

Contributions to Improved Hard- and Soft-Decision Decoding in Speech and Audio Codecs

Von der Fakultät für Elektrotechnik, Informationstechnik, Physik
der Technischen Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung des Grades einer Doktorin

der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation

von Sai Han

aus Shandong, China

eingereicht am: 29.04.2016

mündliche Prüfung am: 25.08.2016

1. Referent: Prof. Dr.-Ing. Tim Fingscheidt
Technische Universität Carolo-Wilhelmina zu Braunschweig
 2. Referent: Prof. Dr.-Ing. Tom Bäckström
Friedrich-Alexander-Universität Erlangen-Nürnberg
- Vorsitzender: Prof. Dr.-Ing. Thomas Kürner
Technische Universität Carolo-Wilhelmina zu Braunschweig

Mitteilungen aus dem Institut für Nachrichtentechnik der
Technischen Universität Braunschweig

Band 47

Sai Han

**Contributions to Improved Hard- and Soft-Decision
Decoding in Speech and Audio Codecs**

Shaker Verlag
Aachen 2016

Bibliographic information published by the Deutsche Nationalbibliothek

The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at <http://dnb.d-nb.de>.

Zugl.: Braunschweig, Techn. Univ., Diss., 2016

Editor of this volume:

Prof. Dr.-Ing. Tim Fingscheidt
Institute for Communications Technology
Technische Universität Braunschweig
Schleinitzstrasse 22
38106 Braunschweig
Germany
e-mail: fingscheidt@ifn.ing.tu-bs.de
phone: +49-531-391-2485
fax: +49-531-391-8218

Copyright Shaker Verlag 2016

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the publishers.

Printed in Germany.

ISBN 978-3-8440-4927-5

ISSN 1865-2484

Shaker Verlag GmbH • P.O. BOX 101818 • D-52018 Aachen

Phone: 0049/2407/9596-0 • Telefax: 0049/2407/9596-9

Internet: www.shaker.de • e-mail: info@shaker.de

Acknowledgments

This thesis was written during my doctorate studies at the Institute for Communications Technology (Institut für Nachrichtentechnik, IfN) of Technische Universität Braunschweig. It would not have been possible to finish it without the support of several people. It is my great pleasure to thank all the people who supported me for completing this work.

Firstly, I would like to express my sincere gratitude to my supervisor Prof. Dr.-Ing. Tim Fingscheidt for his continuous support and supervision. By our fruitful and deep discussions, his helpful suggestions, valuable feedback, and encouragement contributed a significant part to the success of this thesis.

Next, I am grateful to Prof. DSc. Tom Bäckström for being the co-examiner of this thesis as well as for his interest in this research work. I would also like to express my appreciation to Prof. Dr.-Ing. Thomas Kürner for being the chair of the examination board.

Moreover, I would like to thank my colleagues from IfN for their helpfulness and for providing a very friendly work atmosphere. Particularly, I am thankful to Samy Elshamy M.Sc., Rudolf Görke, Patrick Meyer M.Sc., Dr.-Ing Florian Pflug, Dipl.-Ing. Simon Receveur, Peter Transfeld M.Sc., Dr.-Ing Huajun Yu, Ziyue Zhao M.Eng. for proofreading chapters of this thesis. In addition, I would like to thank Dr.-Ing Patrick Bauer, Dr.-Ing. Kin Lien Chee, Dr.-Ing. Balázs Fodor, Marc-André Jung M.Sc., Timo Lohrenz M.Sc., Dr.-Ing Volker Märgner, and Dr.-Ing. Florian Pflug for their valuable discussions and suggestions during my time at IfN. I am thankful to my student Chao Tian for his contributions to parts of this work as well.

Finally, I would like to give my special thanks to my parents Jingfan Liu and Jinfu Han, my sister Ran Han, and my husband Shaodong Qin for their endless and unconditional love, support, patience, and understanding for the entire phase of my doctorate studies in Germany.

Braunschweig, September 2016

Sai Han

Abstract

Source coding is an essential part in digital communications. In error-prone transmission conditions, even with the help of channel coding, which normally introduces delay, bit errors may still occur. Single bit errors can result in significant distortions. Therefore, a robust source decoder is desired for adverse transmission conditions. Compared to the traditional hard-decision (HD) decoding and error concealment, soft-decision (SD) decoding offers a higher robustness by exploiting the source residual redundancy and utilizing the bit-wise channel reliability information. Moreover, the quantization codebook index can be either mapped to a fixed number of bits using fixed-length (FL) codes, or a variable number of bits employing variable-length (VL) codes. The codebook entry can be either fixed over time or time-variant. However, using a fixed scalar quantization codebook leads to the same performance for correlated and uncorrelated processes. This thesis aims to improve the performance of speech and audio codecs with FL and VL codes.

The thesis can be divided into three main parts: Firstly, the concept of FL/SD decoding is applied to the Adaptive Multi-Rate Narrowband (AMR-NB) and AMR Wideband (AMR-WB) speech codecs, which are widely used in mobile speech communications. In addition, new approaches exploiting both interframe and intraframe redundancy for the spectral envelope parameters in both codecs are proposed. The speech quality is significantly improved, both for AMR-NB and AMR-WB. Secondly, the links between the FL/SD and VL/SD decoding algorithms are derived. The tradeoffs of the two SD decoding approaches are discussed. Both the FL/SD and VL/SD decoding methods are applied to High-Efficiency Advanced Audio Coding (HE-AAC), which is optimized for low bit rate applications, such as mobile music streaming and digital radio broadcasting. Supported by subjective listening tests, the audio quality shows a tremendous enhancement. Finally, a new decoding approach is proposed to improve the scalar quantization performance for correlated processes. Exploiting the source correlation and employing only a predictor at the receiver side, a time-variant quantization codebook can be generated. This proposed approach is advantageously applicable in both error-free and error-prone transmission conditions, both with HD and SD decoding. It has also been applied to the G.726 and G.722 Adaptive Differential Pulse Code Modulation (ADPCM) speech codecs, which are utilized in cordless and IP telephony. An improved speech quality is observed.

Kurzfassung

Quellencodierung ist ein wesentlicher Bestandteil digitaler Kommunikationssysteme. Bei fehlerhaften Übertragungsbedingungen können auch mit Hilfe der Kanalcodierung, die normalerweise eine Verzögerung bewirkt, noch Bitfehler auftreten. Einzel-Bitfehler können zu signifikanten Signalverzerrungen führen. Daher ist ein robuster Quellendecoder für schlechte Übertragungsbedingungen erwünscht. Gegenüber der herkömmlichen Hard-Decision (HD)-Decodierung und Fehlerverdeckung bietet die Soft-Decision (SD)-Decodierung eine höhere Robustheit durch die Nutzung der Restredundanz der Quelle und die Verwendung der bitweisen Kanaluverlässigkeitsinformation. Außerdem kann der Quantisierungs-Codebuchindex entweder einer festen Anzahl von Bits unter Verwendung einer festen Codewortlänge (FL), oder einer variablen Anzahl von Bits mit variabler Codewortlänge (VL) zugeordnet werden. Der Codebucheintrag selbst kann entweder konstant über der Zeit oder zeitvariant sein. Jedoch führt ein Codebuch unter Verwendung einer festen skalaren Quantisierung zu gleiches Performanz für korrelierte wie für unkorrelierte Prozesse. Diese Arbeit zielt darauf ab, die Leistung von Sprach- und Audio-Codecs mit FL- und VL-Codes zu verbessern.

Die Arbeit lässt sich in drei Hauptteile unterteilen: Zunächst wird das Konzept der FL/SD-Decodierung auf den Adaptive Multi-Rate Narrowband (AMR-NB) und AMR Wideband (AMR-WB) Sprach-Codec angewandt, die in der mobilen Sprachkommunikation weit verbreitet sind. Zusätzlich werden für die spektrale Einhüllende neue Ansätze unter Nutzung von Inter- und Intra-Rahmen-Redundanz vorgestellt. Die Sprachqualität wird sowohl für AMR-NB als auch für AMR-WB deutlich verbessert. Als Zweites werden mathematische Bezüge zwischen FL/SD- und VL/SD-Decodierungsalgorithmen abgeleitet. Dabei wird der Trade-off der beiden SD-Decodierungs-Ansätze diskutiert. Sowohl das FL/SD-, als auch das VL/SD-Decodierungsverfahren werden im Zuge des High-Efficiency Advanced Audio Coding (HE-AAC) eingesetzt, welches für Anwendungen mit niedrigen Bitraten, wie dem mobilen Musik-Streaming oder dem Digitalen Rundfunk, optimiert ist. Im Rahmen subjektiver Hörtests zeigt die Audioqualität eine enorme Verbesserung. Schließlich wird ein neuer Dekodierungs-Ansatz zur Verbesserung der skalaren Quantisierungsleistung von korrelierten Prozessen vorgestellt. Durch Ausnutzung der Quellen-Korrelation und unter Verwendung eines Prädiktors im Empfänger kann ein zeitvariables Codebuch erzeugt werden. Dieser vorgeschlagene Ansatz kann vorteilhaft sowohl bei fehlerfreien, als auch bei fehlerbehafteten Übertragungsbedingungen eingesetzt werden, und zwar sowohl im Zuge der HD- wie auch der SD-Decodierung. Er wird auch auf die beiden Adaptive-Differential-Pulse-Code-Modulation (ADPCM)-Sprachcodecs G.726 und G.722 angewandt, welche in der Schnurlos- und der IP-Telefonie standardisiert sind. Dabei wird eine verbesserte Sprachqualität erzielt.

Contents

Acknowledgments	i
Abstract	iii
Kurzfassung	v
1 Introduction	1
1.1 Concept of Robust Source Decoding	1
1.2 Brief Overview of Some Important Speech and Audio Codecs	2
1.3 Outline of the Thesis	3
2 Aspects of Source Parameter Encoding and Decoding	5
2.1 Source Parameter Models	5
2.1.1 Modeling Unquantized Parameters	5
2.1.2 Quantization of Parameters	6
2.1.3 Modeling Quantized Parameters	7
2.2 Channel Models	9
2.2.1 Additive White Gaussian Noise (AWGN) Channel With Binary Phase-Shift Keying (BPSK)	10
AWGN Channel	10
BPSK Modulation	10
2.2.2 Gilbert-Elliott Channel	11
2.3 Fixed-Length (FL) Soft-Decision (SD) Decoding of Parameters	12
2.3.1 Hard-Decision (HD) Decoding	13
2.3.2 SD: <i>A Posteriori</i> Probabilities (APPs)	13
Channel Transition Probabilities	15
Predictive <i>A Priori</i> Probabilities	15
<i>A Posteriori</i> Probabilities (APPs)	16
Forward-Backward Algorithm (BCJR Algorithm)	16
2.3.3 SD: Parameter Estimation	17
2.4 Summary	17
3 Improving AMR Narrowband and Wideband by Fixed-Length Soft- Decision Decoding	19
3.1 AMR Speech Coding	19
3.1.1 AMR Narrowband	20

	AMR-NB Encoding	20
	AMR-NB Decoding	20
	Line Spectral Frequencies (LSFs)	21
	Adaptive Codebook Index (Pitch Delay)	21
	Adaptive Codebook and Fixed Codebook Gains	22
	Fixed Codebook Index	22
	Bit Allocation Table	22
3.1.2	AMR Wideband	22
	AMR-WB Encoding	23
	AMR-WB Decoding	23
	Immittance Spectral Frequencies (ISFs)	23
	Adaptive Codebook Index (Pitch Delay)	24
	Vector-Quantized Codebook Gain (VQ Gain)	24
	Fixed Codebook Index	24
	Bit Allocation Table	24
3.2	Standard Error Concealment in AMR	25
3.2.1	AMR Narrowband	25
3.2.2	AMR Wideband	25
3.3	New AMR Parameter Soft-Decision Decoding	25
3.3.1	AMR Narrowband	26
	Line Spectral Frequencies (LSFs)	27
	Adaptive Codebook Index (Pitch Delay)	29
	Adaptive Codebook and Fixed Codebook Gains	30
	Complexity Evaluation	30
3.3.2	AMR Wideband	31
	Immittance Spectral Frequencies (ISFs)	32
	Adaptive Codebook Index (Pitch Delay)	33
	Vector-Quantized Codebook Gain (VQ Gain)	33
	Other Parameters	34
	Complexity Evaluation	34
3.4	Simulation Setup and Results	34
3.4.1	Simulation Setup	35
3.4.2	Simulation Results	35
	AMR Narrowband	36
	AMR Wideband	40
3.5	Summary	43
4	Variable-Length Soft-Decision Decoding	45
4.1	Introduction	45
4.2	Variable-Length (VL) Hard-Decision Decoding	48
4.3	Soft-Decision (SD) Decoding: Trellis Representation	49
	Definitions of Stage Boundaries and State Intervals	50
4.4	SD: <i>A Posteriori</i> Probabilities and Source Symbol Estimation	53
4.4.1	<i>A Posteriori</i> Probabilities (APPs)	53
	Forward Recursion	53

Backward Recursion	55
4.4.2 Source Symbol Estimation	56
4.5 From VL/SD Decoding to FL/SD Decoding	56
APPs With AK0	57
APPs With AK1	57
4.6 Simulation Setup and Results	58
4.6.1 Simulation Setup	58
4.6.2 Simulation Results	58
4.7 Summary	62
5 Improving HE-AAC by Soft-Decision Decoding	65
5.1 Introduction	65
5.2 HE-AAC Audio Coding	67
5.2.1 AAC Encoding	67
Filterbank	67
Psychoacoustic Model and Reduction of Psychoacoustic Requirements	68
Determination of Scale Factors and Quantization of Spectral Coef-	
ficients	68
Noiseless Coding	73
Out of Bits Prevention	76
5.2.2 AAC Decoding	76
5.2.3 HE-AAC Bit Stream Structure	77
5.3 Standard Error Concealment in HE-AAC	79
5.4 New HE-AAC Parameter Soft-Decision (SD) Decoding	79
5.4.1 FL/SD Decoding of the Global Gain	80
5.4.2 VL/SD Decoding of the Scale Factors	80
5.4.3 VL/SD Decoding of the Quantized Spectral Coefficients	81
New Trellis Representation	81
<i>A Posteriori</i> Probabilities and Source Symbol Estimation	84
5.5 Simulation Setup and Results	86
5.5.1 Simulation Setup	86
Instrumental Measurement and Subjective Listening Test	86
5.5.2 Simulation Results	89
Single Parameter Distorted	89
Two Parameters Distorted	92
Three Parameters Distorted	94
Subjective Listening Test Results	96
5.6 Summary	97
6 Low-Rate Fixed-Length Hard- and Soft-Decision Decoding With Time-	99
Variant Codebooks	99
6.1 Introduction	100
6.2 New Predictive Hard-Decision Decoding	102
6.3 Analysis and Optimization of the Prediction Error PDF	104
6.4 New Predictive Soft-Decision Decoding	106

6.5	Simulation Setup and Results	107
6.5.1	Simulation Setup	107
6.5.2	Simulation Results	107
6.6	Summary	112
7	Improving G.726 and G.722 ADPCM by Hard-Decision Decoding With Time-Variant Codebooks	113
7.1	G.726 and G.722 ADPCM Speech Coding	113
7.1.1	G.726	114
	G.726 ADPCM Encoding	114
	G.726 ADPCM Decoding	115
7.1.2	G.722	115
	G.722 ADPCM Encoding	115
	G.722 ADPCM Decoding	116
7.2	New Hard-Decision Decoding With Time-Variant Codebooks	118
7.2.1	Error-Free Transmission Conditions	119
	An Outline	119
	Application to ADPCM	119
	Normalized Least-Mean-Squares (NLMS) Algorithm	121
7.2.2	Hard-Decision Decoding	121
7.3	Simulation Setup and Results	121
7.3.1	Simulation Setup	122
7.3.2	Simulation Results	123
	G.726	123
	G.722	128
7.4	Summary	130
8	Conclusions	131
A	APP Calculation of Exploiting Both Interframe and Intraframe Redundancy in AMR-NB	133
B	Time-Variant Quantization Codebook	137
B.1	New Reconstruction Levels	137
B.2	Receiver-Sided Prediction Error Variance	138
C	Audio Database	141
D	PEAQ ODG Results	143
E	Simulation Results in Tabular Forms	145
	List of Symbols	163
	List of Abbreviations	169
	Bibliography	171

Chapter 1

Introduction

1.1 Concept of Robust Source Decoding

In digital communications, source coding transforms original source signals into a stream of bits for the purposes of transmission or storage in a channel [Proakis, 2001, Gersho and Gray, 1992]. At the receiver side, the bit stream is decoded and further reconstructed to recover the original source information. In unreliable transmission conditions (i.e., noisy channels), the bit stream may be distorted and the original source information may not be correctly recovered at the receiver side. The bit stream can be protected against transmission distortions using channel coding. This introduces some redundancy in the bit stream, which can be utilized at the receiver side to correct transmission errors [Bossert, 1999]. However, bit errors may still occur and result in degradation of the speech or audio quality at the receiver side. In addition, usually some delay is introduced. Therefore, a robust source decoder is necessary for adverse transmission conditions.

Residual redundancy is observed within the source signals (or codec parameters). Shannon mentioned that the redundancy of the source can be utilized for robust decoding at the receiver side [Shannon, 1948]:

“However, any redundancy in the source will usually help if it is utilized at the receiving point. In particular, if the source already has a certain redundancy and no attempt is made to eliminate it in matching the channel, this redundancy will help combat noise.”

In the receiver, error concealment techniques are often used to conceal the erroneous effect from the channel [Fingscheidt and Vary, 2001]. Typically a bad frame indicator marking a frame as good or bad is employed in error concealment [ITU-T, 1999, ITU-T, 2006, ETSI, 1996, 3GPP, 2001a]. However, the bad frame indicator only represents a coarse channel reliability information. Compared to the traditional hard-decision (HD) decoding method or error concealment techniques, soft-decision (SD) decoding [Fingscheidt and Vary, 2001, Kliewer and Thobaben, 2005] provides a higher robustness by exploiting the residual redundancy in terms of the *a priori* knowledge, expecting some soft information, which represents the bit-wise channel reliability information.

Moreover, quantization plays an important role in digital communications [Jayant and Noll, 1984]. It assigns a sequence of signal samples to a sequence of discrete values in a certain range, and thereby achieves a certain compression. The quantized signal samples are represented by the corresponding quantization codebook indices, with either fixed-length codes (FLCs) or variable-length codes (VLCs). The codebook entries can be either fixed over time or time-variant. However, for correlated source processes, source correlation cannot be exploited by a scalar quantization with time-invariant codebook entries, and without any form of predictive encoding, since it is designed in the same way both for correlated and uncorrelated processes.

This thesis aims to improve the performance of source codecs, in particular of speech and audio codecs. For erroneous transmission conditions, the SD decoding algorithms can be applied to speech and audio codecs, both for FLCs and VLCs. In order to improve the scalar quantization performance for correlated processes, a new decoding approach generating a time-variant scalar quantization codebook is proposed. This approach is advantageously applicable in both error-free and error-prone transmission conditions, both with HD and SD decoding. For the purpose of further validating the performance of the proposed approach, it is applied to speech codecs for both error-free and error-prone transmission conditions.

1.2 Brief Overview of Some Important Speech and Audio Codecs

Waveform coding is considered as an early precursor of speech coding, where speech samples are quantized and transmitted, known as pulse code modulation (PCM) (ITU-T Recommendation G.711 [ITU-T, 1972]) and Adaptive Differential Pulse Code Modulation (ADPCM) (ITU-T Recommendation G.726 [ITU-T, 1990] for narrowband speech (sampling frequency $f_s = 8$ kHz), and G.722 [ITU-T, 1988] for wideband speech ($f_s = 16$ kHz)).

Currently the most widely used speech coding scheme, follows the so-called hybrid coding approach according to the code-excited linear prediction (CELP) algorithm, which is adopted in the Adaptive Multi-Rate (AMR) [3GPP, 1999a] and AMR Wideband (AMR-WB) [3GPP, 2001b] speech codecs. AMR (or AMR-NB) and AMR-WB are two important speech codecs widely used in mobile communications today.

Aiming at providing high speech quality at low bit rates and offering good quality for generic audio signals, the ITU-T Recommendation G.718 [ITU-T, 2008] is designed for internet protocol (IP) transport applications on fixed, wireless, and mobile networks. It supports bit rates from 8 kbps to 32 kbps and is highly robust to packet losses. Recommendation G.718 includes a two-stage coding structure supporting narrowband or wideband signals: The lower two layers are on the basis of the CELP approach, while in higher layers transform coding according to the overlap-add modified discrete cosine transform (MDCT) is used. G.718 is also interoperable with the AMR-WB standard [Jelinek et al., 2009].

The Moving Picture Experts Group (MPEG) has standardized several audio compression formats: Starting with the most famous MPEG-1 or MPEG-2 Audio Layer III standard, known as MP3, and followed by advanced audio coding (AAC), the MPEG-4 High-Efficiency Advanced Audio Coding (HE-AAC) [ISO/IEC, 2005] was designed to offer high audio quality at a low bit rate. In order to merge the general audio coding and speech coding into one solid design, the MPEG Unified Speech and Audio Coding (USAC) [ISO/IEC, 2012] was designed, especially for applications with limited or varying bandwidth. It is based on the predecessors MP3, AAC, and HE-AAC, but works more efficiently [Neuendorf et al., 2013]. Instead of using Huffman coding (as in HE-AAC), arithmetic coding is used in USAC, which is still a kind of variable-length coding.

The currently newest standard is the Enhanced Voice Services (EVS) [3GPP, 2014a] codec which offers equally high quality both for generic audio and speech [Ericsson, 2014]. As the successor of the current mobile HD speech codec AMR-WB, EVS extends the audio bandwidth up to 20 kHz and provides better compression efficiency and robustness against transmission errors. It supports narrowband, wideband, super-wideband, and fullband audio, with the bit rates ranging from 5.9 to 128 kbps. With the evolution of mobile systems, the EVS codec is particularly designed for Long-Term Evolution (LTE), but it still brings benefits to any circuit-switched system and voice over IP (VoIP) [Nokia, 2015]. Finally it is interoperable with AMR-WB.

For erroneous transmission conditions, in order to improve the speech and audio quality after decoding, the concept of soft-decision decoding can be applied to any codec for those applications where bit errors may occur. For the codecs G.718, USAC, and EVS, where packet loss is likely to occur, much research focuses on frame loss concealment. In this thesis, the important and still very popular ADPCM, AMR-NB, AMR-WB, and HE-AAC codecs are studied.

1.3 Outline of the Thesis

This thesis is divided into three main parts:

- The first part focuses on the FL/SD decoding, including the theoretical description of the FL/SD decoding algorithm inspired from [Fingscheidt and Vary, 2001] and the application of FL/SD decoding to speech codecs and an audio codec.
- The second part concentrates on the VL/SD decoding, including a retrospective look of the VL/SD decoding algorithm having its roots in [Kliewer and Thobaben, 2005] and the application of VL/SD decoding to an audio codec.
- The last part is mainly about an improved decoding approach generating a time-variant scalar quantization codebook, including the theoretical elaboration and its application to speech codecs.

Chapter 2 begins with a general description of source parameter encoding and decoding. First the source parameter models including the modeling of unquantized parameters, the quantization approaches, and the modeling of quantized parameters are introduced. Thereafter, the channel models and the modulation technique used in this thesis are

described. Finally, the traditional FL/HD decoding method and the existing FL/SD decoding approach used in this thesis is briefly presented.

Chapter 3 is dedicated to the application of FL/SD decoding to the AMR-NB and AMR-WB speech codecs. Moreover, new approaches utilizing both interframe and intraframe redundancy for the spectral envelope parameters in both AMR-NB and AMR-WB are proposed. The performance of using the FL/SD decoding algorithm in AMR-NB and AMR-WB, and employing the corresponding FL/HD decoding method or the standard error concealment scheme is compared.

Chapter 4 introduces the VL/SD decoding method, and provides a new insight of the definitions of stage boundaries and state intervals in the trellis representation. In addition, the links between the VL/SD decoding and FL/SD decoding approaches are explored. At the end, the performance tradeoffs of VL/SD and FL/SD decoding depending on block lengths, source correlation, and quantization bit rates are discussed.

Chapter 5 elaborates the application of FL/SD decoding to the parameter global gain and of VL/SD decoding to the scale factor parameters and quantized spectral coefficients in MPEG-4 HE-AAC. Furthermore, for the quantized spectral coefficients with unsigned Huffman codebooks, a new trellis representation considering sign bits is proposed. The performance comparison between SD decoding, HD decoding, and standard error concealment is discussed.

Chapter 6 proposes a new decoding approach to improve the quantization performance for correlated processes, even for error-free transmission conditions. Without changing the standard encoder and based on a predictor at the receiver side, a time-variant quantization codebook is generated. The performance of using the standard and proposed approaches both for error-free and error-prone transmission conditions is presented. In adverse transmission conditions, both HD and SD decoding schemes are adopted.

Chapter 7 emphasizes the application of the improved scalar quantization approach in Chapter 6 to the G.726 and G.722 ADPCM speech codecs, both for error-free and error-prone transmission conditions, with a focus on HD decoding. The performance difference between employing the standard and the proposed decoders is depicted.

Chapter 8 presents concluding remarks of this thesis.

Chapter 2

Aspects of Source Parameter Encoding and Decoding

In digital communications, for transmission and storage purposes, the source parameters are transformed to a bit stream at the encoder side and recovered at the decoder side, which are known as source encoding and decoding, respectively. This chapter presents the general aspects of source parameter encoding and decoding. For the source parameter encoding, the source parameter models including modeling of unquantized parameters, quantization of parameters, and modeling of quantized parameters are explained in Section 2.1. The transmission channel models and a relevant modulation technique are introduced in Section 2.2. At the end, the hard-decision (HD) and soft-decision (SD) decoding approaches are presented in Section 2.3, with the focus on fixed-length codes (FLCs) in this chapter. Decoding approaches for variable-length codes (VLCs) will be explained in Chapter 4.

2.1 Source Parameter Models

This section describes the relevant source parameter models. For the source parameter encoding, the autoregressive (AR) process, which is usually used for modeling of unquantized parameters, is first introduced. Thereafter, different quantization approaches are presented, and the quantization approaches of the codec parameters used in this thesis are listed. The Markov process used to model the quantized parameters with the statistical knowledge (i.e., *a priori* knowledge) is described afterwards. Finally, the channel models, the traditional fixed-length hard-decision (FL/HD) decoding method, and the existing fixed-length soft-decision (FL/SD) decoding used in this thesis are explained.

2.1.1 Modeling Unquantized Parameters

In speech coding, considering a simplified model of the vocal tract, the unquantized codec parameters are often modeled based on the AR process [P. Vary, 2005]. The block diagram

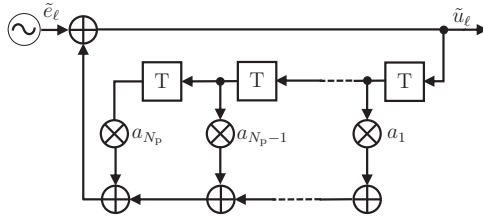


Figure 2.1: Block diagram of the N_p th-order autoregressive process [Jayant and Noll, 1984]).

of the AR process of order N_p ($\text{AR}(N_p)$) is shown in Fig. 2.1. The $\text{AR}(N_p)$ process output parameter \tilde{u}_ℓ at frame $\ell \in \{1, 2, \dots\}$ ¹ satisfies [Jayant and Noll, 1984]

$$\tilde{u}_\ell = \tilde{e}_\ell + \sum_{\varkappa=1}^{N_p} a_{\varkappa} \cdot \tilde{u}_{\ell-\varkappa}, \quad (2.1)$$

typically with a zero-state initialization. Herein, \tilde{e}_ℓ represents the innovation, a_{\varkappa} denotes the predictor coefficient, with $\varkappa \in \{1, 2, \dots, N_p\}$. Assuming the speech signals are stationary for short time intervals of 20 to 400 ms, the predictor coefficients can be adapted on a short-term basis and determined with linear prediction techniques, either with the auto-correlation method or the covariance method [P. Vary, 2005]. The efficient Levinson-Durbin algorithm is often used for the auto-correlation method.

The AR process implies that the successive parameters are correlated. For a stationary random process, the *correlation coefficient* of the parameters \tilde{u}_ℓ at the current frame index ℓ and $\tilde{u}_{\ell-\varkappa}$ at the frame index $\ell - \varkappa$ is defined by [A. Papoulis, 2002]

$$\rho_{\tilde{u}_\ell, \tilde{u}_{\ell-\varkappa}} = \frac{E\{\tilde{u}_\ell \cdot \tilde{u}_{\ell-\varkappa}\} - E^2\{\tilde{u}_\ell\}}{\sigma_{\tilde{u}}^2}, \quad (2.2)$$

with $E\{\cdot\}$ being the expectation value. Note that the *correlation coefficients* are not the same as the values of the predictor coefficients in the $\text{AR}(N_p)$ process. However, for the $\text{AR}(1)$ process, using the auto-correlation method, the only predictor coefficient a can be obtained by [P. Vary, 2005]

$$a = a_1 = \frac{\varphi_{\tilde{u}\tilde{u}}(1)}{\varphi_{\tilde{u}\tilde{u}}(0)} = \frac{E\{\tilde{u}_\ell \cdot \tilde{u}_{\ell-1}\}}{E\{\tilde{u}_\ell^2\}}, \quad (2.3)$$

which leads to $a = \rho_{\tilde{u}_\ell, \tilde{u}_{\ell-1}}$, if the expectation value fulfills $E\{\tilde{u}_\ell\} = 0$.

2.1.2 Quantization of Parameters

In digital communications, quantization is needed for analog-to-digital conversion and for source coding as well [P. Vary, 2005]. A sequence of parameters is assigned to a sequence of discrete values in a certain range. After quantization, the unquantized parameters are represented by the corresponding quantization codebook indices. As mentioned in [Gray and Neuhoff, 1998], three categories of quantizers can be identified as:

¹Without loss of generality, a frame-based parameter with frame index ℓ is assumed in this chapter.

- Scalar quantization [Lloyd, 1982, Max, 1960] which assigns a single input (unquantized) parameter to a single output (quantized) parameter, or vector quantization [Linde et al., 1980] which maps a vector of input parameters to a vector of output parameters;
- Quantizers with fixed-rate coding having the same codeword length for each quantization codebook index [Oliver et al., 1948, Hui and Neuhoff, 2001], or quantizers with variable-rate coding having variable codeword length and providing a better compression performance [Lookabaugh et al., 1993];
- Quantizers being memoryless with a fixed quantization codebook [Lloyd, 1982, Max, 1960, Kleijn and Hagen, 1996], or quantizers having memory with either a time-invariant or a time-variant codebook, with the memory standing for the statistical properties of the source process [Jayant and Noll, 1984, Jayant, 1974, McDonald, 1966].

The above quantization approaches used for the parameters of speech and audio codecs in this thesis are listed as follows:

- (1) Scalar quantization: Adaptive codebook and fixed codebook gains in the Adaptive Multi-Rate Narrowband (AMR-NB) speech codec, and spectral coefficients in the High-Efficiency Advanced Audio Coding (HE-AAC) audio codec;
- (2) Vector quantization: Line spectral frequencies (LSFs) in AMR-NB, immittance spectral frequencies (ISFs) and vector-quantized codebook gain (VQ gain) in the AMR Wideband (AMR-WB) speech codec;
- (3) Fixed-rate quantization: Parameters mentioned above in items (1), (2);
- (4) Variable-rate quantization: Spectral coefficients in the HE-AAC audio codec;
- (5) Memoryless quantization: Parameters mentioned above in items (1)-(5);
- (6) Quantizers having memory: Difference signal in the Adaptive Differential Pulse Code Modulation (ADPCM) speech codec.

2.1.3 Modeling Quantized Parameters

The quantized parameter can be modeled as a Markov process of N th order:

$$P(\mathbf{x}_\ell \mid \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N}, \mathbf{x}_{\ell-N-1}, \dots) = P(\mathbf{x}_\ell \mid \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N}), \quad (2.4)$$

with \mathbf{x}_ℓ being the bit combination of the quantized parameter at frame ℓ . The statistical knowledge of a Markov process is known as the joint probabilities $P(\mathbf{x}_\ell, \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N})$ or the conditional probabilities $P(\mathbf{x}_\ell \mid \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N})$, which are defined as *a priori* probabilities or *a priori* knowledge [Fingscheidt and Vary, 2001]. Typically, modeling the quantized parameter as a zeroth-order and first-order Markov process, a zeroth-order *a priori* knowledge (AK0) $P(\mathbf{x}_\ell^{(i)})$ and first-order *a priori* knowledge (AK1) $P(\mathbf{x}_\ell^{(i)} \mid \mathbf{x}_{\ell-1}^{(j)})$ can be obtained, with $\mathbf{x}_\ell^{(i)}$ and $\mathbf{x}_{\ell-1}^{(j)}$ representing the bit combinations from the current frame ℓ and the previous frame $\ell - 1$, respectively.

The *a priori* knowledge is obtained in the training process in advance, by processing a large database. After counting and normalizing the occurrence frequencies of different

pairs of quantizer output symbols, the AK0 term can be computed by

$$P(\mathbf{x}_\ell^{(i)}) = \sum_{j=0}^{2^M-1} P(\mathbf{x}_\ell^{(i)}, \mathbf{x}_{\ell-1}^{(j)}), \quad (2.5)$$

with M being the quantization bit rate (i.e., the number of bits for transmitting a single parameter). According to the chain rule [A. Papoulis, 2002], the AK1 term can be obtained by

$$P(\mathbf{x}_\ell^{(i)} | \mathbf{x}_{\ell-1}^{(j)}) = \frac{P(\mathbf{x}_\ell^{(i)}, \mathbf{x}_{\ell-1}^{(j)})}{\sum_{f=0}^{2^M-1} P(\mathbf{x}_\ell^{(f)}, \mathbf{x}_{\ell-1}^{(j)})}. \quad (2.6)$$

As mentioned in Chapter 1, Shannon noted the value of utilizing the redundancy in the source at the receiver side [Shannon, 1948]. Once the *a priori* knowledge is known, the residual redundancy (in bits²), which can be utilized for error concealment, is worth being observed. Modeling the quantized parameter as a zeroth-order Markov process, the entropy representing the minimum bits needed for coding a parameter can be calculated by [Jayant and Noll, 1984]

$$H(\mathbf{x}_\ell) = - \sum_{i=0}^{2^M-1} P(\mathbf{x}_\ell^{(i)}) \log_2 P(\mathbf{x}_\ell^{(i)}). \quad (2.7)$$

Assuming an N th-order Markov process, the conditional entropy is given by

$$H(\mathbf{x}_\ell | \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N}) = - \sum_{i=0}^{2^M-1} \sum_{j=0}^{2^M-1} \dots \sum_{h=0}^{2^M-1} P(\mathbf{x}_\ell^{(i)}, \mathbf{x}_{\ell-1}^{(j)}, \dots, \mathbf{x}_{\ell-N}^{(h)}) \cdot \log_2 P(\mathbf{x}_\ell^{(i)} | \mathbf{x}_{\ell-1}^{(j)}, \dots, \mathbf{x}_{\ell-N}^{(h)}). \quad (2.8)$$

As a consequence, the residual redundancy is written as [Jayant and Noll, 1984]

$$\Delta R = M - H(\mathbf{x}_\ell | \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N}). \quad (2.9)$$

Following [Fingscheidt, 2008], (2.9) is determined by $\Delta R = \Delta R_d + \Delta R_c$, with the *utilizable, distribution-dependent* residual redundancy

$$\Delta R_d = M - H(\mathbf{x}_\ell), \quad (2.10)$$

and the *utilizable, correlation-dependent* residual redundancy ΔR_c (also known as the mutual information $\mathcal{I}(\mathbf{x}_\ell; \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N})$) being obtained by

$$\Delta R_c = H(\mathbf{x}_\ell) - H(\mathbf{x}_\ell | \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N}). \quad (2.11)$$

For an AR source process, the Markov process order N is not necessarily the same as the AR process order N_p . After the quantization of the parameters taken from an AR process, the Markov property from (2.4) may not fully hold, due to the distortion by the

²For the simplicity of description, the unit bit is used instead of bit per parameter.

M	N							
	1	2	3	4	5	6	7	8
1	0.406	0.426	0.432	0.434	0.435	0.436	0.436	0.437
2	0.777	0.801	0.804	0.807	0.811	0.819	0.834	0.862

(a) $M \in \{1, 2\}$ bit

$M=3$ bit, N						$M=4$ bit, N		
1	2	3	4	5	6	1	2	3
1.032	1.040	1.045	1.066	1.124	1.263	1.146	1.156	1.209

(b) $M \in \{3, 4\}$ bit

Table 2.1: Residual redundancy ΔR_c for different quantization bit rates $M \in \{1, 2, 3\}$ and different Markov process orders N ; Gaussian AR(1) samples with $a=0.9$, subject to Lloyd-Max quantization assumed.

quantization error, especially for low rate quantization with large quantization error. The conditional entropy satisfies [A. Papoulis, 2002]

$$M \geq H(\mathbf{x}_\ell) \geq H(\mathbf{x}_\ell | \mathbf{x}_{\ell-1}) \geq H(\mathbf{x}_\ell | \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N}) \geq H(\mathbf{x}_\ell | \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-(N+\varkappa')}), \quad (2.12)$$

with $\varkappa' \in \{0, 1, 2, \dots\}$. For an M -bit scalar Lloyd-Max quantizer applied to Gaussian AR(1) samples with $a = 0.9$, the residual redundancies ΔR in (2.9) and ΔR_c in (2.11) increase with the rise of Markov order N , as the ΔR_c results show in Tab. 2.1. In other words, more redundancy can be exploited by modeling a higher order Markov process. Moreover, if $H(\mathbf{x}_\ell | \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-N}) = H(\mathbf{x}_\ell | \mathbf{x}_{\ell-1}, \dots, \mathbf{x}_{\ell-(N+\varkappa')})$, $\varkappa' > 0$, this implies no statistical dependence between \mathbf{x}_ℓ and $\mathbf{x}_{\ell-(N+\varkappa')}$ [Fingscheidt, 1998]. In that case, no extra residual redundancy can be utilized with the assumption of a Markov process order higher than N . In addition, if the bit rate is high enough, the *utilizable, correlation-dependent* residual redundancy ΔR_c for $N = N_p$ already turns out to be approximately the total *correlation-dependent* residual redundancy³ [Fingscheidt, 2008].

2.2 Channel Models

The channel is a medium used to transmit a signal from the transmitter side to the receiver side. The transmitted signals can be influenced by the interference on the channel. The mathematical models are found to reflect the characteristics of the transmission channel easily [Jayant and Noll, 1984]. The channel models and modulation technique adopted in this thesis are described in this section.

³Note that 4 bit is not high enough for this.

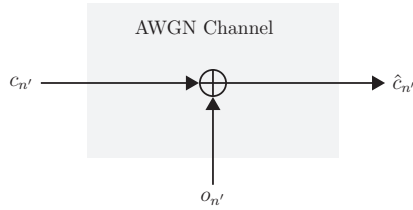


Figure 2.2: The block diagram of an AWGN channel [Proakis, 2001].

2.2.1 Additive White Gaussian Noise (AWGN) Channel With Binary Phase-Shift Keying (BPSK)

AWGN Channel

The additive white Gaussian noise (AWGN) channel is an often used channel model in digital communications, with signals being distorted by additive white Gaussian noise. As depicted in Fig. 2.2, the received (real-valued) symbol $\hat{c}_{n'}$ is obtained by adding the transmitted modulated symbol $c_{n'}$ and the noise $o_{n'}$, with n' being the symbol time index. The white Gaussian noise signal $o_{n'}$ satisfies a constant power spectral density and a zero-mean Gaussian distribution. The probability density function (PDF) of the noise $o_{n'}$ is defined by [Proakis, 2001]

$$p(o_{n'}) = \frac{1}{\sqrt{2\pi}\sigma_{o_{n'}}} \cdot \exp\left(-\frac{(o_{n'})^2}{2\sigma_{o_{n'}}^2}\right), \quad (2.13)$$

with the variance of the noise $\sigma_{o_{n'}}^2 = N_0/2$, which is the constant noise power spectral density. The white Gaussian noise signals are statistically independent from each other. Therefore, an AWGN channel is considered as a memoryless channel. The conditional PDF of the received symbol given the transmitted symbol can be written as [Proakis, 2001]

$$p(\hat{c}_{n'} | c_{n'}) = \frac{1}{\sqrt{\pi N_0}} \cdot \exp\left(-\frac{(\hat{c}_{n'} - c_{n'})^2}{N_0}\right). \quad (2.14)$$

BPSK Modulation

In digital communications, modulation allows the (digital) bit stream (i.e., message signal) to be transmitted over a bandpass frequency range in the channel, by varying the carrier signal. Phase-shift keying modulation varies the phase of the carrier signal, while the amplitude and the frequency remain the same. As the signal space diagram for binary phase-shift keying (BPSK) signals in Fig. 2.3 shows, the BPSK modulated symbols are represented by $c_{n'} \in \{c^{(0)} = +\sqrt{E_b}, c^{(1)} = -\sqrt{E_b}\}$ corresponding to the transmitted bipolar bit +1 and -1, respectively [Proakis, 2001]. Herein, the signal energy per bit E_b equals the signal energy per symbol E_s (i.e., the symbol time index n' is just the bit index).

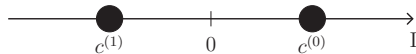


Figure 2.3: The signal space diagram of BPSK signals (I-axis: in phase component) [Proakis, 2001].

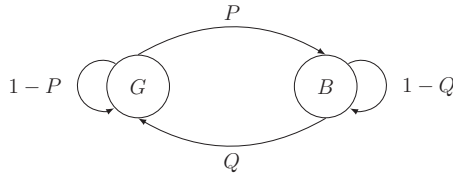


Figure 2.4: The state diagram of a Gilbert-Elliott channel [Gilbert, 1960].

For BPSK modulation, the log-likelihood ratio is defined as [Proakis and Salehi, 2001]

$$L(\hat{c}_{n'}) = \log \frac{\mathbb{p}(\hat{c}_{n'} | c_{n'} = c^{(0)})}{\mathbb{p}(\hat{c}_{n'} | c_{n'} = c^{(1)})}. \quad (2.15)$$

Applying (2.14) to (2.15) and assuming the assignment of the bit $\hat{x}_\ell(m)$ (m th bit in the received bit combination $\hat{\mathbf{x}}_\ell$ at frame ℓ) to $\hat{c}_{n'}$, it can be derived that⁴ [Hagenauer, 1980, Hagenauer, 1995]

$$\begin{aligned} L(\hat{x}_\ell(m)) &= L(\hat{c}_{n'}) = \log \frac{\exp(-(\hat{c}_{n'} - \sqrt{E_b})^2/N_0)}{\exp(-(\hat{c}_{n'} + \sqrt{E_b})^2/N_0)} \\ &= 4 \cdot \frac{E_b}{N_0} \cdot \frac{\hat{c}_{n'}}{\sqrt{E_b}}. \end{aligned} \quad (2.16)$$

The factor $1/\sqrt{E_b}$ plays a normalization role, therefore, the $L(\hat{x}_\ell(m))$ is proportional to the received symbol $\hat{c}_{n'}$. Herein, $\frac{E_b}{N_0}$ is the so-called signal-to-noise ratio (SNR) per bit [Proakis, 2001] reflecting the channel condition.

2.2.2 Gilbert-Elliott Channel

The Gilbert-Elliott (GE) channel model [Gilbert, 1960] is widely used to describe burst errors in transmission channels. A GE channel is a time-variant channel with two channel states: good (G) and bad (B). The state changing process is considered as a first-order Markov process. As it is shown in the state diagram in Fig. 2.4, the transition probability from good to bad is denoted as P , while the one from bad to good is represented by Q . Correspondingly, the transition probability from good to good and bad to bad is represented by $1 - P$ and $1 - Q$, respectively. For a steady state, the stationary state probabilities P_G and P_B satisfy [Turin, 1998]

$$(P_G, P_B) = (P_G, P_B) \cdot \begin{pmatrix} 1 - P & P \\ Q & 1 - Q \end{pmatrix}, \quad (2.17)$$

⁴Note that $\hat{x}_\ell(m) = \text{sign}(L(\hat{x}_\ell(m))) = \text{sign}(\hat{c}_{n'})$.

and $P_G + P_B = 1$, which results in $P_G = \frac{Q}{P+Q}$ and $P_B = \frac{P}{P+Q}$ [Hagenauer, 1980]. The average bit error probability of a GE channel can be calculated by [Hagenauer, 1980]

$$\overline{\text{BER}} = P_G \cdot \text{BER}_G + P_B \cdot \text{BER}_B, \quad (2.18)$$

with BER_G and BER_B being related to the respective SNRs $E_b/N_0|_{\text{good}}$ and $E_b/N_0|_{\text{bad}}$ for the good and bad channel state.

2.3 Fixed-Length (FL) Soft-Decision (SD) Decoding of Parameters

Error concealment is often used to conceal effects from transmissions over error-prone channels and to improve the robustness of the system. The common ground of the error concealment adopted in plain speech waveform transmission [ITU-T, 1999], ADPCM-coded speech [ITU-T, 2006, Serizawa and Nozawa, 2002], and hybrid speech coders such as Global System for Mobile Communications (GSM) [ETSI, 1992a, ETSI, 1995], GSM Enhanced Fullrate (EFR) [ETSI, 1996], AMR-NB [3GPP, 1999b], and AMR-WB [3GPP, 2001a] is that each of the received frames is marked as good or bad by a bad frame indicator (BFI), which can be considered as a coarse channel reliability information. Signal substitution is performed once a bad frame is indicated. Accordingly, when a bad frame is indicated by the BFI, the whole frame will still be replaced although only some less important bits might be distorted. On the other hand, when a good frame is indicated, the whole frame will be regarded as reliable even if a lot of bits are corrupted.

