>
</table>

It is seen that AAC-MDTC-VQ-SC outperforms the other schemes at all base rates. Most of the gain comes from the vector quantisation of scalefactors, as the difference between AAC-MDTC and AAC-MDTC-VQ is higher than the difference between AAC-MDTC-VQ and AAC-MDTC-VQ-SC. However, the difference between AAC-MDTC-VQ and AAC-MDTC-VQ-SC is audible, especially at lower base rates. One interesting result is that the performance of AAC-MDTC at 64 kb/s and AAC-MDTC-VQ-SC 48 kb/s are similar in terms of PEAQ. This shows the potential of combination of vector quantisation of scalefactors and proposed side information compression scheme in the application of multiple description coding to AAC.

5.6 Comparison in lossy channel conditions

5.6.1 Experiment setup

This subsection compares the developed schemes' performance with the single description coding in various loss rates. Different base coding rates are also considered. To provide a fair comparison, single description coder uses the sophisticated error concealment technique called GAPES in the case
of packet loss [57]. Multiple description schemes also employ this algorithm when both descriptions are lost. Furthermore, instead of assuming that it is sent by reliable retransmission, the parameter updating information in multiple description schemes is accepted to be transmitted with the ordinary channels. This side information is placed in 4 different packets where ordinary audio frames are transmitted; 2 repetitions per channel. The bit rate required for transmitting updating information is 3460 bits/s in MDTC case and 2933 bits/s in MDVQ case.

The experiments are designed such that in multiple description coders, a fixed amount of redundancy is added to duplicate the side information and to code the parameter updating information. The additional redundancy is either used for increasing the base rate or for increasing the side decoder precision. At the left-end of the graphs, all of the additional redundancy is used for increasing the base rate. At the right-end of the graphs, all of the additional redundancy is used for increasing side decoder precision. For the single description coders, all of the redundancy is used for increasing the base rate. This experiment setup not only compares multiple description coders with the single description coder but also provides the trade-off points between side decoder precision and main decoder precision. The packet loss patterns are obtained using RUDEN and CRUDE network simulation programs [101].

5.6.2 Experiments at the base rate of 64 kb/s

Figure 5.9 shows the PEAQ results at various packet loss rates. In multiple description coders, redundancy of 10 kb/s is added for the duplication of side information and the parameter updating information. The additional redundancy of 15 kb/s is shared between the base coder and the side coder. All configurations have a total rate of 89 kb/s.

Figure 5.9: Performance of SDC and MDC schemes at the base rate of 64 kb/s as a function of redundancy
5.6.3 Experiments at the base rate of 32 kb/s

Figure 5.10 shows the PEAQ results at various packet loss rates. In multiple description coders, redundancy of 8 kb/s is added for the duplication of side information and the parameter updating information. The additional redundancy of 11 kb/s is shared between the base coder and side coder. All configurations have a total rate of 51 kb/s.

Figure 5.10: Performance of SDC and MDC schemes at the base rate of 32 kb/s as a function of redundancy.


5.6.4 Discussion

In Figure 5.9 (a), it is seen that the performances of MDTC and unconstrained MDVQ drop as more redundancy is allocated to side decoder precision. This is expected as there is no benefit in shifting the redundancy from base coder to side precision when there is no packet loss. At Figure 5.9 (b), MDTC offers a range of redundancy points which are better than single description coder. In Figure 5.9 (c), unconstrained MDVQ is seen to outperform both single description and MDTC schemes. This difference is more pronounced as the loss rate gets higher. It is noticeable that performance of MDTC peaks around and sharply drops after the redundancy rate of 6 kb/s. This is because in MDTC, side decoder precision can not be improved after a certain threshold and there is no point in shifting the redundancy to side decoder precision from the base coder. On the other hand, since the side decoder performance of unconstrained MDVQ increases more gradually and constantly, the peak point in overall performance differs depending on the loss rate.
At the base rate of 32 kb/s, the general picture is similar to 64 kb/s. However, multiple description coders outperform single description coder at loss rates higher than 2%. This is because at lower base rates, redundancy is better exploited by increasing the base rate rather than increasing the side decoder precision. The results also imply that without the side information compression techniques, multiple description schemes would not bring any advantage over single description coding at low loss rates and low base rates.

Informal listening tests at points where the PEAQ performances of multiple description and single descriptions schemes are close, such as at 1% packet loss rate and at 64 kb/s base rate, show that the slight steady drop in the quality of multiple description scheme is preferable to the impulsive “clicks” due to the packet losses in single description coding.

5.7 Conclusion

In this chapter, several techniques have been suggested to improve the performance of established multiple description schemes in AAC context. The performance of MDTC, a very effective multiple description coding technique at low redundancy rates, has been improved with a parameter updating technique. With this technique, variance, redundancy allocation and pairing are updated regularly. Employing a moving average - vector quantisation technique, it is ensured that there isn’t significant overhead in terms of bit rate. Another technique which allows pairings between coefficients from different bands has been suggested and shown to be especially effective as the redundancy increases.

In order to increase robustness to high loss rates, unconstrained MDVQ has been suggested to exploit high redundancy. Methods for redundancy allocation between the coefficients and designing side Huffman codebooks have been suggested. Moreover, a parameter updating scheme has been proposed. With a differential coding approach, this scheme improves the performance without introducing a significant overhead.

In AAC, side information occupies an important percentage of the overall bit rate especially at low bit rates. Since the side information is duplicated in multiple description coding, side information compression is crucial to apply multiple description coding at low bit rates. Consequently, a scheme where Huffman codebook numbers are encoded with respect to scalefactors has been suggested. This scheme is combined with another side information compression scheme leading to improved gains.

A detailed comparison of MDTC, MDVQ and standard single description AAC has been made. It has been shown that at high base rates, multiple description schemes outperform single description coding even at low loss rates. As expected, at low packet loss rates MDTC provides operating points which
lead to better results than MDVQ. On the contrary, at high packet loss rates, MDVQ provides better operating points since it exploits high redundancy rates efficiently.

Overall, it has been shown that with the help of sophisticated optimisation and side information compression techniques, multiple description techniques offer strong error robustness for AAC and there are useful operating points where bit rate sacrificed from base coder can be efficiently allocated to side decoder precision.
Chapter 6

6 Scalable multiple description coding

6.1 Introduction

Heterogeneous networks, various terminals and varying available bandwidth conditions characterise today's multimedia communication systems. Users connect to diverse networks using a wide range of devices with different preferences at anywhere and anytime. Heterogeneity of terminals, networks, and various user preferences impose nontrivial challenges to the video communication systems. In order to deal with these challenges, several encoders can produce bit streams at different specifications (i.e., temporal, spatial, and quality) and these bit streams can be delivered to the requesting devices. However, this method can pose increased storage requirements and unacceptable delays. Therefore, scalable coding is developed as an attractive solution to cope with this heterogeneity problem. Encoding is performed only once and decoding can be performed several times depending on the requirements of network, terminals, and users with scalable coding. Scalable coding allows encoding a multimedia bit stream into a base layer and a number of enhancement layers, which may enhance temporal resolution, spatial resolution, or quality of the content transmitted.

The standards and publications about scalable speech/audio coding are limited when compared to video coding. MPEG 4 offers an audio scalability framework which offers a flexible range of combinations of different codecs [41]:

88
Chapter 6 Quantization for scalable multiple description coding

- AAC layers only
- Narrow-band CELP layer plus AAC
- TwinVQ base layer plus AAC

Depending on the application, either of these possibilities can provide optimum performance. In all cases where good speech quality is a requirement for the case of reception of the core layer only (for example, in a digital broadcasting system using hierarchical channel coding), the speech codec base layer is preferred. If, on the other hand, music should be of reasonable quality for a very low bit rate core layer (for example, Internet streaming of music using scalability), the TwinVQ base layer provides the best quality. If the base layer is allowed to work at higher bitrates like 16 kb/s, a system built from AAC layers only delivers the best overall performance.

At low bitrates, mono transmission may be preferred at the same total bitrates. For higher bit rates, however, stereo transmission is virtually a requirement today. Therefore, stereo enhancement layers can be added as enhancement layers to stereo lower layers.

There are more codec specific scalability schemes for audio coding. Bit Sliced arithmetic coding (BSAC) extension to AAC offers fine step (1 kb/s) scalability. It differs from AAC lossless coding kernel such that the quantized values are not Huffman coded, but arithmetically coded in bit slices. Embedded Audio Coder [102] developed by Microsoft offers a scalable codec with low noise MDCT and implicit psychoacoustic masking technology.

In speech coding, there are a few proposed codecs that offer scalability. G.729.1 [103] offers bandwidth and SNR scalability based on the core layer G.729, which employs CELP algorithm. Enhancement layers employ Time-Domain Bandwidth Extension (TDBWE) and MDCT algorithms. G.718 [104] is a speech and music codec especially designed for Internet traffic, providing scalability using CELP algorithm in the first two layers, and applies Factorial Pulse Coding (FPC) to MDCT coefficients in the enhancement layers.

In this chapter, an SNR based scalability coding scheme where the trade-off between the first and second layer is parametrically controlled is introduced. This scheme is inspired by the parametrically controlled multiple description system proposed in [105].

Like the single layer codecs, multiple description coding can increase the robustness of scalable codecs to packet losses. There are some published papers in the video coding literature studying scalable multiple description coding. In [106], an SNR oriented scalability/multiple description codec...
producing a base layer and two multiple descriptions is proposed. In contrast, Zhao et al [107] proposed a scheme where the base layer is comprised of two descriptions and the same enhancement layer is used for base layer descriptions. In speech and audio coding literature, there aren't any reported scalable codecs that provide multiple descriptions at the same time. In this chapter, a scheme which provides scalability and multiple descriptions for both layers at the same time is proposed.

In section 6.2, the work in [105] is studied and necessary hints for the design of scalable multiple description coding are obtained. Adaptive robustness to packet losses is also studied. In section 6.3, the proposed scalability scheme is introduced and the details of the optimisation by modifying the LBG algorithm are given. Adaptive scalability is studied as well. In section 6.4, the system utilising scalability and multiple description coding at the same time is introduced. This system is parametrically optimised by also modifying the LBG algorithm. Adaptation is again applied. The chapter is finalised by drawing conclusions in section 6.5.

6.2 Quantisation for multiple description coding

6.2.1 Theory

When two quantisers, A and B, are independent, their selection criteria for minimizing individual MSE are as follows:

\[ i^* = \arg \min_i \left[ (C_A(i) - x)^2 \right], \quad j^* = \arg \min_j \left[ (C_B(j) - x)^2 \right] \]  

(6.1)

Kakouros et al [105] proposed a scheme where both quantisers are used to obtain a better combined (central) performance than the individual performances. In order to achieve this, when both descriptions are received, the decoder uses a linear combination of the corresponding codewords to obtain a representation of the input signal such that

\[ \hat{x} = \alpha C_A(i^*) + (1 - \alpha) C_B(j^*) \]  

(6.2)

where \( \alpha \) is weighting factor ranging between 0 and 1 accounting for the fact the descriptions may be of different quality. When only one description is received, only that description is used for the signal representation. In order to obtain a trade-off between the quality of the individual descriptions and the central description, the encoder employs a parameter called \( \gamma \) ranging between 0 and 1 such that
Chapter 6 Quantisation for scalable multiple description coding

\[(i^*, j^*) = \arg \min_{i,j} \gamma \left[ (\alpha C_A(i) + (1-\alpha)C_B(j)) - x \right]^2 + (1-\gamma) \left[ ((C_A(i) - x)^2 + (C_B(j) - x)^2) \right] \]  

(6.3)

In the training process, the LBG algorithm is modified so that a weighted combination of individual and central distortions is minimised. The weighted distortion combination is given as

\[ D = \sum_x \left\{ \gamma \left[ \alpha C_A(i) + (1-\alpha)C_B(j) - x \right]^2 + (1-\gamma) \left[ ((C_A(i) - x)^2 + (C_B(j) - x)^2) \right] \right\} \]  

(6.4)

where \( \alpha \) and \( \gamma \) are the weighting parameters used in the training. The updating formulas for updating codebooks \( C_A(i) \) and \( C_B(i) \) follow (6.4):

\[ C_A(i) = \gamma \cdot \frac{1}{\alpha} \left[ x - (1-\alpha)C_B(j_i) \right] + (1-\gamma) \cdot E_x \{ x \} \]  

(6.5)

\[ C_B(j) = \gamma \cdot \frac{1}{\alpha} \left[ x - (1-\alpha)C_A(i_j) \right] + (1-\gamma) \cdot E_x \{ x \} \]  

(6.6)

### 6.2.2 Experiments

In [105], extensive experiments have been done to optimize the multiple description coder given different packet loss rates. In this thesis, some experiments in quantization for multiple description coding have been done to obtain the necessary information for the design of the scalable multiple description coder which will be explained in the subsequent subsections.