Instead of receiving hard-decided bits as in traditional HD decoding, SD decoding, which expects soft information representing bit-wise channel reliability information from the demodulator [Cuperman et al., 1994] or channel decoder [Hagenauer and Hoehner, 1989, Huber and Ruppel, 1990], has been considered as a robust means for error concealment. The residual redundancy observed in quantizer outputs (Section 2.1.3) can be utilized for SD decoding. In the 90's, using the soft information in the form of log-likelihood ratios (LLRs), Fingscheidt and Vary proposed a (fixed-length) soft-decision source decoding paradigm [Fingscheidt and Vary, 1996, Fingscheidt and Vary, 2001], which has been applied to speech and audio decoders: A-law PCM and GSM Fullrate speech coding [Fingscheidt and Vary, 2001], G.726 and G.722 ADPCM [Fingscheidt, 2003, López-Oller et al., 2016], high-quality PCM audio [Pflug and Fingscheidt, 2011a], AMR-WB [Han et al., 2013], and also distributed automatic speech recognition (DSR) [Ion and Hüb-Umbach, 2006, Peinado et al., 2005]. Moreover, much research has focused on joint or iterative source-channel coding with soft-decision reconstruction, such as mixed-excitation linear predictive (MELP) speech coding [Fazel and Fuja, 2003], image and video coding in JPEG 2000 [Fresia and Caire, 2010] and H.264/AVC [Nasruminallah and Hanzo, 2009], respectively.

Along with the standardization of speech and audio codecs in the packet-switched networks, some research also focuses on packet loss concealment. There are mainly two

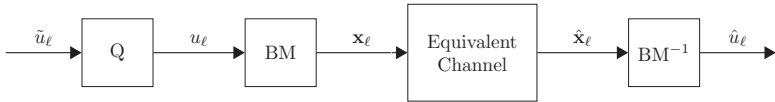


Figure 2.5: The block diagram of the transmission system with hard-decision decoding [Fingscheidt and Vary, 2001].

time-domain concealment paradigms [Lecomte et al., 2015a]: the waveform-based approach such as, e.g., time scale modification [Roucos and Wilgus, 1985], and the most commonly used parameter-based approach (as it is applied in the G.718 codec). The most recent enhanced voice services (EVS) codec includes several packet loss concealment approaches related to signal classification, spectral envelope representation, algebraic code excited linear prediction (ACELP) modes, modified discrete cosine transform (MDCT) modes, and bandwidth extension [3GPP, 2014b]. The detailed description can be found in [3GPP, 2014b, Lecomte et al., 2015b]. Note that this thesis focuses on the *bit* error concealment, where not the whole frame is considered as being lost. Although SD decoding algorithms can directly be applied to packet loss conditions, considering the LLRs to be zero, the SD decoding approach may need further modification for the needs of packet loss applications.

This section presents the traditional HD decoding and the SD decoding as proposed in [Fingscheidt and Vary, 2001]. In addition, different to [Fingscheidt and Vary, 2001] only with a forward recursion, the forward-backward algorithm (BCJR algorithm) [Bahl et al., 1974] is also described.

2.3.1 Hard-Decision (HD) Decoding

The block diagram of the transmission system with HD decoding is depicted in Fig. 2.5. The unquantized source parameter (or source sample, source symbol, source signal) \hat{u}_ℓ is first quantized to u_ℓ with quantization codebook index $i \in \mathcal{I} = \{0, 1, \dots, 2^M - 1\}$. Via bit mapping (BM), the quantized parameter u_ℓ is represented by the bit combination $\mathbf{x}_\ell = (x_\ell(0), x_\ell(1), \dots, x_\ell(m), \dots, x_\ell(M-1))$, with $x_\ell(m) \in \{-1, +1\}$ in bipolar notation. The BM is assumed to be based on the natural binary code (NBC) [Jayant and Noll, 1984]. After transmission over the equivalent channel which may include channel coding/decoding, modulation/demodulation, the inverse bit mapping (BM^{-1}) is performed at the receiver side. It transforms the received bit combination $\hat{\mathbf{x}}_\ell$ to the received parameter \hat{u}_ℓ , based on the quantization codebook. In this thesis, the equivalent channel either is an AWGN channel with BPSK modulation or a GE channel, both without any channel coding.

2.3.2 SD: A *Posteriori* Probabilities (APPs)

Instead of receiving the bit combination $\hat{\mathbf{x}}_\ell$ as in HD decoding, the LLRs representing bit-wise channel reliability information are expected by the SD decoding. The block di-

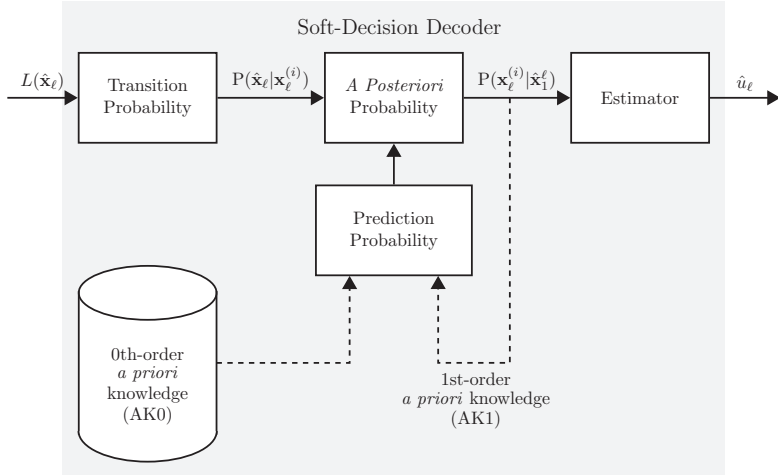


Figure 2.6: The block diagram of a soft-decision decoder [Fingscheidt and Vary, 2001].

agram of SD decoding from [Fingscheidt and Vary, 2001, Fingscheidt, 2008] is depicted in Fig. 2.6. The core of SD decoding is to compute the *a posteriori* probabilities (APPs) $P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1})$, which represent the probability of a possibly transmitted bit combination $\mathbf{x}_\ell^{(i)}$ given the bit combinations $\hat{\mathbf{x}}_1^\ell = (\hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}) = (\hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_{\ell-1}, \dots, \hat{\mathbf{x}}_1)$ received in frames $1, 2, \dots, \ell$. According to the chain rule, the APPs $P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1})$ can be derived by

$$\begin{aligned} P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}) &= \frac{P(\hat{\mathbf{x}}_\ell | \mathbf{x}_\ell^{(i)}, \hat{\mathbf{x}}_1^{\ell-1}) \cdot P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_1^{\ell-1}) \cdot P(\hat{\mathbf{x}}_1^{\ell-1})}{P(\hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1})} \\ &= \frac{P(\hat{\mathbf{x}}_\ell | \mathbf{x}_\ell^{(i)}, \hat{\mathbf{x}}_1^{\ell-1}) \cdot P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_1^{\ell-1})}{P(\hat{\mathbf{x}}_\ell | \hat{\mathbf{x}}_1^{\ell-1})}. \end{aligned} \quad (2.19)$$

Assuming a memoryless channel, (2.19) can be further written as

$$P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}) = \frac{1}{C} \cdot P(\hat{\mathbf{x}}_\ell | \mathbf{x}_\ell^{(i)}) \cdot P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_1^{\ell-1}), \quad (2.20)$$

with $C = P(\hat{\mathbf{x}}_\ell | \hat{\mathbf{x}}_1^{\ell-1})$ being a constant concerning indices $i \in \mathcal{I}$. In practice, the constants⁵ C is used to ensure $\sum_{i \in \mathcal{I}} P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}) = 1$, which results in $C = \sum_{f \in \mathcal{I}} P(\hat{\mathbf{x}}_\ell | \mathbf{x}_\ell^{(f)}) \cdot P(\mathbf{x}_\ell^{(f)} | \hat{\mathbf{x}}_1^{\ell-1})$. The APPs in (2.20) include the *channel transition probabilities* $P(\hat{\mathbf{x}}_\ell | \mathbf{x}_\ell^{(i)})$ and the so-called *predictive a priori probabilities* (*prediction probabilities*) $P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_1^{\ell-1})$.

⁵In this thesis, the constant C is always used to normalize the sum of all left-hand-side probabilities of the equation to one, and it may have different values in different equations.

Channel Transition Probabilities

Note that from [Fingscheidt, 2008, Pflug and Fingscheidt, 2013a, Pflug, 2013], it is known that regardless what functionalities being included in the equivalent channel (memoryless or with memory), a memoryless channel is assumed by SD decoding. The *channel transition probabilities* (*transition probabilities*), which represent the probabilities of a received bit combination $\hat{\mathbf{x}}_\ell$, given a probably transmitted bit combination $\mathbf{x}_\ell^{(i)}$, are calculated by

$$P(\hat{\mathbf{x}}_\ell | \mathbf{x}_\ell^{(i)}) = \prod_{m=0}^{M-1} P(\hat{x}_\ell(m) | x_\ell^{(i)}(m)), \quad (2.21)$$

with $\hat{x}_\ell(m)$ being the received hard-decided m th bit of $\hat{\mathbf{x}}_\ell$ and $x_\ell^{(i)}(m)$ being a possibly transmitted m th bit of $\mathbf{x}_\ell^{(i)}$, respectively. The conditional probability $P(\hat{x}_\ell(m) | x_\ell^{(i)}(m))$ can be written as

$$P(\hat{x}_\ell(m) | x_\ell^{(i)}(m)) = \begin{cases} 1 - \text{BER}_\ell(m), & \text{if } \hat{x}_\ell(m) = x_\ell^{(i)}(m), \\ \text{BER}_\ell(m), & \text{else,} \end{cases} \quad (2.22)$$

with the bit error probability BER being obtained by [Hagenauer, 1995]

$$\text{BER}_\ell(m) = \frac{1}{1 + \exp(|L(\hat{x}_\ell(m))|)}. \quad (2.23)$$

Predictive *A Priori* Probabilities

The residual redundancy is exploited in terms of the *a priori* knowledge and further used to compute the *predictive a priori probabilities* (*prediction probabilities*). Modeling the quantized parameter as an N th-order Markov process, an N th-order *a priori* knowledge is stored with $2^{M(N+1)}$ words of memory. The choice of N is usually determined by considering the tradeoffs between the complexity and the performance of the SD decoder, as well as by the observed residual redundancy [Fingscheidt and Vary, 2001]. If the corresponding complexity is acceptable for an SD decoder, the SD decoding performance can be better improved with a higher Markov process order N , since more redundancy is exploited (see (2.12)).

The *prediction probabilities* $P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_1^{\ell-1})$ imply a possibly transmitted bit combination $\mathbf{x}_\ell^{(i)}$, given the past received bit combinations $\hat{\mathbf{x}}_1^{\ell-1}$ from frame 1 to $\ell-1$.

Modeling the quantized parameter as a zeroth-order Markov process, the prediction probability is simply

$$P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_1^{\ell-1}) = P(\mathbf{x}_\ell^{(i)}). \quad (2.24)$$

According to the chain rule, the prediction probability of the first-order Markov process

can be derived as follows [Fingscheidt, 1998, Fingscheidt, 2008]:

$$\begin{aligned}
P(\mathbf{x}_\ell^{(i)} \mid \hat{\mathbf{x}}_1^{\ell-1}) &= \sum_{j \in \mathcal{I}} P(\mathbf{x}_\ell^{(i)}, \mathbf{x}_{\ell-1}^{(j)} \mid \hat{\mathbf{x}}_{\ell-1}, \hat{\mathbf{x}}_1^{\ell-2}) \\
&= \frac{\sum_{j \in \mathcal{I}} P(\mathbf{x}_\ell^{(i)}, \mathbf{x}_{\ell-1}^{(j)}, \hat{\mathbf{x}}_{\ell-1}, \hat{\mathbf{x}}_1^{\ell-2})}{P(\hat{\mathbf{x}}_{\ell-1}, \hat{\mathbf{x}}_1^{\ell-2})} \\
&= \frac{1}{C} \cdot \sum_{j \in \mathcal{I}} P(\mathbf{x}_\ell^{(i)} \mid \mathbf{x}_{\ell-1}^{(j)}) \cdot P(\hat{\mathbf{x}}_{\ell-1} \mid \mathbf{x}_{\ell-1}^{(j)}) \cdot P(\mathbf{x}_{\ell-1}^{(j)} \mid \hat{\mathbf{x}}_1^{\ell-2}).
\end{aligned} \tag{2.25}$$

In addition, due to the high complexity using AK1 for digital audio samples quantized with 16 bits per sample [Pflug and Fingscheidt, 2011b] or uncoded high quality speech signals with $M \geq 16$ bits [Pflug and Fingscheidt, 2012], the prediction probabilities can also be obtained based on higher-order adaptive prediction using the normalized least-mean-squares (NLMS) algorithm.

A *Posteriori* Probabilities (APPs)

Typically, the APPs with AK0 and AK1 are used. According to (2.24), the APP in (2.20) with AK0 turns out to be

$$P(\mathbf{x}_\ell^{(i)} \mid \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}) = \frac{1}{C} \cdot P(\hat{\mathbf{x}}_\ell \mid \mathbf{x}_\ell^{(i)}) \cdot P(\mathbf{x}_\ell^{(i)}). \tag{2.26}$$

Moreover, applying (2.25) to (2.20), the APP with AK1 becomes

$$P(\mathbf{x}_\ell^{(i)} \mid \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}) = \frac{1}{C} \cdot P(\hat{\mathbf{x}}_\ell \mid \mathbf{x}_\ell^{(i)}) \cdot \sum_{j \in \mathcal{I}} P(\mathbf{x}_\ell^{(i)} \mid \mathbf{x}_{\ell-1}^{(j)}) \cdot P(\mathbf{x}_{\ell-1}^{(j)} \mid \hat{\mathbf{x}}_{\ell-1}, \hat{\mathbf{x}}_1^{\ell-2}), \tag{2.27}$$

with $P(\mathbf{x}_{\ell-1}^{(j)} \mid \hat{\mathbf{x}}_{\ell-1}, \hat{\mathbf{x}}_1^{\ell-2})$ being the APP from the previous frame.

Note that the APP with AK1 can also be derived directly as [Fingscheidt and Vary, 2001]:

$$\begin{aligned}
P(\mathbf{x}_\ell^{(i)} \mid \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}) &= \frac{\sum_{j \in \mathcal{I}} P(\mathbf{x}_\ell^{(i)}, \mathbf{x}_{\ell-1}^{(j)}, \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1})}{P(\hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1})} \\
&= \frac{1}{C} \cdot P(\hat{\mathbf{x}}_\ell \mid \mathbf{x}_\ell^{(i)}) \cdot \sum_{j \in \mathcal{I}} P(\mathbf{x}_\ell^{(i)} \mid \mathbf{x}_{\ell-1}^{(j)}) \cdot P(\mathbf{x}_{\ell-1}^{(j)} \mid \hat{\mathbf{x}}_{\ell-1}, \hat{\mathbf{x}}_1^{\ell-2}).
\end{aligned} \tag{2.28}$$

Forward-Backward Algorithm (BCJR Algorithm)

Using AK1, the APP calculation in (2.27) is a forward recursion only considering the past information. Taking also the future information into account, assuming a block of B consecutive parameters (i.e., $\ell \in \{1, 2, \dots, B\}$), the APPs with AK1 can be calculated by the forward-backward algorithm (BCJR algorithm) [Bahl et al., 1974]:

$$P(\mathbf{x}_\ell^{(i)} \mid \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}) = \frac{\alpha_\ell(i) \cdot \beta_\ell(i)}{\sum_{f \in \mathcal{I}} \alpha_\ell(f) \cdot \beta_\ell(f)}, \tag{2.29}$$

with $\alpha_\ell(i) = P(\hat{\mathbf{x}}_1^\ell, \mathbf{x}_\ell^{(i)})$ being calculated in the forward recursion and $\beta_\ell(i) = P(\hat{\mathbf{x}}_{\ell+1}^B | \mathbf{x}_\ell^{(i)})$ being computed in a backward recursion, respectively.

The forward recursion is processed as

$$\alpha_\ell(i) = P(\hat{\mathbf{x}}_\ell | \mathbf{x}_\ell^{(i)}) \cdot \sum_{j \in \mathcal{I}} P(\mathbf{x}_\ell^{(i)} | \mathbf{x}_{\ell-1}^{(j)}) \cdot \alpha_{\ell-1}(j), \quad (2.30)$$

with the initial value $\alpha_1(i) = P(\hat{\mathbf{x}}_1 | \mathbf{x}_1^{(i)}) \cdot P(\mathbf{x}_1^{(i)})$. In (2.30), $P(\mathbf{x}_\ell^{(i)} | \mathbf{x}_{\ell-1}^{(j)})$ is the AK1 term, and $P(\hat{\mathbf{x}}_\ell | \mathbf{x}_\ell^{(i)})$ denotes the channel transition probabilities based on (2.21).

The backward recursion can be derived by

$$\beta_\ell(i) = \sum_{h \in \mathcal{I}} P(\mathbf{x}_{\ell+1}^{(h)} | \mathbf{x}_\ell^{(i)}) \cdot P(\hat{\mathbf{x}}_{\ell+1} | \mathbf{x}_{\ell+1}^{(h)}) \cdot \beta_{\ell+1}(h), \quad (2.31)$$

with the initial value $\beta_B(i) = 1$.

Note that the AK1 term is required by the forward-backward algorithm. If the AK1 term $P(\mathbf{x}_{\ell+1}^{(h)} | \mathbf{x}_\ell^{(i)})$ is replaced by the AK0 term $P(\mathbf{x}_{\ell+1}^{(h)})$, the BCJR algorithm turns out reduce to only the forward recursion, due to the fact that the backward probability $\beta_\ell(i)$ is then independent of i (i.e., $\beta_\ell(i)$ is then the same for all $i \in \mathcal{I}$ at frame ℓ). Considering $\beta_\ell(i)$ as a constant, $\beta_\ell(i)$ is canceled out in the numerator and denominator in (2.29), with only the forward probability $\alpha_\ell(i)$ being taken into account, which leads to the APP calculation (2.26).

2.3.3 SD: Parameter Estimation

As shown in Fig. 2.6, after the calculation of the APPs, an estimation of the parameter is performed. According to the corresponding quantization codebook, the parameter can be estimated either by minimum mean-square error (MMSE) estimation [Melsa and Cohn, 1978]:

$$\hat{u}_\ell = \sum_{i \in \mathcal{I}} u^{(i)} \cdot P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}), \quad (2.32)$$

with $u^{(i)}$ being the quantization codebook entry for quantization index i , or by maximum *a posteriori* (MAP) estimation:

$$\hat{u}_\ell = u^{(i^{\text{opt}})} \quad \text{with } i^{\text{opt}} = \arg \max_i P(\mathbf{x}_\ell^{(i)} | \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1}). \quad (2.33)$$

2.4 Summary

Some general aspects of source parameter encoding and decoding are introduced in this chapter. The modeling of unquantized codec parameters is often based on an autoregressive (AR) process. The source parameters are quantized for transmission and storage purposes. Three categories of quantization approaches are differentiated. The quantized parameters can be modeled as a Markov process, with the statistical description of the Markov process being given by the *a priori* knowledge. Utilizing the *a priori* knowledge

and the bit-wise channel reliability information, the soft-decision (SD) decoding approach can offer a better robustness of the source decoder than using hard-decision (HD) decoding. This is done by performing an *a posteriori* probability (APP) calculation and a parameter estimation. The SD decoding approach depicted in this chapter will be applied to the Adaptive Multi-Rate Narrowband (AMR-NB) and AMR Wideband (AMR-WB) speech codecs in Chapter 3.

Chapter 3

Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

A good speech quality is desired for error-prone transmission conditions in mobile communications. The fixed-length soft-decision (FL/SD) source decoding presented in Section 2.3 can be advantageously applied to mobile communications. In this chapter, the FL/SD decoding scheme is applied to the Adaptive Multi-Rate Narrowband (AMR-NB) and AMR Wideband (AMR-WB) speech decoders. Section 3.1 introduces the standard AMR speech coding including an analysis of the encoding and decoding in AMR-NB and AMR-WB. The standard error concealment proposed in 3GPP Technical Specifications TS 26.091 and TS 26.191 is described in Section 3.2. In Section 3.3, the application of FL/SD decoding to codec parameters is described in detail and followed by comparing the computational complexity of all proposed SD decoding algorithms. Finally, the simulation setup and simulation results are presented and discussed in Section 3.4.

3.1 AMR Speech Coding

Different to narrowband speech communication with speech frequencies in the range of 300...3400 Hz, wideband speech communication covering frequencies range of 50...7000 Hz offers a better speech quality and intelligibility. The Adaptive Multi-Rate Narrowband (AMR-NB) [3GPP, 1999a] and AMR Wideband (AMR-WB) [3GPP, 2001b] are two speech codecs applied to Global System for Mobile Communications (GSM), Universal Mobile Telecommunications System (UMTS), and Long-Term Evolution (LTE) [Nokia, 2015]. A flexible network-controlled rate operation is allowed for AMR-NB and AMR-WB to select the best operation mode to meet the channel requirements. Both AMR-NB and AMR-WB employ the widely used speech coding scheme algebraic code excited linear prediction (ACELP) coding. In this section, the analysis of the encoding, decoding, and the codec parameters in the AMR-NB and AMR-WB speech codecs is presented.

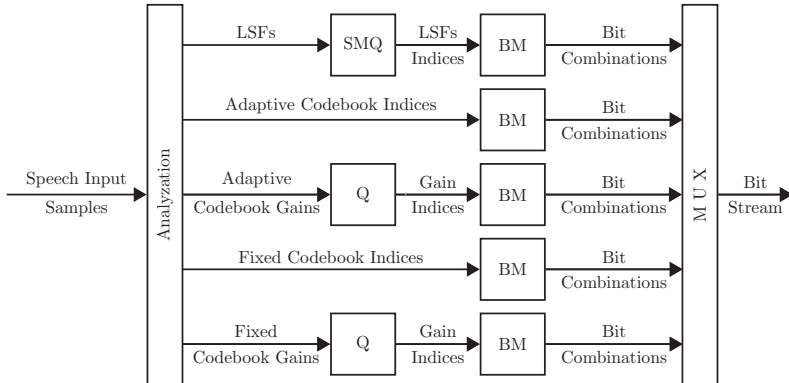


Figure 3.1: The block diagram of the AMR-NB encoder; the split matrix quantization (SMQ) is specific to the 12.2 kbps mode [Tian, 2015].

3.1.1 AMR Narrowband

The AMR-NB speech codec has been standardized as 3GPP Technical Specification TS 26.090 with a sampling rate of 8 kHz and eight modes varying bit rates from 4.75 kbps to 12.2 kbps [3GPP, 1999a]. The widely used 12.2 kbps mode is compatible with the GSM Enhanced Fullrate (EFR) speech codec and is therefore focused in this thesis. The AMR-NB encoding and decoding in [3GPP, 1999a] is briefly described as follows.

AMR-NB Encoding

The block diagram of the AMR-NB encoder operating at 12.2 kbps mode is depicted in Fig. 3.1. In each frame, a block of 160 speech samples (20 ms per frame) with 13-bit uniform pulse code modulation (PCM) format is transformed to a block of bit stream with 244 bits. Based on the ACELP coding model, an analysis of speech input samples is first performed to extract parameters: two sets of line spectral frequencies (LSFs) for each frame; the adaptive code book index (pitch delay), the adaptive codebook gain (pitch gain), the fixed codebook gain, and the fixed codebook index for each subframe. Via a corresponding quantizer/lookup table, the parameters are represented by the associated quantization codebook/lookup table indices. After bit mapping (BM), the bit combinations of all the codec parameters are finally multiplexed and transmitted.

AMR-NB Decoding

Fig. 3.2 shows the block diagram of the AMR-NB decoder. The received bit stream is first demultiplexed to bit combinations of the corresponding codec parameters. With inverse bit mapping (BM^{-1}) and a corresponding quantization codebook/lookup table,

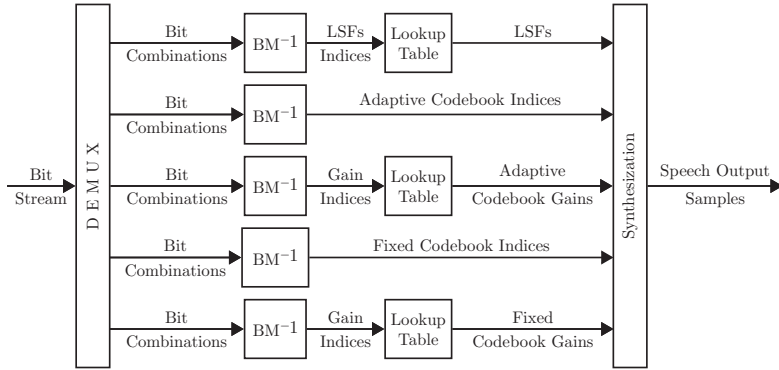


Figure 3.2: The block diagram of the AMR-NB decoder [Tian, 2015].

the bit combinations are transformed to codec parameters, which are finally synthesized to speech output samples. The analysis of each parameter [3GPP, 1999a] is elaborated in the following.

Line Spectral Frequencies (LSFs)

For the 12.2 kbps mode, the linear prediction (LP) analysis is performed twice per frame, which leads to 20 LP filter coefficients. For the purpose of quantization and interpolation, the two sets of LP coefficients are transformed to the LSF representation. Using a first-order moving-average (MA) predictor, the two residual LSF vectors, which are calculated by subtracting the predicted LSF vector from the mean-removed LSF vectors, are jointly quantized employing split matrix quantization (SMQ). The matrix consisting of two residual LSF vectors is split into 5 submatrices of $4 (2 \times 2)$ entries, with two elements from each LSF residual vector. The 5 LSF submatrices are (vector) quantized with 7, 8, 8+1 (sign bit), 8, and 6 bits, respectively.

Adaptive Codebook Index (Pitch Delay)

The adaptive codebook search and fixed codebook search are performed on a subframe basis. The parameters of the adaptive codebook include the adaptive codebook index (pitch delay) and the adaptive codebook gain. A number of fractional values in the range of $[17\frac{3}{6}, 94\frac{3}{6}]$ with the resolution being $\frac{1}{6}$, and integers in the range of $[95, 143]$ is used for the pitch delay in *odd* subframes and encoded with 9 bits. For *even* subframes, the relative delay of the previous *odd* subframe is encoded with 6 bits and in the range of $[T_1 - 8, T_1 + 7\frac{1}{2}]$ with a resolution of $\frac{1}{6}$. Herein, T_1 is the nearest integer to the pitch delay from the previous (*odd*) subframe.

Track	Pulse indices	Pulse positions
1	0,5	0,5,10,15,20,25,30,35
2	1,6	1,6,11,16,21,26,31,36
3	2,7	2,7,12,17,22,27,32,37
4	3,8	3,8,13,18,23,28,33,38
5	4,9	4,9,14,19,24,29,34,39

Table 3.1: Potential (non-zero) pulse positions for the fixed (algebraic) codebook in AMR-NB with 12.20 kbps mode [3GPP, 1999a].

Adaptive Codebook and Fixed Codebook Gains

The adaptive codebook gain is quantized by a 4-bit non-uniform scalar quantizer. A fourth-order MA prediction is used for the quantization of the fixed codebook gain. The correction factor which is the ratio between the (true) fixed codebook gain and the estimated gain is actually quantized by a 5-bit quantization codebook.

Fixed Codebook Index

In the ACELP coding model, an algebraic codebook is used as the fixed codebook, which has all-zero codewectors except a few non-zero pulses with the amplitude being +1 or -1. The 8 kHz sampling frequency leads to 40 samples per subframe. As shown in Tab. 3.1, the 40 positions are divided into 5 tracks of interleaved positions, with 2 non-zero pulses in each track (a total of 10 non-zero pulses). This algebraic structure of the fixed codebook requires no codebook storage (i.e., no lookup tables for the codebook *vectors*). Each pulse position in one track is encoded with 3 bits and an additional 1 bit is needed to represent the sign of the first pulse. An opposite sign is recognized if the second pulse position is smaller than the first pulse position, while the same sign is determined if the second pulse position is larger. Therefore, a total of 35 bits are used for encoding the fixed (algebraic) codebook index.

Bit Allocation Table

The bit allocation of the 20-ms frame in AMR-NB with 12.2 kbps mode is shown in Tab. 3.2.

3.1.2 AMR Wideband

In 2001, AMR-WB was standardized as 3GPP Technical Specification TS 26.190 with 16 kHz sampling rate and nine operating bit rates ranging from 6.6 kbps to 23.85 kbps [3GPP, 2001b]. Because of the ACELP coding scheme, the encoder and decoder in AMR-WB are similar to those in AMR-NB. Therefore, Figs. 3.1 and 3.2 can also be referenced for the block diagram of the encoder and decoder in AMR-WB. Without too much loss of generality, the investigation in this thesis focuses on the widely-used 12.65 kbps mode,

Parameters	First subframe	Second subframe	Third subframe	Fourth subframe	Total per frame
2 LSP sets					38
Pitch delay	9	6	9	6	30
Adaptive codebook gain	4	4	4	4	16
Fixed codebook index	35	35	35	35	140
Fixed codebook gain	5	5	5	5	20
Total					244

Table 3.2: Bit allocation table of one frame for AMR-NB operating at 12.2 kbps mode [3GPP, 1999a].

with 253 bits in each frame. The AMR-WB encoding and decoding processes in [3GPP, 2001b] are listed below.

AMR-WB Encoding

In the AMR-WB encoder, a block of 320 speech samples (16-bit linear encoded PCM format) corresponding to a 20-ms frame with four subframes is analyzed to extract various parameters: a voice activity detector (VAD) flag and the immittance spectral frequencies (ISFs) for each frame; an LTP filtering flag, the adaptive codebook index (pitch delay), the fixed (algebraic) codebook index, the VQ gain for each subframe. The analysis of these parameters is performed at 12.8 kHz sampling rate, which leads to 64 samples in each subframe.

AMR-WB Decoding

The decoding process is similar to the one in AMR-NB. The reconstructed speech samples are finally upsampled to 16 kHz. In the following, the relevant codec parameters are analyzed.

Impittance Spectral Frequencies (ISFs)

In the frequency domain, a number of 16 LP filter coefficients is first transformed to ISFs for quantization and interpolation purposes. Thereafter, applying a first-order moving-average prediction, the residual ISF vector is quantized by a combination of split vector quantization (SVQ) and multistage vector quantization (MSVQ). The residual ISF vector, which is the difference between the mean-removed ISF vector and the predicted residual vector, is first split into two subvectors of 9 and 7 dimensions, with each subvector being quantized in two stages. Both subvectors are quantized with 8 bits in the first stage. In the second stage, for the operation at 12.65 kbps, the quantization error vectors are split into three subvectors (each with 3 dimensions and 6 bits, 7 bits, and 7 bits, respectively) and two subvectors (3 and 4 dimensions, both with 5 bits), respectively.

Parameters	First subframe	Second subframe	Third subframe	Fourth subframe	Total per frame
VAD flag					1
ISFs					46
LTP-filtering flag	1	1	1	1	4
Pitch delay	9	6	9	6	30
Fixed codebook index	36	36	36	36	144
VQ gain	7	7	7	7	28
Total					253

Table 3.3: Bit allocation table of one frame for AMR-WB operating at 12.65 kbps mode [3GPP, 2001b].

Adaptive Codebook Index (Pitch Delay)

For the AMR-WB 12.65 kbps mode, the pitch delays are encoded with 9 bits and 6 bits for *odd* and *even* subframes, respectively. For the pitch delay in the *odd* (first and third) subframes, a total number of 512 pitch delays include fractional values with an interval of $\frac{1}{4}$ in the range of $[34, 127\frac{3}{4}]$, an interval of $\frac{1}{2}$ in the range of $[128, 159\frac{1}{2}]$, and integers in the range of $[160, 231]$. Just as in AMR-NB, the relative difference to the previous *odd* subframe is 6-bit encoded in the *even* subframes. The pitch delay with an interval of $\frac{1}{2}$ is in the range of $[T_1 - 8, T_1 + 7\frac{1}{2}]$, with T_1 being the nearest integer to the pitch delay of the previous subframe.

Vector-Quantized Codebook Gain (VQ Gain)

The adaptive codebook gain and correction factor are two-dimensional vector-quantized using a 7-bit codebook in each subframe, with the correction factor being the ratio between the true fixed codebook gain and an estimated fixed codebook gain.

Fixed Codebook Index

Similar to AMR-NB, in each subframe, the 64 pulse positions (representing the fixed codebook indices) are predefined and divided into 4 tracks. For the 12.65 kbps mode, the codevectors include 8 non-zero pulses, with 2 pulses in each track. Each pulse position is encoded with 4 bits, which results in a number of 36 bits being used for the fixed (algebraic) codebook index.

Bit Allocation Table

The bit allocation of 20-ms frame for the AMR-WB operating at 12.65 kbps mode is listed in Tab. 3.3.

3.2 Standard Error Concealment in AMR

In 3GPP Technical Specifications TS 26.091 [3GPP, 1999b] and TS 26.191 [3GPP, 2001a], the error concealment techniques are based on a bad frame indicator (BFI), which is used to mark the frame either as bad/lost (BFI=1), or good (BFI=0). Once a bad/lost frame is indicated (BFI=1), a substitution is performed either using repetition or extrapolation of the previous good frames. In this section, the standard error concealment in AMR-NB and AMR-WB from [3GPP, 1999b] and [3GPP, 2001a] is described in brief.

3.2.1 AMR Narrowband

For the error concealment in AMR-NB, the LSFs are based on the values of the previous frame and a (predefined) average LSF vector. The pitch delays are replaced by the values from the fourth subframe of the previous frame. The adaptive and fixed codebook gains are replaced by attenuating the values from the previous subframes. The substitution thereby gradually results in muting. Moreover, the fixed (algebraic) codebook indices are used as they are received even in a bad frame, while they are randomly chosen (in the range of $[-1, +1]$) when no data at all is received.

In addition, if the BFI of the current received frame is set to 0 (error-free) but BFI is equal to 1 for the previous received frame, the adaptive and fixed codebook gains of the current frame are taken from the last good subframe at times.

3.2.2 AMR Wideband

Two frame types: `SPEECH_BAD` for an erroneous frame and `SPEECH_LOST` for a lost frame are distinguished for the error concealment of pitch delays, VQ gains, and fixed (algebraic) codebook indices in AMR-WB. Therefore, if BFI is set to one, different ways of substitution will be applied for the erroneous or lost case. For the `SPEECH_BAD` case, the pitch delays are just taken as the current received at times, while they are taken from the previous good frame at times when `SPEECH_LOST` is set. The fixed (algebraic) codebook indices are used as they are received once `SPEECH_BAD` is set, while being randomly chosen for the `SPEECH_LOST` case.

Moreover, for the case of BFI=0 at the current received frame and BFI=1 at the previous frame, the fixed codebook gains are based on the values from the last good subframe at times. Accordingly, the reliability of a frame (error-free, erroneous, or lost) is distinguished only by the value of the BFI.

3.3 New AMR Parameter Soft-Decision Decoding

Instead of using a BFI, SD decoding adopting bit-wise channel reliability information can offer better speech quality. Utilizing soft information, the performance of AMR-

Parameters	LSF submatrices (index η)					Pitch delays		Adaptive codebook gains	Fixed codebook gains
	1	2	3	4	5	<i>odd</i>	<i>even</i>		
ΔR_d (bits)	1.78	1.14	0.97	0.84	1.34	0.54	0.90	0.34	0.63
ΔR_c (bits)	0.40	0.79	0.89	0.53	0.39	1.39	0.40	0.18	0.59
ΔR (bits)	2.18	1.93	1.86	1.37	1.72	1.93	1.30	0.52	1.22

Table 3.4: Contributions to residual redundancy ΔR of parameters LSFs (interframe redundancy), pitch delays, adaptive and fixed codebook gains in AMR-NB: ΔR_d being the *utilizable, distribution-dependent* residual redundancy (equation (2.10)), and ΔR_c being the *utilizable, correlation-dependent* residual redundancy (equation (2.11)).

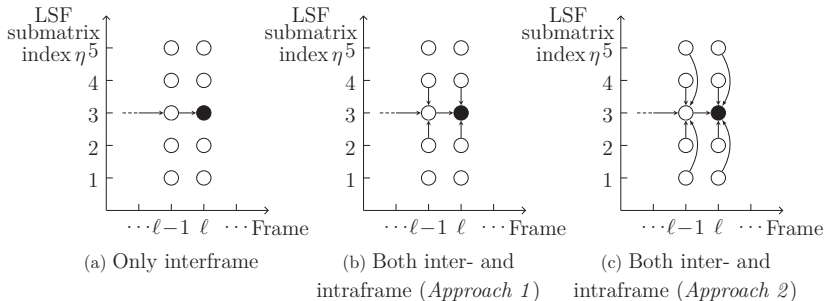


Figure 3.3: Exploiting interframe and intraframe redundancy of LSF submatrices in terms of AK1 *a priori* knowledge.

NB and AMR-WB has been improved in [Lahouti and Khandani, 2007, Othman et al., 2007, Othman et al., 2009]. However, not all important codec parameters have been improved by utilizing soft information. In this section, a more robust AMR-NB or AMR-WB decoding is proposed by applying the SD decoding in Section 2.3 to the relevant codec parameters in AMR-NB and AMR-WB. Additionally, approaches exploiting both interframe and intraframe residual redundancy of the parameters LSFs and ISFs are also proposed and further implemented in the AMR-NB and AMR-WB decoders. Finally, the computational complexity is discussed.

3.3.1 AMR Narrowband

In this section, the application of the SD decoding to the AMR-NB codec parameters LSFs, pitch delays, adaptive and fixed codebook gains is elaborated. Modeling the corresponding quantized parameters as zeroth-order and first-order Markov processes, the residual redundancy based on (2.9) and (2.11) in Section 2.1.3 for each parameter is depicted in Tab. 3.4. Note that the interframe redundancy for LSFs is considered in Tab. 3.4.

LSF submatrices	$\eta = 1$				$\eta = 2$			$\eta = 3$		$\eta = 4$
	$\bar{\eta} = 2$	$\bar{\eta} = 3$	$\bar{\eta} = 4$	$\bar{\eta} = 5$	$\bar{\eta} = 3$	$\bar{\eta} = 4$	$\bar{\eta} = 5$	$\bar{\eta} = 4$	$\bar{\eta} = 5$	$\bar{\eta} = 5$
$\Delta R_{c,\text{intra}}$ (bits)	0.59	0.29	0.19	0.20	0.67	0.38	0.26	0.64	0.29	0.37

Table 3.5: Intraframe residual redundancy between two LSF submatrices η and $\bar{\eta}$.

Line Spectral Frequencies (LSFs)

(1) Exploiting Interframe Residual Redundancy

The 5 LSF submatrices $\text{LSF}_{\ell,\eta}$ at frame ℓ are quantized with 7, 8, 8+1, 8, and 6 bits, with $\eta \in \{1, 2, \dots, 5\}$ being the LSF submatrix index. As shown in Fig. 3.3, the black filled circle ($\eta = 3$) represents the current LSF submatrix index η . A number of 128, 256, 512, 256, and 64 *a posteriori* probabilities (APPs) can be obtained respectively. Performing the APP calculations for all 5 submatrices in parallel, (2.26) and (2.27) are calculated five times in each frame with an effective number of bits $M_\eta \in \{7, 8, 9, 8, 6\}$, respectively. Note that for the submatrix 3 with 8 bits plus one sign bit, a number of 512 (2^9) APPs is computed, adopting a codebook with a number of 512 codebook entries (i.e., 256 positive and their corresponding negative values). Fig. 3.3(a) depicts the exploitation of the interframe residual redundancy of LSF submatrix in terms of first-order *a priori* knowledge (AK1).

(2) Exploiting Inter- and Intraframe Residual Redundancy

Considering the correlation between different LSF submatrices, in analogy to (2.11), not only the interframe but also the intraframe residual redundancy of LSF submatrices can be observed by

$$\Delta R_{c,\text{intra}} = H(\mathbf{x}_{\ell,\eta}) - H(\mathbf{x}_{\ell,\eta} | \mathbf{x}_{\ell,\bar{\eta}}), \quad (3.1)$$

with $\bar{\eta}$ being the other LSF submatrix indices, excluding η . The $\Delta R_{c,\text{intra}}$ results are depicted in Tab. 3.5. Two approaches can be derived for exploiting both inter- and intraframe redundancy.

(2a) Approach 1: Exploiting Interframe Redundancy and Intraframe Redundancy of Only Adjacent LSF Submatrices

The interframe redundancy and the intraframe redundancy of only adjacent LSF submatrices are taken into account in *Approach 1* (see Fig. 3.3(b)). The APPs become a probably transmitted bit combination $\mathbf{x}_{\ell,\eta}^{(i)}$ given the past received bit combinations from frame 1 to ℓ of three LSF submatrices (the current calculated η and two adjacent $\eta - 1, \eta + 1$), and satisfy (see Appendix A)

$$\begin{aligned}
 P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell) &= \frac{1}{C} \cdot P(\hat{\mathbf{x}}_{\ell,\eta} | \mathbf{x}_{\ell,\eta}^{(i)}) \\
 &\cdot \sum_{f=0}^{2^{M_{\eta-1}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell-1,\eta}^{(f)}) \cdot P(\mathbf{x}_{\ell-1,\eta}^{(f)} | \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-1}) \\
 &\cdot \sum_{j=0}^{2^{M_{\eta-1}-1}} P(\hat{\mathbf{x}}_{\ell,\eta-1} | \mathbf{x}_{\ell,\eta-1}^{(j)}) \cdot P(\mathbf{x}_{\ell,\eta-1}^{(j)} | \mathbf{x}_{\ell,\eta}^{(i)}) \\
 &\cdot \sum_{h=0}^{2^{M_{\eta+1}-1}} P(\hat{\mathbf{x}}_{\ell,\eta+1} | \mathbf{x}_{\ell,\eta+1}^{(h)}) \cdot P(\mathbf{x}_{\ell,\eta+1}^{(h)} | \mathbf{x}_{\ell,\eta}^{(i)}), \quad (3.2)
 \end{aligned}$$

with the term $P(\mathbf{x}_{\ell-1,\eta}^{(f)} | \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-1})$ being the APPs from the previous frame $\ell - 1$ for LSF submatrix η and the terms $P(\hat{\mathbf{x}}_{\ell,\eta-1} | \mathbf{x}_{\ell,\eta-1}^{(j)})$, $P(\hat{\mathbf{x}}_{\ell,\eta+1} | \mathbf{x}_{\ell,\eta+1}^{(h)})$ being the transition probabilities calculated from (2.21), and f, j, h being the $M_{\eta-}$, $M_{\eta-}$, and $M_{\eta+}$ -bit quantization indices for LSF submatrix $\eta, \eta - 1, \eta + 1$, respectively.

From Fig. 3.3(b) and (3.2), it can be seen that the interframe redundancy in terms of the *a priori* knowledge $P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell-1,\eta}^{(f)})$ and the intraframe redundancy in terms of the *a priori* knowledge $P(\mathbf{x}_{\ell,\eta}^{(j)} | \mathbf{x}_{\ell,\eta}^{(i)})$, with $\bar{\eta} \in \{\eta - 1, \eta + 1\}$, are exploited in *Approach 1*.

(2b) *Approach 2: Exploiting Interframe Redundancy and Intraframe Redundancy of Both Adjacent and Non-Adjacent LSF Submatrices*

Different to *Approach 1*, the intraframe redundancy of both adjacent and non-adjacent LSF submatrices is considered in *Approach 2*, as shown in Fig. 3.3(c). According to the chain rule, taking $\eta=3$ as an example¹, the APPs representing a possible transmitted bit combination $\mathbf{x}_{\ell,\eta}^{(i)}$ given the past received bit combinations $(\hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell)$ from frame 1 to ℓ of all the five LSF submatrices can be calculated by (see Appendix A)

$$\begin{aligned}
 P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell) &= \frac{1}{P(\mathbf{x}_{\ell,\eta}^{(i)})^4} \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell) \\
 &\cdot \sum_{j=0}^{2^{M_{\eta-2}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta-2}^{(j)}) \cdot P(\mathbf{x}_{\ell,\eta-2}^{(j)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell) \\
 &\cdot \sum_{f=0}^{2^{M_{\eta-1}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta-1}^{(f)}) \cdot P(\mathbf{x}_{\ell,\eta-1}^{(f)} | \hat{\mathbf{x}}_{1,\eta-1}^\ell) \\
 &\cdot \sum_{h=0}^{2^{M_{\eta+1}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta+1}^{(h)}) \cdot P(\mathbf{x}_{\ell,\eta+1}^{(h)} | \hat{\mathbf{x}}_{1,\eta+1}^\ell) \\
 &\cdot \sum_{r=0}^{2^{M_{\eta+2}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta+2}^{(r)}) \cdot P(\mathbf{x}_{\ell,\eta+2}^{(r)} | \hat{\mathbf{x}}_{1,\eta+2}^\ell). \quad (3.3)
 \end{aligned}$$

Note that the term $P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell)$ with $\eta \in \{1, 2, 3, 4, 5\}$ is actually the APP calculation (2.27) only using interframe redundancy. From Fig. 3.3(c) and (3.3), it can be seen that

¹Note that the LSF submatrix index $\eta = 3$ is taken as an example to describe the equations in this thesis, the other values of η can be easily derived.