MSE may not reflect the true trade-off between the side and central distortions accurately with respect to the parameter \( g \) since the changes in central distortion will be much smaller than the side distortions. Therefore, in the experiments, in addition to MSE, bit rates of the single description coder which lead to the same distortion with side and central distortions are also provided as measure of performance. This approach is adopted also in the subsequent subsections.

Firstly, assuming descriptions with equal precision and consequently taking \( \alpha \) as 0.5, the effect of \( \gamma \) on the performance is studied. \( \gamma \) values used in the training and the testing are same in this subsection. A configuration of 4-4 is used. Table 6.1 illustrates the effect of \( \gamma \) on the side and main
decoder performance. In addition to the MSE, the equivalent bit rate of a single description coder which gives the same performance is shown. It is seen that even in the case of $\gamma=1.0$, where both descriptions are trained only to minimize the central distortion, there is some loss (~0.3 bits) in the central performance with respect to a 8 bit single description coder. When $\gamma$ is changed to 0.99 from 1.0, there is a significant increase in the side decoder performance. This is not only evident from the drop in the distortion but also from the increase in the precision (~1.7 bits). In contrast to ~1.7 bits gain in side decoder performance, there is only ~0.6 bits loss in central decoder performance. This finding is in accordance with the principal notion of multiple description coding which suggests that a small amount of redundancy added to a system which is designed to minimize the central distortion is better exploited by dedicating it to the side decoder performance rather than the central decoder performance. As $\gamma$ is decreased towards 0.5, we obtain a more balanced trade-off between side and central decoder performances. However, if we approach from the other end, as $\gamma$ is increased towards 0.5, there is ~0.6 bits gain in the central decoder performance in contrast to only ~0.2 bits loss in the side decoder performance. It becomes evident that a small amount of redundancy added to a system which is designed to minimize the side distortions is better exploited by dedicating it to the central decoder performance rather than the side decoder performance. It can be concluded that unless the packet loss rate is extremely low or high, it is logical to choose an intermediate value of $\gamma$.

Table 6.1 Side and central distortions as a function of $\gamma$

<table>
<thead>
<tr>
<th>$\gamma$</th>
<th>Side distortion</th>
<th>Central distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>2.86 (~0 bits)</td>
<td>0.019 (~7.7 bits)</td>
</tr>
<tr>
<td>0.99</td>
<td>0.92 (~1.7 bits)</td>
<td>0.029 (~7.1 bits)</td>
</tr>
<tr>
<td>0.97</td>
<td>0.53 (~2.4 bits)</td>
<td>0.039 (~6.6 bits)</td>
</tr>
<tr>
<td>0.9</td>
<td>0.32 (~3.3 bits)</td>
<td>0.060 (~5.9 bits)</td>
</tr>
<tr>
<td>0.5</td>
<td>0.23 (~3.8 bits)</td>
<td>0.082 (~5.4 bits)</td>
</tr>
<tr>
<td>0.0</td>
<td>0.21 (~4.8 bits)</td>
<td>0.13 (~4.8 bits)</td>
</tr>
</tbody>
</table>
Chapter 6 Quantisation for scalable multiple description coding

Figure 6.1 shows the performance of varying $\gamma$ at different packet loss rates in logarithmic scale. It is evident that changing $\gamma$ is very effective in terms of optimizing the system performance with respect to the packet loss rate.

![Figure 6.1](image)

Secondly, the effect of using side descriptions with different precision is studied. Figure 6.2 shows the performance of configurations with different allocations of precisions. All configurations have $\gamma$ and $\alpha$ optimised for the loss rate tested. At 0% loss rate, 6-2 configuration marginally outperforms the others, whereas 5-3 again marginally provides the best performance at 1%. At 2% and higher loss rates 4-4 outperforms the others where the gap increases with the loss rate. The results indicate that marginal benefit of using descriptions with unequal precision at very low loss rates is not worth the loss in performance at medium and high loss rates. Consequently, side descriptions with equal precisions will be used in the subsequent sections.
6.2.3 Adaptive multiple description coding

While an encoder is in operation, packet loss rate can change. If the packet loss rate is available to the encoder, it can adapt to the new conditions by changing $\gamma$ value in the quantisation process. To test this approach, an experiment where the performances of adaptive and non-adaptive schemes was performed. To provide a reference, the configuration where the codebooks are optimised according to the tested packet loss rate is included in the comparison as well. Figure 6.3 shows that the adaptive scheme provides close performance to the optimised scheme without leading to possible synchronisation loss and extra codebook storage like the optimised scheme. It can be concluded that if an intermediate value of $\gamma$ is chosen for the training, the system can adapt to different packet loss rates effectively by changing $\gamma$ in the actual coding.
Chapter 6 Quantisation for scalable multiple description coding

6.3 Quantisation for scalable coding

6.3.1 Theory

Traditionally, training MDVQ codebooks are done sequentially or iteratively/simultaneously. In sequential design, each stage (layer) is trained using a database of quantisation error vectors from the previous layer. Although not intentionally, this approach is fully scalable since it tries to maximise the performance of the designed layer without considering the performance of the further enhancement layers. On the other hand, performance of the layer 2 is not optimal since it is not optimised in the design of the intermediate layers.

Alternatively, iterative training can be employed. In this approach, one layer is optimised given all the other layers. This is done such that a different layer is optimised in each iteration. Although this scheme provides a better alternative in terms of the layer 2 distortion, performances of the intermediate layers are compromised.

In this thesis, a different approach where the trade-off between the performances of layer 2 and the intermediate layers is parametrically controlled is used. For the sake of simplicity, only two layers will be considered. However the idea can be easily extended to a higher number of layers.

Assuming full search, a multistage encoder will select the output following the criteria:
In this thesis, (6.7) is proposed to be modified as follows:

\[(i^*, j^*) = \arg_{i,j} \min[(C_1(i) + C_2(j)) - x]^2\] (6.7)

Therefore, the encoder both considers the distortion at layer 1 and the distortion at layer 2 in selecting the set of the codevectors. As \(\beta\) approaches 1, the encoder only considers the layer 2 distortion. As \(\beta\) approaches 0, it principally considers the distortion at layer 1, and at layer 2, it selects the codevector which minimises the layer 2 distortion given the selection in layer 1. Intermediate values of \(\beta\) provide a trade-off between the layer 1 and layer 2 distortions. In the training process, the iterative design algorithm is modified so that a weighted combination of layer 1 and layer 2 distortions is minimised. The weighted distortion combination is given as

\[D = \sum_x [\beta_i [C_1(i) + C_2(x)] - x]^2 + (1 - \beta_i) [C_1(i) - x]^2\] (6.9)

where \(\beta_i\) is the weighting parameter used in the training. Updating formulas for layer 1 and 2 in iterative training are as follows respectively:

\[C_1(i) = \sum_{x \in \mathcal{S}_1} E \{[x - C_2(x)]\} \] (6.10)
\[C_2(i) = \sum_{x \in \mathcal{S}_2} E \{[x - C_1(x)]\} \] (6.11)

We propose to modify (6.11) with the following formula:

\[C_1(i) = \beta_i \sum_{x \in \mathcal{S}_1} E \{[x - C_2(x)]\} + (1 - \beta_i) \sum_{x \in \mathcal{S}_2} E \{x\} \] (6.12)

With this formula it is ensured that the training algorithm takes both layer 1 and layer 2 distortions into account in the updating of layer 1. The updating formula for layer 2 remains unchanged.

### 6.3.2 Experiments

The experiment in this subsection focuses on the effect of parameter \(\beta\) on the performance of layer 1 and layer 2 distortions. It is seen from Table 6.2 that even in the case of \(\beta = 1.0\), where both layers are trained only to minimise the layer 2 distortion, there is some loss (~0.3 bits) in the layer 2 performance.
with respect to a 8 bit single description coder. When $\beta$ is changed to 0.99 from 1.0, there is a significant increase in the layer 2 performance. This is not only evident from the drop in the distortion but also from the increase in the precision (~0.4 bits). In contrast to ~0.4 bits gain in the layer 1 performance, there is only ~0.02 bits loss in the layer 2 performance. However, if we approach from the other end, as $\beta$ is increased towards 0.5, there is ~0.16 bits gain in the layer 2 performance in the expense of only ~0.05 bits loss in layer 1 performance. It can be concluded that unless the quality of layer 1 or layer 2 is overwhelmingly more important than the other, it is logical to choose an intermediate value of $\beta$.

**Table 6.2 The effect of $\beta$ on the layer 1 and 2 performances**

<table>
<thead>
<tr>
<th></th>
<th>$\beta = 1.0$</th>
<th>$\beta = 0.99$</th>
<th>$\beta = 0.97$</th>
<th>$\beta = 0.9$</th>
<th>$\beta = 0.5$</th>
<th>$\beta = 0.001$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Layer 1 distortion</td>
<td>0.48 (~2.7 bits)</td>
<td>0.38 (~3.1 bits)</td>
<td>0.29 (~3.5 bits)</td>
<td>0.24 (~3.8 bits)</td>
<td>0.217 (~3.95 bits)</td>
<td>0.215 (4 bits)</td>
</tr>
<tr>
<td>Layer 2 distortion</td>
<td>0.0197 (~7.7 bits)</td>
<td>0.0199 (~7.68 bits)</td>
<td>0.0215 (~7.53 bits)</td>
<td>0.023 (~7.4 bits)</td>
<td>0.026 (~7.23 bits)</td>
<td>0.028 (~7.07 bits)</td>
</tr>
</tbody>
</table>

**6.3.3 Adaptive scalability**

It is possible that the client characteristics of a network is variable such that the demand for the layer 1 and layer 2 outputs change. In this case, the encoder can optimise its output with respect to the new client profile by varying $\beta$. Table 6.3 shows the results of scheme which is trained at $\beta_0 = 0.97$ and tested at varying $\beta$. It is seen that at $\beta$ values greater than 0.97, layer 1 distortion is slightly lower than the original scheme where same training and testing are applied. This is in the expense of a loss in the precision of the full rate coder. At $\beta$ values less than 0.97, full rate precision is slightly better than the original scheme in the expense of the significant loss in the precision of layer 1 coder. In general, the results indicate that with the proposed adaptation scheme, the encoder would be able to answer the changing demand characteristics although in a limited sense.
Chapter 6 Quantisation for scalable multiple description coding

Table 6.3 The effect of adaptation on the layer 1 and 2 performances

<table>
<thead>
<tr>
<th></th>
<th>$\beta$ = 1.0</th>
<th>$\beta$ = 0.99</th>
<th>$\beta$ = 0.97</th>
<th>$\beta$ = 0.9</th>
<th>$\beta$ = 0.5</th>
<th>$\beta$ = 0.001</th>
</tr>
</thead>
<tbody>
<tr>
<td>Layer 1 distortion</td>
<td>0.44</td>
<td>0.36</td>
<td>0.29</td>
<td>0.265</td>
<td>0.247</td>
<td>0.236</td>
</tr>
<tr>
<td></td>
<td>(~2.7 bits)</td>
<td>(~3.1 bits)</td>
<td>(~3.5 bits)</td>
<td>(~3.61)</td>
<td>(~3.76 bits)</td>
<td>(~3.82 bits)</td>
</tr>
<tr>
<td>Layer 2 distortion</td>
<td>0.0205</td>
<td>0.021</td>
<td>0.0215</td>
<td>0.0226</td>
<td>0.0255</td>
<td>0.0273</td>
</tr>
<tr>
<td></td>
<td>(~7.62 bits)</td>
<td>(~7.57 bits)</td>
<td>(~7.53 bits)</td>
<td>(~7.44 bits)</td>
<td>(~7.27 bits)</td>
<td>(~7.12 bits)</td>
</tr>
</tbody>
</table>

6.4 Scalable multiple description coding

6.4.1 Theory

In this subsection, the problem of designing a scalable multiple-description coder is going to be studied. This scheme is named as Quantisation for Scalable Multiple Description Coding. Figure 6.4 shows the output of the proposed 2-description and 2-layer scalable multiple-description coder. This system provides 2 layers and provides robustness to packet losses for both of the layers by outputting 4 different quantisation indexes. Robustness is provided by the diversity applied to the description A and description B packets. If the decoder is intended to receive only layer 1, layer 2 packets are dropped.