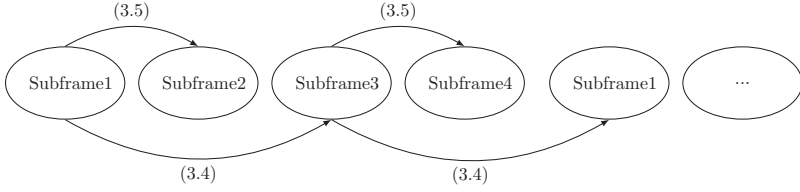


Figure 3.4: The process of the APP calculation for the pitch delay.

the interframe redundancy in terms of the *a priori* knowledge $P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell-1,\eta}^{(j)})$ for all five LSF submatrices and the intraframe redundancy in terms of the *a priori* knowledge $P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\bar{\eta}}^{(j)})$ is exploited in *Approach 2*, with $\bar{\eta} \in \{\eta - 2, \eta - 1, \eta + 1, \eta + 2\}$.

Finally, based on (2.26), (2.27), (3.2), or (3.3), each LSF submatrix can be estimated by an minimum mean-square error (MMSE) estimator (2.32), with the respective M_η -bit quantization codebook.

Adaptive Codebook Index (Pitch Delay)

The pitch delay requires 512 and 64 APPs for 9 bit and 6 bit coded *odd* and *even* subframes, respectively. The relative difference to the previous *odd* subframe is actually encoded in the *even* subframe. Therefore, for the left-hand-side APP calculation with AK1 in (2.27) of both *odd* and *even* subframes, the right-hand-side past APP term in (2.27) should always be taken from the previous *odd* subframe. Similar to the work in [Han et al., 2013], Fig. 3.4 shows the APP calculation of pitch delay for *odd* subframes and *even* subframes. The APP calculation of *odd* subframes ($\lambda = 1, 3, 5, \dots$) can be performed by (recursion as (2.27)!):

$$P(\mathbf{x}_\lambda^{(i)} | \hat{\mathbf{x}}_\lambda, \hat{\mathbf{x}}_1^{\lambda-2}) = \frac{1}{C} \cdot P(\hat{\mathbf{x}}_\lambda | \mathbf{x}_\lambda^{(i)}) \cdot \sum_{j=0}^{511} P(\mathbf{x}_\lambda^{(i)} | \mathbf{x}_{\lambda-2}^{(j)}) \cdot P(\mathbf{x}_{\lambda-2}^{(j)} | \hat{\mathbf{x}}_{\lambda-2}, \hat{\mathbf{x}}_1^{\lambda-4}), \quad (3.4)$$

with $\hat{\mathbf{x}}_\lambda$ being the current received bit combination of the *odd* subframe at global subframe index λ and $i \in \{0, 1, \dots, 511\}$.

Differently, the APP calculation of *even* subframes ($\lambda = 2, 4, 6, \dots$) (no recursion!) turns out to be:

$$P(\mathbf{y}_\lambda^{(i)} | \hat{\mathbf{y}}_\lambda, \hat{\mathbf{x}}_1^{\lambda-1}) = \frac{1}{C} \cdot P(\hat{\mathbf{y}}_\lambda | \mathbf{y}_\lambda^{(i)}) \cdot \sum_{j=0}^{511} P(\mathbf{y}_\lambda^{(i)} | \mathbf{x}_{\lambda-1}^{(j)}) \cdot P(\mathbf{x}_{\lambda-1}^{(j)} | \hat{\mathbf{x}}_{\lambda-1}, \hat{\mathbf{x}}_1^{\lambda-3}), \quad (3.5)$$

with $i \in \{0, 1, \dots, 63\}$, $\hat{\mathbf{y}}_\lambda$ being the current received bit combination of the *even* subframe at the global subframe index λ , and $\hat{\mathbf{x}}_{\lambda-1}$ being the bit combination of the previous *odd* subframe at the global subframe index $\lambda - 1$. Note that the right-hand-side APP term in (3.5) is exactly the same as the left-hand-side APP term in (3.4) at the last subframe $\lambda - 1$.

Accordingly, the training process is different for the *odd* and *even* subframes. As shown in Fig. 3.5, the AK1 term $P(\mathbf{x}_\ell^{(i)} | \mathbf{x}_{\ell-1}^{(j)})$ should be obtained separately for *odd* and

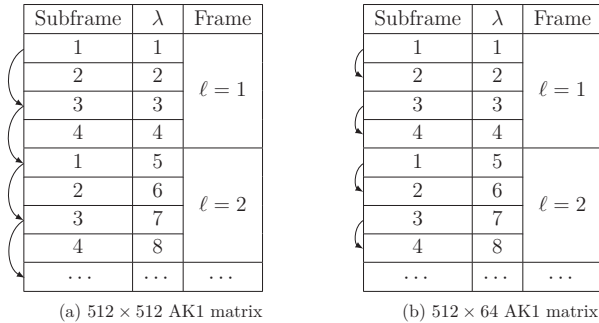


Figure 3.5: Training process of the *a priori* knowledge for the pitch delay: Generation of AK1 matrix for *odd* (a) and *even* (b) subframes [Han et al., 2013].

even subframes: For the calculation of the 512×512 AK1 matrix for *odd* subframes, the *odd* subframes are counted in pairs ($\lambda = 1$ and $\lambda = 3$; $\lambda = 3$ and $\lambda = 5$; ...), while the *even* subframes are counted in pairs ($\lambda = 1$ and $\lambda = 2$; $\lambda = 3$ and $\lambda = 4$; ...) for the calculation of the 512×64 AK1 matrix for *even* subframes.

In addition, the residual redundancy for the *even* subframes can be obtained by $\Delta R_c = H(\mathbf{y}_\lambda) - H(\mathbf{y}_\lambda | \mathbf{x}_{\lambda-1})$, with

$$H(\mathbf{y}_\lambda | \mathbf{x}_{\lambda-1}) = - \sum_{i=0}^{63} \sum_{j=0}^{511} P(\mathbf{y}_\lambda^{(i)}, \mathbf{x}_{\lambda-1}^{(j)}) \cdot \log_2 P(\mathbf{y}_\lambda^{(i)} | \mathbf{x}_{\lambda-1}^{(j)}). \quad (3.6)$$

As can be seen in Tab.3.4, ΔR_c for *even* subframes is very small due to the relative coding of the delay. It is found that the maximum *a posteriori* (MAP) estimator (2.33) is appropriate for the estimation of pitch delays.

Adaptive Codebook and Fixed Codebook Gains

For the 5-bit quantized correction factor of the fixed codebook gain and the 4-bit quantized adaptive codebook gain, a number of 32 and 16 APPs from (2.26) or (2.27) is required, respectively. The received correction factor and fixed codebook gain can be estimated by an MMSE estimator from (2.32), with $M = 5$ and $M = 4$, respectively.

Complexity Evaluation

The computational complexity of APP calculations (2.26), (2.27), (3.2), and (3.3) for a specific LSF submatrix η is discussed in this section, since the other parameters share the same APP calculation equations (2.26) and (2.27), but just replace the quantization bit rate M . Considering the multiplications (without the multiplication with the normalization constant $\frac{1}{c}$), the number of necessary arithmetic operations², the ROM³ requirements

²As in (3.2) and (3.3), $\eta = 3$ is taken as an example here, the other values can be obtained straightforwardly.

³read-only memory

Equation	Arithmetic operations	ROM requirements (words)	RAM requirements (words)
(2.26)	2^{M_η}	2^{M_η}	2^{M_η}
(2.27)	$2^{M_\eta} \cdot (2^{M_\eta} + 1)$	2^{2M_η}	$2 \cdot 2^{M_\eta}$
(3.2)	$2^{M_\eta} \cdot (2^{M_\eta} + 2^{M_{\eta-1}} + 2^{M_{\eta+1}} + 3)$	$2^{M_\eta} \cdot (2^{2M_{\eta-1}} + 2^{M_\eta} + 2^{2M_{\eta+1}})$	$2^{M_\eta} + 2^{M_{\eta-1}} + 2^{M_\eta} + 2^{M_{\eta+1}}$
(3.3)	$2^{M_\eta} \cdot (2^{M_\eta} + 1 + 2^{M_{\eta-2}} \cdot (2^{M_{\eta-2}} + 1 + 1) + 2^{M_{\eta-1}} \cdot (2^{M_{\eta-1}} + 1 + 1) + 2^{M_{\eta+1}} \cdot (2^{M_{\eta+1}} + 1 + 1) + 2^{M_{\eta+2}} \cdot (2^{M_{\eta+2}} + 1 + 1) + 5)$	$2^{M_\eta} + 2^{2M_{\eta-2}} + 2^{2M_{\eta-1}} + 2^{2M_{\eta+1}} + 2^{2M_{\eta+2}} + 2^{M_\eta} \cdot (2^{2M_{\eta-2}} + 2^{2M_{\eta-1}} + 2^{2M_{\eta+1}} + 2^{2M_{\eta+2}})$	$2 \cdot (2^{M_{\eta-2}} + 2^{M_{\eta-1}} + 2^{M_\eta} + 2^{M_{\eta+1}} + 2^{M_{\eta+2}})$

Table 3.6: The number of the arithmetic operations, ROM requirements, and RAM requirements in AMR-NB.

(for the *a priori* probabilities), and the RAM⁴ requirements are listed in Tab. 3.6.

Note that the number of arithmetic operations in (3.3) also considers the APP calculation based on (2.27), which finally results in a number of $2^{M_\eta} \cdot (2^{M_\eta} + 2 + (2^{M_{\eta-2}} + 1)^2 + (2^{M_{\eta-1}} + 1)^2 + (2^{M_{\eta+1}} + 1)^2 + (2^{M_{\eta+2}} + 1)^2)$ arithmetic operations. The AMR-NB codec with a frame length of 20 ms results in a number of 50 frames per second. Therefore, given a specific LSF submatrix $\eta = 3$ with $M_\eta = 9$ bit, a computing power of about 0.26, 13, 26, and 3929 MOPS⁵ is needed for (2.26), (2.27), (3.2), and (3.3), respectively.

Moreover, in (2.27), both the recursive process and the channel transition probabilities need a number of 2^{M_η} words for the RAM. For the APP calculation with (3.2), a number of 2^{M_η} and $2^{M_{\eta-1}} + 2^{M_\eta} + 2^{M_{\eta+1}}$ words is required for the recursion and transition probabilities, respectively. Due to the APP calculation (2.27) being included in (3.3), apart from a number of $2^{M_{\eta-2}} + 2^{M_{\eta-1}} + 2^{M_\eta} + 2^{M_{\eta+1}} + 2^{M_{\eta+2}}$ words being required for the recursion in (3.3), the same number of words is also desired for the transition probabilities.

It can be concluded that the complexity cost of *Approach 2* with (3.3) is undoubtedly much higher than the other approaches. In contrast, the complexity of exploiting the inter- and intraframe redundancy with *Approach 1* is not that noticeable.

3.3.2 AMR Wideband

The application of SD decoding to the AMR-WB speech decoder is very similar to AMR-NB. The residual redundancy ΔR_d , ΔR_c , and ΔR based on (2.9) and (2.11) for the codec parameters ISFs, pitch delays, and VQ gains is shown in Tab. 3.7. Herein, only the interframe redundancy of ISFs is considered.

⁴random-access memory

⁵million operations per second

3. Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

Parameters	ISF subvectors							Pitch delays		VQ gains
	1	2	3	4	5	6	7	<i>odd</i>	<i>even</i>	
ΔR_d (bits)	1.04	0.80	0.21	0.16	0.21	0.08	0.11	0.24	0.23	1.36
ΔR_c (bits)	1.66	1.10	0.13	0.06	0.07	0.03	0.02	1.27	0.21	1.03
ΔR (bits)	2.70	1.90	0.34	0.22	0.28	0.11	0.13	1.51	0.44	2.39

Table 3.7: Residual redundancy of parameters ISFs (interframe redundancy), pitch delays, and VQ gains in AMR-WB.

ISF subvectors	$\eta' = 1$						$\eta' = 2$				
	$\bar{\eta}' = 2$	$\bar{\eta}' = 3$	$\bar{\eta}' = 4$	$\bar{\eta}' = 5$	$\bar{\eta}' = 6$	$\bar{\eta}' = 7$	$\bar{\eta}' = 3$	$\bar{\eta}' = 4$	$\bar{\eta}' = 5$	$\bar{\eta}' = 6$	$\bar{\eta}' = 7$
$\Delta R_{c,\text{intra}}$ (bits)	1.08	0.51	0.36	0.38	0.12	0.10	0.15	0.11	0.16	0.31	0.24

(a) $\eta' \in \{1, 2\}$

ISF subvectors	$\eta' = 3$				$\eta' = 4$			$\eta' = 5$		$\eta' = 6$
	$\bar{\eta}' = 4$	$\bar{\eta}' = 5$	$\bar{\eta}' = 6$	$\bar{\eta}' = 7$	$\bar{\eta}' = 5$	$\bar{\eta}' = 6$	$\bar{\eta}' = 7$	$\bar{\eta}' = 6$	$\bar{\eta}' = 7$	$\bar{\eta}' = 7$
$\Delta R_{c,\text{intra}}$ (bits)	0.06	0.04	0.02	0.02	0.08	0.01	0.01	0.03	0.02	0.02

(b) $\eta' \in \{3, 4, 5, 6\}$

Table 3.8: Intraframe residual redundancy between two ISF subvectors η' and $\bar{\eta}'$.

Immittance Spectral Frequencies (ISFs)

(1) Exploiting Interframe Residual Redundancy

The 7 ISF subvectors with the quantization bit rate being 8, 8, 6, 7, 7, 5, 5 lead to a number of 256, 256, 64, 128, 128, 32, 32 APPs, respectively. As mentioned in Section 3.1.2, due to the SVQ and MSVQ adopted for the quantization of ISFs, only the first two ISF subvectors have relatively high redundancy (see Tab. 3.7). Performing the APP calculations for all seven subvectors in parallel, (2.26) or (2.27) is calculated seven times in each frame [Han et al., 2013], with $M_{\eta'} \in \{8, 8, 6, 7, 7, 5, 5\}$ and $\eta' \in \{1, 2, \dots, 7\}$ being the ISF subvector index.

(2) Exploiting Inter- and Intraframe Residual Redundancy

Defining the subvector indices $\bar{\eta}'$ excluding η' , and replacing η by η' , $\bar{\eta}$ by $\bar{\eta}'$ as well in (3.1), the intraframe residual redundancy between two ISF subvectors η' and $\bar{\eta}'$ can be computed accordingly. Due to the SVQ and MSVQ schemes adopted in the ISF quantization, it is found that most intraframe residual redundancy $\Delta R_{c,\text{intra}}$ can be exploited between ISF subvectors $\eta' = 1$ and $\bar{\eta}' \in \{2, 3, 4, 5\}$; subvectors $\eta' = 2$ and $\bar{\eta}' \in \{6, 7\}$. Those redundancies are marked as bold in Tab. 3.8, while the term $\Delta R_{c,\text{intra}}$ of all the other combinations is very small (in a range of 0.01 to 0.16). Correspondingly, different to the *Approach 1* in AMR-NB, instead of exploiting the residual redundancy between adjacent ISF subvectors, the intraframe redundancy in terms of the *a priori* knowledge

$P(\mathbf{x}_{\ell, \bar{\eta}'}^{(j)} | \mathbf{x}_{\ell, \eta'}^{(i)})$ is exploited, with $\bar{\eta}'$ and η' satisfying

$$\begin{cases} \bar{\eta}' = 2, 3, 4, 5, & \text{if } \eta' = 1, \\ \bar{\eta}' = 1, 6, 7, & \text{if } \eta' = 2, \\ \bar{\eta}' = 1, & \text{if } \eta' = 3 \text{ or } \eta' = 4 \text{ or } \eta' = 5, \\ \bar{\eta}' = 2, & \text{if } \eta' = 6 \text{ or } \eta' = 7. \end{cases} \quad (3.7)$$

Furthermore, on the basis of (3.2) and taking $\eta' = 2$ and $\eta' = 3$ as examples⁶, the APP calculation turns out to be

$$\begin{aligned} P(\mathbf{x}_{\ell, 2}^{(i)} | \hat{\mathbf{x}}_{1,2}^{\ell}, \hat{\mathbf{x}}_{1,1}^{\ell}, \hat{\mathbf{x}}_{1,6}^{\ell}, \hat{\mathbf{x}}_{1,7}^{\ell}) &= \frac{1}{C} \cdot P(\hat{\mathbf{x}}_{\ell, 2} | \mathbf{x}_{\ell, 2}^{(i)}) \\ &\cdot \sum_{f=0}^{2^{M_2-1}} P(\mathbf{x}_{\ell, 2}^{(f)} | \mathbf{x}_{\ell-1, 2}^{(f)}) \cdot P(\mathbf{x}_{\ell-1, 2}^{(f)} | \hat{\mathbf{x}}_{1,2}^{\ell-1}, \hat{\mathbf{x}}_{1,1}^{\ell-1}, \hat{\mathbf{x}}_{1,6}^{\ell-1}, \hat{\mathbf{x}}_{1,7}^{\ell-1}) \\ &\cdot \sum_{j=0}^{2^{M_1-1}} P(\hat{\mathbf{x}}_{\ell, 1} | \mathbf{x}_{\ell, 1}^{(j)}) \cdot P(\mathbf{x}_{\ell, 1}^{(j)} | \mathbf{x}_{\ell, 2}^{(i)}) \\ &\cdot \sum_{h=0}^{2^{M_6-1}} P(\hat{\mathbf{x}}_{\ell, 6} | \mathbf{x}_{\ell, 6}^{(h)}) \cdot P(\mathbf{x}_{\ell, 6}^{(h)} | \mathbf{x}_{\ell, 2}^{(i)}) \\ &\cdot \sum_{r=0}^{2^{M_7-1}} P(\hat{\mathbf{x}}_{\ell, 7} | \mathbf{x}_{\ell, 7}^{(r)}) \cdot P(\mathbf{x}_{\ell, 7}^{(r)} | \mathbf{x}_{\ell, 2}^{(i)}), \end{aligned} \quad (3.8)$$

and

$$\begin{aligned} P(\mathbf{x}_{\ell, 3}^{(i)} | \hat{\mathbf{x}}_{1,3}^{\ell}, \hat{\mathbf{x}}_{1,1}^{\ell}) &= \frac{1}{C} \cdot P(\hat{\mathbf{x}}_{\ell, 3} | \mathbf{x}_{\ell, 3}^{(i)}) \\ &\cdot \sum_{f=0}^{2^{M_3-1}} P(\mathbf{x}_{\ell, 3}^{(f)} | \mathbf{x}_{\ell-1, 3}^{(f)}) \cdot P(\mathbf{x}_{\ell-1, 3}^{(f)} | \hat{\mathbf{x}}_{1,3}^{\ell-1}, \hat{\mathbf{x}}_{1,1}^{\ell-1}) \\ &\cdot \sum_{j=0}^{2^{M_1-1}} P(\hat{\mathbf{x}}_{\ell, 1} | \mathbf{x}_{\ell, 1}^{(j)}) \cdot P(\mathbf{x}_{\ell, 1}^{(j)} | \mathbf{x}_{\ell, 3}^{(i)}). \end{aligned} \quad (3.9)$$

Finally, the MMSE estimation (2.32) is found as the appropriate estimation method for a certain ISF subvector with the corresponding $M_{\eta'}$ being M [Han et al., 2013].

Adaptive Codebook Index (Pitch Delay)

The application of SD decoding to the pitch delay in AMR-WB is performed in the same way as in AMR-NB, as detailed in Section 3.3.1 [Han et al., 2013].

Vector-Quantized Codebook Gain (VQ Gain)

The 7-bit encoded VQ gain leads to a number of 128 APPs. Compared to using an MAP estimator, the received VQ gain comprising adaptive codebook gain and the correction

⁶The APPs for the other η' values are $P(\mathbf{x}_{\ell, 1}^{(i)} | \hat{\mathbf{x}}_{1,1}^{\ell}, \hat{\mathbf{x}}_{1,2}^{\ell}, \hat{\mathbf{x}}_{1,3}^{\ell}, \hat{\mathbf{x}}_{1,4}^{\ell}, \hat{\mathbf{x}}_{1,5}^{\ell})$, $P(\mathbf{x}_{\ell, 4}^{(i)} | \hat{\mathbf{x}}_{1,4}^{\ell}, \hat{\mathbf{x}}_{1,1}^{\ell})$, $P(\mathbf{x}_{\ell, 5}^{(i)} | \hat{\mathbf{x}}_{1,5}^{\ell}, \hat{\mathbf{x}}_{1,1}^{\ell})$, $P(\mathbf{x}_{\ell, 6}^{(i)} | \hat{\mathbf{x}}_{1,6}^{\ell}, \hat{\mathbf{x}}_{1,2}^{\ell})$, $P(\mathbf{x}_{\ell, 7}^{(i)} | \hat{\mathbf{x}}_{1,7}^{\ell}, \hat{\mathbf{x}}_{1,2}^{\ell})$, and can be derived straightforwardly.

factor can be better estimated by an MMSE estimator (2.32) [Han et al., 2013], with $M = 7$.

Other Parameters

Due to the random nature of the fixed excitation, a sufficient amount of residual parameter redundancy cannot be exhibited, in other words, the zeroth-order *a priori* knowledge (AK0) and AK1 *a priori* knowledge is very likely to be uniform. Moreover, as mentioned in [Othman et al., 2005], the bits of the fixed codebook indices are not very much sensitive to channel errors, due to the algebraic codebook structure. If one of the fixed codebook indices is received with error, only one of the 10 (AMR-NB) or 8 (AMR-WB) non-zero pulse positions is distorted. Therefore, the fixed codebook indices are simply hard-decision decoded as in [3GPP, 1999a] and [3GPP, 2001b].

Moreover, the VAD flag in each frame and the LTP filtering flag in each subframe in AMR-WB are not sensitive to errors. As a result, the SD decoding approach is not applied to these parameters. However, the MAP estimator (2.33) would be an appropriate choice in case estimation is required [Han et al., 2013].

In addition, the application of SD decoding to the other modes can be implemented straightforwardly, only adapting the summation length in the APP calculations and estimation to the actual quantization bit rate of the parameters.

Complexity Evaluation

Similar to AMR-NB, the complexity for a specific ISF subvector with index η' is investigated here. Replacing M_η by $M_{\eta'}$ in Tab.3.6, the APP calculation with AK0 (2.26) and AK1 considering interframe redundancy (2.27) is the same as in AMR-NB. As a result, only the complexity of APPs with inter- and intraframe redundancy is presented here. Considering that the residual redundancy is exploited between ISF subvectors 1 and 2, 3, 4, 5, subvectors 2 and 6, 7, and taking $M_{\eta'} \in \{8, 8, 6, 7, 7, 5, 5\}$ into account as well, in analogy to (3.8), the complexity for $\eta' = 1$ which has the most complicated equation is depicted here, the others can be derived easily. The number of arithmetical operations reaches $2^{M_1} \cdot (2^{M_1} + 2^{M_2} + 2^{M_3} + 2^{M_4} + 2^{M_5} + 5)$, and results in 10 MOPS for the 20-ms frame with AMR-WB, compared to about 3 MOPS for using only the interframe redundancy (2.27). The *a priori* probabilities require a ROM with the size of $2^{M_1} \cdot (2^{M_1} + 2^{M_2} + 2^{M_3} + 2^{M_4} + 2^{M_5})$ words. For the recursion and transition probabilities, a RAM with 2^{M_1} and $2^{M_1} + 2^{M_2} + 2^{M_3} + 2^{M_4} + 2^{M_5}$ words is needed, respectively.

3.4 Simulation Setup and Results

The performance of using the standard HD decoder and the proposed SD decoder is compared in this section, both for AMR-NB and AMR-WB. The simulations include two parts: distorting a single parameter and impairing the bits of all the parameters.

3.4.1 Simulation Setup

The NTT monaural speech database [NTT, 1994] with 16 kHz sampling rate has been used for training and testing of AMR-WB in the simulations. The speech signals downsampled⁷ from 16 kHz to 8 kHz with a low-pass finite impulse response (FIR) filter are chosen to be the speech database for AMR-NB. A number of 1920 speech signals including 20 languages (except British English) is used for training, each with 96 speech files spoken by 4 male and 4 female speakers. For testing, 96 English speech signals spoken by 4 male and 4 female British native speakers, each with a length of 8 s are used. All signals have been normalized to -26 dBov.

In order to evaluate the performance, the perceptual evaluation of speech quality (PESQ) (P.862) [ITU-T, 2001] and its wideband extension (WB PESQ) (P.862.2) [ITU-T, 2007] are adopted as instrumental measures for AMR-NB and AMR-WB, respectively. Moreover, the successor of PESQ, the perceptual objective listening quality assessment (POLQA) [ITU-T, 2011] has also been used for performance evaluation. For POLQA, the modes **narrowband e1**⁸ and **super wideband** have been selected for AMR-NB and AMR-WB, respectively. A higher mean opinion score (MOS) of PESQ, WB PESQ, and POLQA implies a better speech quality.

For both AMR-NB and AMR-WB, the Gilbert-Elliott (GE) channel model along with additive white Gaussian noise (AWGN) and binary phase-shift keying (BPSK) is employed in the simulations. The term $E_b/N_0|_{\text{bad}}$ for the bad channel state indicating bad frames is varied from -5 dB to 10 dB. In addition, for error concealment (EC), the good frames (good GE channel state) and bad frames (bad GE channel state) are easily distinguished by the header information in each frame in the bit stream.

On the one hand, the GE channel with $P=0.11, Q=0.89$ is adopted to investigate the performance of distorting each individual parameter, with $E_b/N_0|_{\text{good}} \rightarrow \infty$ (error-free) for the good channel state indicating good frames. In that case, only one specific parameter is distorted by AWGN, while the other parameters are error-free. Correspondingly, for the standard error concealment, the BFI triggers the error concealment only for that specific parameter. All other parameters remain correctly received.

On the other hand, for the experiments distorting all bits, $E_b/N_0|_{\text{good}} = 10$ dB is assumed. The percentage of bad frames P_{bad} and the average error burst lengths in the GE channel model with the used different GE parameters (P, Q) are listed in Tab. 3.9.

3.4.2 Simulation Results

In this section, the traditional HD decoding is denoted as HD, while the standard error concealment is represented by EC. The SD decoding for the corresponding parameters includes four different options:

⁷ITU-T software tools [ITU-T, 2005], `filter.exe`, version 3.3, command line option: `"filter - down - qHQ2"`.

⁸`e1` stands for electrical recordings.

GE parameters	P_{bad}	P_{good}	Average error burst length (# of frames)
$P=0.11, Q=0.89$	10%	90%	1
$P=0.62, Q=0.38$	50%	50%	1.6
$P=0.9, Q=0.1$	82%	18%	5.1

Table 3.9: The percentage of bad and good frames, and the average error burst length in a Gilbert-Elliott channel with different GE parameters.

- AK0 implies the employment of the zeroth-order *a priori* knowledge, for both AMR-NB and AMR-WB;
- AK1_Inter represents the utilization of the first-order *a priori* knowledge, with the interframe redundancy being exploited for LSFs in AMR-NB or ISFs in AMR-WB;
- Based on AK1_Inter, AK1_Inter+Intra (Apr.1) and AK1_Inter+Intra (Apr.2) denote exploitation of both inter- and intraframe redundancy of LSF submatrices in AMR-NB, with Apr.1 and Apr.2 referring to *Approach 1* and *Approach 2* as proposed in Section 3.3.1;
- Based on AK1_Inter, AK1_Inter+Intra indicates the approach proposed in Section 3.3.2 exploiting both inter- and intraframe redundancy of ISF subvectors in AMR-WB.

AMR Narrowband

(1) *Single Distorted Parameter*

The MOS results of PESQ and POLQA for separately distorted individual parameters (adaptive codebook gain, fixed codebook gain, pitch delay, and LSFs) in AMR-NB are shown in Fig. 3.6⁹. Note that AK1_Inter+Intra (Apr.1) and AK1_Inter+Intra (Apr.2) are only valid for LSFs. Compared to HD, the superiority of SD can be clearly observed for all the channel conditions and all the parameters. Comparing HD for four parameters in Fig. 3.6, it can be stated that the LSFs are most sensitive to bit errors. In contrast, the adaptive code book gain is more robust to bit errors, along with the smallest redundancy ($\Delta R = 0.52$ in Tab. 3.4), only a slight improvement can be observed by using SD decoding.

Comparing HD with EC, it can be seen that error concealment outperforms HD decoding only if $E_b/N_0|_{\text{bad}} < 1$ dB for the fixed codebook gain, the pitch delay, and the LSFs, while for the adaptive codebook gain HD is always better than EC in the investigated range of $E_b/N_0|_{\text{bad}} \in [-5, 10]$ dB. The reason is explained in the following subsection (2).

For LSFs, compared to AK1_Inter, the speech quality can be further improved by employing both inter- and intraframe redundancy of LSF submatrices, the MOS gap between AK1_Inter+Intra and AK1_Inter is up to 0.14 and 0.17 points for PESQ and POLQA, respectively. The maximum MOS difference between *Approach 2* and *Approach 1* is only 0.05 points, because most intraframe redundancy exists between the adjacent LSF

⁹In this thesis, the simulation results depicted with figures are shown in tabular forms in Appendix E, minor differences between the results can be observed there.

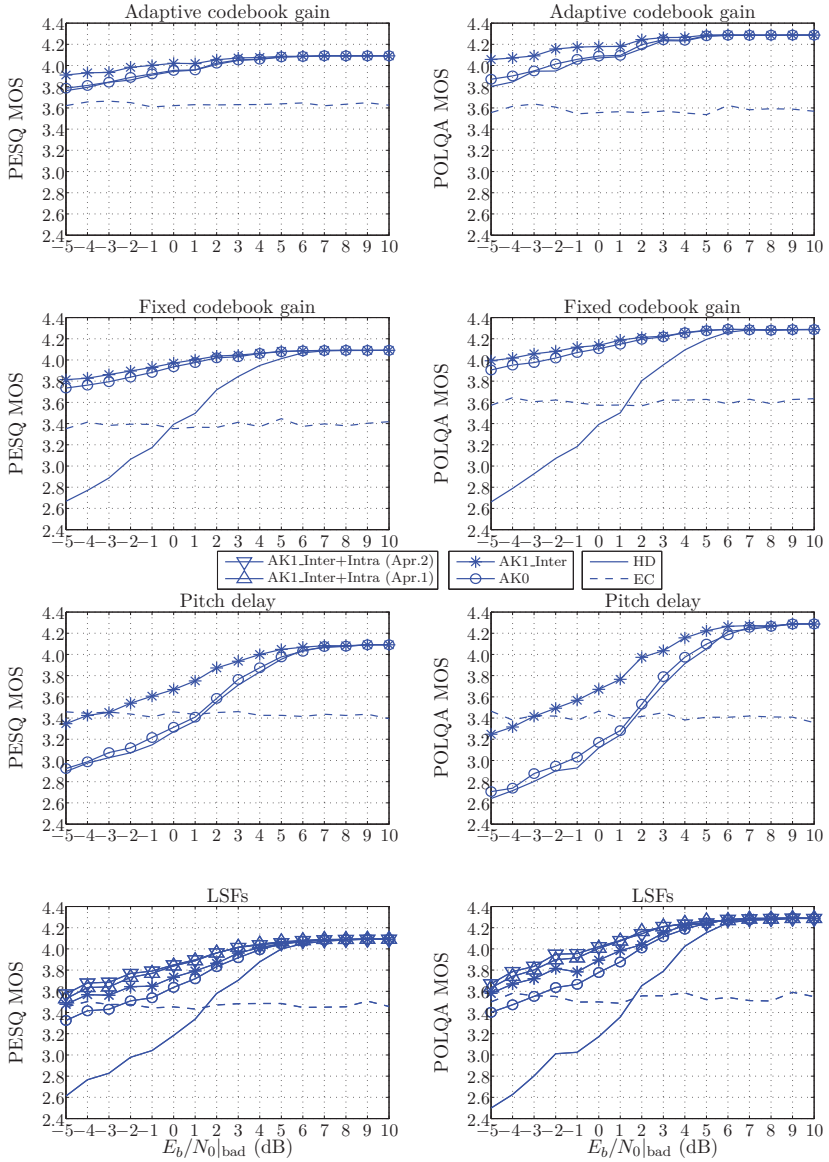


Figure 3.6: PESQ and POLQA MOS results for AMR-NB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} \rightarrow \infty$, $\mathbf{P}=0.11$, $\mathbf{Q}=0.89$. Only a single parameter is individually corrupted (see Tabs. E.1 and E.2).

AMR-NB parameters	PESQ		POLQA	
	MOS gain	$E_b/N_0 _{\text{bad}}$ gain	MOS gain	$E_b/N_0 _{\text{bad}}$ gain
Adaptive codebook gain	0.15	4 dB	0.26	4.1 dB
Fixed codebook gain	0.58	7.9 dB	0.61	8.1 dB
Pitch delay	0.32	4 dB	0.49	4 dB
LSFs	0.46	7.1 dB	0.60	7.2 dB

Table 3.10: The largest MOS gains and $E_b/N_0|_{\text{bad}}$ gains of any of the SD decoding methods vs. the best of HD and EC for AMR-NB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} \rightarrow \infty$, $P=0.11$, $Q=0.89$. Each single parameter is individually corrupted.

submatrices, as listed in Tab. 3.5.

For each parameter, the largest MOS gain and $E_b/N_0|_{\text{bad}}$ gain of any of the SD decoding methods vs. the best of HD and EC can be easily seen in Tab. 3.10. Comparing the four parameters, the best improvement is achieved by the fixed codebook gain, with 0.61 MOS gain and 8.1 dB $E_b/N_0|_{\text{bad}}$ gain.

(2) *All Bits Distorted*

The PESQ MOS and POLQA MOS results for AMR-NB transmission over three different GE channels with $P=0.9$, $P=0.62$, and $P=0.11$ are shown in Fig. 3.7. First the MOS scores show an ascending tendency with the decrease of P , due to a smaller P_{bad} . Supported by informal listening tests, a remarkable improvement can be observed by using SD decoding from all cases. For error-free transmission conditions, the MMSE or MAP estimation in SD decoding leads to the correctly estimated value automatically and results in the same MOS score (4.09 for PESQ and 4.29 for POLQA) as in HD decoding. Moreover, using AK1_Inter+Intra, the MOS score is increased by up to 0.13 points for PESQ and 0.11 points for POLQA. Comparing the AK1_Inter+Intra (Apr.1) with AK1_Inter+Intra (Apr.2), the MOS score is again only slightly increased (up to 0.03 for PESQ and 0.04 for POLQA) by using *Approach 2*.

In addition, given a certain GE channel model, a similar performance is observed using the reference error concealment for all the channel conditions¹⁰. The optimal switch between BFI=1 and BFI=0 is $E_b/N_0|_{\text{bad}}=-1$ dB, -1 dB, and 2 dB for $P=0.9$, $P=0.62$, and $P=0.11$, respectively. Regardless of channel conditions, the error concealment performance increases along with the decrease of the value of P . The reason is that compared to the GE channel with a smaller value of P which leads to a smaller P_{bad} (i.e., fewer percentages of bad frames), more percentages of bad frames are replaced for a GE channel with a higher value of P . Moreover, when channel conditions are good, the performance of error concealment marking the frame as bad is very poor, even if only a few bit errors occur in the frame. As a result, the disadvantage of error concealment can be clearly concluded.

¹⁰The performance of the error concealment with $P=0.9, Q=0.1$ is too bad to satisfy the speech material requirement of POLQA (a typical speech burst duration being 1 to 3s and at least 3s active speech [HEAD acoustics GmbH, 2012]), the POLQA calculation is unsuccessful at times. As a result, only PESQ is used to measure the error concealment performance for GE channel with $P=0.9, Q=0.1$.

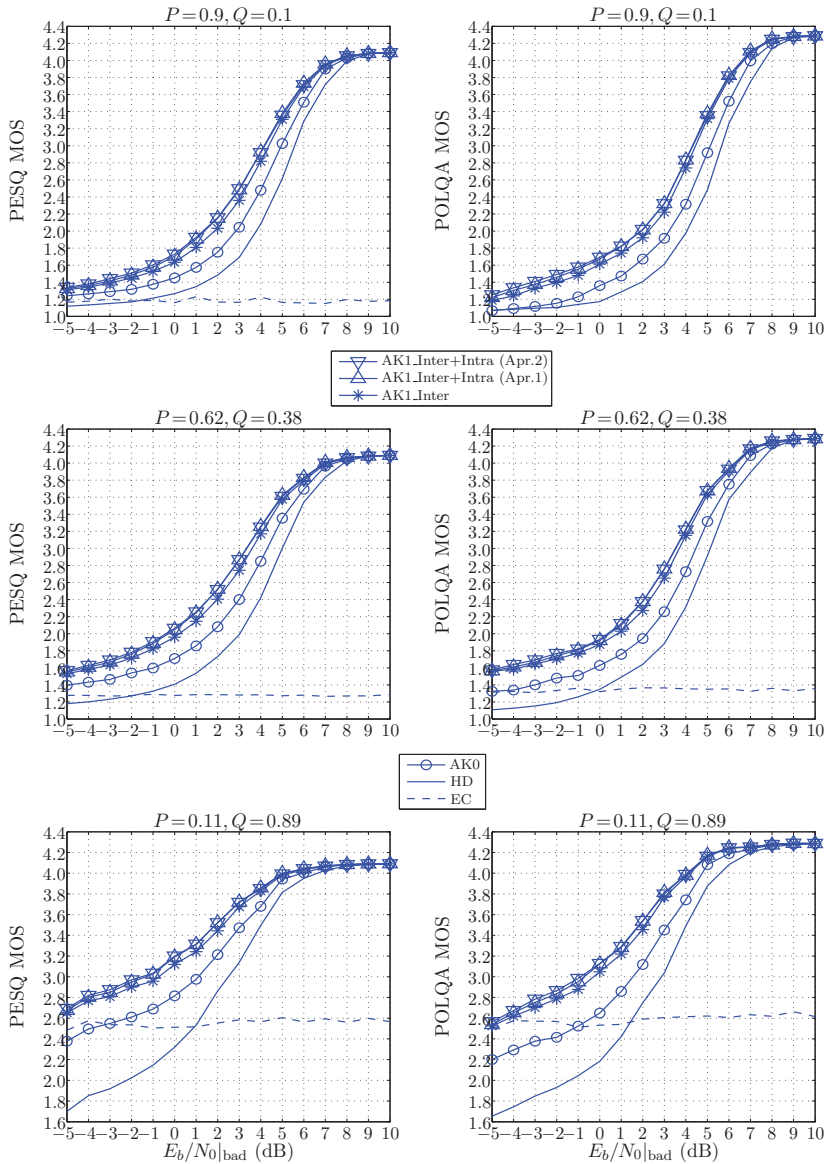


Figure 3.7: PESQ and POLQA MOS results for **AMR-NB** transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB (see Tabs. E.3 and E.4).

GE parameters	PESQ		POLQA	
	MOS gain	$E_b/N_0 _{\text{bad}}$ gain	MOS gain	$E_b/N_0 _{\text{bad}}$ gain
$P=0.9, Q=0.1$	0.84	6.1 dB	0.87	6 dB
$P=0.62, Q=0.38$	0.88	6.1 dB	0.90	6.5 dB
$P=0.11, Q=0.89$	0.78	6.5 dB	0.80	6.5 dB

Table 3.11: The largest MOS gains and $E_b/N_0|_{\text{bad}}$ gains of any of the SD decoding methods vs. the best of HD and EC for AMR-NB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB.

AMR-WB parameters	PESQ		POLQA	
	MOS gain	$E_b/N_0 _{\text{bad}}$ gain	MOS gain	$E_b/N_0 _{\text{bad}}$ gain
ISFs	0.50	6.2 dB	0.59	6.2 dB
Pitch delay	0.52	6 dB	0.82	7.2 dB
VQ gain	0.89	7.5 dB	1.01	6.2 dB

Table 3.12: The largest MOS gains and $E_b/N_0|_{\text{bad}}$ gains of any of the SD decoding methods vs. the best of HD and EC for AMR-WB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} \rightarrow \infty$, $P=0.11, Q=0.89$. Single parameter is individually corrupted.

Comparing the MOS scores of PESQ and POLQA for a given GE channel, it can be found that PESQ and POLQA lead to a similar performance, but with a higher POLQA MOS score for error-free transmission conditions. For each GE channel, in comparison to the best of HD and EC, the best improvements caused by SD decoding methods are concluded in Tab. 3.11, which show that the MOS improvement between any of the SD decoding method, and the best of HD and EC reaches up to 0.9 points and the $E_b/N_0|_{\text{bad}}$ gain is up to 6.5 dB.

AMR Wideband

(1) *Single Distorted Parameter*

Fig. 3.8 shows the simulation results of separately corrupting the individual parameters (ISFs, pitch delay, and VQ gain) in AMR-WB. Similar to AMR-NB, the MOS scores are significantly increased by SD decoding. The VQ gain is shown to be most vulnerable to bit errors, and has been improved most by SD decoding because of its high redundancy (see Tab. 3.7). The MOS gain between AK1_Inter+Intra and AK1_Inter of ISFs is up to 0.08 points for PESQ and 0.06 points for POLQA. This is because only a small intraframe residual redundancy can be exploited, as shown in Tab. 3.8. The best enhancements between any of the SD decoding methods, and the best of HD and EC for each individual parameter are depicted in Tab. 3.12.

(2) *All Bits Distorted*

The simulation results of AMR-WB transmissions over three different GE channels are depicted in Fig. 3.9. Similar to AMR-NB, the MOS scores of both PESQ and POLQA

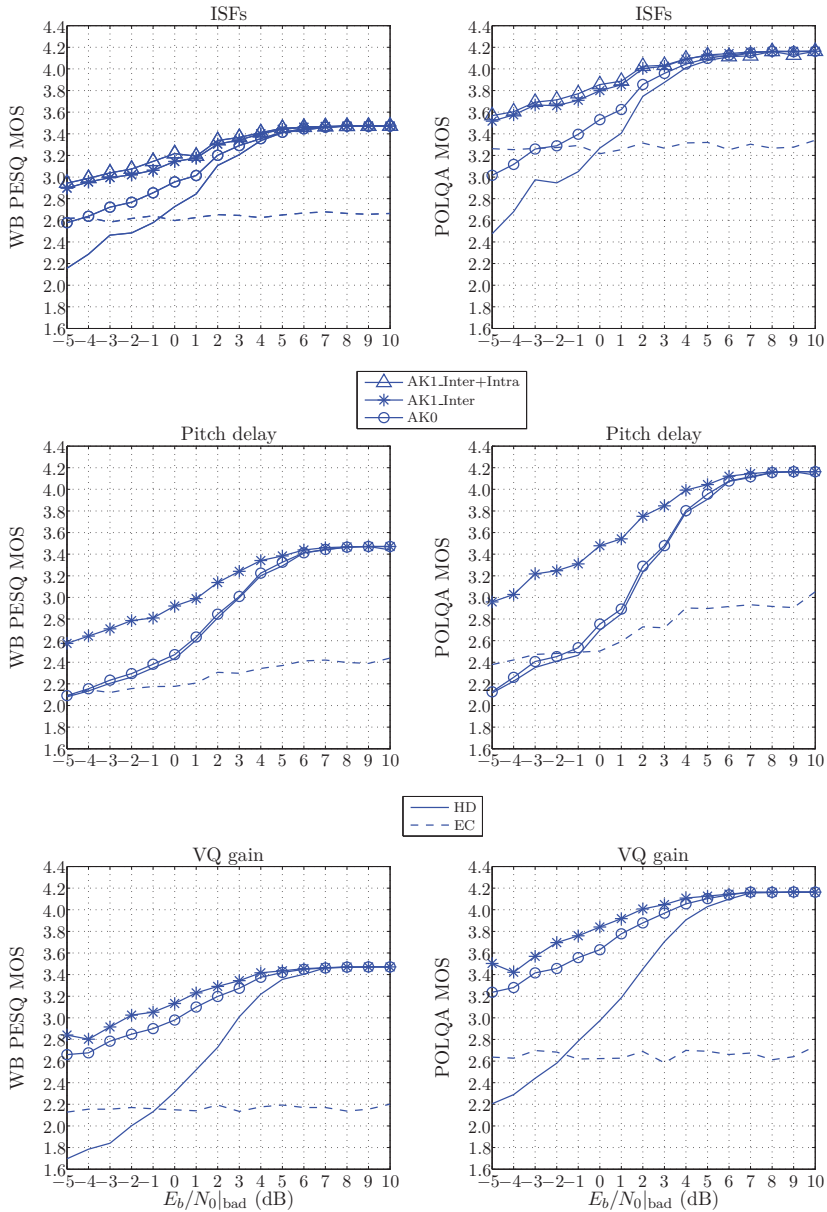


Figure 3.8: PESQ and POLQA MOS results for **AMR-WB** transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} \rightarrow \infty$, $P=0.11$, $Q=0.89$. Only a single parameter is individually corrupted (see Tabs. E.5 and E.6).