![Figure 6.4 Proposed encoder output structure](image-url)
Chapter 6 Quantisation for scalable multiple description coding

There are 6 different scenarios about how a decoder could possibly use the output sent by the encoder. 

Scenario 1: In this scenario, the decoder expects layer 1 packets and receives both of them successfully (Figure 6.5).

![Figure 6.5 Scenario 1](image1)

Scenario 2: In this scenario, the decoder expects layer 1 packets however only receives description A (Figure 6.6).

![Figure 6.6 Scenario 2](image2)

Scenario 3: In this scenario, the decoder expects layer 1 packets however only receives description B (Figure 6.7).

![Figure 6.7 Scenario 3](image3)

Scenario 4: In this scenario, the decoder expects 2 layers and successfully receives them (Figure 6.8).

![Figure 6.8 Scenario 4](image4)
Scenario 5: In this scenario, the decoder expects full information however only receives description A (Figure 6.9).

![Description A Layer 1](Description A Layer 2)

Figure 6.9 Scenario 5

Scenario 6: In this scenario, the decoder expects full information however only receives description B (Figure 6.10).

![Description B Layer 1](Description B Layer 2)

Figure 6.10 Scenario 6

The scalable multiple description encoder should take all these scenarios into account when choosing the quantisation indexes. This is a complex problem where many constraints need to be satisfied as much as possible at the same time. The proposed quantisation criterion is as follows:

\[
(i^*, j^*, k^*, l^*) = \arg \min_{i,j} \beta \gamma [0.5(C_{1A}(i) + C_{1B}(j) + C_{2A}(k) + C_{2B}(l)) - x]^2 \\
+ \beta(1-\gamma)[(C_{1A}(i) + C_{2A}(k) - x)^2 + (C_{1B}(j) + C_{2B}(l) - x)^2] \\
+(1-\beta)(1-\gamma)[(C_{1A}(i) - x)^2 + (C_{1B}(j) - x)^2] \\
+(1-\beta)(1-\gamma)[(C_{1A}(i) - x)^2 + (C_{1B}(j) - x)^2]
\]

In this equation, first line corresponds to scenario 4, second line corresponds scenario 5 and 6, third line corresponds to scenario 1 and fourth line corresponds to scenario 2 and 3. Parameter \( \gamma \) controls the trade-off between the individual descriptions and the central description (in both layers). Parameter \( \beta \) controls the trade-off between the layer 1 and full precision outputs (for both individual and central descriptions). In the training process, the LBG algorithm is modified so that a weighted combination of distortions due to the listed scenarios is minimised. The weighted distortion metric is given as

\[
D = \sum_x [\beta \gamma [0.5(C_{1A}(i) + C_{1B}(j) + C_{2A}(k) + C_{2B}(l)) - x]^2
\]

(6.14)
where \( \beta_i \) and \( \gamma_i \) are the parameters used in the training. The updating formulas for updating codebooks \( C_{1A}(i) \), \( C_{1B}(j) \), \( C_{2A}(k) \) and \( C_{2B}(l) \) follow (6.14):

\[
C_{1A}(i) = \beta_i \gamma_i \cdot \frac{E}{x \in \mathbb{S}_{1A}} \left\{ [2.5(C_{1A}(i) + C_{2A}(k) + C_{2B}(l))] \right\} \tag{6.15}
\]

\[
+ \beta_i (1 - \gamma_i) \cdot \frac{E}{x \in \mathbb{S}_{2A}} \left\{ [x - C_{2A}(k)]^2 \right\}
\]

\[
+ (1 - \beta_i) \cdot \gamma_i \cdot \frac{E}{x \in \mathbb{S}_{1B}} \left\{ [2.5C_{1B}(j)]^2 \right\}
\]

\[
+ (1 - \gamma_i) (1 - \beta_i) \cdot \frac{E}{x \in \mathbb{S}_{1A}} [x]
\]

where line 1 corresponds to scenario 4, line 2 corresponds to scenario 5, line 3 corresponds to scenario 1 and line 4 corresponds to scenario 2.

\[
C_{1B}(j) = \beta_i \gamma_i \cdot \frac{E}{x \in \mathbb{S}_{1B}} \left\{ [2.5(C_{1A}(i) + C_{2A}(k) + C_{2B}(l))] \right\} \tag{6.16}
\]

\[
+ \beta_i (1 - \gamma_i) \cdot \frac{E}{x \in \mathbb{S}_{2A}} \left\{ [x - C_{2A}(k)]^2 \right\}
\]

\[
+ (1 - \beta_i) \cdot \gamma_i \cdot \frac{E}{x \in \mathbb{S}_{1A}} \left\{ [2.5C_{1B}(j)]^2 \right\}
\]

\[
+ (1 - \gamma_i) (1 - \beta_i) \cdot \frac{E}{x \in \mathbb{S}_{1B}} [x]
\]

where line 1 corresponds to scenario 4, line 2 corresponds to scenario 6, line 3 corresponds to scenario 1 and line 4 corresponds to scenario 3.

\[
C_{2A}(k) = \gamma_i \cdot \frac{E}{x \in \mathbb{S}_{2A}} \left\{ [2.5(C_{1A}(i) + C_{1B}(j) + C_{2B}(l))] \right\} \tag{6.17}
\]

\[
+ (1 - \gamma_i) \cdot \frac{E}{x \in \mathbb{S}_{2A}} \left\{ [x - C_{2B}(l)]^2 \right\}
\]

where line 1 corresponds to scenario 4 and line 2 corresponds to scenario 5.

\[
C_{2B}(l) = \gamma_i \cdot \frac{E}{x \in \mathbb{S}_{2B}} \left\{ [2.5(C_{1A}(i) + C_{1B}(j) + C_{2A}(k))] \right\} \tag{6.18}
\]

\[
+ (1 - \gamma_i) \cdot \frac{E}{x \in \mathbb{S}_{2B}} \left\{ [x - C_{2B}(l)]^2 \right\}
\]
where line 1 corresponds to scenario 4 and line 2 corresponds to scenario 6.