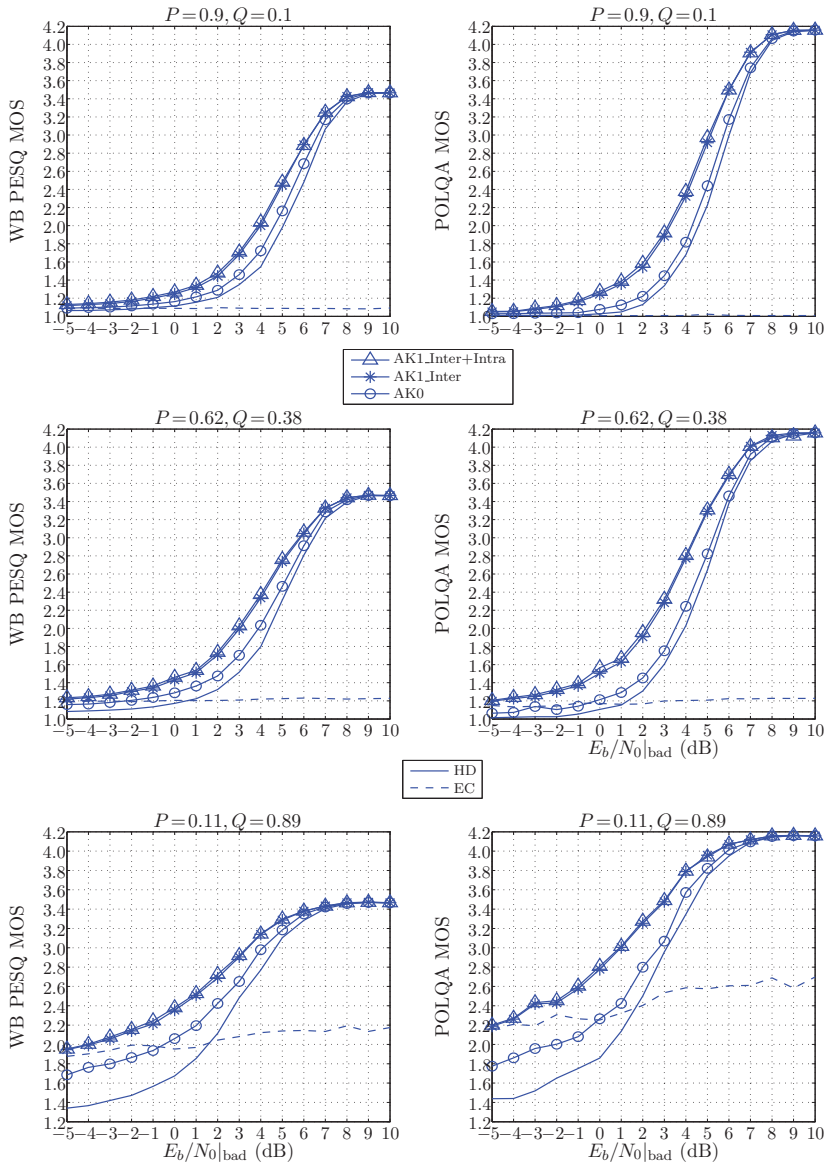


Figure 3.9: PESQ and POLQA MOS results for AMR-WB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB (see Tabs. E.7 and E.8).

GE parameters	PESQ		POLQA	
	MOS gain	$E_b/N_0 _{\text{bad}}$ gain	MOS gain	$E_b/N_0 _{\text{bad}}$ gain
$P=0.9, Q=0.1$	0.5	5.2 dB	0.75	6 dB
$P=0.62, Q=0.38$	0.58	6.1 dB	0.77	6.5 dB
$P=0.11, Q=0.89$	0.61	4.9 dB	0.78	4.8 dB

Table 3.13: The largest MOS gains and $E_b/N_0|_{\text{bad}}$ gains of any of the SD methods vs. the best of HD and EC for AMR-WB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB.

have been considerably improved by SD decoding. In AMR-WB, compared to AK1Inter, the approach of exploiting both inter- and intraframe redundancy only leads to up to gains of 0.04 MOS points for PESQ and 0.06 points for POLQA. The largest MOS gain and $E_b/N_0|_{\text{bad}}$ gain of any of the SD decoding methods vs. the best of HD and EC for each GE channel can be easily seen in Tab.3.13. It can be stated that the performance gap between SD decoding and the best of HD and EC reaches up to 0.78 MOS points and an $E_b/N_0|_{\text{bad}}$ ratio of 6.5 dB.

In Fig. 3.9, for $P = 0.9$ or $P = 0.62$, the standard error concealment (EC) reveals a similar very poor performance for all the channel conditions. However, for $P = 0.11$, the MOS scores in good channel conditions are slightly higher than the values in bad channel conditions. This is due to the fact that for bad frames, the parameter pitch delay is taken unmodified at times (see pitch delay in Fig. 3.8) and the fixed codebook indices are always employed as they are received. This effect is noticeable for $P = 0.11$ because of fewer bad frames (small P_{bad} in Tab. 3.9).

3.5 Summary

In this chapter, the fixed-length soft-decision (FL/SD) decoding described in Section 2.3 is applied to the AMR-NB and AMR-WB speech decoders. The codec parameters in Adaptive Multi-Rate Narrowband (AMR-NB) and AMR Wideband (AMR-WB) are first analyzed. The standard error concealment in AMR-NB and AMR-WB are described thereafter, which shows that only a coarse channel reliability information has been utilized so far. The application of FL/SD decoding to the codec parameters is elaborated afterwards. Moreover, approaches exploiting both inter- and intraframe redundancy for the parameters line spectral frequencies (LSFs) in AMR-NB and immittance spectral frequencies (ISFs) in AMR-WB are also proposed, which have further enhanced the performance. Compared to the traditional hard-decision (HD) decoding or the standard error concealment, by employing SD decoding the speech quality can be improved significantly by up to 0.9 MOS or up to 6.5 dB E_b/N_0 for AMR-NB. For AMR-WB, a similar improvement of mean opinion score (MOS) and E_b/N_0 gains up to 0.78 points and 6.5 dB is observed, respectively. Therefore, robust AMR-NB and AMR-WB decoders have been found to deal with adverse transmission conditions.

Chapter 4

Variable-Length Soft-Decision Decoding

In Chapter 3, for a given codec parameter, the quantized parameter is represented by a fixed-length code (FLC) providing in each frame or regular subframe grid pattern (even, odd, etc.) the same codeword length (i.e., quantization bit rate). In contrast, variable-length codes (VLCs) map each parameter instance to a variable number of bits and offer a lower bit rate resulting in a better compression performance. But inevitably, VLCs are very vulnerable to transmission errors. Similar to fixed-length soft-decision decoding (FL/SD) decoding in Chapters 2 and 3, VL/SD decoding utilizing bit-wise channel reliability information can improve robustness. VLCs are widely used in audio coding and have been adopted, e.g., in the MPEG-4 High-Efficiency Advanced Audio Coding (HE-AAC) audio codec. In order to improve the HE-AAC performance for error-prone transmission conditions in Chapter 5, the VL/SD decoding approach is detailed in this chapter. Moreover, the performance tradeoffs of FL/SD and VL/SD decoding depending on block length, quantization bit rate, and source correlation are also discussed in this chapter. Section 4.1 introduces some background of VL/SD decoding and the transmission system. The traditional VL/HD decoding is explained in Section 4.2. Section 4.3 shows the trellis representation for VL/SD decoding. The calculation of the *a posteriori* probabilities (APPs) and source symbol estimation is presented in Section 4.4. The links between VL/SD and FL/SD decoding are described in Section 4.5. Finally, the simulation results are shown in Section 4.6.

4.1 Introduction

In media transmission, the same message can be represented either by fixed-length codes (FLCs) allowing the same codeword length for each source symbol [Oliver et al., 1948, Hui and Neuhoff, 2001], or by variable-length codes (VLCs) which assign each source symbol to a variable number of bits [Lookabaugh et al., 1993]. Compared to FLCs, VLCs offer a lower bit rate and achieve a higher coding efficiency. For storage/transmission in reliable (error-free) conditions, the correct start and end bit positions (in the bit stream) of

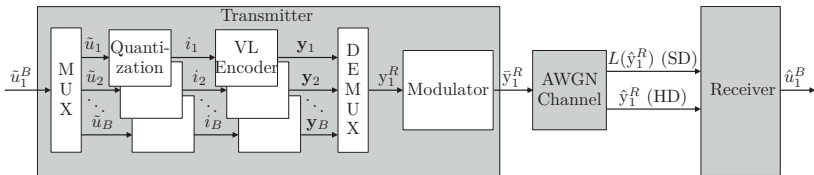


Figure 4.1: Block diagram of the transmission system for VLCs [Han and Fingscheidt, 2014a], showing the transmission of a block of parameters \tilde{u}_1^B . Note that a number of B symbols is involved, since each \tilde{u}_n belongs to a certain time index n .

each symbol with corresponding codeword can be identified. However, for unreliable transmission conditions, due to the variable length of each codeword, the start and end bit positions may not be correctly recognized at the receiver. In that case, error propagation becomes a serious problem for VLCs, even single bit errors can lead to significant signal distortions over a long time. Therefore, a robust VL source decoder is desired for adverse transmission conditions.

For the traditional variable-length hard-decision (VL/HD) source decoding, error concealment is offered as a further option in [ISO/IEC, 2005], but only with a binary (very coarse) reliability information per frame (similar to the bad frame indicator (BFI) in Section 3.2). As mentioned in Section 2.3, instead of receiving hard-decided bits in HD decoding, soft-decision (SD) decoding has been considered as a more robust means for error concealment, which expects the soft information representing bit-wise channel reliability information. The concept of SD decoding cannot only be applied to FL decoding, but also to VL decoding.

The VL/SD decoding approach is either non-trellis-based [Wen and Villasenor, 1999], or using the Viterbi algorithm based on a graphical representation or a trellis representation [Park and Miller, 2000, Demir and Sayood, 1998, Murad and Fuja, 1998]. Based on the BCJR algorithm [Bahl et al., 1974], a bit-level soft-in/soft-out VL/SD decoder has been proposed by [Bauer and Hagenauer, 2001, Thobaben and Kliewer, 2005], with the help of the trellis representation from [Balakirsky, 1997]. The BCJR algorithm [Bahl et al., 1974] has also been applied to a symbol-level soft-in/soft-out VL/SD decoder [Rainer and Hagenauer, 2000], which is based on an intuitive trellis representation. Based on the work in [Rainer and Hagenauer, 2000] which models the source symbol as a zeroth-order Markov process, Kliewer and Thobaben proposed a VL/SD decoder which applies a first-order Markov source model and offers a higher robustness in the case of temporal correlations of the source symbols [Kliewer and Thobaben, 2005].

VLCs are widely used for video, image, and audio coding. In [Xiang et al., 2003], a soft decoder with bit-level trellis representation for Huffman codes is proposed for iterative decoding of JPEG-coded images. VL/SD decoding has also been applied to H.264 and H.263+ Recommendations [Bergeron and Lamy-Bergot, 2004, Lee et al., 2005]. Using the method from [Rainer and Hagenauer, 2000], the scale factors of the MPEG-AAC are reconstructed by VL/SD decoding [Derrien et al., 2008]. It is found that the number of scale factors (i.e., *block length* in this chapter) varies in each frame. However, both [Rainer

z		Column index			
		$j_c = 0$	$j_c = 1$	$j_c = 2$	$j_c = 3$
Row index	$j_r = 0$	7	7	4	9
	$j_r = 1$	15	15	3	3

(a) Decimal representation

z		Column index			
		$j_c = 0$	$j_c = 1$	$j_c = 2$	$j_c = 3$
Row index	$j_r = 0$	1 1 1	1 1 1	1 0 0	10 0 1
	$j_r = 1$	11 1 1	11 1 1	0 1 1	0 1 1

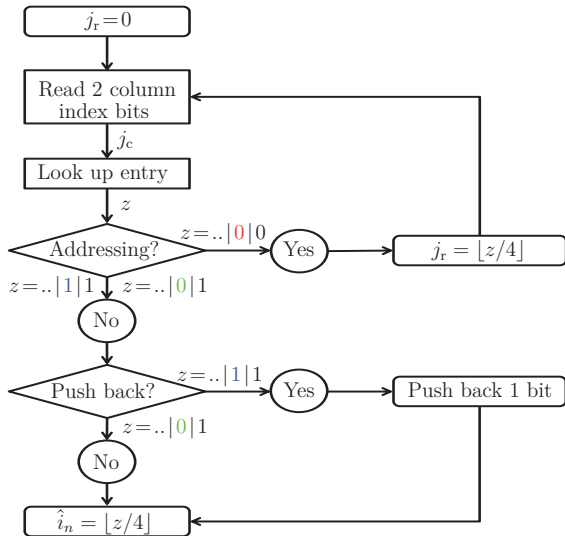
(b) Binary representation

Table 4.1: Example of a VL/HD decoder lookup table in decimal and binary representation.

and Hagenauer, 2000] and [Kliewer and Thobaben, 2005] only show the results for a block length of 100, although the block length must play a significant role in the performance of VLCs. Therefore, it is interesting to study the influence of block length and quantization in VL/SD decoding. In this chapter, we adopt the VL/SD decoding approach from [Kliewer and Thobaben, 2005] and present it in more detail. We also provide some new insight into the definitions of the stage boundaries and state intervals in the trellis representation. Moreover, the performance tradeoffs of FL/SD and VL/SD decoding for different block lengths, quantization bit rates, and source correlation are discussed.

The block diagram of the transmission system with VL/SD decoding is shown in Fig. 4.1 [Han and Fingscheidt, 2014a]. The (unquantized) source symbols \tilde{u}_1^B inside a block are first multiplexed $\tilde{u}_1^B = (\tilde{u}_1, \dots, \tilde{u}_n, \dots, \tilde{u}_B)$, with $n \in \{1, 2, \dots, B\}$ being the symbol time index and B being the block length. Subsequently, each source symbol is fixed-rate quantized and represented by a corresponding M bit quantization codebook index $i \in \mathcal{I} = \{0, 1, \dots, 2^M - 1\}$. A VL encoder (e.g., a Huffman code) maps each quantization index i to a variable-length bit combination $\mathbf{y}_n \in \{0, 1\}^{N^{(i)}}$, with $N^{(i)}$ being the codeword length of the quantization codebook index i . After demultiplexing, the resulting unipolar bit stream \mathbf{y}_1^R is transformed to a stream of bipolar modulation symbols¹ $\bar{\mathbf{y}}_1^R = (\bar{\mathbf{y}}(1), \bar{\mathbf{y}}(2), \dots, \bar{\mathbf{y}}(R)) = ((\bar{\mathbf{y}}_1)^T, \dots, (\bar{\mathbf{y}}_n)^T, \dots, (\bar{\mathbf{y}}_B)^T) = \{-1, +1\}^R$ by a binary phase-shift keying (BPSK) modulator, with $\bar{\mathbf{y}}_n$ being the bipolar representation of \mathbf{y}_n , symbol $()^T$ denoting the transpose, and the length R being the number of all bits in the block. Thereafter, transmission takes place over an additive white Gaussian noise (AWGN) channel. For conventional HD decoding, the received hard-decided bipolar bit stream $\hat{\mathbf{y}}_1^R = \{-1, +1\}^R$ is analyzed starting from bit positions 1. The VL bit combinations are transformed to corresponding quantizer indices according to a VL *decoder-sided* lookup table (different to the VL transmitter-sided codebook). In contrast, the VL/SD decoder expects log-likelihood ratios (LLRs) $L(\hat{\mathbf{y}}_1^R) = (L(\hat{\mathbf{y}}(1)), L(\hat{\mathbf{y}}(2)), \dots, L(\hat{\mathbf{y}}(R))) \in \mathbb{R}^R$ which represent the channel reliability information. The details of VL/HD decoding are explained in the following Section 4.2.

¹In this notation we mix a limited-length sequence with a sequence of a certain number of row vectors.

Figure 4.2: Flowchart of VL/HD decoding providing \hat{i}_n .

4.2 Variable-Length (VL) Hard-Decision Decoding

The classical HD demodulator converts the received symbols into a stream of unipolar bits $\hat{y}_1^R = \{0, 1\}^R$. Based on a *decoder-sided* lookup table, they are converted to the corresponding received VL codebook indices \hat{i}_n and the quantized symbol \hat{u}_n . The decimal and binary representation of the lookup table is depicted in Tab. 4.1, with a simplified example of four Huffman codewords [Huffman, 1952] $\mathbf{y}^{(\hat{i}_n)}$ being $\mathbf{y}^{(0)} = (1, 0, 1)$, $\mathbf{y}^{(1)} = (0)$, $\mathbf{y}^{(2)} = (1, 1)$, $\mathbf{y}^{(3)} = (1, 0, 0)$ ($\hat{i}_n \in \mathcal{I} = \{0, 1, \dots, 3\}$ in this example). The row and column indices of the tables are denoted as j_r and j_c , respectively. Three types of values (marked blue, green, and red) are shown in Tab. 4.1. In Tab. 4.1(b), the last two bits of each table entry determine whether the corresponding leading 1 or 2 bits (..) represent a Huffman codebook index \hat{i}_n (in case of ..|1|1, ..|0|1), or a row index j_r (in case of ..|0|0). Moreover, the table entries belonging to an odd codeword length $N^{(\hat{i}_n)} \in \{1, 3\}$ always occur twice in the table one after the other in the same row (this is case for ..|1|1). If the length of the codeword $\mathbf{y}^{(\hat{i}_n)}$ is even ($N^{(\hat{i}_n)} = 2$), the last two bits of the table entry are ..|0|1.

The flowchart of VL/HD decoding is depicted in Fig. 4.2. Decoding starts with setting the row index to $j_r = 0$. The two-bit column index j_c is obtained by reading two bits from the bit stream (starting with the first two (i.e., leftmost) bits of Huffman codewords). Thereafter, according to j_r and j_c , the corresponding table entry z is identified in Tab. 4.1(a) or Tab. 4.1(b). If the last two bits of the entry are 0|0, a new row index needs to be addressed by computing $j_r = \lfloor z/4 \rfloor$, which is a right shift of z by two bits (i.e., the leading bits). Consequently, two more bits are read to obtain the new j_c . If no further

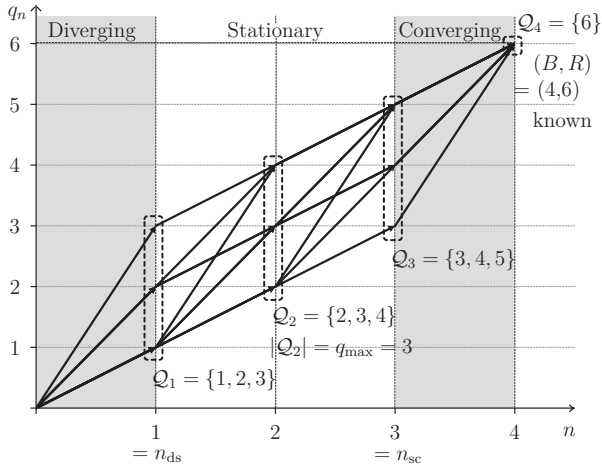


Figure 4.3: Trellis representation for $B=4$, $R=6$, both are known. The Huffman codeword time index is n , and the total number of bits until time n is q_n . The VLC is $\mathbf{y}^{(0)} = (1, 0, 1)$, $\mathbf{y}^{(1)} = (0)$, $\mathbf{y}^{(2)} = (1, 1)$, $\mathbf{y}^{(3)} = (1, 0, 0)$ [Han and Fingscheidt, 2014a].

addressing is necessary (i.e., the last two bits are $1|1$ or $0|1$), the current table entry, right-shifted by two bits, will be taken as the codebook index $\hat{i}_n = \lfloor z/4 \rfloor$. In addition, the last two bits being $1|1$ implies that the codeword length is odd. Since two bits are read each time, the second bit has to be pushed back and used as the starting bit for the next process of reading two bits.

As an example for a received bit stream $\mathbf{y}_1^5 = (1, 0, 0, 1, 1)$, reading the first two bits $(1, 0)$ result in $j_c = 2$. Locating $j_r = 0$ and $j_c = 2$ in Tab. 4.1, $z = 1|0|0$ is found. Based on Fig. 4.2, addressing is needed and $j_r = 1$ is obtained. Thereafter, two more bits $(0, 1)$ are read to get new $j_c = 1$, $z = 11|1|1$ is therefore found in Tab. 4.1. According to $\hat{i}_n = \lfloor z/4 \rfloor$ in Fig. 4.2, we get $\hat{i}_1 = 3$. In addition, 1 bit is pushed back, which means that the next two bits $(1, 1)$ will be read. Finally, according to $j_c = 3$ and resetting $j_r = 0$, $z = 10|0|1$ is found in Tab. 4.1. Known from Fig. 4.2, $\hat{i}_2 = 2$ is obtained without pushing one bit back. Therefore, the bit stream assigns to two quantization codebook indices 3 and 2.

After describing the VL/HD decoding, the VL/SD decoding is described in the following Sections 4.3 and 4.4.

4.3 Soft-Decision (SD) Decoding: Trellis Representation

In erroneous transmission conditions, due to the variable length codewords, the bit positions in the bit stream at the receiver at each time index may not be correctly localized. In order to obtain a robust VL decoder for adverse transmission conditions,

all the possible bit positions at each time index must be considered. Fig. 4.3 shows the trellis representation from [Rainer and Hagenauer, 2000]. According to this trellis representation, in the bit stream y_1^R , all the possible positions of the last bit of \mathbf{y}_n at the current symbol time n and of \mathbf{y}_{n-1} at the previous symbol time $n-1$ are denoted as state $q_n = \nu \in \mathcal{Q}_n$ and $q_{n-1} = \mu \in \mathcal{Q}_{n-1}$, respectively. In other words, at symbol time n , the state changes from μ to ν , which is due to the transmission of the bit combination \mathbf{y}_n with a length $\nu - \mu$. Fig. 4.3 shows an example of block length $B = 4$ with 2 bit quantization, and the total number of $R = 6$ with Huffman codewords $\mathbf{y}^{(0)} = (1, 0, 1)$, $\mathbf{y}^{(1)} = (0)$, $\mathbf{y}^{(2)} = (1, 1)$, $\mathbf{y}^{(3)} = (1, 0, 0)$. The VL codeword of the quantization index i is represented by $\mathbf{y}_n^{(i)} = (y_n^{(i)}(1), \dots, y_n^{(i)}(m), \dots, y_n^{(i)}(N^{(i)}))$, correspondingly, with $y_n^{(i)}(m) \in \{0, 1\}$ being the m th bit of $\mathbf{y}_n^{(i)}$.

As can be seen from Fig. 4.3, the time-varying trellis can be divided into three distinct stages: *Diverging* (the number of states in \mathcal{Q}_n increases along with n), *stationary* (the number of states in \mathcal{Q}_n remains the maximum number of states q_{\max}), and *converging* (the number of states decreases towards the known number of symbols and bits in the block (B, R)).

Due to the various possibilities for the pair (B, R) , not all the codeword lengths $N^{(i)}$ are necessarily needed in the trellis representation. Imagine, e.g., $(B, R) = (4, 4)$, where obviously only codeword $\mathbf{y}^{(1)} = (0)$ has been received four times, or $(B, R) = (4, 5)$, where only one of the four received codewords consisted of 2 bits (here $\mathbf{y}^{(2)} = (1, 1)$). A further example is $(B, R) = (4, 11)$, where obviously three codewords must have been of length 3 bits, while one was of 2 bits length: $\mathbf{y}^{(2)} = (1, 1)$. Therefore, we define $N'_{\min} = \min_{i \in \mathcal{I}} N^{(i)}$ as the minimum length of all the Huffman codewords, while N'_{\min} is the minimum length of the Huffman codewords used in the trellis representation. Correspondingly, $N'_{\max} = \max_{i \in \mathcal{I}} N^{(i)}$ and N'_{\max} denote the maximum codeword length of all the Huffman codewords and the Huffman codewords used in the trellis representation, respectively. It is found that [Han and Fingscheidt, 2015a]

$$N'_{\min} = \max(R - N_{\max} \cdot (B-1), N_{\min}), \quad (4.1)$$

and

$$N'_{\max} = \min(R - N_{\min} \cdot (B-1), N_{\max}). \quad (4.2)$$

The maximum number of states q_{\max} of any of the sets \mathcal{Q}_n with $n \in \{1, 2, \dots, B\}$ in a trellis representation is also a function of R, B, N'_{\max} , and N'_{\min} , which can be found in Tab. 4.2. Note that in Fig. 4.3, $N'_{\min} = N_{\min} = 1$, $N'_{\max} = N_{\max} = 3$, and $q_{\max} = 3$.

Definitions of Stage Boundaries and State Intervals

Since three different stages exist in the trellis representation (as shown in Fig. 4.3), the stage boundaries in terms of symbol time n and the definitions of state intervals in each stage should be different, which can be easily seen in Tab. 4.2.

In the *diverging* stage, the transitions from state q_{n-1} to q_n with *all* the codeword lengths from N'_{\min} to N'_{\max} are possible. As a result, the state at symbol time n starts from $\nu_{\text{start}} = N'_{\min} \cdot n$ and ends at $\nu_{\text{end}} = N'_{\max} \cdot n$, which spans the state interval $[\min \mathcal{Q}_n,$

	$R < N'_{\max} \cdot (B-1)$	$R \geq N'_{\max} \cdot (B-1)$	$B = 3, R = 2 \cdot N'_{\max}$ $N'_{\max} \neq \frac{3}{2} N'_{\min}$
q_{\max}	$R - N'_{\min} \cdot B + 1$	$N'_{\max} \cdot B - R + 1$	Λ
n_{ds}		$\lceil \frac{q_{\max}-1}{N'_{\max}-N'_{\min}} \rceil$	1
n_{sc}	$\lfloor \frac{N'_{\max} \cdot B - R}{N'_{\max} - N'_{\min}} \rfloor$	$\lfloor \frac{R - N'_{\min} \cdot B}{N'_{\max} - N'_{\min}} \rfloor$	2
Diverging		$[N'_{\min} \cdot n, N'_{\max} \cdot n]$	$[N'_{\min}, N'_{\max}]$
State Stationary	$[N'_{\min} \cdot n, N'_{\min} \cdot n + R - N'_{\min} \cdot B]$	$[N'_{\max} \cdot n - N'_{\max} \cdot B + R, N'_{\max} \cdot n]$	$[R - N'_{\max}, R - N'_{\min}]$
Converging	$[R - N'_{\max} \cdot B + N'_{\max} \cdot n, R - N'_{\min} \cdot B + N'_{\min} \cdot n]$		$[R]$

Table 4.2: Definitions of the maximum number of states q_{\max} , stage boundaries n_{ds} and n_{sc} , and state intervals $[\min \mathcal{Q}_n, \max \mathcal{Q}_n]$ for the diverging, stationary, and converging stages.

$\max \mathcal{Q}_n]$. Because of the limitation of the maximum number of states q_{\max} , the number of states at the symbol time n_{ds} , which is the boundary between the diverging and the stationary stage, satisfies

$$(N'_{\max} - N'_{\min}) \cdot n_{\text{ds}} + 1 = q_{\max}, \quad (4.3)$$

which results in $n_{\text{ds}} = \lceil \frac{q_{\max} - 1}{N'_{\max} - N'_{\min}} \rceil$ (=1 in Fig. 4.3).

For the *stationary* stage, the number of states equals q_{\max} , which means $\nu_{\text{end}} = \nu_{\text{start}} + q_{\max} - 1$. In addition, the ν_{start} and ν_{end} satisfy

$$\nu_{\text{start}} = N'_{\min} \cdot n, \text{ if } R < N'_{\max} \cdot (B-1), \quad (4.4a)$$

$$\nu_{\text{end}} = N'_{\max} \cdot n, \text{ if } R \geq N'_{\max} \cdot (B-1), \quad (4.4b)$$

with the corresponding ending and starting states being calculated according to the value of q_{\max} .

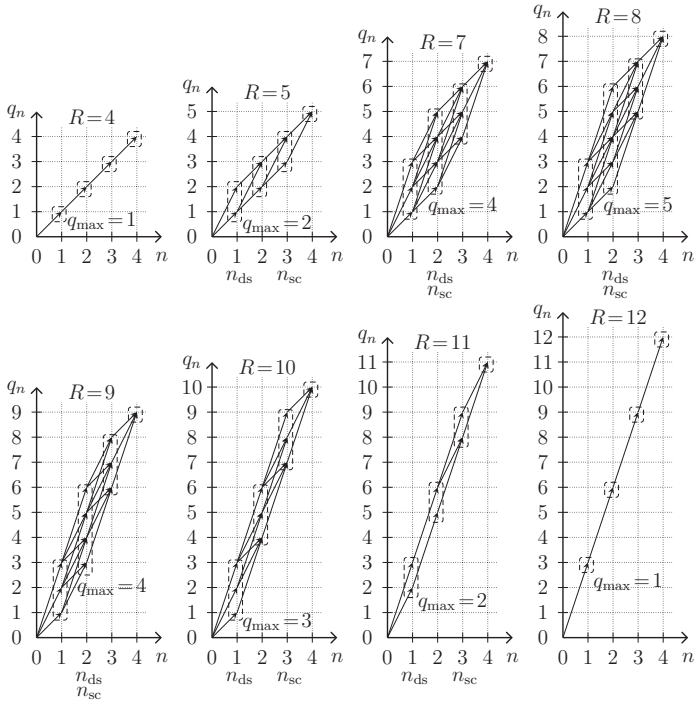
In the *converging* stage, with the starting state ν_{start} at symbol time $n = n_{\text{sc}}$ and the maximum codeword length N'_{\max} for the state transitions in the subsequent symbol times $(n+1, \dots)$, we have $\nu_{\text{start}} + N'_{\max} \cdot (B-n) = R$. Consequently, it can be derived that $\nu_{\text{start}} = R - N'_{\max} \cdot B + N'_{\max} \cdot n$. Similarly, for the ending state ν_{end} at symbol time n , with the minimum codeword length N'_{\min} for the following state transitions, we have $\nu_{\text{end}} + N'_{\min} \cdot (B-n) = R$, which results in $\nu_{\text{end}} = R - N'_{\min} \cdot B + N'_{\min} \cdot n$. Correspondingly, for the symbol time boundary n_{sc} of stationary and converging stages, the number of states satisfies

$$N'_{\min} \cdot n_{\text{sc}} + N'_{\max} \cdot (B - n_{\text{sc}}) \geq R, \text{ if } R < N'_{\max} \cdot (B-1), \quad (4.5a)$$

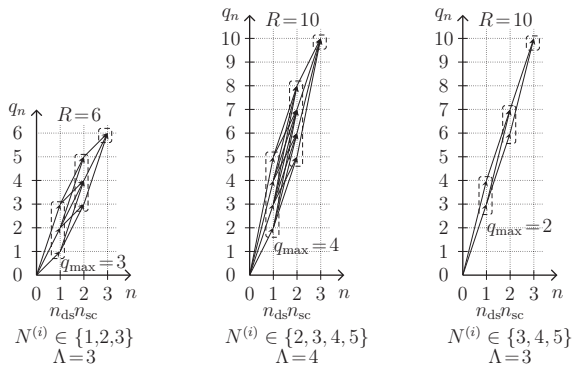
$$N'_{\max} \cdot n_{\text{sc}} + N'_{\min} \cdot (B - n_{\text{sc}}) \leq R, \text{ if } R \geq N'_{\max} \cdot (B-1). \quad (4.5b)$$

The resulting n_{sc} can be seen in Tab. 4.2 (=3 in Fig. 4.3).

With $B=4$ and the VLCs being the same as in Fig. 4.3 ($N^{(i)} \in \{1, 2, 3\}$), the trellis representation for all the other possibilities of R is shown in Fig. 4.4(a), while $R=6$ is depicted in Fig. 4.3. Note that for $R \in \{7, 8, 9\}$, $n_{\text{ds}} = n_{\text{sc}}$, which implies no stationary stage in the trellis representation. If $R=4$ or $R=12$, the trellis representation becomes a straight line, since $N'_{\min} = N'_{\max}$, the VLCs in fact are FLCs.



(a) $B=4$, with the same Huffman codewords as $R=6$ in Fig. 4.3 and $N^{(i)} \in \{1, 2, 3\}$



(b) $B=3$, with different Huffman codewords

Figure 4.4: Trellis representation for VLCs with block length $B = 3$ and $B = 4$, with different Huffman codewords.

The definitions mentioned above are valid for all the values of B and R , except for² $B = 3$, $R = 2 \cdot N_{\max}$, $N_{\max} \neq \frac{3}{2}N_{\min}$, as shown in Fig. 4.4(b) and the last column in Tab. 4.2, with three different examples of Huffman codewords (i.e., different ranges of $N^{(i)}$, or different values of N_{\max}). Each stage covers one symbol time. The states for the diverging and stationary stages fall in $[N'_{\min}, N'_{\max}]$ and $[R - N'_{\max}, R - N'_{\min}]$, respectively. The value of q_{\max} is now the number of different-length Huffman codewords used in the trellis representation Λ (i.e., how many different values of $N^{(i)}$ occur in the range of $[N'_{\min}, N'_{\max}]$). Note that $N'_{\max} = 4$ ($\neq N_{\max}$) in the third plot of Fig. 4.4(b).

4.4 SD: A *Posteriori* Probabilities and Source Symbol Estimation

The block length B , the bit stream length R , and the LLRs will be required by the VL/SD decoder. As in the FL/SD decoder, the core of the VL/SD decoder is to calculate the *a posteriori* probabilities (APPs) which are further used for source symbol estimation.

4.4.1 A *Posteriori* Probabilities (APPs)

The APP calculation in this section is according to the modified BCJR algorithm from [Kliewer and Thobaben, 2005]. At symbol time n , the APPs of a probably transmitted quantizer bit combination $\mathbf{y}_n = \mathbf{y}^{(i)}$, given the received bit stream $\hat{\mathbf{y}}_1^R$ is calculated by

$$P(\mathbf{y}_n = \mathbf{y}^{(i)} | \hat{\mathbf{y}}_1^R) = \frac{1}{C} \cdot \sum_{\mu \in \mathcal{Q}_{n-1}} \sum_{\nu \in \mathcal{Q}_n} \alpha_{n-1}(\mu) \cdot \beta_n(\nu) \cdot \gamma_n(i, \mu, \nu), \quad (4.6)$$

with the forward probability $\alpha_{n-1}(\mu) = p(q_{n-1} = \mu, \hat{\mathbf{y}}_1^\mu)$, backward probability $\beta_n(\nu) = p(\hat{\mathbf{y}}_{\nu+1}^R | q_n = \nu)$, and $\gamma_n(i, \mu, \nu) = P(\hat{\mathbf{y}}_{\mu+1}^\nu, \mathbf{y}_n = \mathbf{y}^{(i)}, q_n = \nu | q_{n-1} = \mu, \hat{\mathbf{y}}_1^\mu)$. Moreover, the received bit stream from bit position μ to ν is denoted as $\hat{\mathbf{y}}_\mu^\nu = (\hat{\mathbf{y}}(\mu), \hat{\mathbf{y}}(\mu+1), \dots, \hat{\mathbf{y}}(\nu)) = \{-1, +1\}^{\nu-\mu+1}$. The state μ is an element of state set \mathcal{Q}_{n-1} at the previous symbol time $n-1$, while $\nu \in \mathcal{Q}_n$ denotes the current symbol time n , respectively. The constant $C = \sum_{i \in \mathcal{I}} \sum_{\mu \in \mathcal{Q}_{n-1}} \sum_{\nu \in \mathcal{Q}_n} \alpha_{n-1}(\mu) \cdot \beta_n(\nu) \cdot \gamma_n(i, \mu, \nu)$ is used to normalize the sum over the APPs to one.

Forward Recursion

The forward recursion can be derived by

$$\alpha_n(\nu) = \sum_{\mu \in \mathcal{Q}_{n-1}} \sum_{i \in \mathcal{I}} \alpha_{n-1}(\mu) \cdot \gamma_n(i, \mu, \nu), \quad (4.7)$$

with the initial value $\alpha_0(0) = 1$, and

$$\gamma_n(i, \mu, \nu) = P(\hat{\mathbf{y}}_{\mu+1}^\nu | \mathbf{y}_n = \mathbf{y}^{(i)}) \cdot P(\mathbf{y}_n = \mathbf{y}^{(i)}, q_n = \nu | q_{n-1} = \mu, \hat{\mathbf{y}}_1^\mu), \quad (4.8)$$

which includes a *channel term* and a *source probability distribution term*.

²If $N_{\max} = \frac{3}{2}N_{\min}$, then $N'_{\min} = N'_{\max}$, the trellis representation in this case is a straight line.

(1) *Channel Term*

Assuming a memoryless channel, the channel term, which describes the probability of the received bit stream $\hat{y}_{\mu+1}^\nu$ given a probably transmitted quantizer bit combination $\mathbf{y}_n = \mathbf{y}^{(i)}$, can be obtained by the conditional bit probability (as the channel transition probabilities in Section 2.3.2)

$$P(\hat{y}_{\mu+1}^\nu | \mathbf{y}_n = \mathbf{y}^{(i)}) = \prod_{m=1}^{N^{(i)}} P(\hat{y}(\mu + m) | y^{(i)}(m)), \quad (4.9)$$

with $\hat{y}(\mu + m)$ being the received hard-decided bit at bit position $\mu + m$ and

$$P(\hat{y}(\mu + m) | y^{(i)}(m)) = \begin{cases} 1 - \text{BER}(m), & \text{if } \hat{y}(\mu + m) = \bar{y}^{(i)}(m), \\ \text{BER}(m), & \text{else.} \end{cases} \quad (4.10)$$

Herein, $\bar{y}^{(i)}(m)$ is the bipolar representation of $y^{(i)}(m)$ (i.e., the m th bit of codeword $\mathbf{y}^{(i)}$). The bit error probability can be obtained by $\text{BER}(m) = \frac{1}{1 + \exp(|L(\hat{y}(\mu + m))|)}$, with $L(\hat{y}(\mu + m))$ being calculated based on (2.16).

(2) *Source Probability Distribution Term*

Modeling the quantized symbols either as a zeroth-order Markov process, or as a first-order Markov process, the *source probability distribution term* (i.e., the latter term of the right-hand-side of (4.8)) can be obtained either with the zeroth-order *a priori* knowledge (AK0) or the first-order *a priori* knowledge (AK1). The AK0 term $P(\mathbf{y}_n = \mathbf{y}^{(i)})$ and AK1 term $P(\mathbf{y}_n = \mathbf{y}^{(i)} | \mathbf{y}_{n-1} = \mathbf{y}^{(j)})$ can be obtained in analogy to (2.5) and (2.6) in Section 2.1.3.

(2a) *Source Probability Distribution Term With AK0*

Using AK0, the latter term of the right-hand-side of (4.8) can be written as

$$P(\mathbf{y}_n = \mathbf{y}^{(i)}, q_n = \nu | q_{n-1} = \mu, \hat{y}_1^\mu) = \frac{1}{C_1'(\mu)} \begin{cases} P(\mathbf{y}_n = \mathbf{y}^{(i)}), & \text{if } \nu - \mu = N^{(i)}, \\ 0, & \text{else,} \end{cases} \quad (4.11)$$

with the normalization³ $C_1'(\mu) = \sum_{\nu' \in \mathcal{Q}_n} \sum_{\substack{i \in \mathcal{I} \\ N^{(i)} = \nu' - \mu}} P(\mathbf{y}_n = \mathbf{y}^{(i)})$.

(2b) *Source Probability Distribution Term With AK1*

Considering source correlation, the *source probability distribution term* in (4.8) can be computed by

$$P(\mathbf{y}_n = \mathbf{y}^{(i)}, q_n = \nu | q_{n-1} = \mu, \hat{y}_1^\mu) = \sum_{j \in \mathcal{I}} P(\mathbf{y}_n = \mathbf{y}^{(i)}, q_n = \nu | \mathbf{y}_{n-1} = \mathbf{y}^{(j)}, q_{n-1} = \mu) \cdot P(\mathbf{y}_{n-1} = \mathbf{y}^{(j)} | q_{n-1} = \mu, \hat{y}_1^\mu). \quad (4.12)$$

³ $C_1'(\mu) = 1$ in the diverging stage.

The first term on the right-hand-side of (4.12) can be calculated by the AK1 term:

$$\begin{aligned} & \text{P}(\mathbf{y}_n = \mathbf{y}^{(i)}, q_n = \nu \mid \mathbf{y}_{n-1} = \mathbf{y}^{(j)}, q_{n-1} = \mu) = \\ & \frac{1}{C_1(\mu, j)} \begin{cases} \text{P}(\mathbf{y}_n = \mathbf{y}^{(i)} \mid \mathbf{y}_{n-1} = \mathbf{y}^{(j)}), & \text{if } \nu - \mu = N^{(i)}, \\ 0, & \text{else,} \end{cases} \end{aligned} \quad (4.13)$$

with the normalization $C_1(\mu, j) = \sum_{\nu' \in \mathcal{Q}_n} \sum_{\substack{i \in \mathcal{I} \\ N^{(i)} = \nu' - \mu}} \text{P}(\mathbf{y}_n = \mathbf{y}^{(i)} \mid \mathbf{y}_{n-1} = \mathbf{y}^{(j)})$.

According to the chain rule, the latter term on the right-hand-side of (4.12) can be expressed by

$$\text{P}(\mathbf{y}_{n-1} = \mathbf{y}^{(j)} \mid q_{n-1} = \mu, \hat{y}_1^\mu) = \frac{\text{P}(\mathbf{y}_{n-1} = \mathbf{y}^{(j)}, q_{n-1} = \mu, \hat{y}_1^\mu)}{\sum_{f \in \mathcal{I}} \text{P}(\mathbf{y}_{n-1} = \mathbf{y}^{(f)}, q_{n-1} = \mu, \hat{y}_1^\mu)}. \quad (4.14)$$

For a given $\mathbf{y}_{n-1} = \mathbf{y}^{(j)}$ and $q_{n-1} = \mu$, the state $q_{n-2} = \kappa \in \mathcal{Q}_{n-2}$ is uniquely determined by μ and the codeword length of index j , yielding $q_{n-2} = \mu - N^{(j)}$. Therefore, according to the definition of α and γ in (4.6), it can be derived that

$$\begin{aligned} \text{P}(\mathbf{y}_{n-1} = \mathbf{y}^{(j)}, q_{n-1} = \mu, \hat{y}_1^\mu) &= \text{P}(\mathbf{y}_{n-1} = \mathbf{y}^{(j)}, q_{n-1} = \mu, q_{n-2} = \kappa, \hat{y}_1^\mu) \\ &= \gamma_{n-1}(j, \kappa, \mu) \cdot \alpha_{n-2}(\kappa). \end{aligned} \quad (4.15)$$

Applying (4.15) to (4.14), the latter term on the right-hand-side of (4.12) can be written as

$$\text{P}(\mathbf{y}_{n-1} = \mathbf{y}^{(j)} \mid q_{n-1} = \mu, \hat{y}_1^\mu) = \frac{1}{C_2(\mu)} \alpha_{n-2}(\kappa) \cdot \gamma_{n-1}(j, \kappa, \mu), \quad (4.16)$$

with $C_2(\mu) = \sum_{f \in \mathcal{I}} \alpha_{n-2}(\kappa = \mu - N^{(f)}) \cdot \gamma_{n-1}(f, \kappa = \mu - N^{(f)}, \mu)$. It can be seen that each C_2 value corresponds to one state μ . Moreover, if $\kappa \notin \mathcal{Q}_{n-2}$, the terms $\alpha_{n-2}(\kappa)$ and $\gamma_{n-1}(j, \kappa, \mu)$ will be zero.

Finally, note that with $i \in \mathcal{I}$, not all the transitions from $q_{n-1} = \mu$ to $q_n = \mu + N^{(i)}$ are available both in the stationary and the converging stages of the VLC trellis representation in Fig. 4.3. As a result, $C_1'(\mu)$ in (4.11) and $C_1(\mu, j)$ in (4.13) are not always equal to one.

Backward Recursion

Considering the states $\nu \in \mathcal{Q}_n$ at the current symbol time n and $\omega \in \mathcal{Q}_{n+1}$ at the next symbol time $n+1$, the backward recursion is processed as

$$\beta_n(\nu) = \sum_{\omega \in \mathcal{Q}_{n+1}} \sum_{h \in \mathcal{I}} \gamma'_{n+1}(h, \nu, \omega) \cdot \beta_{n+1}(\omega), \quad (4.17)$$

with the initial value of β being $\beta_B(R) = 1$ and

$$\gamma'_{n+1}(h, \nu, \omega) = \text{P}(\hat{y}_{\nu+1}^\omega \mid \mathbf{y}_{n+1} = \mathbf{y}^{(h)}) \cdot \text{P}(\mathbf{y}_{n+1} = \mathbf{y}^{(h)}, q_{n+1} = \omega \mid q_n = \nu), \quad (4.18)$$

which includes a *channel term* and a *source probability distribution term*.

Similar to the forward recursion, the *channel term* is obtained by

$$\mathbb{P}(\hat{y}_{\nu+1}^{\omega} | \mathbf{y}_{n+1} = \mathbf{y}^{(h)}) = \prod_{m=1}^{N^{(h)}} \mathbb{P}(\hat{y}(\nu + m) | y^{(h)}(m)). \quad (4.19)$$

The *source probability distribution term* can be derived by

$$\mathbb{P}(\mathbf{y}_{n+1} = \mathbf{y}^{(h)}, q_{n+1} = \omega | q_n = \nu) = \frac{1}{C_3(\nu)} \begin{cases} \mathbb{P}(\mathbf{y}_{n+1} = \mathbf{y}^{(h)}), & \text{if } \omega - \nu = N^{(h)}, \\ 0, & \text{else.} \end{cases} \quad (4.20)$$

Considering the special case of stationary and converging stages, the normalization is processed as $C_3(\nu) = \sum_{\omega' \in \mathcal{Q}_{n+1}} \sum_{\substack{f \in \mathcal{I} \\ N^{(f)} = \omega' - \nu}} \mathbb{P}(\mathbf{y}_{n+1} = \mathbf{y}^{(f)})$.

4.4.2 Source Symbol Estimation

The minimum mean-square error (MMSE) estimation can maximize the value of the SNR. Therefore, after calculating the APPs, the MMSE is adopted in this chapter [Han and Fingscheidt, 2014a] to estimate the source symbol

$$\hat{u}_n = \sum_{i \in \mathcal{I}} u^{(i)} \cdot \mathbb{P}(\mathbf{y}_n = \mathbf{y}^{(i)} | \hat{y}_1^R), \quad (4.21)$$

applying the APP from (4.6), with $u^{(i)}$ being the quantization codebook entry corresponding to the quantization codebook index i .

4.5 From VL/SD Decoding to FL/SD Decoding

The links between VL/SD and FL/SD decoding are discussed in this section. Applying the APP calculation of VL/SD decoding in Section 4.4.1 to FLCs, a block of B consecutive source symbols, each with N bits, is defined. As $R = 4$ and $R = 12$ with $B = 4$ shown in Fig. 4.4(a), the trellis representation turns out to be a straight line through points $(n, N \cdot n)$, with $n \in \{1, 2, \dots, B\}$ being the symbol time index. For each symbol time index n , only one state q_n exists. Moreover, the variable-length bit combination \mathbf{y}_n turns out to be the fixed-length bit combination $\mathbf{x}_n \in \{0, 1\}^N$. The received bit stream \hat{y}_1^R can be substituted by \hat{x}_1^R , with the total number of bits $R = B \cdot N$. Correspondingly, the received bit sequence $\hat{y}_{\mu+1}^{\nu}$ at symbol time n , which results in changing the state from $q_{n-1} = \mu \in \mathcal{Q}_{n-1}$ to $q_n = \nu \in \mathcal{Q}_n$, is now represented by the bit combination $\hat{\mathbf{x}}_n$ received at symbol time n . The received stream \hat{y}_1^{μ} is replaced by the bit stream $\hat{\mathbf{x}}_1^{\mu} = (\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_{n-1})$ received in symbol times $1, 2, \dots, n-1$, with $\mu = N \cdot (n-1)$. Therefore, the calculation of APPs in (4.6) turns out to be [Han and Fingscheidt, 2015a]

$$\begin{aligned} \mathbb{P}(\mathbf{x}_n = \mathbf{x}^{(i)} | \hat{\mathbf{x}}_1^R) &= \frac{1}{C} \cdot \alpha_{n-1}(\mu) \cdot \beta_n(\nu) \cdot \gamma_n(i, \mu, \nu) \\ &= \frac{\alpha_{n-1}(\mu) \cdot \beta_n(\nu) \cdot \gamma_n(i, \mu, \nu)}{\sum_{f \in \mathcal{I}'} \alpha_{n-1}(\mu) \cdot \beta_n(\nu) \cdot \gamma_n(f, \mu, \nu)} \\ &= \frac{1}{C} \cdot \gamma_n(i, \mu, \nu), \end{aligned} \quad (4.22)$$

with i being the corresponding codebook index for FLCs. Note that $\mu = N \cdot (n - 1)$ and $\nu = N \cdot n$ are now unambiguously defined for any symbol time n , therefore the double sum in (4.6) vanishes. Equation (4.22) indicates that the APP calculation for FLCs is actually the computation of the γ term (4.8). Including the *channel term* and the *source probability distribution term*, the γ calculation (4.8) for FLCs becomes

$$\gamma_n(i, \mu, \nu = \mu + N) = P(\hat{\mathbf{x}}_n | \mathbf{x}_n = \mathbf{x}^{(i)}) \cdot P(\mathbf{x}_n = \mathbf{x}^{(i)} | \hat{\mathbf{x}}_1^\mu). \quad (4.23)$$

The *channel term* $P(\hat{\mathbf{x}}_n | \mathbf{x}_n = \mathbf{x}^{(i)})$ can be obtained based on (4.9), with $N^{(i)}$ being equal to the fixed N . The second term, the *source probability distribution term*, reduces from (4.11) or (4.12) to the given simple form again since the states q_n at each symbol time instant n are unambiguous with probability one.

APPs With AK0

The calculation of the *source probability distribution term* with AK0 in (4.11) for FLCs does not show dependency from previously (received) bits $\hat{\mathbf{x}}_1^\mu$, therefore it turns out to be

$$P(\mathbf{x}_n = \mathbf{x}^{(i)} | \hat{\mathbf{x}}_1^\mu) = P(\mathbf{x}_n = \mathbf{x}^{(i)}). \quad (4.24)$$

As a result, applying (4.24) to (4.23) and (4.22), and noting that $\alpha_{n-1}(\mu) \cdot \beta_n(\nu)$ do not depend on f and there cancel each other out in the numerator and the denominator in (4.22), the APPs can be calculated by

$$P(\mathbf{x}_n = \mathbf{x}^{(i)} | \hat{\mathbf{x}}_1^R) = \frac{1}{C} \cdot P(\hat{\mathbf{x}}_n | \mathbf{x}_n = \mathbf{x}^{(i)}) \cdot P(\mathbf{x}_n = \mathbf{x}^{(i)}), \quad (4.25)$$

which actually results in the same APP calculation as in the FL/SD decoding approach (2.26) in Section 2.3.2.

APPs With AK1

The *source probability distribution term* with AK1 (4.12) for FLCs accordingly is⁴:

$$P(\mathbf{x}_n = \mathbf{x}^{(i)} | \hat{\mathbf{x}}_1^\mu) = \sum_{j \in \mathcal{I}} P(\mathbf{x}_n = \mathbf{x}^{(i)} | \mathbf{x}_{n-1} = \mathbf{x}^{(j)}) \cdot P(\mathbf{x}_{n-1} = \mathbf{x}^{(j)} | \hat{\mathbf{x}}_1^\mu). \quad (4.26)$$

Applying (4.26) to (4.23) and (4.22), we obtain

$$P(\mathbf{x}_n = \mathbf{x}^{(i)} | \hat{\mathbf{x}}_1^R) = \frac{1}{C} \cdot P(\hat{\mathbf{x}}_n | \mathbf{x}_n = \mathbf{x}^{(i)}) \cdot \sum_{j \in \mathcal{I}} P(\mathbf{x}_n = \mathbf{x}^{(i)} | \mathbf{x}_{n-1} = \mathbf{x}^{(j)}) \cdot P(\mathbf{x}_{n-1} = \mathbf{x}^{(j)} | \hat{\mathbf{x}}_1^\mu), \quad (4.27)$$

which again leads to the same APP calculation as in the FL/SD decoding (2.27) in Section 2.3.2.

⁴Note that only a pure forward recursion is included in the APP calculation for the FLCs, no backward recursion as in VL/SD decoding is performed, due to the fact that both the forward recursion term $\alpha_{n-1}(\mu)$ and the backward recursion term $\beta_n(\nu)$ are canceled out in the numerator and denominator in (4.22). Only the $\gamma_n(i, \mu, \nu)$ term is left, which is actually a forward recursion [Kliever and Thobaben, 2005].

4.6 Simulation Setup and Results

The performance tradeoffs of the VL/SD and FL/SD decoding depending on the quantization bit rate, block length, and source correlation are discussed in this section.

4.6.1 Simulation Setup

The source symbols are taken from the AR(1) process with the i.i.d.⁵ zero-mean, unit-variance Gaussian innovation and the correlation coefficient being 0.9. Lloyd-Max quantizers with 2, 4, and 6 bit are employed in three separate simulations. For the training process to obtain the AK0 and AK1 terms, a number of 10^8 source symbols is utilized in all simulations. A number of 64000 fixed source symbols is used for testing in all simulations. In each simulation, the source symbols are divided into block lengths B of 1, 2, 4, 8, 16, 32, 64, and 128, respectively. Each block is transmitted over different channel realizations for a given range of E_b/N_0 . At the end, the performance is evaluated according to the global SNR of the fixed 64000 source symbols. Moreover, all terms in the formulae in Section 4.4.1 are implemented in the log domain in order to avoid the numerical issues in good channel transmission conditions. The APPs are transformed back to the linear domain before estimating a source symbol (4.21).

Considering the different bit rates between VLCs and FLCs, a fair comparison between VL and FL decoding needs to be ensured. Therefore, the *same (average) energy per source symbol* E_s is assumed for all simulations: $M \cdot E_b^{\text{FLC}} = E_s = \bar{M} \cdot E_b^{\text{VLC}}$ for an M bit quantizer, with $\bar{M} \in \{1.9896, 3.8093, 5.8697\}$ being the average Huffman codeword length⁶ of the VLCs for $M \in \{2, 4, 6\}$ bit. In other words, in the dB domain, $E_b/N_0 = E_s/N_0 - 10 \cdot \log_{10}(\bar{M})$ for VLCs and $E_b/N_0 = E_s/N_0 - 10 \cdot \log_{10}(M)$ for FLCs. The 3 quantizers shall be tested in roughly the same E_b/N_0 range, therefore E_s/N_0 is varied in a range of [-2, 13] dB, [1, 16] dB, and [3, 18] dB in the simulations for $M \in \{2, 4, 6\}$ bit, respectively. Moreover, based on the AK0 term $P(\mathbf{y}_n = \mathbf{y}^{(i)})$, the VLCs are generated by the standard Huffman coding [Huffman, 1952], which leads to three separate Huffman codebooks for $M \in \{2, 4, 6\}$ bit.

The simulation results of SD decoding with three algorithms are compared in this section: the VL/SD decoding (from Section 4.4.1), the FL/SD decoding (from Section 2.3.2) with only forward recursion [Fingscheidt and Vary, 2001], and the original BCJR algorithm [Bahl et al., 1974] having both forward and backward recursions.

4.6.2 Simulation Results

The SNR results of the $M = 2, 4$, and 6 bit quantized Gaussian AR(1) process with different block lengths B are shown in Figs. 4.5, 4.6, and 4.7, respectively. In all three

⁵independently and identically distributed

⁶The average Huffman codeword length is obtained by $\sum_{i=0}^{2^M-1} N^{(i)} \cdot P(\mathbf{y}_n = \mathbf{y}^{(i)})$.

figures, HD decoding is denoted as HD, SD decoding with zeroth-order and first-order *a priori* knowledge are represented as AK0 and AK1. For FL/SD decoding, the APP calculation with only forward recursion is identified as FL (FA), while FL (FBA) represents the original BCJR algorithm⁷.

As shown in all figures, both for FLCs and VLCs, if no source correlation but histogram-related residual redundancy is used (AK0), SD decoding outperforms HD decoding in all cases. Further improvements can be observed if the source correlation is utilized in addition (AK1). However, as expected, for $B=1$ with all the quantization bit rates, using AK0 leads to the same performance as using AK1, both for VL/SD and FL/SD decoding (either with FBA or FA), since temporal correlation is not modeled between blocks. Moreover, for the worst channel conditions, all SD results show a minimum of SNR = 0 dB [Fingscheidt and Vary, 2001], which can be explained by the muting of the parameters as a consequence of MMSE estimation.

For FL/SD decoding with AK1, FL (FBA) outperforms FL (FA) for all the quantization bit rates and all the block lengths except $B=1$, since the performance of FL (FBA) is the same as FL (FA) if $B=1$.

The influence of the block length for FLCs and VLCs can be investigated by varying the block length. For a given channel condition, on the one hand, using either HD or AK0, the performance for FL decoders with different block lengths remains constant, for all the quantizers. On the other hand, increasing the block length, an enhanced performance can be observed for FL (FBA)/AK1 and FL (FA)/AK1 due to a reduced influence of initialization at the block boundaries. In addition, this effect is also noticeable for VL/AK1 with shorter block lengths ($B \leq 16$), especially with medium to good channel conditions, for all the quantizers. One possible reason for VL/AK1 is that the initialization (using AK0) may dominate if the block length is short enough, while more previous information is available for long block lengths. In contrast, for both VL/HD and VL/AK0, the VL decoders always lose performance for longer block lengths, because of more error propagation in identifying symbol boundaries. This effect is valid for VL/AK1 with longer block lengths ($B > 16$) as well.

The comparison between FLCs and VLCs for HD, AK0, and AK1 depending on the block length and quantization bit rate is discussed below.

(1) *FL/HD vs. VL/HD*

In general, the FL/HD exceeds the VL/HD scheme, especially for long block lengths. However, for $M=4$ bit at very bad channel qualities (see Fig. 4.6), for very short block lengths ($B \leq 4$) VL/HD performs even slightly better than the FL/HD approach. The intersection of FL/HD and VL/HD in terms of E_s/N_0 is decreasing along with the increase of block length, until $E_s/N_0 \approx 1.5$ dB for $B=4$ is reached. This shows that in such bad transmission conditions, the VL/HD scheme offers a comparable robustness as the FL/HD scheme for $M=4$ bit.

⁷Note that as mentioned in Section 2.3.2, the FL (FBA)/AK0 in fact is the FL (FA)/AK0, since no source correlation is taken into account.

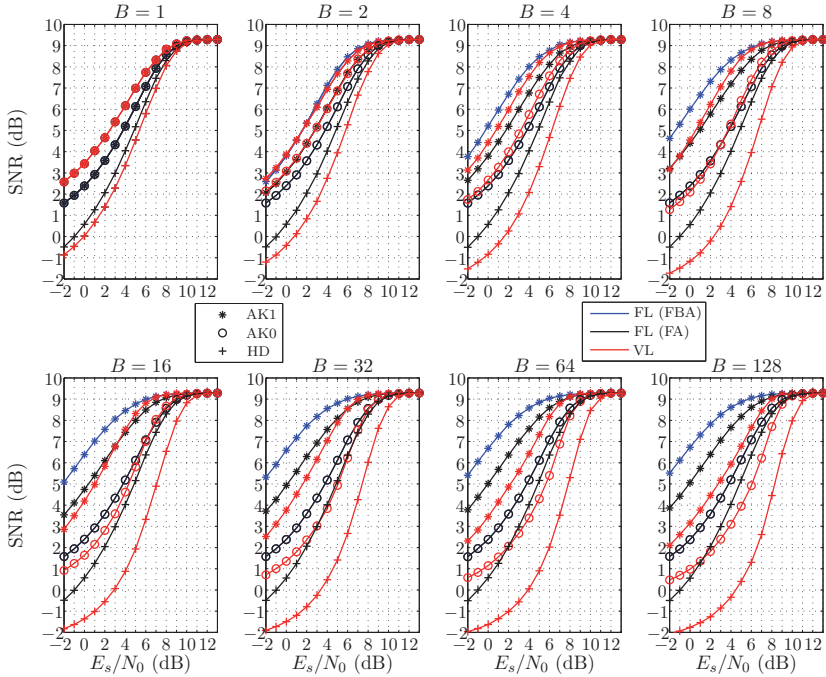


Figure 4.5: Simulation results for an $M=2$ bit quantized Gaussian AR(1) process with different block lengths B (see Tabs. E.9 and E.10).

(2) *FL/AK0 vs. VL/AK0*

The comparison between $FL/AK0^8$ and $VL/AK0$ reveals a different behavior for different quantization bit rates, even though $FL/AK0$ outperforms the $VL/AK0$ scheme in many cases. However, $VL/AK0$ exceeds the $FL/AK0$ for $M=2$ bit (Fig. 4.5) with $B \leq 4$ and $M=4$ bit (Fig. 4.6) with $B=1$ for all channel conditions. Moreover, for the other cases with $M=2$ and 4 bit, the $VL/AK0$ approach performs better than the $FL/AK0$ method only for short block lengths ($B \leq 16$) with medium to good channel qualities (compare to [Park and Miller, 2000]). The quality crossover between $VL/AK0$ and $FL/AK0$ with respect to E_s/N_0 increases along with the rise of block length. Finally, for $M=6$ bit (Fig. 4.7), the superiority of $VL/AK0$ is only valid for $B=1$ with $E_s/N_0 \geq 12$ dB and $B=2$ with $E_s/N_0 \geq 15$ dB. Using $AK0$ is equivalent to using $AK1$ for uncorrelated source symbols, therefore, it can be stated that if SD decoding is employed in the above specific conditions and the use of an AWGN channel, VLC may be a better choice than FLC only for uncorrelated sources. This effect is especially noticeable for lower rate quantization and shorter block lengths, with medium to good channel qualities.

⁸ $FL(FBA)/AK0=FL(FA)/AK0$

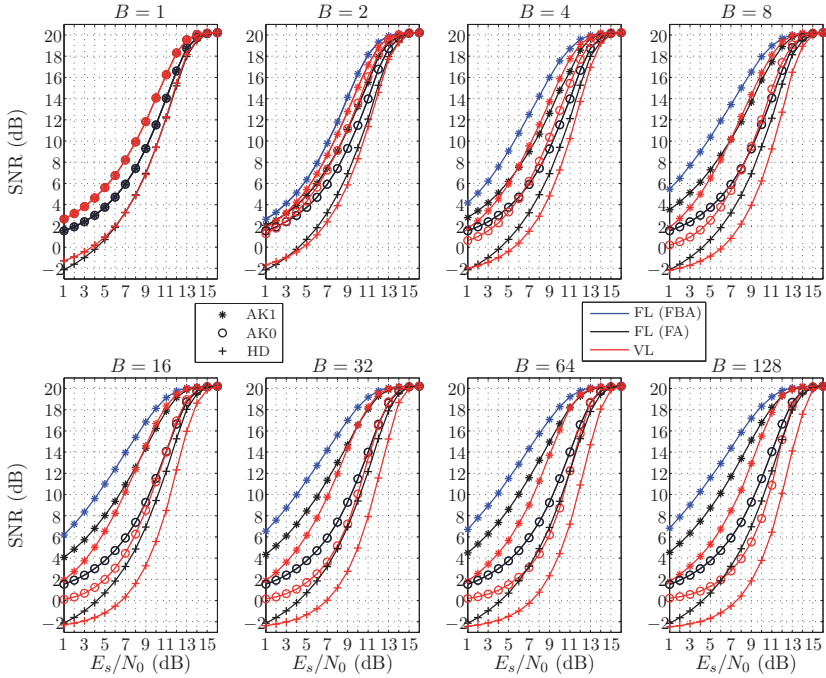


Figure 4.6: Simulation results for an $M=4$ bit quantized Gaussian AR(1) process with different block lengths B (see Tabs. E.11 and E.12).

(3) *FL/AK1 vs. VL/AK1*

Comparing FL (FBA)/AK1 and VL/AK1 with all three quantizers, it can be observed that FL (FBA)/AK1 performs better than VL/AK1 for most conditions, except the following three cases (none of them practically relevant due to the very short block length):

- $M=2$: $B=1$ with all channel conditions and $B=2$ with $E_s/N_0 \leq 0$ dB;
- $M=4$: $B=1$ with all channel conditions;
- $M=6$: $B=1$ with $E_s/N_0 \geq 12$ dB.

Moreover, comparing FL (FA)/AK1 and VL/AK1, firstly for $M=2$ bit (Fig. 4.5), it is found that VL/AK1 exceeds FL (FA)/AK1 for $B \leq 4$ for all channel conditions and $B = 8, 16, 32$ for medium to good channel conditions. Secondly, if $M=4$ bit (Fig. 4.6), VL/AK1 performs better than the FL (FA)/AK1 for $B = 1$ in all channel conditions and $B > 1$ at medium to good channel qualities. The intersection of VL/AK1 and FL (FA)/AK1 is increased from $E_s/N_0 = 2$ dB to $E_s/N_0 = 12$ dB along with increasing the block length from $B = 2$ to $B = 128$. Finally, for the case of $M=6$ bit (Fig. 4.7), FL (FA)/AK1 either outperforms VL/AK1 or performs similar to VL/AK1.

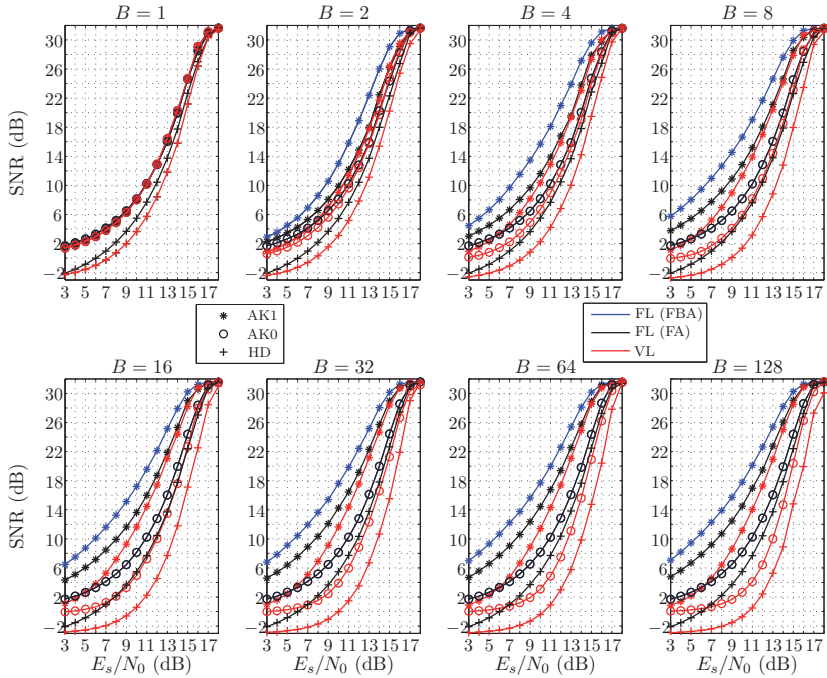


Figure 4.7: Simulation results for an $M=6$ bit quantized Gaussian AR(1) process with different block lengths B (see Tabs. E.13 and E.14).

As a result, for correlated source symbols, the FL (FBA) scheme turns out to be more powerful than the VL/SD method for $B > 1$ in general.

4.7 Summary

The variable-length soft-decision (VL/SD) decoding approach is presented in this chapter. Different to fixed-length hard-decision (FL/HD) decoding in Chapter 2, VL/HD decoding needs a *specific decoder-sided* lookup table. After introducing the VL/HD decoding process, VL/SD decoding adopted from [Kliewer and Thobaben, 2005] is presented in this chapter in many details, including the trellis representation with detailed definitions of stage boundaries and state intervals, the *a posteriori* probability (APP) calculation, and source symbol estimation. Moreover, the links between FL/SD and VL/SD decoding are discussed, which show that the APP calculation of VL/SD decoding for FLCs in fact is equivalent to the APP calculation in Chapter 2. Finally, the performance tradeoffs of FL/SD and VL/SD decoding under different conditions (block length, quantization bit rate, and source correlation) are discussed. The simulation results show that the VL/SD

decoding approach outperforms the FL/SD decoding method for uncorrelated sources and short block lengths with medium to good channel qualities, especially for a lower quantization bit rate. For correlated sources, the FL/SD decoding scheme is a better choice than the VL/SD decoding scheme in most cases, excluding low rate quantization with very short block length, under good channel conditions. The approach presented in this chapter will be applied in Chapter 5 to improve the audio codec performance.

Chapter 5

Improving HE-AAC by Soft-Decision Decoding

As mentioned in Chapter 4, variable-length codes (VLCs) are adopted in the MPEG-4 High-Efficiency Advanced Audio Coding (HE-AAC) audio codec offering a better coding efficiency. HE-AAC has originally been designed for error-free transmission and storage conditions. In HE-AAC, the global gain parameter global gain is coded with fixed-length codes (FLCs), while the scale factors and quantized spectral coefficients are coded with VLCs. Due to the low robustness of HE-AAC transmission over bit error-prone channels, a robust HE-AAC decoder employing both FL/soft-decision (SD) and VL/SD decoding is proposed in this chapter. Section 5.1 introduces some background to the HE-AAC audio codec. The standard HE-AAC audio coding is described in more detail in Section 5.2, including AAC encoding, AAC decoding, and the bit stream structure of HE-AAC. The HE-AAC error concealment from 3GPP Technical Specification TS 26.402 is explained in Section 5.3. The applications of FL/SD decoding (from Section 2.3) to the global gain and VL/SD decoding (from Section 4.4) to the scale factors and the quantized spectral coefficients are presented in Section 5.4. Section 5.5 presents the simulation setup and discusses the simulation results.

5.1 Introduction

Based on the advanced audio coding (AAC), the MPEG-4 High-Efficiency Advanced Audio Coding (HE-AAC) audio codec has significantly increased the coding efficiency and has been optimized for low bit rate applications. HE-AAC can deliver high audio quality even for applications limited in storage capacity or transmission bandwidth, such as digital radio and TV broadcasting, mobile music and TV (e.g., music downloads, music streaming, the audio part of a mobile TV broadcasting system), etc. [Herre and Dietz, 2008].

The core of the HE-AAC is the AAC perceptual waveform coding, which can regenerate the waveform of the original audio input signal with high fidelity by using a minimum

amount of data and considering psychoacoustic principles. Due to the fact that the human ear is less sensitive to the high frequency range of audio signals, instead of transmitting the upper part of the spectrum with AAC, the spectral band replication (SBR), which performs the high frequency reconstruction and envelope adjustment, reproduces the higher-frequency components on the basis of the components from the lower frequency range. Furthermore, the parametric stereo (PS) compression tool ensures efficient stereo coding at very low bit rates [Herre and Dietz, 2008].

Several so-called audio profiles have been defined in ISO/IEC 14496-3:2005 [ISO/IEC, 2005]. As a superset of the AAC profile containing the audio object type AAC LC (AAC in its low-complexity (LC) version), the high efficiency AAC profile incorporates the audio object types AAC LC and SBR. The first release of HE-AAC (HE-AAC v1) combines AAC-LC and SBR, while PS is further incorporated in HE-AAC v2 [Herre and Dietz, 2008]. HE-AAC v2 has also been standardized as 3GPP Enhanced aacPlus general audio codec in 3GPP Technical Specification TS 26.401 [3GPP, 2004a], which implements the high efficiency AAC profile at level 2 restricting the number of used channels to a maximum of two. In addition, HE-AAC operates in a dual rate mode, which means that the AAC encoder operates at half of the encoding sampling rate $f_s/2$, with f_s being the sampling rate of SBR. Therefore, for the high efficiency AAC profile at level 2, the maximum AAC sampling rate is restricted to 24 kHz if SBR is used, and to 48 kHz in case that SBR is not being used.

As mentioned in [Herre and Dietz, 2008], the SBR and PS tools were designed for the demand of audio coding with increased coding efficiency at low bit rates (24 kbps per audio channel). The typical bit rates used for HE-AAC v2 are 24 to 32 kbps [Fraunhofer-Gesellschaft, 2013]. The HE-AAC v2 audio codec is adopted in the digital audio broadcasting (DAB+) system, which allows bit rates from 8 to 192 kbps per audio channel [Hoeg and Lauterbach, 2009, Kozamernik, 1995, Eureka-147, 1997]. Therefore, considering a focus on low bit rate applications (e.g., mobile audio streaming), HE-AAC operating at 24 kbps¹ with 48 kHz sampling rate is adopted in this thesis, with a focus on monophonic audio signals. In other words, HE-AAC v1 including the AAC and SBR tools are employed in this thesis. For stereo signals, in addition the PS tool will have to be used (i.e., HE-AAC v2). Note that the approach proposed in this chapter is applied to the AAC part, which implies that the SBR (or PS) part is assumed error-free. Therefore, the proposed approach can be applied to AAC, HE-AAC v1, or HE-AAC v2. Moreover, in the SBR tool, the parameters envelope scale factors and the noise floor scale factors are delta coded using a number of 10 different Huffman codebooks [3GPP, 2004b, ISO/IEC, 2005]. The stereo parameters inter intensity difference (IID) and inter channel coherence (ICC) in the PS part are coded with two respective Huffman codebooks [3GPP, 2004c, ISO/IEC, 2005]. The approach mentioned in Section 5.4.2 could therefore be applied to the above mentioned parameters, as long as the complexity is acceptable.

In order to improve the coding efficiency, VLCs are adopted in the HE-AAC bit stream. However, as mentioned in Section 4.1, significant signal distortions can occur even because of single bit errors, for bit error-prone (unreliable) transmission conditions.

¹One example of application is a DAB+ radio sport station in Melbourne, Australia [Wohnort, 2013].

Error propagation turns out to be a serious problem for an HE-AAC decoder. As a result, a robust source decoder is desired for the application of HE-AAC under unreliable transmission conditions (e.g., applications of streaming audio over wireless networks).

Several methods have been proposed to increase the robustness of HE-AAC. Changing the bit stream syntax, HE-AAC can be combined with the error protection tool applied in the error resilient audio object types (e.g., ER AAC LC) in the ISO/IEC standard [ISO/IEC, 2005], which provides unequal error protection (UEP) capability by employing a forward error correction (FEC) and/or a cyclic redundancy check (CRC). As a further option specified in 3GPP Technical Specification TS 26.402 [3GPP, 2004d], error concealment performs an interpolation of the corrupted spectral coefficients still before frequency-to-time conversion. However, the corruption of spectral coefficients is communicated by only a binary reliability information per frame, which is a very coarse BFI-like message. In addition, a delay of one frame is introduced by that error concealment.

Sections 2.3 and 4.1 pointed out that soft-decision (SD) decoding can be considered as a more robust means for bit error concealment, which expects soft information representing bit-wise channel reliability information. In AAC, the global gain parameter is FL-coded with 8 bits, while the scale factors and quantized spectral coefficients are both VL-coded. In [Derrien et al., 2008], VL/SD decoding is applied to the scale factors in AAC, however, only assuming the bits of the scale factors and spectral coefficients to be corrupted. In this chapter, the bits of the global gain, the scale factors, and the quantized spectral coefficients are assumed to be distorted. On this basis, the FL/SD decoding from Section 2.3 is applied to the global gain, while the VL/SD decoding from Section 4.4 is applied to the scale factors and the quantized spectral coefficients.

5.2 HE-AAC Audio Coding

The core of the HE-AAC audio coding is the AAC coding algorithm. This section introduces AAC encoding, AAC decoding, and the HE-AAC bit stream structure described in [3GPP, 2004e] and [ISO/IEC, 2005] in detail.

5.2.1 AAC Encoding

A simplified block diagram of the AAC encoder is depicted in Fig. 5.1 [Salomon and Motta, 2010], which includes a filterbank, the determination of scale factors and quantization of spectral coefficients, and both FL and VL encoders.

Filterbank

The modified discrete cosine transform (MDCT) filterbank transforms the time domain input audio signals to the frequency domain, providing the spectral coefficients

$$\tilde{X}_k = 2 \cdot \sum_{n=0}^{K-1} s_n \cdot \cos\left(\frac{2\pi}{K}\left(n + \frac{K/2+1}{2}\right)\left(k + \frac{1}{2}\right)\right), \quad (5.1)$$

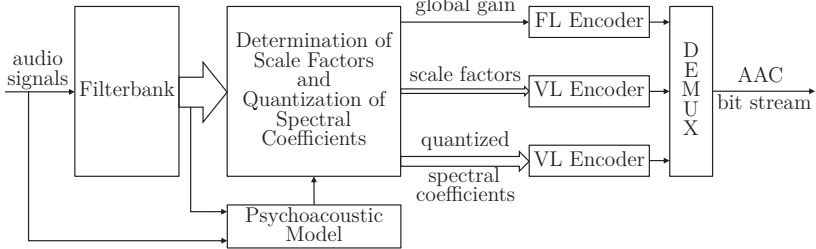


Figure 5.1: Simplified block diagram of an AAC encoder [Han and Fingscheidt, 2015a].

with the spectral coefficient index $k \in \mathcal{K} = \{0, \dots, \frac{K}{2} - 1\}$, the sample time index n , the windowed input audio sequence s_n , and either $K = 2048$ for the long window type, or $K = 256$ for the short window type.

Psychoacoustic Model and Reduction of Psychoacoustic Requirements

A threshold related to the maximum quantization noise just being masked by the signal energy is calculated by the psychoacoustic model [ISO/IEC, 2004]. Based on the threshold, a reduction of the psychoacoustic requirements (by increasing the threshold) is necessary, due to the fact that the requirements of the psychoacoustic model are normally too strong for the desired bit rate. The threshold reduction strategy first modifies the thresholds, then estimates the bit demand, and finally calculates the reduction values.

The psychoacoustic model works along with quantization and coding. The resulting (increased) threshold θ_b and (unquantized) spectral coefficients \tilde{X}_k from the MDCT calculation are required for the following process of quantization and coding, with $b \in \mathcal{B} = \{0, 1, \dots, B-1\}$ being the scale factor band index, which will be explained in the following in detail.

Determination of Scale Factors and Quantization of Spectral Coefficients

Before explaining the quantization and coding procedure, first four terminologies used in AAC are introduced: scale factor band, gain value, scale factor, and global gain. In AAC, with the help of the so-called *gain values*² sf_{gb} , the spectral coefficients \tilde{X}_k are quantized. For the purpose of controlling the quantization noise in the frequency domain, the spectrum is divided into several groups of consecutive spectral coefficients, named as *scale factor bands*. The width of each scale factor band (i.e., the number of spectral coefficients X_k) is based on the critical bands of the human auditory system, with each scale factor band containing an integer multiple of four spectral coefficients. The maximum number of scale factor bands³ B_{\max} and the width of each scale factor band are determined by the sampling frequency and the window type, as defined in Tabs. 4.117 (for long window type) and 4.118 (for short window type) in [ISO/IEC, 2005] for AAC

² sf_{gb} is denoted as `scfGain` in [3GPP, 2004e].

³For AAC operating at 24 kHz sampling rate, $B_{\max} = 47$ for $K = 2048$ and $B_{\max} = 15$ for $K = 256$.

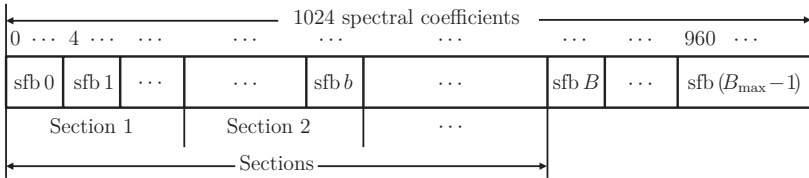


Figure 5.2: Scale factor bands (sfbs) for long window types with a window length of $K = 2048$ for AAC operating at 24 kHz sampling rate. On top the lowest spectral coefficient index $k_{\min,b}$ of each sfb b is shown.

operating at 24 kHz sampling rate. Fig. 5.2 gives an example of the scale factor bands for a long window type with a window length of $K = 2048$ at 24 kHz sampling rate (Tab. 4.117 in [ISO/IEC, 2005]), with sfb being the abbreviation of scale factor band. The meaning of sections will be explained along with the noiseless coding further on. Additionally, in order to reduce the amount of transmitted information, not all scale factor bands are transmitted. The number of transmitted scale factor bands B implies that the spectral coefficients \tilde{X}_k from scale factor band B to $B_{\max}-1$ are all 0, and neither scale factors nor quantized spectral coefficients are transmitted. In each scale factor band $b \in \mathcal{B} = \{0, 1, \dots, B-1\}$, the first (lowest) and the last (highest) spectral coefficient indices of the scale factor band b are denoted as $k_{\min,b}$ and $k_{\max,b}$, respectively, with $k_{\min,b}$ and $k_{\max,b}$ being defined in Tabs. 4.117 and 4.118 in [ISO/IEC, 2005]. Fig. 5.2 shows the lowest spectral coefficient index $k_{\min,b} \in \{0, 4, \dots, 960\}$. Each scale factor band b shares one gain value sf_{g_b} applied to change the amplitude of all spectral coefficients in that scale factor band. Correspondingly, $sf_b = g - sf_{g_b}$ is called the *scale factor* of scale factor band b , with g being the so-called *global gain*, which is a common reference value to all scale factor bands.

After the filterbank and psychoacoustic model, in combination with the determination of scale factors sf_b , the spectral coefficients \tilde{X}_k are quantized to X_k [3GPP, 2004e]. In each frame⁴, this iterative process is on the basis of a search for an optimal gain value $sf_{g_b} = g - sf_b$ for each scale factor band b . According to the threshold θ_b , starting with a first guess of sf_{g_b} , first the preliminary optimal sf_{g_b} is found by searching the minimum quantization distortion (steps (1)-(3) as described below). Subsequently, the optimal sf_{g_b} satisfying a compromise between a small bit demand for the coding of scale factors and a small distortion needs to be found (step (4)). Once the optimal gain values sf_{g_b} are determined, the corresponding quantized spectral coefficients X_k are also obtained. Finally, based on the sf_{g_b} for each scale factor band, the global gain g is chosen as the maximum value of sf_{g_b} over all the scale factor bands. The scale factor sf_b for each scale factor band is correspondingly derived at the end (step (5)). The details of each step are described as follows.

The flowchart of the first search to find the preliminary optimal gain values sf_{g_b} is shown in Fig. 5.3. Fig. 5.4 depicts the flowchart of the final determination of the global gain, the scale factors, and the quantized spectral coefficients.

⁴In the following, the frame index ℓ is omitted, unless it is explicitly mentioned.

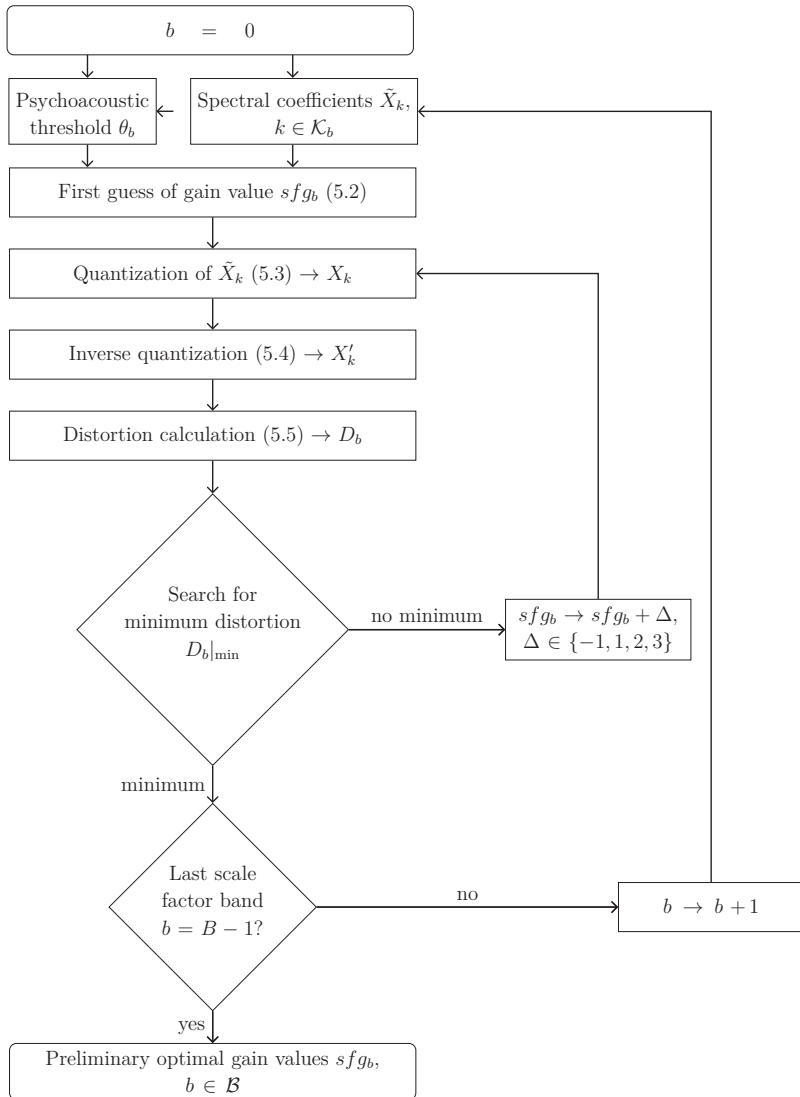


Figure 5.3: Flowchart of the search for a preliminary optimal $sf g_b$ for each scale factor band b in the core AAC encoder.

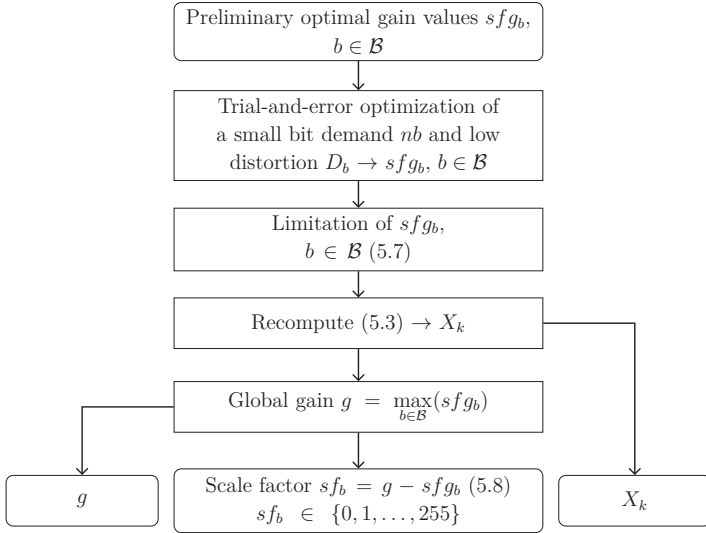


Figure 5.4: Flowchart in continuation of Fig. 5.3 of the final determination of the global gain g , the scale factors sf_b , and the quantized spectral coefficients X_k in the AAC core encoder.

(1) *First Guess*

In each scale factor band b , by the first guess of sf_{g_b} , the quantization noise should be approximately equivalent to the threshold θ_b , which is calculated from the threshold reduction strategy. The first guess of sf_{g_b} is processed by employing (5.1) and

$$sf_{g_b} = \left\lceil 8.8585 \cdot (\log_{10}(6.75 \cdot \theta_b)) - \log_{10} \left(\sum_{k \in \mathcal{K}_b} \sqrt{|\tilde{X}_k|} \right) \right\rceil, \quad (5.2)$$

with $k \in \mathcal{K}_b = \{k_{\min,b}, \dots, k_{\max,b}\}$.

(2) *Quantization and Inverse Quantization of Spectral Coefficients*

The spectral coefficients \tilde{X}_k in scale factor band b are quantized to

$$X_k = \text{sign}(\tilde{X}_k) \cdot \left\lceil \left((|\tilde{X}_k|) \cdot 2^{-\frac{1}{4} \cdot sf_{g_b}} \right)^{\frac{3}{4}} + 0.4054 \right\rceil = X_k(sf_{g_b}), \quad (5.3)$$

with X_k being the quantized spectral coefficient⁵. The quantization and so-called *inverse quantization* are always jointly processed to calculate and compare the quantization error. The inverse quantization, which is also performed in the decoder (see Fig. 5.7), is processed by

$$X'_k = \text{sign}(X_k(sf_{g_b})) \cdot |X_k(sf_{g_b})|^{\frac{4}{3}} \cdot 2^{\frac{1}{4} \cdot sf_{g_b}} = X'_k(sf_{g_b}), \quad (5.4)$$

with X'_k being the so-called inversely quantized spectral coefficients.

⁵The maximum absolute value is defined as 8191.

(3) *Scale Factor Improvement by Quantization of Spectral Coefficients*

As shown in Fig. 5.3, after calculating the $sf g_b$ in (5.2), the quantized spectral coefficients X_k and the inversely quantized spectral coefficients X'_k are sequentially computed by (5.3) and (5.4), respectively. Thereafter, the distortion⁶ D_b in scale factor band b is defined by

$$D_b = \sum_{k \in \mathcal{K}_b} (\tilde{X}_k - X'_k)^2. \quad (5.5)$$

Increasing and decreasing the values of $sf g_b$ ($sf g_b \rightarrow sf g_b + \Delta$, $\Delta \in \{-1, 1, 2, 3\}$), the minimum distortion is found by $D_b|_{\min} = \min_{sf g_b} \sum_{k \in \mathcal{K}_b} (\tilde{X}_k - X'_k(sf g_b))^2$. After searching for all the scale factor bands, the preliminary optimal gain values $sf g_b$ are obtained, with $b \in \mathcal{B}$.

(4) *Scale Factor Difference Reduction*

In each frame, the *differential scale factor* $dsf_b = sf_b - sf_{b-1}$, which is the difference of the scale factors between two adjacent scale factor bands is actually coded. To accomplish this, a smaller difference requiring less bits needs to be found. As shown in Fig. 5.4, based on the values of the preliminary optimal $sf g_b$ in each scale factor band from Fig. 5.3, the search in this iterative process is to further find an optimal $sf g_b$ which satisfies a compromise between a small estimated bit demand (i.e., number of bits required) nb and a small distortion D_b , in a kind of trial-and-error fashion. The estimation of the bit demand nb is computed by:

$$nb = \begin{cases} 0.7 \cdot nl \cdot ld, & \text{if } ld \geq c_1, \\ 0.7 \cdot nl \cdot ld \cdot (c_2 + c_3 \cdot ld), & \text{else.} \end{cases} \quad (5.6)$$

Herein, the estimated number of spectral lines⁷ nl , which will not be zero after the quantization, is already computed in the reduction of psychoacoustic requirements: $nl = \left(\sum_{k \in \mathcal{K}_b} \sqrt{|\tilde{X}_k|} \right) / \left(\frac{en_b}{|\mathcal{K}_b|} \right)^{0.25}$; $ld = \log_2(en_b) - 0.375 \cdot sf g_b$, with the scale factor band energy en_b being already calculated in the psychoacoustic model: $en_b = \sum_{k \in \mathcal{K}_b} \tilde{X}_k^2$; $c_1 = \log_2(8)$, $c_2 = \log_2(2.5)$, and $c_3 = 1 - c_2/c_1$. Note that (5.3), (5.4), (5.5), and (5.6) are always recalculated once $sf g_b$ is varied.

(5) *Final Scale Factor Determination*

The optimal gains $sf g_b$ shall be differentially coded in the form of scale factors $sf_b = g - sf g_b$. The maximum difference between two optimal gains is $\max_{b \in \mathcal{B}}(sf g_b) - \min_{b' \in \mathcal{B}}(sf g_{b'}) \leq 60$. As a result, after obtaining $sf g_b$ for all the scale factor bands, all the values of $sf g_b$ are limited to

$$sf g_b \leq \min_{b \in \mathcal{B}}(sf g_b) + 60, \quad (5.7)$$

with $\min_{b \in \mathcal{B}}(sf g_b)$ being the minimum value of the $sf g_b$ over all the scale factor bands. If the value of $sf g_b$ is changed, the quantized spectral coefficients X_k are correspondingly

⁶ D_b is denoted as `sfbDist` in [3GPP, 2004e].