### 6.3.4 Experiments

All codebooks are represented using 3 bits. Experiments in this subsection are carried out by varying $\beta$ and $\gamma$. Firstly, the special case of $\beta = \gamma = 1$ is going to be examined. In this case, all 4 codebooks $(C_{1A}(i), C_{1B}(j), C_{2A}(k), C_{2B}(l))$ are optimised exclusively to minimise the central description in layer 2. The resulting side and central distortions for both layers are given in Table 6.4. It is seen that although all codebook are jointly trained only to minimize the layer 2 central distortion, there is some loss (~0.7 bits) in the central performance with respect to a 12 bit direct quantiser.

<table>
<thead>
<tr>
<th>$\beta = \gamma = 1$</th>
<th>Side distortion</th>
<th>Central distortion</th>
<th>Side distortion</th>
<th>Central distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Layer 1</td>
<td>Layer 1</td>
<td>Layer 2</td>
<td>Layer 2</td>
</tr>
<tr>
<td>$\beta = \gamma = 1$</td>
<td>2.27</td>
<td>0.59</td>
<td>2.27</td>
<td>0.0021</td>
</tr>
<tr>
<td>(0 bits)</td>
<td>(~2.4 bits)</td>
<td>(0 bits)</td>
<td>(~11.4 bits)</td>
<td></td>
</tr>
</tbody>
</table>

Next, keeping $\beta=1$, several $\gamma$ values are tested. The results for $\gamma=0.996$ is given in Table 6.5. There is a significant increase in the precision of the side description of layer 2 (~2.7 bits), in the expense of only (~1.0 bits) in the central description of layer 2.

<table>
<thead>
<tr>
<th>$\beta = 1$, $\gamma = 0.996$</th>
<th>Side distortion</th>
<th>Central distortion</th>
<th>Side distortion</th>
<th>Central distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Layer 1</td>
<td>Layer 1</td>
<td>Layer 2</td>
<td>Layer 2</td>
</tr>
<tr>
<td>$\beta = 1$, $\gamma = 0.996$</td>
<td>1.25</td>
<td>0.56</td>
<td>0.49</td>
<td>0.0041</td>
</tr>
<tr>
<td>(~1.2 bits)</td>
<td>(~2.5 bits)</td>
<td>(~2.7 bits)</td>
<td>(~10.4 bits)</td>
<td></td>
</tr>
</tbody>
</table>

The results for $\gamma=0.99$ is given in Table 6.6. The gain in the precision of the layer 1 side description proceeds (~0.7 bits); however at the expense of more balanced trade-off (~0.7 bits drop in the precision of layer 2 central distortion). It seems that when compared to quantisation for multiple...
description coding, in quantisation of scalable multiple description coding it is more penalising to decrease $\gamma$ in terms of the central performance. This can be attributed to the extra complexity in quantisation for scalable multiple description coding due to the 4 codebooks used.

**Table 6.6 Performance, $\beta = 1$, $\gamma = 0.99$**

<table>
<thead>
<tr>
<th></th>
<th>Side distortion</th>
<th>Central distortion</th>
<th>Side distortion</th>
<th>Central distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Layer 1</td>
<td>Layer 1</td>
<td>Layer 2</td>
<td>Layer 2</td>
</tr>
<tr>
<td>$\beta = 1$</td>
<td>1.18 (1.3 bits)</td>
<td>0.57 (~2.5 bits)</td>
<td>0.32 (~3.4 bits)</td>
<td>0.0059 (~9.7 bits)</td>
</tr>
</tbody>
</table>

Next, keeping $\gamma = 1$, several values of $\beta$ are tested. Since $\gamma = 1$, side decoder performances are very bad at both layers. As $\beta$ is decreased to 0.99 (Table 6.7), there is a significant increase in the layer 1 central performance (~2 bits) with a relatively smaller drop in the layer 2 central performance (~0.5 bits).

**Table 6.7 Performance, $\beta = 0.99$, $\gamma = 1$**

<table>
<thead>
<tr>
<th></th>
<th>Side distortion</th>
<th>Central distortion</th>
<th>Side distortion</th>
<th>Central distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Layer 1</td>
<td>Layer 1</td>
<td>Layer 2</td>
<td>Layer 2</td>
</tr>
<tr>
<td>$\beta = 0.99$</td>
<td>1.89 (&lt;1 bits)</td>
<td>0.19 (~4.2 bits)</td>
<td>2.01 (&lt;0 bits)</td>
<td>0.0031 (~10.9 bits)</td>
</tr>
</tbody>
</table>

As $\beta$ is further decreased to 0.9 (Table 6.8), there is again significant increase in the layer 1 central performance (~1.0 bits) with a relatively smaller drop in the layer 2 performance (~0.3 bits)
Chapter 6 Quantisation for scalable multiple description coding

Table 6.8 Performance, $\beta = 0.9$, $\gamma = 1$

<table>
<thead>
<tr>
<th></th>
<th>Side distortion Layer 1</th>
<th>Central distortion Layer 1</th>
<th>Side distortion Layer 2</th>
<th>Central distortion Layer 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta = 0.9$</td>
<td>1.8</td>
<td>0.088</td>
<td>2.2</td>
<td>0.0035</td>
</tr>
<tr>
<td>$\gamma = 1$</td>
<td>($&lt; 1$ bits)</td>
<td>($\sim 5.2$ bits)</td>
<td>($&lt; 0$ bits)</td>
<td>($\sim 10.6$ bits)</td>
</tr>
</tbody>
</table>

Taking the experiments in this subsection and the previous two subsections into account, two operating points will be investigated. Firstly, operating point of $\beta = 0.99$, $\gamma = 0.996$ is shown in Table 6.9. In the expense of $\sim 0.8$ bits loss in the layer 2 central performance, this configuration provides $\sim 1.3$ bits improvement in layer 1 central performance, $\sim 1.5$ bit improvement in layer 1 side performance and $\sim 2.5$ bits in layer 2 side performance. Assuming that $\beta = 0.99$, equalising $\gamma$ to 0.996 to gives best performance among all $\gamma$ values for layer 1, at packet loss rate $\%1$ and for layer 2, at packet loss rate $\%0.5$.