⁷ nb , ld , and nl are denoted as `nBitsestim`, `ldRatio`, and `nLines` in [3GPP, 2004e], respectively.

Global gain g	Codebook index i_g	Codeword
0	0	00000000
1	1	00000001
...
255	255	11111111

Table 5.1: Global gain codebook. The codewords are given in binary notation and the codeword length is always 8 bits (FL code).

recalculated by (5.3). As a result, all the final quantized spectral coefficients X_k are simultaneously determined during the process.

At the end, the global gain value g is chosen as the maximum value of $sf g_b$ for all the scale factor bands: $g = \max_{b \in \mathcal{B}}(sf g_b)$.

Due to $sf g_b = g - sf_b$, the scale factors are finally obtained by

$$sf_b = g - sf g_b. \quad (5.8)$$

The resulting global gain g , scale factors sf_b , and quantized spectral coefficients X_k will be coded by the subsequent noiseless coding stage.

Noiseless Coding

The redundancy of the scale factors and the quantized spectrum is reduced by noiseless coding. As shown in Tab. 5.1 with a binary representation for the codewords, the global gain g is coded by an 8 bit PCM FLC, with the global gain codebook index $i_g \in \mathcal{I}_g = \{0, 1, \dots, 2^8 - 1\}$.

The differential scale factor⁸ $dsf_b = sf_b - sf_{b-1}$ is VL-coded with a differential scale factor Huffman codebook, which is shown in Tab. 5.2 with a hexadecimal representation for the codewords. The differential scale factor satisfies $dsf = i_s - 60$, with the differential scale factor Huffman codebook index $i_s \in \mathcal{I}_s = \{0, 1, \dots, 120\}$ (Tab. 4.A.1 in [ISO/IEC, 2005]).

Sectioning is performed for the coding of the quantized spectral coefficients X_k , by a so-called greedy merge algorithm which minimizes the number of bits for the quantized spectral coefficients [3GPP, 2004e, ISO/IEC, 2004]. As shown in Fig. 5.2, a number of 1024 coefficients in a frame is segmented into sections. For each section of the spectrum, the best spectrum Huffman codebook (from Tab. 4.A.2 to Tab. 4.A.12 in [ISO/IEC, 2005]) minimizing the total number of bits for all the quantized spectral coefficients is found. Note that the section boundaries can only be found at the boundaries of scale factor bands to maintain the coding efficiency. Correspondingly, the spectrum Huffman codebook identifier (1, 2, ..., 11) and the length of the section (i.e., how many scale factor bands in that section) need to be transmitted.

In AAC, for each scale factor band, there is always an integer multiple of four spectral coefficients. A certain number of t -tuples ($t \in \{2, 4\}$) of unsigned or signed quantized

⁸Note that $dsf_0 = sf_0 - g = -sf g_0$.

Differential scale factor dsf	Codebook index i_s	Codeword	Codeword length
-60	$-60 + 60 = 0$	3ffe8	18
-59	$-59 + 60 = 1$	3ffe6	18
...
0	$0 + 60 = 60$	0	1
...
60	$60 + 60 = 120$	7fff3	19

Table 5.2: Differential scale factor Huffman codebook (Tab. 4.A.1 in [ISO/IEC, 2005]). The codewords are given in the hexadecimal notation and the codeword length in number of bits (VL code).

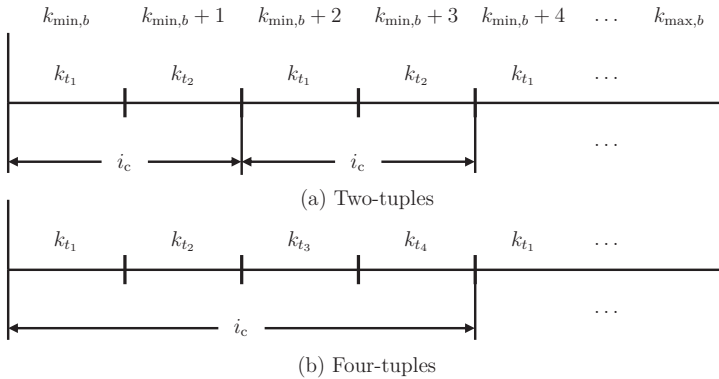


Figure 5.5: Decoding of the quantized spectral coefficients X_k in scale factor band b , with two-tuples (a) and four-tuples (b) Huffman codebooks.

spectral coefficients X_k is jointly coded by the corresponding one of the 11 different spectrum Huffman codebooks (Tab. 4.A.2 to Tab. 4.A.12 in [ISO/IEC, 2005]). For each spectrum Huffman codebook, the tuple size t , the largest absolute value $lav = \max |X_k|$, and whether the codebook is unsigned or signed are defined in Tab. 4.132 in [ISO/IEC, 2005]. According to the magnitude of X_k (for the unsigned codebook) or the value of $X_k + lav$ (for the signed codebook), the quantized spectral coefficients X_k are transformed (in pairs or 4-tuples) to the corresponding Huffman codebook indices i_c , or Huffman codewords. In case an unsigned Huffman codebook is used, after the Huffman codewords representing the absolute values of the quantized spectral coefficients, the sign bits of each non-zero quantized spectral coefficient are appended to the associated codewords. In other words, for each X_k a number of 0 to t bits is required for the codebooks with t -tuples.

In order to ease the presentation of the SD decoding of the quantized spectral coefficients X_k in Section 5.4.3, the decoding process of the quantized spectral coefficients is explained in some detail in the following. Fig. 5.5 implies that the decoding of quantized spectral coefficients X_k is performed in ascending order of frequency index from $k = k_{\min,b}$

to $k = k_{\max,b}$ in each scale factor band b . As the VL/hard-decision (HD) decoding described in Section 4.2, the codewords are first transformed to the corresponding spectrum Huffman codebook index i_c , by a *decoder-sided* lookup table. Thereafter, the index i_c is interpreted as a 4-tuple of quantized spectral coefficients by

$$\begin{aligned}
 X_{k_{t_1}} &= \lfloor i_c / \text{mod}^3 \rfloor - \text{off} \\
 i_c &= i_c - (X_{k_{t_1}} + \text{off}) \cdot \text{mod}^3 \\
 X_{k_{t_2}} &= \lfloor i_c / \text{mod}^2 \rfloor - \text{off} \\
 i_c &= i_c - (X_{k_{t_2}} + \text{off}) \cdot \text{mod}^2 \\
 X_{k_{t_3}} &= \lfloor i_c / \text{mod} \rfloor - \text{off} \\
 i_c &= i_c - (X_{k_{t_3}} + \text{off}) \cdot \text{mod} \\
 X_{k_{t_4}} &= i_c - \text{off},
 \end{aligned} \tag{5.9}$$

or as a 2-tuple of quantized spectral coefficients by

$$\begin{aligned}
 X_{k_{t_1}} &= \lfloor i_c / \text{mod} \rfloor - \text{off} \\
 i_c &= i_c - (X_{k_{t_1}} + \text{off}) \cdot \text{mod} \\
 X_{k_{t_2}} &= i_c - \text{off},
 \end{aligned} \tag{5.10}$$

respectively, with $\text{mod} = \text{lav} + 1$ and $\text{off} = 0$ for an unsigned codebook, while $\text{mod} = 2 \cdot \text{lav} + 1$ and $\text{off} = \text{lav}$ for a signed codebook. Moreover, in (5.10), we have $k_{t_2} = k_{t_1} + 1$, with $k_{t_1} \in \{k_{\min,b}, k_{\min,b+2}, k_{\min,b+4}, \dots, k_{\max,b-1}\}$. Similarly, for (5.9), $k_{t_2} = k_{t_1} + 1$, $k_{t_3} = k_{t_1} + 2$, $k_{t_4} = k_{t_1} + 3$, with $k_{t_1} \in \{k_{\min,b}, k_{\min,b+4}, k_{\min,b+8}, \dots, k_{\max,b-3}\}$.

In addition, large quantized spectral values X_k are represented by the **escape** (ESC) Huffman codebook (unsigned spectrum Huffman codebook 11 with 2-tuples, Tab. 4.A.12 in [ISO/IEC, 2005]). The order of reading out the bits of the quantized spectral coefficients is first the Huffman codewords, thereafter a number of 0 to 2 sign bits, and finally a number of 0, 1, or 2 additional bit sequences each being an **escape_sequence** [ISO/IEC, 2005]. When either $X_{k_{t_1}}$ or $X_{k_{t_2}}$ calculated from (5.10) is in a range⁹ of 0 to $\text{lav} - 1$, the calculated value is directly used as the desired value $|X_{k_{t_1}}|$ or $|X_{k_{t_2}}|$. Once either $X_{k_{t_1}}$ or $X_{k_{t_2}}$ equals lav , after reading the possible sign bits, additional bits of an **escape_sequence** are required to represent the large quantized values. The structure of the bit sequence **escape_sequence** is depicted in Fig. 5.6, which starts with the **escape_prefix** having a number of n_p ($n_p \geq 0$) bits all being 1 (1|. . .|1). After a fixed bit with a value of 0, the bit sequence ends by the **escape_word** with a number of $n_p + 4$ bits. Therefore, altogether a number of $2 \cdot n_p + 5$ bits is required for the **escape_sequence**, with the limitation of¹⁰ $2 \cdot n_p + 5 < 22$. Finally, the desired $|X_{k_{t_1}}|$ or $|X_{k_{t_2}}|$ is obtained by adding $2^{(n_p+4)}$ to the integer value of **escape_word** [ISO/IEC, 2005]¹¹. To sum up, we have

$$|X_{k_{t_1}}| = \begin{cases} |X_{k_{t_1}}|, & \text{if } 0 \leq X_{k_{t_1}} < \text{lav}, \\ 2^{(n_p+4)} + \text{int}(\text{escape_word}), & \text{if } X_{k_{t_1}} = \text{lav}. \end{cases} \tag{5.11}$$

⁹For **escape** (ESC) Huffman codebook, $\text{lav} = 16$.

¹⁰ $n_p \in \{0, 1, \dots, 8\}$ and $2 \cdot n_p + 5 \in \{5, 7, 9, \dots, 21\}$.

¹¹For example, $n_p = 0$ with the **escape_sequence** 00000 and 01111 imply $|X_k| = 16$ and 31, respectively. Moreover, $n_p = 1$ with 1000000 and 1011111 denote $|X_k| = 32$ and 63, respectively.

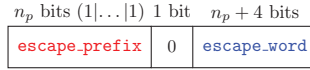


Figure 5.6: The bit sequence (`escape_sequence`) structure for Huffman codebook 11.

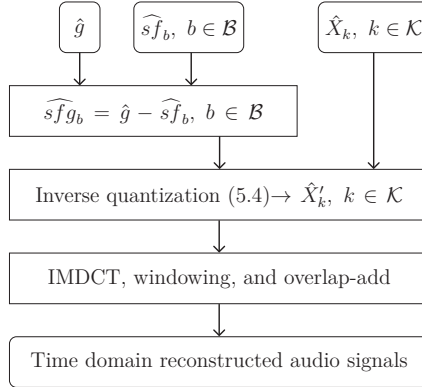


Figure 5.7: Simplified flowchart of the decoding process in AAC in continuation of Fig. 5.4.

Note that (5.11) is also valid for $X_{k_{t_2}}$, by replacing $X_{k_{t_1}}$.

Out of Bits Prevention

The real number of the required bits is determined only after quantizing the spectral coefficients and noiseless coding. If the number of required bits is too high, a reduction of the number of bits is necessary. Increasing the value of global gain, a new quantization and noiseless coding is performed consequently. The real number is finally obtained, when the bit demand is reduced small enough to match the constraints of the bitreservoir, which is calculated in the reduction of the psychoacoustic requirements.

5.2.2 AAC Decoding

Similar to the VL/HD decoding process in Fig. 4.2 in Section 4.2, according to the corresponding *decoder-sided* lookup table for a Huffman codebook¹², the received bit stream is analyzed to extract the received global gain \hat{g} , scale factors $\hat{s}f_b$, and quantized spectral coefficients \hat{X}_k . Thereafter, as the simplified flowchart of the AAC decoding process in Fig. 5.7 shows, based on the value of $\hat{s}fg_b = \hat{g} - \hat{s}f_b$, the inverse quantization in (5.4) is performed to generate the received inversely quantized spectral coefficients \hat{X}'_k . Thereafter, the filterbank module including an inverse modified discrete cosine transform

¹²Note that for the lookup table of differential scale factors in HE-AAC, there are 65 rows, four columns, and $|\mathcal{I}_s| = 121$ Huffman codewords ($\hat{i}_s \in \mathcal{I}_s = \{0, 1, \dots, 120\}$), which results in a potential number of 1 to 7 leading bits (..) in Tab. 4.1 and Fig. 4.2.

(IMDCT), a window, and an overlap-add function transforms the \hat{X}_k^l to the time domain reconstructed audio signals.

5.2.3 HE-AAC Bit Stream Structure

The structure of the HE-AAC bit stream is shown in Fig. 5.8, listing the necessary elements used in the thesis¹³.

The header contains the information of audio object type (AAC LC and SBR), sampling rate (of the core AAC), channel configuration (mono or stereo), extension sampling rate (of SBR), the number of frames, and each frame length.

The data block in each frame defines the element type (i.e., syntactic elements¹⁴: e.g., 0 for **single channel element** (AAC elements for a single audio channel), 6 for **fill element** (SBR elements), 7 for **end** (termination)), the element instance tag (indicating different instances of the same syntactic element, except **fill element**¹⁵), global gain (8 bits), individual channel stream (ICS) data, section data, scale factor data, and spectral data. For the pulse data, temporal noise shaping (TNS) data, and gain control data, each of these starts with a corresponding flag (1 bit), where 0 stands for no data used and 1 indicates data following¹⁶.

The data which is necessary to decode one audio channel is incorporated in the ICS data. The reserve bit is a flag reserved for future use¹⁷. Different applicable transform windows are defined by the sequence of windows and the shape of the window. The window sequences are defined differently (only long, long start, and long stop) for a long window type with 1024 spectral coefficients per long block, and a short window type with eight short blocks per frame, each with 128 spectral coefficients per short block. Depending on the window shape, a Kaiser-Bessel-derived window or a sine window is used for the trailing part of the analysis window. The number of transmitted scale factor bands is determined¹⁸ by B . For the long window type, one additional prediction data flag being 0 is read. For the short window type, the information about the grouping of short windows is represented by the scale factor grouping.

The determination of which spectrum Huffman codebook should be applied and also the number of scale factor bands in each section depends on the section codebook and section length (5 bits for long blocks and 3 bits for short blocks) in the section data, respectively. The section data includes a number of B section codebook identifiers and section lengths.

After the AAC elements, the element type being 6 indicates the start of the SBR elements. The count value implies the initial value of the length of SBR data. If count = 15,

¹³Note that only the terms related to the thesis are explained in detail in this section.

¹⁴Tab. 4.71 in [ISO/IEC, 2005].

¹⁵This is because the subsequent reference is not necessary for the content of **fill element** and it may occur several times [ISO/IEC, 2005].

¹⁶Note that pulse data and gain control data flags are always 0 in [3GPP, 2008].

¹⁷The reserve bit should be 0.

¹⁸ B is denoted as **max.sfb** in [ISO/IEC, 2005].

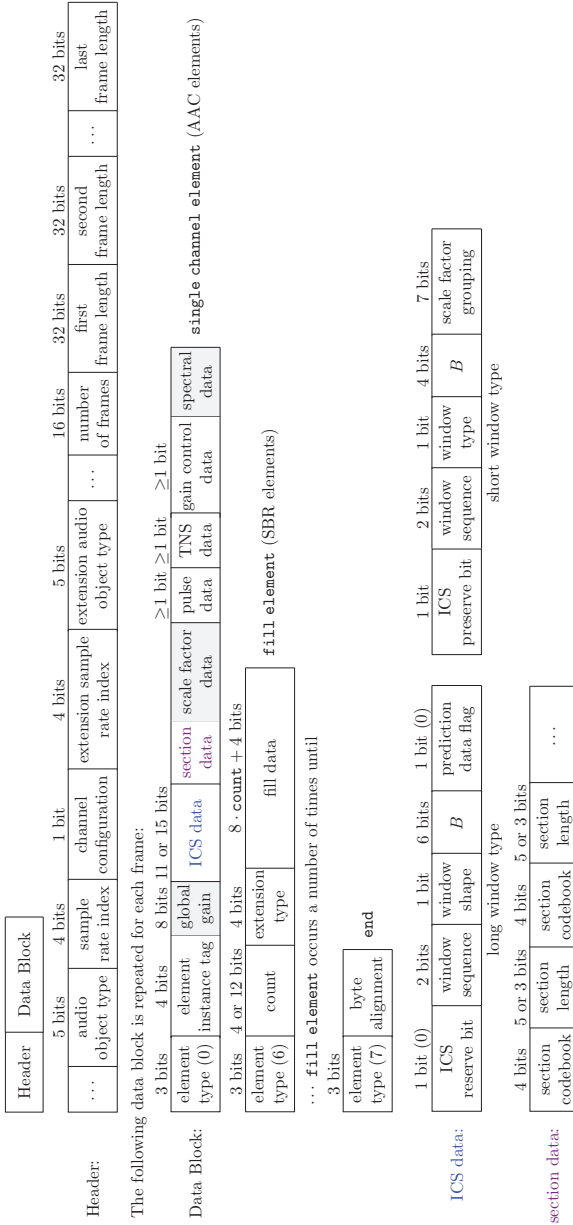


Figure 5.8: HE-AAC bit stream structure.

the integer value of the next 8 bits added to 14 is used as the actual value of count. The SBR elements also include the information of the extension type (e.g., `EXT_SBR_DATA`, `EXT_SBR_DATA_CRC` (SBR with CRC)). A number of 8-count+4 bits is used for the envelope and noise floor data, etc. The `fill element` can occur several times, until the element type being 7 (indicating the termination) is read. Afterwards, byte alignment will be processed to read out all the possible bits left in the current frame.

5.3 Standard Error Concealment in HE-AAC

In order to improve the error robustness of HE-AAC, an error concealment function is included in the AAC core decoder in the 3GPP Technical Specification TS 26.402 [3GPP, 2004d], introducing a decoder delay of one frame. Starting with CRCs and ending in a variety of plausibility checks, various tests are included in the core decoder. Either in case that an invalid bit stream is suggested by any of the checks, or if a corrupted or missing frame is indicated by a `frameOK` flag, error concealment is applied to the received inversely quantized spectral coefficients \hat{X}'_k just before the final frequency-domain-to-time-domain conversion. In the error concealment, the to-be-replaced spectral values \hat{X}'_k of the corrupted/missing frame are always based on the ones from a stored last good frame, which is only updated by a good frame. The signs of the spectral values for the replaced corrupted/missing frame are randomly determined. If only a single frame is corrupted, an interpolation is performed.

Four different states are included in the error concealment procedure [3GPP, 2004d]: `ok` (no consecutive corrupted frames in the current and last two frames), `fade out`, `fade in`, and `muting`. In case that multiple consecutive frames are corrupted, based on the values from the last good frame, a `fade out` incrementally attenuating the spectral values occurs. After fading out five times, the concealment turns to muting (the complete spectrum is set to zero). Once new good frames become available, concealment fades in the new spectral values. The processes of either `muting to fade in`, and of `fade in to ok` also last five frames.

In SBR error concealment, instead of using data from the corrupted bit stream, with an applied decay, a substitute for the corrupted data is generated based on the envelope and noise floor values from the stored last good frame.

5.4 New HE-AAC Parameter Soft-Decision (SD) Decoding

In this section, the FL/SD decoding approach in Section 2.3 is applied to the FL-coded global gain. The VL/SD decoding approach in Section 4.4 is applied to both the VL-coded scale factors and the quantized spectral coefficients. Moreover, considering the unsigned Huffman codebooks with sign bits, a modified trellis representation based on Section 4.3 is proposed.

Parameters	Global gain	Differential scale factors
ΔR_d (bits)	3.03	0.11
ΔR (bits)	4.26	0.15

Table 5.3: Residual redundancy of the global gain and the differential scale factors: ΔR_d being the *utilizable, distribution-dependent* residual redundancy (equation (2.10)), and ΔR being the total residual redundancy (equation (2.9)).

5.4.1 FL/SD Decoding of the Global Gain

In HE-AAC, each audio frame contains one global gain value and a number of B differential scale factors. From Tab. 5.1, it is known that the FL-coded global gain g equals the global gain codebook index $i_g \in \mathcal{I}_g = \{0, 1, \dots, 255\}$. Applying FL/SD decoding to the global gain, according to (2.26) or (2.27), the *a posteriori* probabilities (APPs) of the global gain $P(\mathbf{x}_\ell^{(i_g)} | \hat{\mathbf{x}}_\ell, \hat{\mathbf{x}}_1^{\ell-1})$ can be calculated, with \mathbf{x}_ℓ being the bit combination (i.e., the codeword in Tab. 5.1) of the global gain codebook index i_g at frame ℓ . Similar to (2.32), using the minimum mean-square error (MMSE) estimation and taking the nearest integer value of the estimated floating-point value, the global gain g_ℓ can be estimated [Han and Fingscheidt, 2015a], with $M=8$. In addition, modeling the global gain codebook indices as a zeroth-order or first-order Markov process, the residual redundancies ΔR_d and ΔR based on (2.9) are shown in Tab. 5.3.

5.4.2 VL/SD Decoding of the Scale Factors

The differential scale factor dsf and the corresponding Huffman codebook index i_s satisfy $dsf = i_s - 60$. Therefore, in each frame, the received \widehat{dsf} can be obtained according to a newly estimated differential scale factor codebook index \hat{i}_s using the VL/SD decoding approach presented in Chapter 4. The block length here is the number of the transmitted scale factor bands¹⁹ B . Replacing the symbol time index n by the scale factor band index b in (4.6), the APPs $P(\mathbf{y}_b = \mathbf{y}^{(i_s)} | \hat{\mathbf{y}}_1^{R_s})$ of the differential scale factor band codebook index i_s can be calculated, with \mathbf{y}_b being the bit combination of the differential scale factor codebook index at scale factor band b , and R_s being the total number of bits associated with the scale factors in that frame (the frame being the block containing B scale factor bands). In analogy to (4.21), using the MMSE estimator, the new differential scale factor codebook index $\hat{i}_{s,b}$ can be obtained [Han and Fingscheidt, 2015a]. Finally, the received differential scale factor is computed by $\widehat{dsf}_b = \hat{i}_{s,b} - 60$ and rounded to the nearest integer value. Moreover, similar to the global gain, the residual redundancies ΔR_d and ΔR of the differential scale factor can be seen in Tab. 5.3. The average Huffman codeword length \overline{M} ($\overline{M}=2.21$ bit for scale factors) is adopted for M in (2.9). We observe that the differential scale factors reveal less redundancy than the global gain parameter.

In HE-AAC, the block length B can be extracted from the HE-AAC bit stream, as depicted in Fig. 5.8 (assumed to be error-free). For error-prone transmission, the total

¹⁹Note that in Chapter 4, the symbol time index satisfies $n \in \{1, 2, \dots, B\}$, while the scale factor band index in this chapter meets $b \in \{0, 1, \dots, B-1\}$.

number of bits R_s is an additional side information, leading to only a small relative increase of the bit rate²⁰ [Derrien et al., 2008].

5.4.3 VL/SD Decoding of the Quantized Spectral Coefficients

In order to apply the VL/SD decoding approach to the quantized spectral coefficients, the block length B has to be identified first. As depicted in Fig. 5.2, for a given section, the same spectrum Huffman codebook is used for all the spectral coefficients in that section. According to the section length, the total number of spectral coefficients in one section can be obtained by adding the width of each scale factor band and is denoted as²¹ K_s . In addition, each spectrum Huffman codebook²² index represents t -tuples of quantized spectral coefficients ($t \in \{2, 4\}$). Similar to Section 5.4.2, the VL/SD decoding is supposed to be applied to the spectrum Huffman codebook index i_c generating a new received spectrum Huffman codebook index \hat{i}_c . Replacing i_c by the received \hat{i}_c , the received quantized spectral coefficients ($\hat{X}_{k_{t_1}}, \hat{X}_{k_{t_2}}, \hat{X}_{k_{t_3}}, \hat{X}_{k_{t_4}}$) can be obtained according to (5.9) or (5.10) afterwards. Therefore, for each section in one frame²³, the block length B_c is defined as the number of decoded Huffman codebook indices, with $B_c = K_s/t$.

The APP calculation for different spectrum Huffman codebooks is also distinguished. On the one hand, for signed codebooks, considering the Huffman codebook index as a symbol in the trellis representation in Section 4.3, the APP calculation is the same as in Section 4.4.1. On the other hand, for unsigned codebooks, a number of 0 to t sign bits is appended to the Huffman codewords and has also to be considered as a symbol in the trellis representation.

New Trellis Representation

Based on the trellis representation from Section 4.3, a modified new trellis representation is shown in Fig. 5.9, with an example of $B_c = 4$, $t = 2$, the Huffman codeword length $N^{(i_c)} \in \{1, 2, 3\}$. Fig. 5.9 draws five examples of the total number of bits R_c in one block. The absolute value and the corresponding sign of the quantized spectral coefficients are assumed to share the same *global* symbol index $l = \{1, 2, \dots, B_c\}$. Considering a number of 0 to t sign bits as an individual symbol, the *local* symbol index is denoted as $l_c = \{1, 2, \dots, 2 \cdot B_c\}$, with the *odd* l_c ($l_c = t \cdot l - 1$) related to the absolute value and the *even* l_c ($l_c = t \cdot l$) implying the sign of the received quantized spectral coefficients, respectively.

It is assumed that the *global* symbol time index l includes the *local* symbol index l_c

²⁰Assuming the same number of bits is used for representing the side information in each frame, among the audio test signals used in this thesis, the maximum number of R_s for scale factors per frame is 129 (i.e., 8 bits for representation), the average number of bits per frame is 1029, therefore, there is an increase of about $8/1029 \approx 0.78\%$ of the bit rate.

²¹In K_s , s refers to section.

²²In this section, unless explicitly mentioned, the spectrum Huffman codebook refers to one of the 11 spectrum Huffman codebooks.

²³Note that the block length for the scale factors is on a frame basis, while for quantized spectral coefficients it is on a section basis.

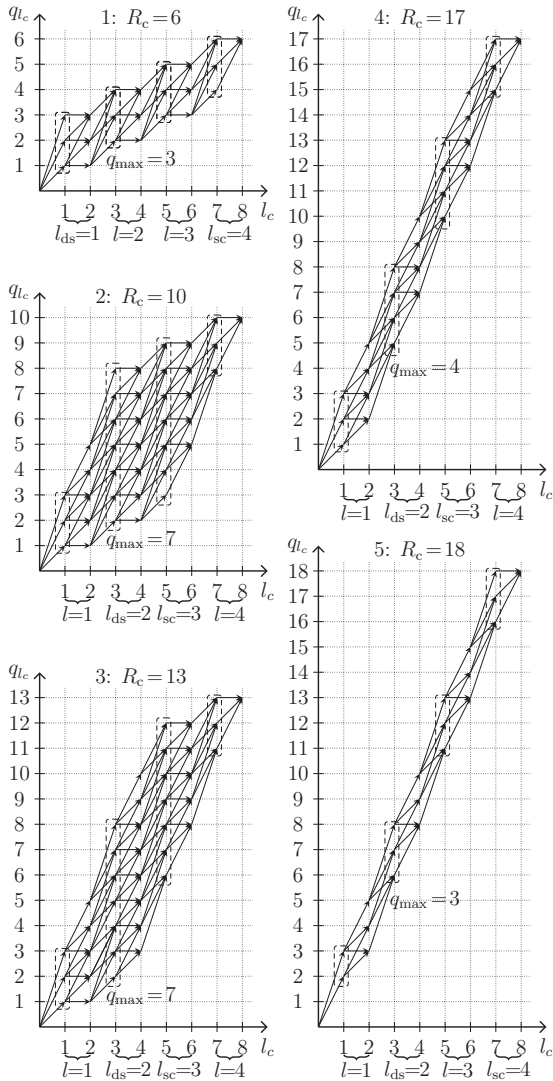


Figure 5.9: New trellis representation for the unsigned spectrum Huffman codebooks, with an example of $B_c=4$, codeword length $N^{(lc)} \in \{1, 2, 3\}$, and $t=2$ ($R_c''=14$, $l''=3$).

<i>Global</i> symbol index	<i>Local</i> symbol index	Implication	State
$l-1$	l_c-2	absolute value	$\mu \in \mathcal{Q}_{l_c-2} = [\mu_{\text{start}}, \mu_{\text{end}}]$
$l-1$	l_c-1	sign	$\mu' \in \mathcal{Q}_{l_c-1} = [\mu'_{\text{start}}, \mu'_{\text{end}}]$
l	l_c	absolute value	$\nu \in \mathcal{Q}_{l_c} = [\nu_{\text{start}}, \nu_{\text{end}}]$
l	l_c+1	sign	$\nu' \in \mathcal{Q}_{l_c+1} = [\nu'_{\text{start}}, \nu'_{\text{end}}]$

Table 5.4: The definition of the *global* symbol index, *local* symbol index, and the state.

	Stage		
	Diverging	Stationary	Converging
ν_{start}	$N'_{\min} \cdot l$	$\max(N'_{\min} \cdot l, R_c - t \cdot (B_c - l + 1) - N'_{\max} \cdot (B_c - l))$	
ν_{end}	$N'_{\max} \cdot l + t \cdot (l-1)$	$\nu_{\text{start}} + q_{\text{max}} - 1$	$R_c - N'_{\min} \cdot (B_c - l)$
ν'_{start}	$\max(\nu_{\text{start}}, R_c - N'_{\max} \cdot (B_c - l) - t \cdot (B_c - l))$		$\min(\nu_{\text{start}} + t, R_c - t \cdot (B_c - l) - N'_{\max} \cdot (B_c - l))$
ν'_{end}	$\min(\nu_{\text{end}} + t, R_c - N'_{\min} \cdot (B_c - l))$		$R_c - N'_{\min} \cdot (B_c - l)$

Table 5.5: Definitions of state intervals (cf. Tab. 4.2) for the diverging, stationary, and converging stages in a new trellis representation considering sign bits.

for the absolute value and $l_c + 1$ for the sign. In analogy to Section 4.3, the implication of the local symbol indices (absolute value or sign) and state notations are detailed in Tab. 5.4. Moreover, from *odd* l_c (absolute value symbol index) to *even* l_c (sign symbol index), the state is possible to change in a range of 0 to t . In contrast, from *even* l_c to *odd* l_c , the state can change in a range of N'_{\min} to N'_{\max} , with N'_{\min} and N'_{\max} being the minimum and maximum Huffman codeword length used in the trellis representation, respectively. It is found that $N'_{\min} = \max(R_c - N_{\max} \cdot (B_c - 1) - t \cdot B_c, N_{\min})$ and $N'_{\max} = \min(R_c - N_{\min} \cdot (B_c - 1), N_{\max})$, with $N_{\min} = \min_{i_c \in \mathcal{I}_c} N^{(i_c)}$ and $N_{\max} = \max_{i_c \in \mathcal{I}_c} N^{(i_c)}$ representing the minimum and maximum codeword length of all the Huffman codewords, respectively.

The definitions of stage boundaries and state intervals should be redefined. The diverging, stationary, and converging stages are discriminated only according to the number of states for *odd* local symbol indices l_c , without considering the number of bits for sign symbol indices. In other words, the stage boundaries l_{ds} and l_{sc} are assumed in terms of the global symbol index l (not the local symbol index l_c).

The stage boundaries and the maximum number of states q_{max} for *odd* local symbol indices l_c can be obtained in the following steps. Note that the maximum number of states for *even* local symbol indices is $q_{\text{max}} + t$. As shown in Fig. 5.9, when $R_c = 13$, $q_{\text{max}} = 7$, the number of states for $l_c = 4$ is 8.

First, an intermediate value R''_c , which satisfies the smallest integer value of R_c with $\lceil \frac{(N_{\max} + t) \cdot B_c + t - R_c}{N_{\max} - N_{\min} + t} \rceil \leq \lfloor \frac{R_c - N_{\min} \cdot B_c + t}{N_{\max} - N_{\min} + t} \rfloor$, needs to be found. The corresponding intermediate l'' is obtained by $l'' = \lfloor \frac{R''_c - N_{\min} \cdot B_c + t}{N_{\max} - N_{\min} + t} \rfloor$.

Thereafter, the l_{sc} and q_{max} for *odd* local symbol indices (not *even* l_c) can be defined

differently as follows:

$$\left\{ \begin{array}{ll}
 1 : l_{sc} = B_c, & \text{if } R_c \leq N_{\min} \cdot B_c + t, \\
 \quad q_{\max} = R_c - N'_{\min} \cdot B_c + 1, & \\
 2 : l_{sc} = \lfloor \frac{(N'_{\max} + t) \cdot B_c - R_c}{N'_{\max} - N'_{\min} + t} \rfloor, & \text{else if } R_c \leq N_{\max} \cdot (B_c - l'') + N_{\min} \cdot l'' \\
 \quad q_{\max} = R_c - N'_{\min} \cdot B_c + 1, & \quad + t \cdot (B_c - l'' + 1) \\
 3 : l_{sc} = \lceil \frac{(N'_{\max} + t) \cdot B_c + t - R_c}{N'_{\max} - N'_{\min} + t} \rceil, & \text{else if } R_c < R'_c, \\
 \quad q_{\max} = (N'_{\max} - N'_{\min} + t) \cdot (B_c - l_{sc}) + t + 1, & \\
 4 : l_{sc} = \lfloor \frac{R_c - N'_{\min} \cdot B_c + t}{N'_{\max} - N'_{\min} + t} \rfloor, & \text{else if } R_c < N_{\max} \cdot B_c + t \cdot (B_c - 1), \\
 \quad q_{\max} = (N'_{\max} + t) \cdot B_c - R + 1, & \\
 5 : l_{sc} = B_c, & \text{else if } R_c \geq N_{\max} \cdot B_c + t \cdot (B_c - 1), \\
 \quad q_{\max} = (N'_{\max} + t) \cdot B_c - R_c + 1. &
 \end{array} \right. \quad (5.12)$$

After q_{\max} is known, we obtain $l_{ds} = \lceil \frac{q_{\max} - 1 + t}{N'_{\max} - N'_{\min} + t} \rceil$.

The state intervals for the diverging, stationary ($[l_{ds}, l_{sc}]$), and converging stage can finally be found in Tab. 5.5²⁴. Five examples of R_c corresponding to the above five cases are plotted in Fig. 5.9.

A *Posteriori* Probabilities and Source Symbol Estimation

The spectrum Huffman codebook is used for the *odd* values of l_c to calculate the APPs for the absolute values of the quantized spectral coefficients, while an additional different codebook should be used for *even* l_c to compute the APPs for the corresponding sign. Considering the sign bits only for non-zero quantized spectral coefficients, a number of 0 to t bits is included in the sign codebook. The sign codebook for $t=2$ is depicted in Tab. 5.6, with the codebook size²⁵ being 9 and x implying no sign bits at the corresponding position. More specifically, $x0$ denotes that the received quantized spectral coefficient $\hat{X}_{k_{t_1}} = 0$ (no sign bits) and $\hat{X}_{k_{t_2}} > 0$ (sign bit 0); $1x$ implies $\hat{X}_{k_{t_1}} < 0$ (sign bit 1) and $\hat{X}_{k_{t_2}} = 0$ (no sign bits). Note that if the sign codebook index is 0 (xx: no sign bits), the channel term in (4.9) in Section 4.4.1 is 1.

Moreover, due to the fact that the number of sign bits and the position of sign bits are determined by the absolute value, in order to distinguish the codebook indices with the same codewords (e.g., codebook indices 1 and 3, or 2 and 4 in Tab. 5.6), for the APP calculation of the sign symbol, instead of considering all sign codebook indices, the actually used sign codebook indices are limited to a small range, based on the corresponding estimated absolute values. More specifically, taking $t=2$ and Tab. 5.6 as an example, if only $\hat{X}_{k_{t_2}}$ is non-zero, the codebook indices 1 and 2 are used, while codebook indices 3 and 4 are taken into account once only $\hat{X}_{k_{t_1}}$ is non-zero, etc.

It is found that instead of using MMSE estimation, the maximum *a posteriori* (MAP) estimator (2.33) provides a better performance, because the estimated value is actually the received Huffman codebook index \hat{i}_c , which is further divided and transformed to new

²⁴The corresponding derivation is based on Section 4.3.

²⁵The codebook size of the sign codebook for $t=4$ is 81.

Sign bits	Codebook index	Codewords	Codeword length
xx	0		0
x0	1	0	1
x1	2	1	1
0x	3	0	1
1x	4	1	1
00	5	00	2
01	6	01	2
10	7	10	2
11	8	11	2

Table 5.6: The sign codebook for 2-tuples of quantized spectral coefficients.

received quantized spectral coefficients. MAP estimation is applied to both the absolute value and the sign for unsigned Huffman codebook, and applicable to the signed Huffman codebook as well.

As mentioned in Section 5.2.1, for the **escape** (ESC) Huffman codebook (spectrum Huffman codebook 11), additional 0 or $2 \cdot n_p + 5$ ($n_p \in \{0, 1, \dots, 8\}$) bits are followed by the sign bits. If the **escape_sequence** is also considered as a symbol index in the trellis representation, a codebook size of 8177 words would be required²⁶. Although it is not necessary to store the whole codebook, but still a number of $n_p + 4$ bits with the maximum value being 12 is required for the **escape_sequence** (12-bit codebook). The complexity would also be dramatically increased. Therefore, SD decoding is not suggested and adopted for the spectrum Huffman codebook 11.

Compared to the VL/SD decoding of scale factors, the complexity for quantized spectral coefficients is significantly increased, due to more spectral data than scale factors, and the SD decoding is applied to a number of 10 spectrum Huffman codebooks as well. Therefore, only the VL/SD decoding with AK0 is adopted for the quantized spectral coefficients, since AK1 introduces more computations. The residual redundancy ΔR_d for the spectrum Huffman codebook 1 to 10 ranges from 0.01 to 0.07, with the average Huffman codeword length \bar{M} is in a range of 2.65 to 5.86.

Note that the total number of bits for each section in each frame should be known by the decoder as a further side information, which leads to a somewhat bigger relative increase of the bit rate compared to VL/SD decoding for scale factors²⁷.

²⁶We have $1 + \sum_{n_p=0}^8 2^{n_p+4} = 8177$, where 1 stands for no additional bits, while $n_p + 4$ is due to the fact that the first $n_p + 1$ bits are fixed in Fig. 5.6. The maximum value of $n_p + 4$ is 12.

²⁷If we assume the same number of bits in each section is used for representing the side information, for the signals used in test, the maximum number of bits R_c for each section is 759 (i.e., 10 bits for representation), the average number of sections per frame is 6, the average number of bits per frame is 1029, as a result, there is an increase of about $60/1029 = 5.8\%$ of the bit rate.

5.5 Simulation Setup and Results

In this section, the performance of HE-AAC with HD and SD decoding is compared. Section 5.5.1 starts with introducing the audio database, the channel models, the instrumental measurement, and the subjective listening tests.

5.5.1 Simulation Setup

For the training process, a number of 15 monaural music pieces with a total length of 89 min is used. A number of 10 monaural fractions of other music pieces with a total length of 68 s is adopted as the test database. The audio data includes excerpts from classical music pieces with a motion-picture soundtrack with music and effects, and with different instruments (organs, strings, brass and percussion instruments). The audio database used for training and test is depicted in Tab. C.1 and Tab. C.2 in Appendix C. Moreover, all signals are 16 bit pulse-code modulated with 48 kHz sampling rate and have been normalized to -26 dBFS (decibels relative to full scale).

The Gilbert-Elliott channel model with additive white Gaussian noise (AWGN) noise and the AWGN channel model are adopted in different simulations. For the GE channel model, $E_b/N_0|_{\text{good}}$ for the good channel state is 10 dB (non-perfect transmission even in the good frames), while $E_b/N_0|_{\text{bad}}$ for the bad frames is varied between 5 dB to 10 dB. This simulation was built up intentionally, in order to provoke the future of more efficient media transmission, where frames are not lost but only a few bit errors occur. In the AWGN channel model, each frame is distorted with a given E_b/N_0 ratio varied in a range from 0 dB to 10 dB. For the purpose of increasing the reliability of the performance measurements, each audio file is transmitted over five different channel realizations, for any given test condition.

Moreover, for the error concealment in 3GPP Technical Specification TS 26.402 [3GPP, 2004d], the `frameOK` flag which distinguishes bad frames (i.e., bad channel state) and good frames (i.e., good channel state) is known by the decoder as an additional input file to the decoder.

In this thesis, the bits associated with the global gain g , the scale factors sf , and the quantized spectral coefficients X_k are corrupted by the erroneous channel, while all the other bits are assumed to be correctly received and with correct known bit position in the HE-AAC bit stream. Therefore, also the knowledge about which of the 11 spectrum Huffman codebooks was used for each scale factor band is known correctly. In addition, the SBR part of the HE-AAC bit stream is assumed to be non-corrupted as well.

Instrumental Measurement and Subjective Listening Test

As mentioned in [Herre and Dietz, 2008], for HE-AAC with parametric coding tools (e.g., SBR) which depart from the waveform coding, instead of the instrumental measurement, a reliable subjective listening test has to be used for assessing audio quality. However,

subjective listening tests are time-consuming, and therefore impractical to test all the conditions. Therefore, apart from a few conditions being tested in a subjective listening test, the global SNR and the perceptual evaluation of audio quality (PEAQ) [ITU-R, 2001] are adopted for instrumental measurements. For PEAQ, the calculated objective difference grade (ODG) ranging from 0 to -4 corresponds to the ITU-R five-grade quality scale from excellent to bad [ITU-R, 2003] (or the impairment scale from imperceptible to very annoying). In addition, before measuring the audio quality, an optimal delay (4166 samples²⁸) compensation of the audio signals is made.

The MUlti Stimulus test with Hidden Reference and Anchor (MUSHRA) [ITU-R, 2014] is a widely used and proper method for the assessment of intermediate audio quality²⁹. The MUSHRA listening test starts with a training phase, which aims to make the listeners familiar with all the sounds experienced during the test. Thereafter, in the evaluation phase, in comparison with the reference sound, the assessors have to grade all the test sounds including one hidden reference, two hidden anchors, and all the sounds under different test conditions (e.g., HD and SD decoding). The hidden reference has to be found and scored as 100. Both two hidden anchors are low-pass filtered versions of the reference signal, with the cut-off frequency being 3.5 kHz (low-range anchor³⁰) and 7 kHz (mid-range anchor), respectively. The grading is based on the continuous quality scale (CQS), the scores varying from 0 to 100 are assigned to 5 equally distributed intervals: bad, poor, fair, good, and excellent (i.e., 20 scores for each interval).

MUSHRA is known to be more suitable for assessing medium and large impairments [ITU-R, 2014]. Different to the methods of absolute category rating (ACR), degradation category rating (DCR), and comparison category rating (CCR) in Recommendation P.800 [ITU-T, 1996], the advantage of MUSHRA is that the listeners can compare the impaired sounds (i.e., test signals) with the reference sound, and also the different degrees of impairment between all test signals straightforwardly. If the difference between test signals is very small, it is difficult to distinguish that difference if the test signals are only compared to the reference. The listeners can switch easily between the reference and any test sound with MUSHRA.

After all the assessors have finished the evaluation, a post-screening method is adopted to exclude the scores of listeners who graded the degraded anchor signal as a high score or considered the obviously degraded signal as the hidden reference. The post-screening method is based on the following criterion [ITU-R, 2014]: If the hidden reference for more than 15% of the test experiments (audio files) is scored lower than 90, or the mid-range anchor for more than 15% of the experiments is graded higher than 90, the evaluation results of that listener are not considered to be used anymore. However, if the mid-range anchor is rated higher than 90 for more than 25% of listeners, the results of those listeners are not removed.

²⁸This value is found by searching for the maximum value of SNR.

²⁹The term intermediate audio quality refers to the quality of audio streaming or digital broadcasting [EBU, 2003].

³⁰The 3.5 kHz low-pass filter should satisfy that the maximum pass band ripple is ± 0.1 dB, the minimum attenuation at 4 kHz is 25 dB, and the minimum attenuation at 4.5 kHz is 50 dB [ITU-R, 2014].

The mean scores and the corresponding 95% confidence intervals are required for the analysis of the results [ITU-R, 2014]. Given the scores observed from listeners, the range where an unknown score likely to fall in can be estimated by the confidence interval [Neyman, 1937]. Note that the true scores are not necessarily included in the confidence interval. When comparing the results from different test conditions (e.g., comparing the performance of HD and SD decoding), the significance of the calculated mean value (from each test condition) can be estimated by the confidence interval. In other words, if the confidence intervals are not overlapping, the difference of the means is significant and can be used to compare the results. Under a given test condition, assuming a number of N_L listeners and N_F test files, the mean MUSHRA score for a specific test file from all the listeners is obtained by

$$\bar{\tau}_{i_F} = \frac{1}{N_L} \cdot \sum_{i_L=1}^{N_L} \tau_{i_L, i_F}, \quad (5.13)$$

with $i_L \in \{1, 2, \dots, N_L\}$ being the listener index, $i_F \in \{1, 2, \dots, N_F\}$ being the test file index, and τ_{i_L, i_F} implying the MUSHRA score from listener i_L for file i_F . The 95% confidence interval is obtained by $[\bar{\tau}_{i_F} - \delta_{i_F}, \bar{\tau}_{i_F} + \delta_{i_F}]$, with $\delta_{i_F} = 1.96 \cdot \frac{\sigma_{i_F}}{\sqrt{N_L}}$, and the standard deviation being calculated by [EBU, 2003]

$$\sigma_{i_F} = \sqrt{\sum_{i_L=1}^{N_L} \frac{(\bar{\tau}_{i_F} - \tau_{i_L, i_F})^2}{(N_L - 1)}}. \quad (5.14)$$

Similarly, considering all the test files and listeners, replacing N_L by $N_L \cdot N_F$, the confidence interval for a given test condition can be computed by $[\bar{\tau} - \delta, \bar{\tau} + \delta]$, with $\delta = 1.96 \cdot \frac{\sigma}{\sqrt{N_L \cdot N_F}}$. Herein, the mean is written as

$$\bar{\tau} = \frac{1}{N_L \cdot N_F} \cdot \sum_{i_F=1}^{N_F} \sum_{i_L=1}^{N_L} \tau_{i_L, i_F}, \quad (5.15)$$

and the standard deviation can be computed by

$$\sigma = \sqrt{\sum_{i_F=1}^{N_F} \sum_{i_L=1}^{N_L} \frac{(\bar{\tau} - \tau_{i_L, i_F})^2}{(N_L \cdot N_F - 1)}}. \quad (5.16)$$

In this thesis, a number of 20 subjects participated in the listening test, which includes 10 experiments corresponding to 10 different audio files from Tab.C.2. For the transmission over five different channel realizations, the audio files with the first and third transmission are randomly assigned to the listeners, with 10 listeners for each channel realization. According to the post-screening method, in total 4 subjects are removed before analyzing the results, because they have graded 2 or 3 reference files with a score lower than 90. As a result, there is a remaining number of $N_L = 16$ listeners and a number of $N_F = 10$ audio files for each test condition. Moreover, a MATLAB interface MUSHRAM [Vincent, 2005] is adopted for the subjective listening test, with a laptop PC, an external sound card (FocusriteScarlett6i6), and a high-quality headphone (AKGK271MKII).

The GE channels with $P=0.9, Q=0.1$ and $E_b/N_0|_{\text{bad}} \in \{5, 6.5\}$ dB with the bit error probability calculated from (2.18) in Chapter 2 being $3.7 \cdot 10^{-3}$ and $8.7 \cdot 10^{-4}$ are selected for the subjective listening test³¹. For the experiment with audio file i_F , the test signals are listed as *file* i_F -**A*-**B*-**C*, with **A* referring to two (global gain and scale factors) or three parameters (global gain, scale factors, and quantized spectral coefficients) being corrupted, **B* implying $E_b/N_0|_{\text{bad}} = 5$ dB or 6.5 dB, and **C* standing for the HD or SD decoding method, respectively. Therefore, for each piece of audio, the combination of all three options leads to 8 different test conditions. Considering one hidden reference and two anchors, altogether 11 test signals are used for each piece of audio.

5.5.2 Simulation Results

The performance of HD decoding and SD decoding is compared in three experiments:

- The bits of a single parameter (i.e., global gain, scale factors, or quantized spectral coefficients) are distorted separately.
- The bits of two parameters (global gain and scale factors) are distorted.
- The bits of all three parameters (global gain, scale factors, and quantized spectral coefficients) are distorted.

In this section, the results of HD decoding is denoted by HD. SD decoding with zeroth-order and first-order *a priori* knowledge is represented by AK0 and AK1, respectively. The simulation results of error concealment from 3GPP TS 26.402 [3GPP, 2004d] is represented by EC. Note that HD, SD, AK0, and AK1 imply the decoding method for the respective distorted one parameter or two, or three parameters in the different experiments.

Single Parameter Distorted

This part shows the simulation results of HE-AAC transmission over an AWGN channel, with only the bits of a single parameter being corrupted. As mentioned in Section 5.3, EC is applied to the received inversely quantized spectral coefficients \hat{X}'_k , just before the conversion from frequency-domain to time-domain, which implies that different to AMR-NB or AMR-WB in Section 3.2, there is no *specific* error concealment for the global gain and scale factor parameters. But the value of \hat{X}'_k is based on the received global gain \hat{g} , scale factors \hat{s}_f , and quantized spectral coefficients \hat{X}_k .

(1) Global Gain and Scale Factors

The PEAQ ODG results of corrupting the bits of only the global gain and only the scale factors are shown in Tab. 5.7. It turns out, however, that the ODG does not reflect the performance very well in most of the conditions. For HD, the ODG values at bad channel conditions are even better (higher) than the values at good channel conditions. Even if $E_b/N_0 = 10$ dB, the ODG value of -2.77 indicates a quality rating of slightly annoying. However, for $E_b/N_0 \geq 6$ dB, the ODG still shows AK1>AK0>HD. PEAQ

³¹The BER of the typical bad and good channel conditions is 10^{-3} and 10^{-4} , respectively [Sperschnieder, 2000].

Parameters	Method	E_b/N_0 (dB)										
		0	1	2	3	4	5	6	7	8	9	10
Global gain	AK1	-3.25	-3.22	-3.18	-3.07	-2.98	-2.90	-2.81	-2.79	-2.77	-2.77	-2.77
	AK0	-2.98	-3.01	-3.13	-3.13	-3.05	-2.94	-2.85	-2.80	-2.78	-2.77	-2.77
	HD	-1.37	-1.39	-1.48	-1.66	-1.93	-2.65	-2.86	-2.83	-2.80	-2.77	-2.77
	EC	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54
Scale factors	AK1	-3.27	-3.25	-3.32	-3.30	-3.32	-3.18	-2.98	-2.83	-2.78	-2.77	-2.77
	AK0	-3.27	-3.25	-3.25	-3.21	-3.29	-3.17	-2.99	-2.82	-2.78	-2.77	-2.77
	HD	-2.02	-2.19	-2.50	-2.70	-3.11	-3.23	-3.19	-2.95	-2.83	-2.78	-2.77
	EC	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54

Table 5.7: ODG results for HE-AAC transmission over an AWGN channel, with either the individual parameter global gain or the scale factors being distorted.

ODG is more suitable for measurements when only the AAC part is used [Rawat et al., 2011, Derrien et al., 2008], without SBR or PS tools.

The corresponding global SNR results are shown in Fig. 5.10. In the AWGN channel model, all the frames are indicated as bad frames. As a result, EC turns to complete muting which results in SNR=0 dB. It can be obviously seen that the performance has been significantly improved by SD decoding for both global gain and scale factors. Compared to HD/EC (the optimal option between HD and EC), the SNR can be increased by up to 14.41 dB for the global gain, while by up to 8.76 dB for the scale factors. The corresponding E_b/N_0 gain reaches up to 5.7 dB and 4.5 dB, respectively. The global gain has been improved more than the scale factors, due to the fact that the global gain shows more redundancy than the scale factors (see Tab. 5.3). However, the results of SD decoding with AK0 are still remarkable for the scale factors, since only 0.11 bit of residual redundancy is observed. This motivates the application of VL/SD decoding to the quantized spectral coefficients, which show very small redundancy (0.01 to 0.07 bit). In addition, the SNR difference between AK1 and AK0 is 1.68 dB for the global gain, while only 0.64 dB is achieved for the scale factors. This is due to the fact that the residual redundancy difference between ΔR and ΔR_d is only 0.04 bit for differential scale factors, while 1.23 bits is observed for the global gain. Considering the tradeoffs of the performance and the complexity of SD decoding, it can be stated that AK1 is the better option for the global gain, while AK0 is better to be chosen for the scale factors.

Supported by an informal listening test, it is found that the audio quality in error concealment experiments is in fact better reflected by the global SNR as compared to ODG. The reason for this not yet widely known fact is that in error concealment typically isolated significant corruptions occur, which are better identified by the global SNR than by the instrumental measurements (e.g., ODG). For *error concealment* purposes, the speech quality has been shown to correlate nicely with the SNR measure [Fingscheidt and Vary, 2001, Pflug and Fingscheidt, 2013b], even for audio coding based on the perceptual methods [Cheng et al., 2002]. The subjective listening test of distorting the bits of two parameters and three parameters will also prove this in the results shown hereafter. Therefore, not all the ODG results for each simulation are listed in this chapter, but they can be found in Appendix D.

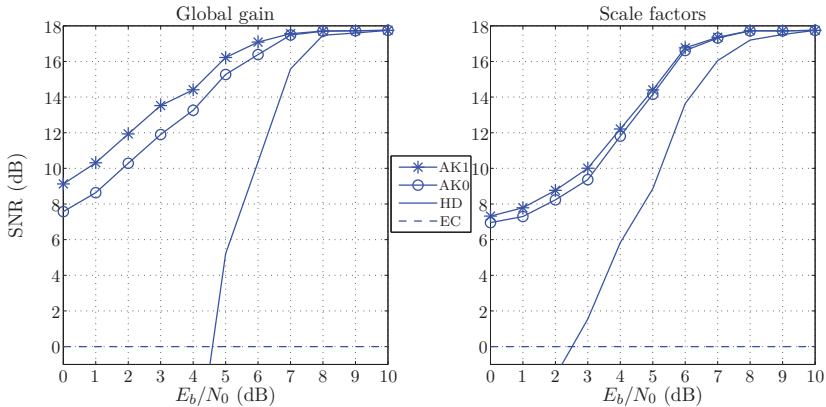


Figure 5.10: Global SNR (dB) results for HE-AAC transmission over an AWGN channel, with the individual parameter global gain or scale factors being distorted separately (see Tab. E.15).

(2) Quantized Spectral Coefficients

As mentioned in Section 5.4.3, VL/SD decoding is not recommended for the quantized spectral coefficients with the spectrum Huffman codebook 11. In order to do a fair comparison between HD and SD decoding, not only the bits of global gain and scale factors, but also the bits of the quantized spectral coefficients with spectrum Huffman codebook 11 are assumed error-free. Moreover, due to the new trellis representation for the unsigned spectrum Huffman codebooks, it is interesting to see the performance if only the bits of the signed codebooks are corrupted (i.e., without new trellis representation). Note that in this part, HD and SD imply the decoding method for the specific spectrum Huffman codebooks. As mentioned above, for the transmission over an AWGN channel, the EC always turns to complete muting and is not listed in this part.

Assuming only the bits of the quantized spectral coefficients with signed codebooks (spectrum Huffman codebook 1, 2, 5, 6) are distorted, the global SNR results of HD and SD decoding are depicted in Tab. 5.8(a). Different to the global gain and scale factors, a slight SNR improvement of up to 1.05 dB is observed. HD even outperforms SD when $E_b/N_0 = 0$ dB. One reason for this observation is because of the small redundancy; another possible reason may be that in contrast to the global gain or scale factors, which take the estimated values directly or add them to some numbers, the estimated spectrum Huffman codebook index \hat{i}_c is further divided into a number of 2 or 4 received quantized spectral coefficients, which may not be able to lead to values closer to the original correct values, especially the quantized spectral coefficients are not unsigned values.

The global SNR results of distorting the bits of 10 spectrum Huffman codebooks are shown in Tab. 5.8(b), with SD.4CBs denoting SD decoding for a number of 4 signed codebooks, and SD.10CBs implying SD for a number of 10 codebooks including both

Method	E_b/N_0 (dB)										
	0	1	2	3	4	5	6	7	8	9	10
SD	8.86	9.60	10.55	11.77	13.16	15.04	16.65	17.35	17.66	17.73	17.74
HD	9.13	9.54	10.23	11.09	12.34	13.99	15.66	16.77	17.52	17.70	17.74

(a) Distorting a number of 4 signed codebooks

Method	E_b/N_0 (dB)										
	0	1	2	3	4	5	6	7	8	9	10
SD_10CBs	0.49	0.86	1.41	2.37	4.02	6.35	9.54	13.15	16.01	17.32	17.71
SD_4CBs	-0.79	-0.25	0.57	1.70	3.53	6.01	9.35	13.13	16.02	17.36	17.71
HD	-0.97	-0.40	0.42	1.52	3.38	5.77	9.03	12.93	15.91	17.32	17.70

(b) Distorting a number of 10 signed and unsigned codebooks (except codebook 11)

Table 5.8: Global SNR (dB) results for HE-AAC transmission over an AWGN channel, with corrupted quantized spectral coefficients (ODG results in Tab. D.1).

signed and unsigned codebooks, respectively. Compared to the best of HD and EC³², the SNR of using SD decoding is increased by up to 0.99 dB. The maximum SNR difference between SD_10CBs and SD_4CBs is 1.28 dB, which shows the advantage of using the new trellis representation from Section 5.4.3.

Two Parameters Distorted

The results above show that the SNR can be significantly improved when applying SD decoding to the global gain and scale factors, while being slightly enhanced for the quantized spectral coefficients. The simulation results of the HE-AAC transmission over an AWGN channel and a GE channel are shown in Tab. 5.9, with the bits of both global gain and scale factors being distorted. Three different SD decoding methods of the global gain g and the differential scale factors dsf with the option AK0 or AK1 are distinguished separately.

(1) Transmission Over an AWGN Channel

The SNR results for the transmission over an AWGN channel is depicted in Tab. 5.9(a). Compared to the best of HD and EC, applying SD decoding with AK0 to both global gain and scale factors (g_AK0/dsf_AK0), the global SNR has been significantly raised by up to 11.46 dB. Adopting AK1 to the global gain (g_AK1/dsf_AK0), the SNR can be improved even more by up to 0.58 dB (compared to g_AK0/dsf_AK0). Moreover, the performance is further enhanced by up to 0.50 dB using AK1 for both global gain and scale factors (g_AK1/dsf_AK1). The E_b/N_0 gain between SD and HD/EC reaches up to 5.8 dB. Therefore, considering the tradeoffs of complexity and performance, it can be stated that by using the method g_AK1/dsf_AK0 , most of the SNR gain can be achieved (up to 11.97 dB).

³²The SNRs of EC are always 0.

Method	E_b/N_0 (dB)										
	0	1	2	3	4	5	6	7	8	9	10
g_AK1/dsf_AK1	5.81	6.26	7.38	8.99	10.78	13.54	16.19	17.18	17.68	17.70	17.74
g_AK1/dsf_AK0	5.55	5.90	6.96	8.49	10.47	13.30	16.06	17.13	17.67	17.70	17.74
g_AK0/dsf_AK0	5.43	5.82	6.75	8.13	10.14	12.79	15.48	17.06	17.66	17.70	17.74
EC	0	0	0	0	0	0	0	0	0	0	0
HD	-18.69	-17.19	-15.53	-10.62	-8.20	1.33	7.67	13.73	16.90	17.36	17.74

(a) Transmission over an AWGN channel

GE parameters	Method	$E_b/N_0 _{\text{bad}}$ (dB)										
		5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0	9.5	10.0
$P=0.9$ $Q=0.1$ $P_{\text{bad}}=82\%$	g_AK1/dsf_AK1	14.04	15.46	16.38	16.88	17.26	17.48	17.70	17.72	17.70	17.74	17.74
	g_AK1/dsf_AK0	13.80	15.42	16.28	16.79	17.22	17.47	17.70	17.71	17.70	17.74	17.74
	g_AK0/dsf_AK0	13.34	14.65	15.86	16.48	17.14	17.45	17.69	17.71	17.70	17.74	17.74
$P=0.62$ $Q=0.38$ $P_{\text{bad}}=50\%$	HD	2.54	9.14	8.75	12.33	14.09	15.85	16.96	17.58	17.51	17.73	17.74
	EC	0.01	0.01	0	0.01	0.02	-0.01	0.02	0.01	0.02	0.01	0
	g_AK1/dsf_AK1	15.34	16.29	17.00	17.16	17.39	17.54	17.71	17.64	17.74	17.74	17.74
$P=0.62$ $Q=0.38$ $P_{\text{bad}}=50\%$	g_AK1/dsf_AK0	15.22	16.23	16.92	17.14	17.37	17.54	17.71	17.64	17.74	17.74	17.74
	g_AK0/dsf_AK0	14.69	15.43	16.68	16.66	17.34	17.54	17.71	17.64	17.74	17.74	17.74
	HD	8.47	11.18	12.15	13.68	15.47	16.54	17.22	17.64	17.60	17.74	17.74
EC	1.15	1.23	1.19	1.12	1.31	1.26	1.19	1.24	1.32	1.24	1.21	

(b) Transmission over a GE channel with $E_b/N_0|_{\text{good}}=10$ dB

Table 5.9: Global SNR (dB) results for HE-AAC transmission over an erroneous channel. Only the global gain and scale factors are being corrupted (ODG results in Tab. D.2).

(2) Transmission Over a GE Channel

As shown in Tab. 5.9(b), two GE channel models with $P=0.9, Q=0.1$ and $P=0.62, Q=0.38$ are adopted in two separate simulations. The corresponding P_{bad} and average error burst lengths can be found in Tab. 3.9.

It is known from Section 5.3 that the error concealment turns to muting after gradually fading out five times. When $P=0.9, Q=0.1$, the average error burst length is 5.1 frames. Therefore, EC results in muting from time to time ($\text{SNR} \approx 0$). In general, for a given GE channel model, a similar performance is observed regardless of channel conditions, due to the received but unused bits in bad frames. In other words, the EC performance is only dependent on the number of bad frames (P_{bad}) and the average error burst length. Moreover, once the frame is marked as bad, the whole frame is always substituted, even though $E_b/N_0|_{\text{bad}}$ is high and an acceptable performance is delivered by HD decoding. This explains why HD outperforms EC for all the channel conditions.

Similar to Tab. 5.9(a), the performance is enhanced by adopting g_AK0/dsf_AK0 , g_AK1/dsf_AK0 , and g_AK1/dsf_AK1 gradually. The E_b/N_0 gain between SD and HD reaches up to about 2 dB for both GE channels, while the SNR gain peaks at 11.50 dB and 6.87 dB for $P=0.9$ and $P=0.62$, respectively. Compared to g_AK1/dsf_AK0 , the SNR of g_AK1/dsf_AK1 is only increased by up to 0.24 dB and 0.12 dB for $P=0.9$ and $P=0.62$, respectively. Considering the high complexity of using AK1 for scale factors, the method g_AK1/dsf_AK0 is considered as the best solution for SD decoding, which improves the SNR by up to 11.27 dB [Han and Fingscheidt, 2015a] for $P=0.9$ and 6.75 dB for $P=0.62$, respectively.

GE parameters	Method	$E_b/N_0 _{\text{bad}}$ (dB)										
		5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0	9.5	10.0
$P=0.9$ $Q=0.1$ $P_{\text{bad}}=82\%$	g_AK1/dsf_AK1	-2.97	0.22	2.90	7.18	10.15	12.18	13.99	16.01	16.48	17.48	17.59
	g_AK1/dsf_AK0	-2.99	0.21	2.88	7.19	10.15	12.19	13.99	16.00	16.48	17.48	17.59
	g_AK0/dsf_AK0	-2.69	0.09	2.85	7.16	10.17	12.16	13.97	16.00	16.47	17.48	17.59
$P=0.62$ $Q=0.38$ $P_{\text{bad}}=50\%$	HD	-6.57	-2.18	-0.45	4.96	8.95	11.22	13.44	15.91	16.25	17.47	17.59
	EC	0.01	0.01	0	0.01	0.02	-0.02	0.02	0.01	0.02	-0.07	0
$P=0.9$ $Q=0.1$ $P_{\text{bad}}=82\%$	g_AK1/dsf_AK0	0.17	3.80	6.19	9.57	12.19	13.82	14.92	16.67	16.97	17.56	17.59
	HD	-3.49	0.86	4.40	7.93	11.38	13.29	14.48	16.68	16.82	17.56	17.59
	EC	1.07	1.22	1.11	1.12	1.23	1.20	1.19	1.16	1.25	1.16	1.14

(a) SNR (dB) results

GE parameters	Method	$E_b/N_0 _{\text{bad}}$ (dB)										
		5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0	9.5	10.0
$P=0.9$ $Q=0.1$ $P_{\text{bad}}=82\%$	g_AK1/dsf_AK1	-3.18	-3.25	-3.22	-3.12	-3.07	-2.92	-2.86	-2.80	-2.82	-2.79	-2.78
	g_AK1/dsf_AK0	-3.18	-3.26	-3.22	-3.12	-3.07	-2.92	-2.86	-2.80	-2.82	-2.79	-2.78
	g_AK0/dsf_AK0	-3.14	-3.23	-3.22	-3.13	-3.07	-2.92	-2.86	-2.80	-2.82	-2.79	-2.78
$P=0.62$ $Q=0.38$ $P_{\text{bad}}=50\%$	HD	-2.53	-2.82	-2.95	-3.10	-3.09	-2.99	-2.92	-2.82	-2.83	-2.79	-2.78
	EC	-3.38	-3.49	-3.42	-3.42	-3.44	-3.46	-3.50	-3.54	-3.45	-3.41	-3.46

(b) ODG results

Table 5.10: Simulation results for HE-AAC transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}}=10$ dB. The global gain, scale factors, and quantized spectral coefficients are being corrupted. The quantized spectral coefficients are HD-decoded.

Three Parameters Distorted

The simulation results of the HE-AAC transmission over a GE channel, with the global gain, scale factors, and quantized spectral coefficients being distorted are discussed in this part.

(1) HD Decoded Quantized Spectral Coefficients (Without Spectral Side Information)

Tab. 5.10 shows the simulation results of applying SD decoding to the global gain and scale factors, while the quantized spectral coefficients are HD-decoded. As presented in Tab. 5.10(a), for $P=0.9, Q=0.1$, compared to HD/EC, three different SD decoding methods g_AK0/dsf_AK0 , g_AK1/dsf_AK0 , g_AK1/dsf_AK1 have enhanced the performance, with the SNR gain and $E_b/N_0|_{\text{bad}}$ gain up to 2.9 dB and 0.56 dB, respectively. Comparing the three methods, the SNR difference between g_AK1/dsf_AK0 and g_AK0/dsf_AK0 is up to 0.12 dB, while the maximum performance gap between g_AK1/dsf_AK1 and g_AK1/dsf_AK0 is only 0.02 dB. Taking into account the complexity and the minor improvement by using the method g_AK1/dsf_AK1 , g_AK1/dsf_AK0 is again assumed as the best option for SD decoding, which increases the SNR by up to 2.88 dB, in comparison to HD/EC. Therefore, only the method g_AK1/dsf_AK0 is adopted for the transmission over a GE channel with $P=0.62, Q=0.38$. The corresponding SNR gain and $E_b/N_0|_{\text{bad}}$ gain between SD decoding and HD/EC reach up to 2.58 dB and 0.41 dB, respectively [Han and Fingscheidt, 2015a].

In addition, similar to the distortion of a single parameter, the ODG results in Tab. 5.10(b) again cannot reflect the performance adequately. Moreover, the ODG results of the three different SD decoding methods show the same performance.

GE parameters	Method	$E_b/N_0 _{\text{bad}}$ (dB)										
		5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0	9.5	10.0
$P=0.9$	$g_AK1/dsf_AK0/X_k_AK0$	-1.32	1.91	4.46	8.45	11.28	13.32	14.56	16.41	16.68	17.50	17.60
	$g_AK1/dsf_AK0/X_k_HD$	-1.54	1.57	4.06	8.16	11.03	13.20	14.44	16.22	16.69	17.50	17.59
$Q=0.1$	HD	-5.73	-1.40	0.32	5.81	9.79	12.09	13.88	16.11	16.47	17.49	17.59
	EC	0.01	0.01	0	0.01	0.02	-0.01	0.02	0.01	0.02	-0.07	0
$P=0.62$	$g_AK1/dsf_AK0/X_k_AK0$	1.58	5.13	7.29	10.73	13.16	14.81	15.46	16.90	17.07	17.55	17.60
	$g_AK1/dsf_AK0/X_k_HD$	1.23	4.82	7.03	10.38	13.06	14.90	15.30	16.84	17.07	17.57	17.59
$Q=0.38$	HD	-2.81	1.74	5.14	8.69	12.26	14.19	14.85	16.85	16.92	17.57	17.59
	EC	1.07	1.22	1.11	1.12	1.23	1.20	1.19	1.16	1.25	1.16	1.14
$P_{\text{bad}} = 82\%$												

Table 5.11: Global SNR (dB) results for HE-AAC transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB. The global gain, scale factors, and quantized spectral coefficients are being corrupted, with spectral side information (ODG results in Tab. D.3).

(2) *SD Decoded Quantized Spectral Coefficients With Spectral Side Information*

As mentioned in Section 5.4.3, the side information (i.e., total number of bits in each section) is required and known by the SD decoding of the quantized spectral coefficients. Therefore, a fair comparison between HD and SD decoding should be performed by both considering the side information of the quantized spectral coefficients. In order to observe the performance improved by applying the SD decoding to the quantized spectral coefficients, two different SD decoding methods are distinguished: $g_AK1/dsf_AK0/X_k_AK0$ and $g_AK1/dsf_AK0/X_k_HD$.

As the simulation results in Tab. 5.11 show, the HD decoding approach with the known spectral side information overtakes the HD decoding without side information in Tab. 5.10(a), for both GE channels. As expected, the EC performance remains the same as in Tab. 5.10(a).

Compared to HD/EC, it can be seen that using $g_AK1/dsf_AK0/X_k_HD$, the SNR is increased by up to 4.14 dB and 3.39 dB (compared to 2.88 dB and 2.58 dB in Tab. 5.10) for $P=0.9$ and $P=0.62$, respectively. The corresponding $E_b/N_0|_{\text{bad}}$ gain reaches up to 0.65 dB and 0.5 dB (compared to 0.56 dB and 0.41 dB in Tab. 5.10). However, in contrast to $g_AK1/dsf_AK0/X_k_HD$, applying SD decoding to the quantized spectral coefficients, the SNR is only slightly improved by a maximum of 0.4 dB when $P=0.9$ and 0.35 dB when $P=0.62$. It seems that SD decoding of the quantized spectral coefficients plays an insignificant role, with an indistinguishable improvement.

(3) *SD Decoded Quantized Spectral Coefficients With Spectral Side Information, Spectrum Huffman Codebook 11 Assumed Error-Free*

In order to exclude the influence of the spectrum Huffman codebook 11 representing the large quantized values which are always HD decoded, and to investigate the improvement by SD decoding purely, similar to Tab. 5.11 in part(2) above, the bits of the spectrum Huffman codebook 11 are assumed error-free in this part. As depicted in Tab. 5.12, the SNR gain and the $E_b/N_0|_{\text{bad}}$ gain between $g_AK1/dsf_AK0/X_k_AK0$ and HD/EC is up to 6.63 dB and 1.25 dB, while the corresponding values between $g_AK1/dsf_AK0/X_k_HD$ and HD/EC is 6.05 dB and 1.2 dB. The maximum SNR difference between the method of X_k with AK0 and X_k with HD is 0.61 dB.

GE parameters	Method	$E_b/N_0 _{\text{bad}}$ (dB)										
		5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0	9.5	10.0
$P=0.9$	$g_AK1/dsf_AK0/X_k_AK0$	6.64	8.42	9.96	12.01	13.44	14.70	16.15	17.06	17.27	17.67	17.71
$Q=0.1$	$g_AK1/dsf_AK0/X_k_HD$	6.06	7.81	9.50	11.64	13.25	14.58	16.07	16.88	17.28	17.67	17.70
$P_{\text{bad}} = 82\%$	HD	-1.43	4.04	4.36	8.58	11.21	13.45	15.48	16.76	17.09	17.66	17.70
	EC	0.01	0.01	0	0.01	0.02	0	0.02	0.01	0.02	0.01	0

Table 5.12: Global SNR (dB) results for HE-AAC transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB. The global gain, scale factors, *and* the bits of quantized spectral coefficients with Huffman codebook 1 to 10 are being corrupted, with spectral side information (ODG results in Tab. D.4).

To sum up, it can be stated that applying FL/SD decoding with AK1 to the global gain and VL/SD decoding with AK0 to the scale factors, the performance can be improved for the HE-AAC transmission over erroneous channel conditions. Further improvement can be observed by adopting VL/SD decoding to the quantized spectral coefficients, but only with a slight enhancement and for the cost of a significantly increased complexity.

Subjective Listening Test Results

The subjective listening test results of the mean MUSHRA score (the center point in the bar) and the associated 95% confidence interval (the bar) of the reference, anchor, and test signals are shown in Fig. 5.11. As mentioned in [EBU, 2003], a more complete comprehension of the performance comparison between different test conditions can be provided by the individual results of each audio file (file 1 to file 10) and the average results considering all the audio files (the column depicted as “all” in Fig. 5.11), separately. The MUSHRA scores of the reference and two anchors are presented in the first plot. The listening test results of two parameters (global gain and scale factors) being corrupted and three parameters (global gain, scale factors, and quantized spectral coefficients) being corrupted are shown in the second and third plot, respectively. In both plots, a GE channel with $P = 0.9, Q = 0.1$ is adopted, HD refers to the corresponding two or three parameters being HD-decoded. Considering the tradeoffs of the performance, complexity, and the amount of side information, the respective optimal SD decoding methods are adopted in the listening test, with SD indicating g_AK1/dsf_AK0 in the second plot, while $g_AK1/dsf_AK0/X_k_HD$ without spectral side information in the third plot, respectively.

It can be seen that the mean score of SD always exceeds HD, for all test files and test conditions. Moreover, some overlaps of the 95% confidence intervals are observed: For file 2 and file 4 when two parameters are corrupted, and for all the files if three parameters are corrupted as well. However, considering all audio files, for a given test condition, the average result shows no overlap of the 95% confidence interval between HD and SD. Therefore, it can be stated that the study of the mean score for all the audio files is sensible. In the following, the mean MUSHRA scores of all the test files will be discussed.

On the one hand, in the case of two corrupted parameters, the MUSHRA score gain between SD and HD reaches 34.19 and 30.56 for $E_b/N_0|_{\text{bad}} = 5$ dB and 6.5 dB, respectively. Correspondingly, the audio quality has been significantly improved from bad to fair, and from fair to good. On the other hand, the performance improvement for three corrupted

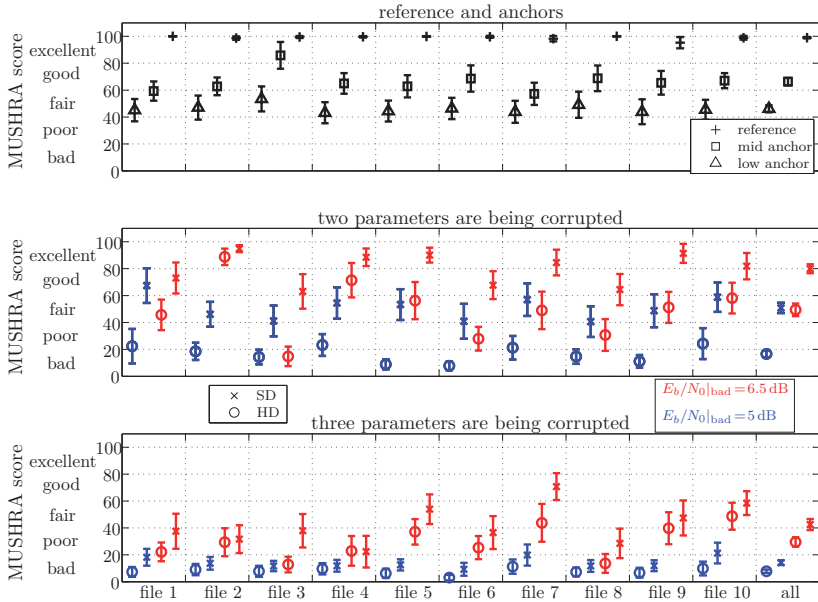


Figure 5.11: MUSHRA results with the mean and the associated 95% confidence interval (see Tabs. E.16 and E.17).

parameters is hardly noticeable. When $E_b/N_{0|bad} = 5$ dB, the score is only increased by 6.41, the quality of both HD and SD remains bad. However, for $E_b/N_{0|bad} = 6.5$ dB, using SD decoding, the audio quality has been improved from poor to fair, with a score improvement of 12.94. The results of the subjective listening test are consistent with the SNR results presented in Tab. 5.9(b) and 5.10(a), which further supports the observation that the audio quality correlates to the global SNR measure quite well.

5.6 Summary

In this chapter, aiming at an increased robustness for HE-AAC transmission over error-prone channels, the SD decoding approaches are applied to the HE-AAC v1, which includes AAC and SBR tools. The standard HE-AAC including AAC encoding, decoding, and the bit stream structure is introduced in detail. Thereafter, the FL/SD decoding approach from Chapter 2 is applied to the FL-coded global gain and the VL/SD decoding method from Chapter 4 is applied to the VL-coded differential scale factors and the quantized spectral coefficients in the AAC part. Moreover, due to the fact that the quantized spectral coefficients with unsigned spectrum Huffman codebooks include sign bits, which are appended to the corresponding codewords, a new trellis representation considering both Huffman codewords and sign bits is suggested in this chapter. The simulation

results show that the audio quality can be significantly enhanced by SD decoding. Considering the tradeoffs between complexity, performance, and the amount of side information, the recommended best SD decoding method is AK1 for the global gain and AK0 for the scale factors. In consistence with the results of a subjective listening test, the global SNR has been improved by up to 2.88 dB if the bits of three parameters are corrupted, while by up to 11.26 dB if the bits of the global gain and scale factors are distorted. In addition, the MUSHRA scores from the subjective listening test show a score improvement of up to 34.19 using SD decoding. Compared to the error concealment in 3GPP TS 26.402, using the proposed approach, not only the audio quality can be dramatically improved, but also no extra delay is introduced.

Chapter 6

Low-Rate Fixed-Length Hard- and Soft-Decision Decoding With Time-Variant Codebooks

In the previous chapters, the codewords for the quantization codebook indices are either fixed-length or variable-length. The common ground is that at the receiver side, the quantization codebook entry for a given quantization codebook index is time-invariant (i.e., memoryless). For correlated input signals, memoryless scalar quantization (e.g., Lloyd-Max quantization (LMQ)) is inefficient, due to the fact that it is designed in the same way for correlated as for uncorrelated processes. Utilizing the source correlation is normally done by predictive quantization using a predictor. Typically, the same predictor is used both in the encoder and the decoder. Moreover, in Adaptive Differential Pulse Code Modulation (ADPCM), although an adaptive predictor and adaptive quantization is used, residual correlation is still observed, which is obviously not utilized by the decoder. In order to obtain codebook entries that are - at a certain point in time - more close to the original unquantized values, the idea is now that given a standard encoder (without an encoder-sided predictor), we can utilize the (residual) correlation and generate a time-variant codebook in the decoder. In this chapter, a new *decoding* approach is proposed to improve the memoryless scalar quantization performance for correlated processes. Using the standard encoder at the transmitter side, a new time-variant quantization codebook is obtained based on a receiver-sided predictor. Section 6.1 introduces the outline of the new approach and the transmission system. The details of the new proposed approach with an hard-decision (HD) decoder are described in Section 6.2. The analysis and optimization of the receiver-sided prediction error probability density function (PDF) is presented in Section 6.3. Section 6.4 shows that the proposed approach can also be applied to the receiver with soft-decision (SD) decoding. Finally, the simulation setup and results are depicted and discussed in Section 6.5.

6.1 Introduction

Concerning the categories of quantization approaches listed in Section 2.1.2, the study in this chapter focuses on fixed-rate scalar quantization. Considerable redundancy can be observed in speech and images [Gray and Neuhoff, 1998], with redundancy related to the statistical source correlation. For correlated processes, the redundancy in terms of source correlation cannot be exploited by memoryless scalar quantization, since this approach is designed in the same way for correlated as for uncorrelated processes. By comparison, predictive quantization [Elias, 1955, Arnstein, 1975] and transform coding [Huang and Schultheiss, 1963] are two major approaches of scalar quantization with memory.

Predictive quantization is widely used in differential pulse code modulation (DPCM) [Jayant, 1974, McDonald, 1966] and ADPCM [Cummiskey et al., 1973, ITU-T, 1990, ITU-T, 1988]. For typical predictive quantization, a quantizer inside a prediction loop is required by the transmitter, with the same predictor being employed in the receiver. The transmitter-sided prediction error, which is the difference between the original input signal and its predicted signal, is quantized at the transmitter side. An open-loop and closed-loop predictive quantization is distinguished by whether the predicted signal is based on the past *original/unquantized* signal or on the past *reconstructed/quantized* signal, respectively. Assuming an $AR(N)$ process with zero-mean Gaussian innovation being used for the source process, the optimal transmitter-sided prediction error variance and predictor coefficients are the same as the variance of the innovation and the $AR(N)$ source process coefficients in open-loop predictive quantization, respectively, while they are supposed to differ from the corresponding values in a closed-loop scheme [Gersho and Gray, 1992, Chang and Gray, 1986]. Since the transmitter-sided prediction error is the input to the quantizer, the scaling of the quantization codebook is further influenced and different in the open-loop and closed-loop schemes. The effects above are more noticeable for low-rate quantization [Gersho and Gray, 1992, Chang and Gray, 1986], while for high-rates the quantization in the prediction loop is negligible. It is found that even though predictive quantization aims to decorrelate the source signals, residual correlation of the transmitter-sided prediction error is still observed [Han and Fingscheidt, 2015b].

Stemming from the idea of predictive quantization, as well as from the linear prediction scheme adopted in the soft-decision (SD) decoder [Pflug and Fingscheidt, 2011b, Pflug and Fingscheidt, 2012], a new *decoding* approach utilizing the source correlation is proposed in this chapter, aiming at an improved memoryless scalar quantization performance for correlated processes. A Gaussian $AR(N)$ process model is assumed for representing the correlated process. In contrast to predictive quantization, a standard Lloyd-Max quantization (LMQ) without any predictor is assumed in the encoder, only with a predictor inside a feedback loop being employed in the decoder. A new reconstruction level can be obtained based on a receiver-sided prediction error probability density function (PDF), generating a time-variant quantization codebook. Similar to the facts outlined above, the receiver-sided predictor coefficients and prediction error variance are not identical with the respective values of the Gaussian $AR(N)$ source signal, because of the feedback loop. For simplicity of description, the focus is on the Gaussian $AR(1)$ process in this thesis.

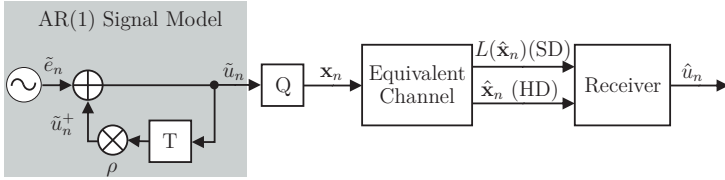


Figure 6.1: Block diagram of the transmission system [Han and Fingscheidt, 2014b].

The transmission system including the AR(1) signal model is introduced in the following.

The block diagram of the transmission system is shown in Fig. 6.1 [Han and Fingscheidt, 2014b]. The correlated source samples $\tilde{u}_0, \tilde{u}_1, \dots, \tilde{u}_n, \dots$ are taken from the AR(1) process with the i.i.d. Gaussian innovation $\tilde{e}_0, \tilde{e}_1, \dots, \tilde{e}_n, \dots$ having zero mean, variance $\sigma_{\tilde{e}}^2$, and sample time index $n \in \mathbb{N}_0$. With the correlation coefficient ρ of the AR(1) process, any source sample satisfies $\tilde{u}_n = \tilde{e}_n + \rho \cdot \tilde{u}_{n-1}$, with $\tilde{u}_0 = \tilde{e}_0$. The (unquantized) sample \tilde{u}_n is quantized to u_n by an M bit standard Lloyd-Max quantizer, and represented by a corresponding quantization index $i_n \in \mathcal{I} = \{0, 1, \dots, 2^M - 1\}$ or a (fixed rate) quantizer bit combination in bipolar notation $\mathbf{x}_n = \{-1, +1\}^M$. After the transmission over the equivalent channel, instead of receiving hard-decided bipolar bit combination $\hat{\mathbf{x}}_n = \{-1, +1\}^M$ in the traditional hard-decision (HD) decoding, the LLRs $L(\hat{\mathbf{x}}_n) = (L(\hat{x}_n(0)), L(\hat{x}_n(1)), \dots, L(\hat{x}_n(m)), \dots, L(\hat{x}_n(M-1))) \in \mathbb{R}^M$ of each received bit $\hat{x}_n(m) \in \{-1, +1\}$ are expected by the receiver utilizing SD decoding [Fingscheidt and Vary, 2001], with bit index $m \in \{0, 1, \dots, M-1\}$.

A transmitter-sided prediction is in fact comprised in the correlated Gaussian AR(1) signal model. According to the past sample \tilde{u}_{n-1} , the predicted sample can be obtained by $\tilde{u}_n^+ = a \cdot \tilde{u}_{n-1}$, with $a = \rho$ being the transmitter-sided predictor coefficient. Correspondingly, the transmitter-sided prediction error is written as $\tilde{u}_n - \tilde{u}_n^+ = \tilde{e}_n$ (equaling the innovation). If we are interested in the PDF of $\tilde{u}_n = \tilde{e}_n + \tilde{u}_n^+$, given a *known* predicted sample \tilde{u}_n^+ at time index n , the sample PDF $p_{\tilde{U}}(\cdot)$ of $\tilde{u}_n = \tilde{e}_n + \tilde{u}_n^+$ conditioned on a known deterministic value \tilde{u}_n^+ turns out to be the transmitter-sided prediction error PDF $p_{\tilde{E}}(\cdot)$ shifted by \tilde{u}_n^+ , which can be written as [Fingscheidt, 1998]

$$p_{\tilde{U}}(\tilde{u}_n | \tilde{u}_n^+) = p_{\tilde{E}}(\tilde{e}_n = \tilde{u}_n - \tilde{u}_n^+) = f(\tilde{u}_n). \quad (6.1)$$

Herein, the shifted PDF $p_{\tilde{E}}(\tilde{e}_n = \tilde{u}_n - \tilde{u}_n^+)$ of the transmitter-sided prediction error is a function of \tilde{u}_n for any given \tilde{u}_n^+ .

Moreover, according to the LMQ centroid condition [Gersho and Gray, 1992], the reconstruction level is the centroid of the region of the unquantized sample PDF $p_{\tilde{U}}(\tilde{u})$ in the i -th quantization interval $[d_i, d_{i+1}]$, with d_i and d_{i+1} being the decision levels. The centroid is marked as the black dot in the upper graph of Fig. 6.2. The reconstruction levels read [Jayant and Noll, 1984]

$$u^{(i)} = \frac{\int_{d_i}^{d_{i+1}} \tilde{u} \cdot p_{\tilde{U}}(\tilde{u}) d\tilde{u}}{\int_{d_i}^{d_{i+1}} p_{\tilde{U}}(\tilde{u}) d\tilde{u}}. \quad (6.2)$$

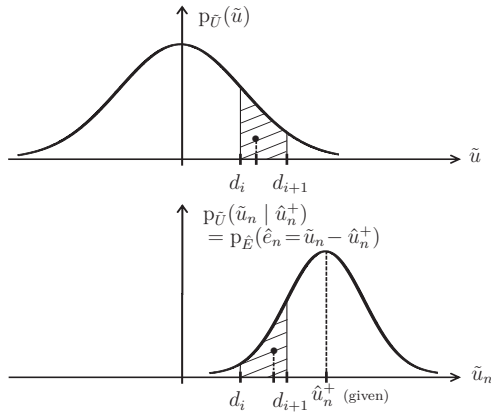


Figure 6.2: The upper plot is the PDF of the unquantized sample \tilde{u}_n , with one interval and its centroid being mashed. The lower plot is the shifted PDF of the receiver-sided prediction error \hat{e}_n [Han and Fingscheidt, 2014b], with the same interval being mashed, but due to the predicted sample \hat{u}_n^+ a different PDF $p_{\tilde{U}}(\tilde{u}_n | \hat{u}_n^+)$ is obtained, resulting in a different quantization interval centroid (i.e., reconstruction level).

6.2 New Predictive Hard-Decision Decoding

Applying the transmitter-sided prediction scheme to the receiver side, the proposed receiver with HD decoding is depicted in Fig. 6.3 [Han and Fingscheidt, 2014b]. For HD decoding, via inverse bit mapping (BM^{-1}), the received bit combination $\hat{\mathbf{x}}_n$ is first transformed to the received quantization index \hat{i}_n . This section starts with the application of the approach in error-free transmission conditions. In that case, the received quantization index \hat{i}_n is equal to the transmitted quantization index¹ i_n .

After estimating the previous sample \hat{u}_{n-1} at the receiver side, the current sample can be predicted by a first order predictor

$$\hat{u}_n^+ = \hat{a} \cdot \hat{u}_{n-1}, \quad (6.3)$$

with \hat{a} being the receiver-sided predictor coefficient and $\hat{u}_{n=0}^+ = 0$. Correspondingly, the receiver-sided prediction error is defined by $\hat{e}_n = \tilde{u}_n - \hat{u}_n^+$.

Replacing \tilde{u}_n^+ and \tilde{e}_n in (6.1) by \hat{u}_n^+ and \hat{e}_n , respectively, and consequently adapting (6.1) to the receiver, we have

$$p_{\tilde{U}}(\tilde{u}_n | \hat{u}_n^+) = p_{\tilde{E}}(\hat{e}_n = \tilde{u}_n - \hat{u}_n^+) = f(\tilde{u}_n). \quad (6.4)$$

Herein, $p_{\tilde{E}}(\hat{e}_n = \tilde{u}_n - \hat{u}_n^+)$ denotes the receiver-sided prediction error PDF $p_{\tilde{E}}(\cdot)$ shifted by the receiver-sided predictor output \hat{u}_n^+ , and is sketched in the lower plot of Fig. 6.2.

Furthermore, for a fixed received quantization index i at time n , considering the reconstruction level $u^{(i)}$ given a known predicted sample \hat{u}_n^+ (from (6.3)), the centroid condition

¹For the simplicity of notation, we will use $i=i_n$ equivalently in this section.