Table 6.9 Performance $\beta = 0.99$, $\gamma = 0.996$

<table>
<thead>
<tr>
<th></th>
<th>Side distortion Layer 1</th>
<th>Central distortion Layer 1</th>
<th>Side distortion Layer 2</th>
<th>Central distortion Layer 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta = 0.996$</td>
<td>1.01</td>
<td>0.25</td>
<td>0.56</td>
<td>0.0046</td>
</tr>
<tr>
<td>$\gamma = 0.99$</td>
<td>($\sim 1.5$ bits)</td>
<td>($\sim 3.7$ bits)</td>
<td>($\sim 2.5$ bits)</td>
<td>($\sim 10.1$ bits)</td>
</tr>
</tbody>
</table>

In Table 6.10 operating point of $\beta = 0.9$ and $\gamma = 0.99$ is shown. When compared to the previous configuration, there is a further increase of $\sim 0.6$ bits in the performance of layer 1 side performance, $\sim 1.2$ increase in the layer 1 central performance and $\sim 0.6$ bits in the layer side performance in the expense of $\sim 0.8$. It is seen that as $\beta$ and $\gamma$ are decreased, the loss in the layer 2 central performance starts not to be compensated by the increase in layer 1 and layer 2 side precisions. Assuming $\beta = 0.9$, equalising $\gamma$ to 0.99 gives the best performance among all $\gamma$ values for layer 1, at a packet loss rate of $\%3$ and for layer 2, at a packet loss rate of $\%3$. 

104
### Chapter 6 Quantisation for scalable multiple description coding

#### 6.3.5 Adaptive scalable multiple description coding

It is possible that the client characteristics of a network are variable such that the demand for the layer 1 and layer 2 outputs change. It is also possible that the packet loss change when the encoder is in operation. The encoder can adapt to these changes by varying $\beta$ and $\gamma$. Table 6.11 shows the results of a scheme which is trained at $\beta = 0.99$, $\gamma = 0.996$ and tested at $\beta = 0.9$, $\gamma = 0.99$. Compared to the configuration which is trained and tested at $\beta = 0.996$, $\gamma = 0.99$, this system provides increase in the layer 1 side precision (−0.2 bits), layer 1 central precision (−0.4 bits), layer 2 side precision (−0.2 bits) at the expense of (−0.6 bits) loss in the precision of layer 2 central distortion. The results indicate adaptation works although in the expense of the significant loss in the layer 2 central distortion.

#### Table 6.11 Performance of adaptive scheme

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>Side distortion</th>
<th>Central distortion</th>
<th>Side distortion</th>
<th>Central distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma$</td>
<td>Layer 1</td>
<td>Layer 1</td>
<td>Layer 2</td>
<td>Layer 2</td>
</tr>
<tr>
<td>0.9</td>
<td>0.75 (−2.1 bits)</td>
<td>0.12 (−4.9 bits)</td>
<td>0.37 (−3.1 bits)</td>
<td>0.0073 (−9.3 bits)</td>
</tr>
<tr>
<td>0.996</td>
<td>0.91 (−1.7 bits)</td>
<td>0.19 (−4.1 bits)</td>
<td>0.48 (−2.7 bits)</td>
<td>0.0065 (−9.5 bits)</td>
</tr>
</tbody>
</table>

#### 6.4 Conclusion

In this chapter, firstly, a previously proposed multiple description coding scheme called quantisation for multiple description coding has been studied. In this scheme the trade-off between the central and side descriptions is controlled by a parameter called $\gamma$. It was found that a small amount of redundancy added to the system which is designed to minimize the central distortion is better...
exploited by dedicating it to the side decoder performance rather than the central decoder performance. Adaptation to different packet loss rates than the one assumed in the training has also been studied. As a future work, the performance of quantisation for multiple description coding can be compared with established multiple description techniques such as MDTC, MDSQ or the theoretical bound.

Secondly, a parametric scheme, quantisation for scalable coding has been introduced. In this scheme, the trade-off between first and second layers is controlled by a parameter called $\beta$. The training of the system is done by the modified LBG algorithm. It has been shown that assuming a 2 layer codebook which is only optimised to minimise the layer 2 distortion, the performance of layer 1 can be significantly improved with only a small drop in the layer performance. An adaptive scalability scheme has also been proposed.

Lastly, the main contribution of the chapter, quantisation for scalable multiple description coding has been proposed. This scheme provides multiple description coding for the available two layers where parameter $\beta$ controls the trade-off between first and second layers and parameter $\gamma$ controls the trade-off between side and central distortions. The foreseen scenarios have been illustrated and their impacts on the training constraints have been shown. Modified LBG algorithm has been used to train the system. The results indicate that the proposed system is capable of provided multiple description and scalability at the same time. Online adaptation has also been studied. The system can become more flexible by defining separate variables for all the constraints considered such as defining a variable for layer 1 side distortion. In this way, the correlation between descriptions in different layers can be controlled separately.
Chapter 7

7 Conclusions

7.1 Preamble

Apart from storage systems, most applications of speech coders imply the transmission of compressed voice over a communication channel. With the emergence of ultra-high speed fixed and wireless networks, streaming audio has become one of the most important applications of the audio coders as well. Depending on whether the system is packet based, on the type of channel coding used, and the characteristics of the transmission channel, the channel errors can manifest themselves in two ways from the source coder point of view. Firstly, individual bits may be corrupted. This is the usual case in wireless environments, and especially when no channel coding is used. Secondly, complete frames may be lost. This can for example be due to packet losses in packet based networks, use of convolutional channel coders, bit errors in the header information, or because of the loss of synchronisation.

When the communication channel is prone to bit errors, which is typically the case for a mobile channel, error concealment techniques are not enough to maintain decent audio quality. The bitstream
Chapter 7. Conclusions

needs to be more protected, and this is achieved through channel coding. The channel coding schemes used for such applications vary a lot depending on the channel and the speech coding scheme used. Generally a combination of error correction coding and CRC for error detection is used.

On the other side, multimedia streaming sets contradictory requirements for transport delays and error robustness. Packet losses should be compensated efficiently to ensure high quality of service. Retransmission is not a viable solution because of the delay constraints and possible feedback implosion. Therefore, schemes adding redundancy to the source payload is a necessity for packet networks as well.

Traditional channel coding approaches aim to completely counteract all channel impairments. However, for communication subject to a smooth distortion measure, a much less strict requirement can be imposed by employing JSCC techniques. In this research, either new or extensions to established JSCC techniques have been developed to cope with both bit errors and packet losses. The philosophy of replacing a traditional source code and channel code with an JSCC is to mitigate the effect of an erasure even if it can not be corrected. The thesis was split into 3 main sections in this regard as:

1. The first aim was to improve the error resilience of speech and audio codecs against bit errors commonly seen in wireless channels. IACC, a JSCC scheme, which does not interfere with the source codec design was introduced. With this scheme, the amount of added redundancy is determined according to the sensitivity of the different values of the source parameters. It was shown that the source characteristics play an important role in the performance of IACC. A combination of IACC and convolutional coding was also presented. The parameters generated by AMR-WB+ audio codec were used to evaluate the performance of IACC based schemes. Subjective tests showed that IACC based schemes outperform conventional convolutional coding in medium to high BERs.

2. The second aim was to improve the performance of multiple description techniques for protecting audio in networks with packet losses. AAC was chosen as the underlying audio codec. Firstly, the performance of MDTC applied in application to spectral coefficients was improved by two extensions. Secondly, the adaptation of unconstrained MDVQ to MDCT coefficients and a method for improving its performance was presented. Thirdly, a joint coding scheme which reduces the side information overload was introduced. Lastly,
the performance of MDVQ, MDTC and single description coding were compared in networks with various packet loss rates giving useful operating points for all these schemes.

3. The last aim was to develop a scalable multiple description scheme. The proposed system provides multiple description for the hierarchical two layers. By modifying the quantisation criteria, the trade-off between the first and second layers and the trade-off between the central and side distortions can be controlled parametrically.

7.2 Concluding Overview

Chapter 2 provided a brief overview of speech and audio coding as well as some fundamental principles. The main criteria for the design of speech and coding algorithms, such as bit-rate, quality, delay and error robustness are discussed. The three main speech coding paradigms and important audio coding techniques are also presented along with brief discussions. After giving an overview of quantisation, the basic building blocks of a perceptual audio coder were introduced. For speech coding, the popular source-filter model was presented.

In chapter 3, the ways for error resilient audio and speech communications were reviewed. The problems in the communication systems can be roughly divided into two groups: Bit errors and packet losses. For bit errors and packet losses, error detection followed by an appropriate error concealment technique can be a perceptually effective technique as long as the loss rate is modest.

For bad channel and network conditions, adding a significant amount of redundancy seems a necessity. Relative advantages of separate source and channel coding and JSCCs were presented. It was emphasized that MDC, which is a type of JSCC specialised for packet networks, can provide performance that decreases gracefully with the increasing loss rate.

In chapter 4, a source optimized channel coding scheme called IACC was presented and its application to AMR-WB+ was evaluated. The main advantage of this scheme over other JSCC techniques is its compatibility with a given source codec. Although pure IACC is only promising at high BERs, the proposed IACC-CC schemes is shown to yield better performance than the benchmark convolutional coder at medium to high BERs. Subjective tests comparing the performance of IACC based schemes
and established convolutional coding proved the superiority of IACC based schemes at high BERs. Chapter 5 presented various new techniques for improving the performance of multiple description techniques in protecting audio in networks with packet losses. Firstly, two methods for improving the performance of MDTC in application to spectral coefficients were proposed. Both of these schemes increased the performance of MDTC in AAC context. Secondly, it was shown that adaptation of unconstrained MDVQ to MDCT is useful at high loss rates. Thirdly, a joint coding scheme for side information which improves the performance of MDC schemes was proposed. Lastly, the performance of MDVQ, MDTC and single description coding were compared in networks with various packet loss rates providing useful operating ranges with respect to the loss rate and how the redundancy is exploited. Overall, the results showed that with the deployment of refined optimisation and side information compression schemes multiple description schemes offer strong error robustness for AAC.

Chapter 6 reviewed a scalable multiple description system via manipulation of conventional single stage quantisation. Secondly, it was shown that quantisation based scalable coding provides a trade-off between first and second layers which can be controlled parametrically. Finally, the design of a quantisation scheme for scalable multiple description coding was given. This scheme provides multiple description coding for the available two layers the trade-off between first and second layers and the trade-off between side and central distortions can be controlled parametrically.

7.3 Future work

- The performance of IACC against convolutional coding should be measured in channels with correlated bit errors. Initial results and the adaptation capability of IACC indicate that the advantage of IACC is greater in channels with burst errors.
- MDC testing in this research was based on PEAQ algorithm. However, measuring the quality of speech and audio codecs in lossy channel is an area which has room for improvement.
- Re-optimisation AAC Huffman codebooks for MDCT can improve the performance.
- The performance of quantisation for multiple description coding can be compared with established multiple description techniques such as MDTC, MDSQ or the theoretical bound.
- Prominent multiple description schemes, dithered delta-sigma quantization and GMDTC which have been shown to approach theoretical limits can be applied to audio coding.
Chapter 7. Conclusions

- The scalable multiple description system can become more flexible by defining separate variables for all the constraints considered such as defining a variable for layer 1 side distortion. In this way, the correlation between descriptions in different layers can be controlled separately.

- The scalable multiple description can be used to represent speech and audio codec parameters, most evidently the LSF coefficients. Since the conventional schemes quantize LSF parameters using multi-stages, the relative central performance loss in the second layer will be confined to the desired parametrical adjustments done for increasing the side decoder and first layer precisions.
List of Publications & Presentations


Bibliography


Bibliography


[34] K. Rosen, Introduction to Information Theory and Data Compression, CRC Press, 2003


[79] K. Kawahara, K. Kumazoe, T. Takine, and Y. Oie, "Forward error correction in ATM networks:

Magazine, IEEE , vol.18, no.5, pp.74-93, Sep 2001
[82] N. S. Jayant and S. W. Christensen, "Effects of packet losses in waveform coded speech and
[83] A. Ingle and V.A. Vaishampayan, "DPCM system design for diversity systems with applications
[84] W. Jiang and A. Ortega, "Multiple description speech coding for robust communication over
lossy packet networks," in Proc. of Int. Conf on Multimedia and Expo (ICME'00), pp. 444-447, 2000
[85] G. Kubin and W.B. Kleijn, "Multiple-description coding (MDC) of speech with an invertible
[87] V. A. Vaishampayan, N. Sloane, and S. Servetto, "Multiple description vector quantization
[88] N. G'ortz and P. Leelapornchai, "Optimization of the index assignment for multiple description