Applying (6.7) to (6.6), the new reconstruction level can be easily and quickly calculated by (see Section B.1 in Appendix B)

$$u_n^{(i)} = \hat{u}_n^+ - \sqrt{\frac{2}{\pi}} \hat{\sigma}_e \cdot \frac{\exp\left(\frac{-(d_{i+1} - \hat{u}_n^+)^2}{2\hat{\sigma}_e^2}\right) - \exp\left(\frac{-(d_i - \hat{u}_n^+)^2}{2\hat{\sigma}_e^2}\right)}{\operatorname{erf}\left(\frac{(d_{i+1} - \hat{u}_n^+)}{\sqrt{2}\hat{\sigma}_e}\right) - \operatorname{erf}\left(\frac{(d_i - \hat{u}_n^+)}{\sqrt{2}\hat{\sigma}_e}\right)}, \quad (6.8)$$

with the help of the error function² $\operatorname{erf}(\cdot)$. The resulting $u_n^{(i)}$ is used as the received sample \hat{u}_n at time index n .

Finally, replacing i by \hat{i}_n in (6.6) and (6.8), the resulting $u_n^{(\hat{i}_n)}$ is adopted as \hat{u}_n for any real (adverse) transmission condition with HD decoding.

6.3 Analysis and Optimization of the Prediction Error PDF

A prediction feedback loop can be observed in the receiver in Fig. 6.3. Therefore, the receiver-sided prediction error variance $\hat{\sigma}_e^2$ in (6.4) and (6.7) should be different from the transmitter-sided prediction error variance σ_e^2 of the AR(1) signal model in (6.1). Similarly, the predictor coefficients at the transmitter and receiver sides are also expected to be different. However, as mentioned in [Gersho and Gray, 1992], given the quantized past, the optimal coefficients are difficult to find. In addition, the reconstruction is assumed to be reasonably good. Therefore, the predictor coefficients in closed-loop predictive quantization are often taken from the values in open-loop predictive quantization.

Three different methods are utilized for the analysis and optimization of the prediction error PDF in error-free transmission conditions:

- (1) A one-dimensional full numerical search for the optimal value of the receiver-sided prediction error variance $\hat{\sigma}_{\text{opt}}^2$ [Han and Fingscheidt, 2014b];
- (2) An *analytical* solution of obtaining the receiver-sided prediction error variance $\hat{\sigma}_e^2$ [Han and Fingscheidt, 2015c];
- (3) A two-dimensional full numerical search to identify both an optimal receiver-sided prediction error variance $\hat{\sigma}_{\text{opt}}^2$ and an optimal predictor coefficient \hat{a}_{opt} [Han and Fingscheidt, 2014c].

Note that the receiver-sided predictor coefficients in (1) and (2) are considered the same as the transmitter-sided predictor coefficients ($\hat{a} = \rho$). Moreover, the full numerical search is performed in advance on some development data set. The optimal values are identified by the maximum SNR and are then used in the receiver. The *analytical* solution (2) is explained in detail in this section.

In order to model the receiver-sided prediction error in error-free transmission conditions, an approximation of the proposed decoder scheme (from Fig. 6.3) without feedback loop is represented by a simplified prediction scheme, as it is shown in Fig. 6.4. In this

² $\operatorname{erf}(a) = \frac{2}{\sqrt{\pi}} \int_0^a \exp(-x^2) dx.$

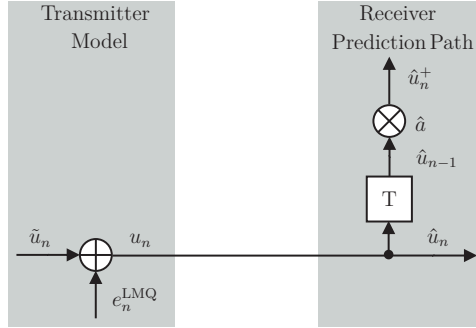


Figure 6.4: A simplified transmission and receiver-sided prediction model without feedback loop (for error-free transmission) [Han and Fingscheidt, 2015c].

scheme, the estimated sample \hat{u}_n is simply considered as the standard LMQ output without any transmission channel error ($\hat{u}_n = u_n$). Therefore, the receiver-sided prediction error turns out to be

$$\hat{e}_n = \tilde{u}_n - \hat{u}_n^+ = \tilde{u}_n - \hat{a} \cdot \hat{u}_{n-1} = \tilde{u}_n - \hat{a} \cdot u_{n-1}. \quad (6.9)$$

Including the Lloyd-Max quantization error $e_n^{\text{LMQ}} = u_n - \tilde{u}_n$, we have

$$\hat{e}_n = \tilde{u}_n - \hat{a} \cdot (\tilde{u}_{n-1} + e_{n-1}^{\text{LMQ}}). \quad (6.10)$$

For the purpose of obtaining the receiver-sided prediction error variance, first the expectation value needs to be calculated as

$$E\{\hat{e}_n\} = E\{\tilde{u}_n\} - \hat{a} \cdot E\{\tilde{u}_{n-1}\} - \hat{a} \cdot E\{e_{n-1}^{\text{LMQ}}\} = 0, \quad (6.11)$$

with the known property of the LMQ that the mean of the total quantization error $E\{e_{n-1}^{\text{LMQ}}\}$ is zero [Gersho and Gray, 1992], and the process of unquantized samples \tilde{u}_n has been assumed to be zero-mean. Thereafter, the receiver-sided prediction error variance can be computed by

$$\hat{\sigma}_e^2 = E\{(\hat{e}_n - E\{\hat{e}_n\})^2\} = E\{\hat{e}_n^2\}. \quad (6.12)$$

Applying (6.10) to (6.12), the variance $\hat{\sigma}_e^2$ can be further written as

$$\begin{aligned} \hat{\sigma}_e^2 &= E\{(\tilde{u}_n - \hat{a} \cdot (\tilde{u}_{n-1} + e_{n-1}^{\text{LMQ}}))^2\} \\ &= E\{\tilde{u}_n^2 + \hat{a}^2 \cdot (e_{n-1}^{\text{LMQ}})^2 - 2\hat{a}\tilde{u}_n \cdot (e_{n-1}^{\text{LMQ}})\} \\ &= E\{\tilde{u}_n^2 + \hat{a}^2 \tilde{u}_{n-1}^2 + \hat{a}^2 \cdot (e_{n-1}^{\text{LMQ}})^2 + 2\hat{a}^2 \tilde{u}_{n-1} e_{n-1}^{\text{LMQ}} - 2\hat{a}\tilde{u}_n \tilde{u}_{n-1} - 2\hat{a}\tilde{u}_n e_{n-1}^{\text{LMQ}}\} \\ &= E\{\tilde{u}_n^2\} + \hat{a}^2 E\{\tilde{u}_{n-1}^2\} + \hat{a}^2 E\{(e_{n-1}^{\text{LMQ}})^2\} \\ &\quad + 2\hat{a}^2 E\{\tilde{u}_{n-1} e_{n-1}^{\text{LMQ}}\} - 2\hat{a} E\{\tilde{u}_n \tilde{u}_{n-1}\} - 2\hat{a} E\{\tilde{u}_n e_{n-1}^{\text{LMQ}}\}, \end{aligned} \quad (6.13)$$

with each term being calculated in Section B.2 in Appendix B, finally resulting in

$$\begin{aligned} \hat{\sigma}_e^2 &= \frac{(1 + \hat{a}^2 - 2\hat{a}\rho) \cdot \sigma_e^2}{1 - \rho^2} - \frac{(\hat{a}^2 - 2\hat{a}\rho) \cdot \sigma_e^2}{\text{SNR}(M) \cdot (1 - \rho^2)} \\ &= \frac{((1 + \hat{a}^2 - 2\hat{a}\rho) \cdot \text{SNR}(M) - (\hat{a}^2 - 2\hat{a}\rho)) \cdot \sigma_e^2}{\text{SNR}(M) \cdot (1 - \rho^2)}, \end{aligned} \quad (6.14)$$

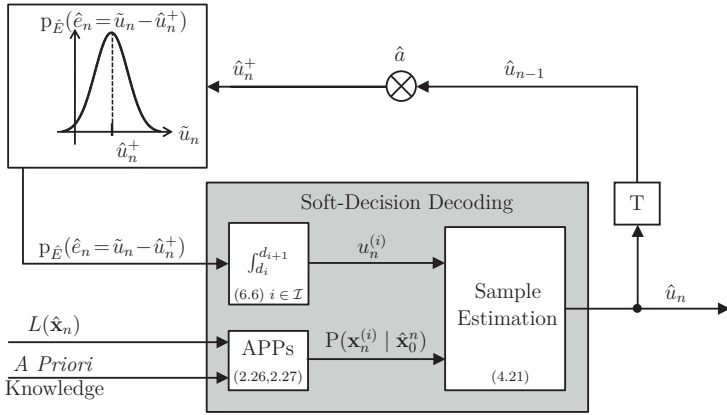


Figure 6.5: Block diagram of the proposed receiver with SD decoding [Han and Fingscheidt, 2014b].

with $\text{SNR}(M) = \frac{\sigma_u^2}{\sigma_{e^{\text{LMQ}}}^2}$ being the LMQ quantizer signal-to-noise ratio³ for different quantizer bit rates M .

In addition, assuming the reconstruction is reasonably good, the receiver-sided predictor coefficient can be regarded as the correlation coefficient of the Gaussian AR(1) process: $\hat{a} = \rho$. In consequence, (6.14) is then simplified to

$$\hat{\sigma}_e^2 = 1 + \frac{\rho^2 \cdot \sigma_e^2}{\text{SNR}(M) \cdot (1 - \rho^2)}. \quad (6.15)$$

6.4 New Predictive Soft-Decision Decoding

The proposed new approach can also be advantageously applied to the receiver with SD decoding [Han and Fingscheidt, 2014b], as depicted in Fig. 6.5, with $u_n^{(i)}$ being the new time-variant quantization codebook entry for quantization index $i \in \mathcal{I}$ and the APPs being calculated from (2.26) or (2.27), respectively. Note that the quantization codebook entry $u^{(i)}$ needs to be replaced by the new time-variant codebook entry $u_n^{(i)}$ in the minimum mean-square error (MMSE) sample estimation (2.32).

Moreover, the difference to the soft-decision decoding approach used in [Pflug and Fingscheidt, 2013a, Pflug and Fingscheidt, 2013b] is explained in the following. In [Pflug and Fingscheidt, 2013a, Pflug and Fingscheidt, 2013b], the prediction error PDF is used to calculate the *predictive a priori probabilities* (*prediction probabilities*) $P(\mathbf{x}_n^{(i)} | \hat{\mathbf{x}}_0^{n-1})$

³The values of $\text{SNR}(M)$ can be found in [Jayant and Noll, 1984, Tab. 4.4 on page 135]. Note that the values from [Jayant and Noll, 1984] are given in the log domain as dB, and the $\text{SNR}(M)$ here is a linear entity.

(see Section 2.3.2) by

$$\begin{aligned} P(\mathbf{x}_n^{(i)} | \hat{\mathbf{x}}_0^{n-1}) &\approx P(u_n^{(i)} | \hat{u}_n^+) \\ &= \int_{d_i}^{d_{i+1}} p_{\hat{E}}(\hat{e}_n = \tilde{u}_n - \hat{u}_n^+) d\tilde{u}_n. \end{aligned} \quad (6.16)$$

Thereafter, together with $L(\hat{\mathbf{x}}_n)$, the *a posteriori* probabilities $P(\mathbf{x}_n^{(i)} | \hat{\mathbf{x}}_0^n)$ can be calculated by (2.20). Finally the sample estimation is performed based on the *original* codebook entries $u^{(i)}$. However, in the new approach in this thesis, the prediction error PDF is used to get *new* codebook entries: (6.16) plays as the denominator in (6.6). The *prediction probabilities* are obtained either by (2.24) using a zeroth-order *a priori* knowledge or by (2.25) using a first-order *a priori* knowledge. Moreover, the predictor coefficients are adaptively updated by normalized least-mean-squares (NLMS) algorithm in [Pflug and Fingscheidt, 2013a, Pflug and Fingscheidt, 2013b], where the predictor with the order of one and fixed predictor coefficient is used here. However, the NLMS algorithm can also be utilized in the ADPCM scheme in Chapter 7.

6.5 Simulation Setup and Results

This section starts with the simulation setup. The performance of using the standard LMQ approach and the proposed approach will be compared afterwards.

6.5.1 Simulation Setup

A number of 10^6 unquantized source samples \tilde{u}_n is taken from an Gaussian AR(1) process with zero-mean and unit-variance ($\sigma_e^2 = 1$) innovation. LMQ codebooks with $M \in \{1, 2, 3, 4, 5\}$ bit are used in the various simulations. Assuming an AWGN channel, the bits are transmitted over different channel realizations for a given E_b/N_0 ratio. The global SNR with respect to the 10^6 source samples \tilde{u}_n is used to evaluate the performance.

For the simulations of SD decoding, a number of 10^8 quantized source samples is used as the training database to obtain the *a priori* knowledge AK0 (2.5) and AK1 (2.6).

6.5.2 Simulation Results

The simulation results of using the standard LMQ and the proposed receiver are discussed in this section.

- (1) *Using the Optimal Receiver-Sided Prediction Error Variances $\hat{\sigma}_{opt}$ and the Receiver-Sided Predictor Coefficients $\hat{u} = \rho$.*

The simulation results for $M = 1, 2, 3$ bit quantized Gaussian AR(1) samples with correlation coefficient $\rho = 0.9$ using both HD and SD decoding are shown in Fig. 6.6, with HD representing the receiver using HD decoding (Fig. 6.3), AK0 and AK1 denoting the use of soft decisions (Fig. 6.5), respectively. In Fig. 6.6, SD decoding outperforms the

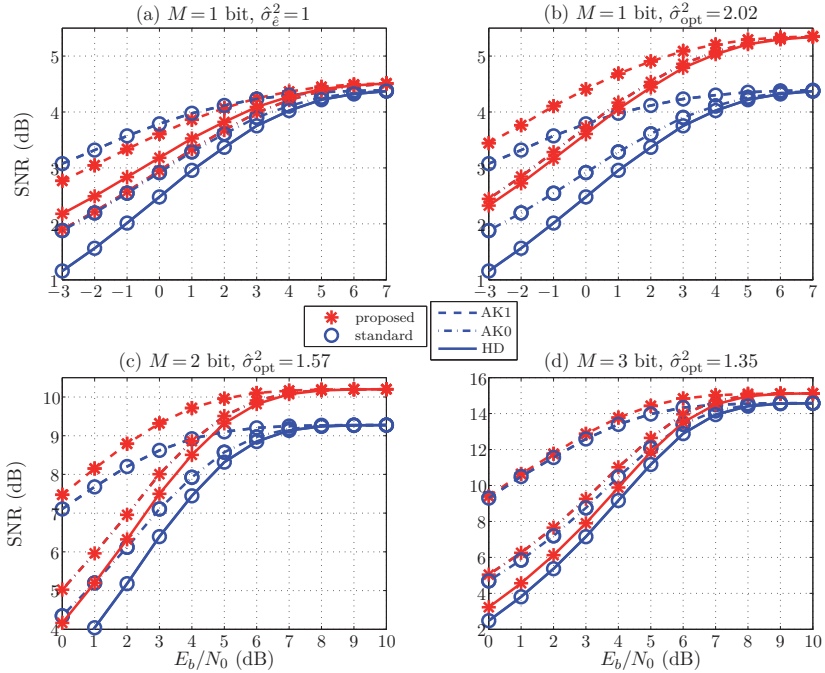


Figure 6.6: Simulation results for $M = 1, 2, 3$ bit quantized Gaussian AR(1) samples with correlation coefficient $\rho = 0.9$ and $\hat{a} = \rho$. Graph (a) shows the use of the transmitter-sided $\sigma_{\hat{\epsilon}}^2 = 1$ also at the receiver ($\hat{\sigma}_{\hat{\epsilon}}^2 = 1$), while graph (b) shows the results for using the optimal $\hat{\sigma}_{\hat{\epsilon}}^2$ from a numerical search (method (1) on page 104), both for $M = 1$. Graphs (c) and (d) show the results with optimal receiver-sided error variances for $M = 2$ and $M = 3$, respectively [Han and Fingscheidt, 2014b] (see Tabs. E.18 and E.19).

corresponding HD decoding, for both standard and proposed approaches. The superiority of SD decoding has already been discussed in Chapter 3, this chapter focuses on the performance comparison between the standard and the proposed HD decoding approach.

As can be seen in graph (a) in Fig. 6.6, using the transmitter-sided prediction error variance $\sigma_{\hat{\epsilon}}^2 = 1$ as the receiver-sided prediction error variance, the HD performance can be improved by the proposed approach. However, as described in Section 6.3, the value of $\sigma_{\hat{\epsilon}}^2 = 1$ is not an optimum variance value for the receiver-sided prediction error. A numerical search over the number range $1.2 \leq \hat{\sigma}_{\hat{\epsilon}} \leq 1.7$ in steps of 0.01 is performed to find the optimal receiver-sided prediction error variance $\hat{\sigma}_{\hat{\epsilon}}^2_{\text{opt}} = 2.02$. As shown in graph (b) of Fig. 6.6, using the optimal prediction error variance $\hat{\sigma}_{\hat{\epsilon}}^2_{\text{opt}}$, the SNR is significantly increased by the proposed approach, both with HD and SD decoding. The SNR gain of using the new approach vs. the standard approach can be improved from 0.1 dB ($\sigma_{\hat{\epsilon}}^2 = 1$) to 1.0 dB ($\hat{\sigma}_{\hat{\epsilon}}^2_{\text{opt}} = 2.02$) in *error-free transmission conditions*. A prominent improvement can

also be observed (see graphs (c) and (d) in Fig. 6.6) by using the new proposed decoder for $M=2$ bit and $M=3$ bit with respective optimized values $\hat{\sigma}_{\text{opt}}^2$, both with HD and SD decoding approaches.

The SNR gain between the proposed and the standard approach with HD decoding is up to 1.17 dB, 1.16 dB, and 0.75 dB for $M=1, 2, 3$ bit respectively, while up to 0.97 dB, 0.93 dB, and 0.56 dB for the receiver with SD decoding (highest gain at highest E_b/N_0). Therefore, for both HD and SD decoding, the performance can be improved by using the time-variant codebooks, especially for lower quantization rates.

- (2) *Using the Analytical Solution of Obtaining the Receiver-Sided Prediction Error $\hat{\sigma}_\varepsilon^2$ vs. the Two-Dimensional Numerical Search for the Optimal $\hat{\sigma}_{\text{opt}}^2$ and Optimal \hat{a}_{opt} .*

In order to evaluate the performance difference between using the *analytical* solution (2) and using the optimal numerically found $\hat{\sigma}_{\text{opt}}^2$, and to compare the performance of employing $\hat{a}=\rho$ and the optimal numerically found \hat{a}_{opt} as well, four different experiments are conducted as follows:

- (a) $\hat{a}=\rho, \hat{\sigma}_{\hat{a}=\rho}^2$: The receiver-sided predictor coefficient \hat{a} is the same as the correlation coefficient ρ , while the corresponding prediction error variance $\hat{\sigma}_{\hat{a}=\rho}^2$ is obtained from the *analytical* solution (6.15); this is method (2) on page 104;
- (b) $\hat{a}=\rho, \hat{\sigma}_{\text{opt}}^2$: A one-dimensional full numerical search for the optimal $\hat{\sigma}_{\text{opt}}^2$ (over the number range $1 \leq \hat{\sigma}_\varepsilon \leq 1.5$ in steps of 0.001) with $\hat{a}=\rho$; this is method (1) on page 104;
- (c) $\hat{a}=\hat{a}_{\text{opt}}, \hat{\sigma}_{\hat{a}=\hat{a}_{\text{opt}}}^2$: A one-dimensional full numerical search (over the number range $\rho-0.1 \leq \hat{a} \leq \rho+0.1$ in steps of 0.001) for the optimal \hat{a}_{opt} , with the corresponding $\hat{\sigma}_{\hat{a}=\hat{a}_{\text{opt}}}^2$ being calculated from the *analytical* solution (6.14);
- (d) $\hat{a}=\hat{a}_{\text{opt}}, \hat{\sigma}_{\text{opt}}^2$: The optimal values are obtained from the two-dimensional numerical search (over the number range $\rho-0.1 \leq \hat{a} \leq \rho+0.1$ in steps of 0.001, and $0.9 \leq \hat{\sigma}_\varepsilon \leq 1.2$ in steps of 0.001)⁴; this is method (3) on page 104.

Tab. 6.1 shows the simulation results for different quantization rates $M \in \{1, 2, 3, 4, 5\}$ in error-free transmission conditions, with the correlation coefficient $\rho=0.9$. It proves again the advantage of using the proposed approach, for all the quantization rates, especially for lower rates. The performance difference between the four experiments is discussed below.

The SNR is first increased by the proposed approach with $\hat{a}=\rho, \hat{\sigma}_{\hat{a}=\rho}^2$ (a) (in a range of 0.13 to 0.9 dB), for all the quantization rates, especially the lower-rate quantization takes profit. For $\hat{a}=\rho$, the SNR difference between using $\hat{\sigma}_{\hat{a}=\rho}^2$ (a) and $\hat{\sigma}_{\text{opt}}^2$ (b) is most recognizable for $M=1$ bit, with 0.16 dB SNR gain by using $\hat{\sigma}_{\text{opt}}^2$, while similar SNR results are observed for the other quantization rates, especially for $M=2$ bit and higher. It can be stated that the employment of the *analytical* solution (6.15) is reasonable, which is especially valid for $M=2$ bit and higher.

⁴Note that the values of the optimum prediction error variance $\hat{\sigma}_{\text{opt}}^2$ are shown in the simulation results, while the optimal search is performed with $\hat{\sigma}_\varepsilon$.

		M					
		1	2	3	4	5	
Standard	SNR	4.40	9.30	14.58	20.20	25.97	
	SNR	5.20	10.20	15.10	20.44	26.10	
Proposed	(a) $\hat{a} = \rho$	$\hat{a} = \rho$	0.9	0.9	0.9	0.9	0.9
		$\hat{\sigma}_{\hat{a}=\rho}^2$	2.55	1.50	1.15	1.04	1.01
	(b) $\hat{a} = \rho$	SNR	5.36	10.20	15.13	20.46	26.11
		$\hat{\sigma}_{\text{opt}}^2$	2.02	1.57	1.35	1.25	1.08
	(c) $\hat{a} = \hat{a}_{\text{opt}}$	SNR	5.26	10.23	15.15	20.48	26.11
		$\hat{a} = \hat{a}_{\text{opt}}$	0.855	0.930	0.945	0.944	0.933
	(d) $\hat{a} = \hat{a}_{\text{opt}}$	$\hat{\sigma}_{\hat{a}=\hat{a}_{\text{opt}}}^2$	2.55	1.51	1.16	1.05	1.02
		SNR	5.47	10.26	15.15	20.48	26.11
	(d) $\hat{a} = \hat{a}_{\text{opt}}$	$\hat{a} = \hat{a}_{\text{opt}}$	0.981	0.965	0.952	0.949	0.940
		$\hat{\sigma}_{\text{opt}}^2$	1.38	1.19	1.100	1.014	0.968

Table 6.1: SNR (in dB) results of the standard LMQ and the proposed LMQ decoder for M bit quantized Gaussian AR(1) samples in error-free transmission conditions, with the correlation coefficient $\rho=0.9$.

Compared to $\hat{a} = \rho$, $\hat{\sigma}_{\hat{a}=\rho}^2$ (a), the SNR is improved by using the optimized predictor coefficient $\hat{a} = \hat{a}_{\text{opt}}$ (c), which also provides a higher SNR than $\hat{a} = \rho$, $\hat{\sigma}_{\text{opt}}^2$ (b) for $M \in \{2, 3, 4\}$ bit. However, in comparison to $\hat{a} = \rho$, $\hat{\sigma}_{\text{opt}}^2$ (b), the SNR drops 0.1 dB by adopting $\hat{a} = \hat{a}_{\text{opt}}$, $\hat{\sigma}_{\hat{a}=\hat{a}_{\text{opt}}}^2$ (c) for $M = 1$ bit. It can be concluded that once the prediction error variance is chosen as the optimal one, using $\hat{a} = \rho$ can already achieve a good performance.

The best improvement is obtained by using $\hat{a} = \hat{a}_{\text{opt}}$ and $\hat{\sigma}_{\text{opt}}^2$ (d) from the two-dimensional numerical search, but only for the lower quantization rates ($M \in \{1, 2\}$). For the higher quantization rates (especially $M = 5$), most gains are already obtained by using ρ as the receiver-sided predictor coefficient and $\hat{\sigma}_{\hat{a}=\rho}^2$ (a) chosen from the *analytical* solution (6.15). At last, compared to the standard approach, using the new approach, the SNR can be increased by up to 1.07, 0.96, 0.57, 0.28, and 0.14 dB for $M \in \{1, 2, 3, 4, 5\}$ bit, respectively.

In error-free transmission conditions, the effect of the correlation is also investigated by varying the correlation coefficients ρ , as shown in Tab. 6.2 for $M=2$ bit quantized Gaussian AR(1) samples. If the process is uncorrelated, $\rho = 0$ leads to $\hat{a}_n^+ = 0$ and consequently results in the standard zero-mean PDF $p_E(\hat{e}_n = \hat{u}_n - 0) \equiv p_{\hat{U}}(\hat{u}_n)$. As a result, the SNR of using the proposed decoder is the same as using the standard LMQ for uncorrelated processes. Moreover, a larger SNR gain is observed with a higher correlation coefficient. It can be seen that four different methods provide exactly the same performance for less correlated samples ($\rho < 0.7$). For highly correlated samples ($\rho \in \{0.7, 0.8, 0.9\}$), both methods (a,b) adopting $\hat{a} = \rho$ lead to very similar SNR results. The SNR is gradually increased by performing the numerical search for the optimal $\hat{a} = \hat{a}_{\text{opt}}$, especially with the two-dimensional search for $\hat{a} = \hat{a}_{\text{opt}}$ and $\hat{\sigma}_{\text{opt}}^2$ (d), but only with a slight enhancement.

		ρ					
		0	0.3	0.5	0.7	0.8	0.9
Standard	SNR	9.30	9.30	9.30	9.30	9.30	9.30
Proposed	(a) SNR	9.30	9.35	9.45	9.66	9.86	10.20
	$\hat{a} = \rho$	0	0.3	0.5	0.7	0.8	0.9
	$\hat{\sigma}_{\hat{a}=\rho}^2$	1	1.01	1.04	1.11	1.21	1.50
	(b) SNR	9.30	9.35	9.45	9.67	9.86	10.20
	$\hat{a} = \rho$	0	0.3	0.5	0.7	0.8	0.9
	$\hat{\sigma}_{\text{opt}}^2$	1	1.02	1.05	1.14	1.27	1.57
	(c) SNR	9.30	9.35	9.45	9.67	9.87	10.23
	$\hat{a} = \hat{a}_{\text{opt}}$	0.005	0.308	0.516	0.730	0.835	0.930
	$\hat{\sigma}_{\hat{a}=\hat{a}_{\text{opt}}}^2$	1	1.01	1.04	1.11	1.21	1.51
	(d) SNR	9.30	9.35	9.45	9.67	9.88	10.26
	$\hat{a} = \hat{a}_{\text{opt}}$	0	0.31	0.52	0.74	0.86	0.97
	$\hat{\sigma}_{\text{opt}}^2$	1	1.01	1.03	1.08	1.13	1.19

Table 6.2: SNR (in dB) results of the standard LMQ and the proposed LMQ decoder for $M=2$ bit quantized Gaussian AR(1) samples having different correlation coefficients ρ in error-free transmission conditions.

Applying the proposed approach to error-prone transmission conditions, the simulation results for $M=1$ and 2 with the correlation coefficient $\rho=0.9$ are shown in Tab. 6.3 (HD decoding). On the one hand, for $M=1$ bit, the SNR results have been gradually increased by the proposed approach with (a) $\hat{a} = \rho$, $\hat{\sigma}_{\hat{a}=\rho}^2$; (c) $\hat{a} = \hat{a}_{\text{opt}}$, $\hat{\sigma}_{\hat{a}=\hat{a}_{\text{opt}}}^2$; (b) $\hat{a} = \rho$, $\hat{\sigma}_{\text{opt}}^2$; (d) $\hat{a} = \hat{a}_{\text{opt}}$, $\hat{\sigma}_{\text{opt}}^2$ for all the channel conditions, as in error-free transmission conditions ($M=1$ bit in Tab. 6.1). On the other hand, when $M=2$ bit, the performance rank follows (b)<(a)<(c)<(d) for bad to medium channel conditions ($E_b/N_0 \leq 5$ dB), while the performance satisfies (a)<(b)<(c)<(d) for medium to good channel conditions. However, the improvement between any two sequential methods is very small, by up to 0.11 dB for $M=1$ bit and only 0.03 dB for $M=2$ bit, respectively. The best improvement is always achieved by using (d) $\hat{a} = \hat{a}_{\text{opt}}$, $\hat{\sigma}_{\text{opt}}^2$. The SNR gain between the proposed and the standard approach reaches up to 1.2 dB for $M=1$ bit and 1.29 dB for $M=2$ bit, respectively.

To sum up, considering the simulation results of the transmission over error-free and error-prone conditions, it can be stated that the approximation from the *analytical* solution in Section 6.3 performs reasonably well for $M > 1$ bit. However, for $M=1$ bit, there is an SNR difference of about 0.21 dB between the approximation (c) and the two-dimensional full numerical search (d) in error-free transmission conditions. In addition, most of the gains are already obtained by taking the predictor coefficient from the source process correlation coefficient ($\hat{a} = \rho$).

E_b/N_0	$M=1$ bit					$M=2$ bit				
	Standard	Proposed				Standard	Proposed			
		(a) $\hat{a}=\rho$ $\hat{\sigma}_{\hat{a}=\rho}^2$	(b) $\hat{a}=\rho$ $\hat{\sigma}_{\text{opt}}^2$	(c) $\hat{a}=\hat{a}_{\text{opt}}$ $\hat{\sigma}_{\hat{a}=\hat{a}_{\text{opt}}}^2$	(d) $\hat{a}=\hat{a}_{\text{opt}}$ $\hat{\sigma}_{\text{opt}}^2$		(a) $\hat{a}=\rho$ $\hat{\sigma}_{\hat{a}=\rho}^2$	(b) $\hat{a}=\rho$ $\hat{\sigma}_{\text{opt}}^2$	(c) $\hat{a}=\hat{a}_{\text{opt}}$ $\hat{\sigma}_{\hat{a}=\hat{a}_{\text{opt}}}^2$	(d) $\hat{a}=\hat{a}_{\text{opt}}$ $\hat{\sigma}_{\text{opt}}^2$
0	2.48	3.49	3.61	3.51	3.68	3.01	4.19	4.16	4.19	4.30
1	2.96	3.94	4.05	3.96	4.13	4.04	5.22	5.20	5.23	5.33
2	3.37	4.31	4.44	4.34	4.52	5.18	6.34	6.32	6.35	6.44
3	3.75	4.65	4.79	4.69	4.88	6.39	7.51	7.50	7.53	7.61
4	4.02	4.89	5.03	4.93	5.13	7.45	8.51	8.51	8.54	8.60
5	4.22	5.06	5.21	5.11	5.32	8.32	9.32	9.32	9.35	9.40
6	4.32	5.14	5.30	5.20	5.41	8.85	9.82	9.82	9.85	9.88
7	4.37	5.18	5.34	5.24	5.46	9.14	10.07	10.08	10.10	10.14
8	4.39	5.20	5.36	5.26	5.47	9.24	10.17	10.17	10.20	10.23
9	4.39	5.20	5.36	5.26	5.48	9.27	10.19	10.20	10.22	10.25
10	4.39	5.20	5.36	5.26	5.48	9.27	10.20	10.20	10.23	10.26

Table 6.3: SNR (in dB) results of the standard LMQ and the proposed LMQ decoder for $M=1$ bit and 2 bit quantized Gaussian AR(1) samples in error-prone transmission conditions (HD decoding), with the correlation coefficient $\rho=0.9$.

6.6 Summary

In order to improve the scalar quantization performance for the correlated processes, a new *decoding* approach employing a predictor only at the receiver side has been proposed in this chapter. In this approach, a time-variant codebook with time-variant reconstruction levels is generated based on a receiver-sided prediction error probability density function (PDF). The analysis and optimization of the prediction error PDF include either a one- or a two-dimensional full numerical search, while also an analytical solution for the prediction error variance is provided. The simulation results show a significant SNR improvement in a range of 0.14 to 1.07 dB in error-free transmission conditions, especially for low-rate scalar quantization of highly correlated processes. The proposed approach is also applicable to error-prone transmission conditions, either with hard-decision (HD) or soft-decision (SD) decoding. Simulations over an additive white Gaussian noise (AWGN) channel show an improvement of up to 1.29 dB SNR. In addition, comparing the simulation results of different methods, the analytical solution is shown to perform reasonably well. The proposed approach is compatible with the standard transmitter-sided Lloyd-Max quantization (LMQ) encoder. The concept of this approach can be applied to speech and audio codecs as will be shown in Chapter 7, as the example of ADPCM speech coding.

Chapter 7

Improving G.726 and G.722 ADPCM by Hard-Decision Decoding With Time-Variant Codebooks

With the standardization in ITU-T Recommendations G.726 and G.722, Adaptive Differential Pulse Code Modulation (ADPCM) is widely used in cordless and internet protocol (IP) telephony. Although adaptive prediction and adaptive quantization is employed in ADPCM, residual correlation of the quantization input signals is still observed. In ADPCM, a fixed scalar quantization codebook/lookup table is adopted. In this chapter, in order to improve the scalar quantization performance in ADPCM, the improved scalar quantization approach proposed in Chapter 6 is applied to the ADPCM decoder, with an unchanged ADPCM encoder. The standard G.726 and G.722 ADPCM speech coding is recapitulated in Section 7.1. Applying the approach from Chapter 6, a new ADPCM decoding scheme with time-variant codebooks is proposed in Section 7.2, performing hard-decision (HD) decoding both in error-free and error-prone transmission conditions. At the end, the simulation setup and results are presented in Section 7.3.

7.1 G.726 and G.722 ADPCM Speech Coding

Adaptive Differential Pulse Code Modulation (ADPCM) is widely used in voice over internet protocol (VoIP), digital enhanced cordless telephony (DECT) [ETSI, 1992b], New Generation DECT [ETSI, 2007a], and cordless advanced technology-internet and quality (CAT-iq). Considering that speech is non-stationary, compared to the DPCM scheme which works well for stationary signals and fixed predictor coefficients, ADPCM including adaptive prediction backwards (APB) and adaptive quantization backwards (AQB) methods [Jayant and Noll, 1984] offers a better speech quality.

For the non-stationary input process with the variance changing over time, a better quantization performance can be achieved by adaptive quantization, especially for low-rate quantization [Jayant and Noll, 1984]. The step size adaptation is based on

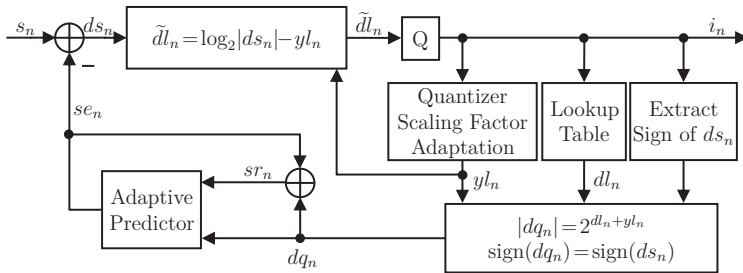


Figure 7.1: Block diagram of the standard G.726 ADPCM encoder (as in [Han and Fingscheidt, 2015b]).

the reconstructed/quantized signal in adaptive quantization with backward estimation (AQB).

For similar reasons of the non-stationary property of speech, compared to using fixed predictor coefficients, the predictor coefficients should be adapted over time leading to a higher prediction gain [Jayant and Noll, 1984]. In APB, the adaptation information is recursively computed according to the previously quantized signals. A predictor order of 10 is normally adequate for speech with a sampling rate of 8 kHz, but a second-order all-pole predictor and a sixth-order all-zero predictor (autoregressive moving-average (ARMA) predictor) are adopted in ADPCM [ITU-T, 1990, ITU-T, 1988].

7.1.1 G.726

In 1990, ADPCM was standardized as ITU-T Recommendation G.726 [ITU-T, 1990], which offers speech quality at different levels and operates at bit rates of 16, 24, 32, and 40 kbps. The G.726 codec [ITU-T, 1990] is the mandatory narrowband codec for New Generation DECT [ETSI, 2007b]. The G.726 ADPCM encoding and decoding from [ITU-T, 1990] will be briefly explained in the following.

G.726 ADPCM Encoding

As the simplified block diagram of the standard G.726 ADPCM encoder depicted in Fig. 7.1 shows, at time index n , the difference signal¹ ds_n is generated by $ds_n = s_n - se_n$, with the original input signal s_n (uniform PCM) and the estimated signal se_n . The difference signal ds_n is transformed to a base-2 logarithmic representation and scaled by a scaling factor yl_n , which is obtained from the quantizer scaling factor adaptation block. The resulting normalized difference signal \tilde{d}_l is scalar quantized and represented by a corresponding quantization index i_n with two, three, four or five bits for the operation at 16, 24, 32 or 40 kbps, respectively. The first bit of i_n implies the sign of the difference signal ds_n . Via a lookup table, the quantization index i_n is assigned to the quantized

¹ ds is denoted as d in [ITU-T, 1990].

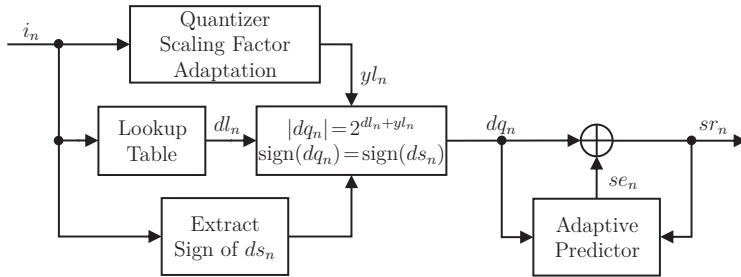


Figure 7.2: Block diagram of the standard G.726 ADPCM decoder (as in [Han and Fingscheidt, 2015b]).

normalized difference signal d_{l_n} . The quantized difference signal d_{q_n} can be obtained by performing $|dq_n| = 2^{d_{l_n} + y_{l_n}}$, with the sign being extracted from i_n , as $\text{sign}(ds_n)$. The summation of the quantized difference signal $d_{q_n} = |dq_n| \cdot \text{sign}(dq_n)$ and the estimated signal se_n results in the reconstructed signal $sr_n = dq_n + se_n$. The estimated signal se_n is generated by an adaptive ARMA predictor in a feedback loop with the input being the quantized difference signal d_{q_n} and the reconstructed signal sr_n .

G.726 ADPCM Decoding

Including the same feedback loop structure as the encoder, the simplified block diagram of the G.726 ADPCM decoder is shown in Fig.7.2. In the decoder and the case of an error-free transmission, the estimated signal se_n , quantized difference signal d_{q_n} , and reconstructed signal sr_n adopt the same values as in the encoder.

7.1.2 G.722

In order to improve the speech quality by extending the bandwidth from narrowband to wideband, the G.722 wideband codec [ITU-T, 1988] is mandatory for New Generation DECT [ETSI, 2007b], and virtually found in all modern IP phones. Subband adaptive differential pulse code modulation (SB-ADPCM) is used in Recommendation G.722 with a bit rate of 64 kbps. Three operation modes are included in G.722 with the coding bit rates of 64, 56, and 48 kbps corresponding to auxiliary data channel bit rates of 0, 8, and 16 kbps, respectively. The 64 kbps mode is adopted in this thesis. In the following we briefly introduce G.722 ADPCM encoding and decoding from [ITU-T, 1988].

G.722 ADPCM Encoding

The block diagram of the standard G.722 ADPCM encoder is depicted in Fig.7.3. The 14 bit uniform digital input signal s_n with 16 kHz sampling rate is first transmitted to quadrature mirror filters (QMFs) consisting of two linear-phase non-recursive digital filters, with n being the time index. The frequency band 0 to 8000 Hz is therefore split into

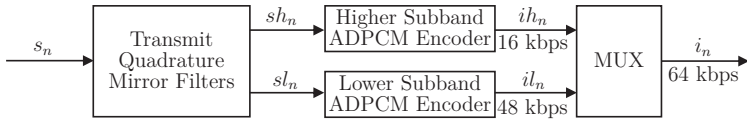


Figure 7.3: Block diagram of the standard G.722 ADPCM encoder in 64 kbps mode [ITU-T, 1988].

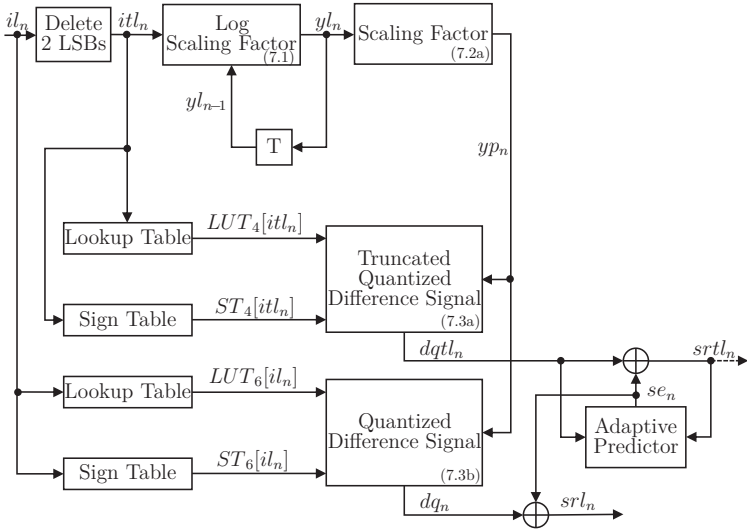


Figure 7.4: Block diagram of the standard G.722 lower subband decoder in 64 kbps mode [ITU-T, 1988].

a lower subband (0 to 4000 Hz) and a higher subband (4000 to 8000 Hz). The outputs of the QMFs are the higher subband signal s_{h_n} and the lower subband signal s_{l_n} , with both signals being sampled at 8 kHz. Thereafter, the signals s_{h_n} and s_{l_n} are transmitted to a higher subband ADPCM encoder delivering a 16 kbps bit stream and a lower subband ADPCM encoder providing a 48 kbps bit stream, respectively. The two output signals are represented by a corresponding 2-bit quantizer index i_{h_n} and a 6-bit quantizer index i_{l_n} . Subsequently, via the multiplexer, the signals i_{h_n} and i_{l_n} are combined to a composite 64 kbps signal and represented by i_n with 8 bits. In addition, a frame size of 10 ms corresponding to 160 samples (each with 8 bits) per frame is defined in [ITU-T, 2006].

G.722 ADPCM Decoding

The G.722 ADPCM decoder performs to some extent the reverse process of the encoder. The block diagrams of the lower and higher subband decoders are depicted in Figs. 7.4 and 7.5, respectively. For the simplicity of description, the signals with the same impli-

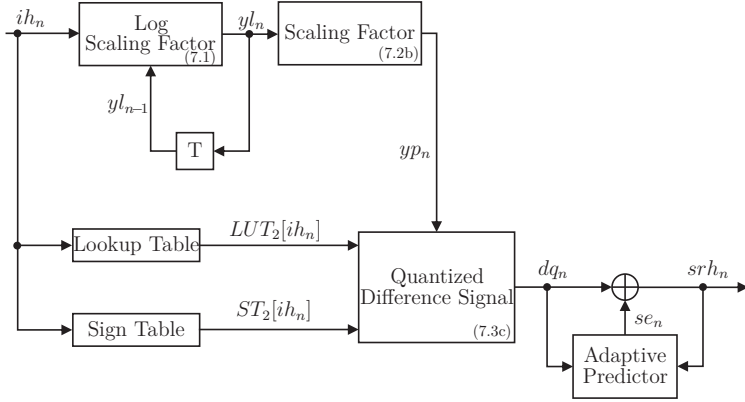


Figure 7.5: Block diagram of the standard G.722 higher subband decoder in 64 kbps mode [ITU-T, 1988].

ation but in different subbands are denoted by the same notations (i.e., y^{l_n} , y_{p_n} , dq_n , se_n) in this section. As shown in Fig. 7.4, the lower subband input signal il_n with 6 bits is truncated by the two least significant bits to itl_n having 4 bits, which is used for the quantizer scaling factor adaptation. The scaling factor is updated in the log domain, with the log scaling factor being calculated by

$$y^{l_n} = \frac{127}{128} \cdot y^{l_{n-1}} + W_M[itl_n]. \quad (7.1)$$

Herein, W_M is the logarithmic scaling factor multiplier in a lookup table (Tab.7 in Recommendation G.722) with $M=4$ for the lower subband. Note that for the higher subband, $W_M[itl_n]$ is replaced by $W_M[ih_n]$ with the higher subband input signal ih_n and $M=2$ (Tab.8 in Recommendation G.722).

Thereafter, the log scaling factor is converted to the linear domain:

$$y_{p_n} = 2^{y^{l_n+2}} \cdot \Delta_{\min} \quad (\text{lower subband}), \quad (7.2a)$$

$$y_{p_n} = 2^{y^{l_n}} \cdot \Delta_{\min} \quad (\text{higher subband}), \quad (7.2b)$$

with Δ_{\min} being equal to half the quantizer step size of the 14 bit analogue-to-digital converter.

Subsequently, according to the inverse adaptive quantizer lookup tables and corresponding sign tables, the truncated quantized difference signal $dqtl_n$ (only in the lower subband) and the quantized difference signal dq_n can be obtained by:

$$dqtl_n = ST_4[itl_n] \cdot LUT_4[itl_n] \cdot y_{p_n} \quad (\text{lower subband}), \quad (7.3a)$$

$$dq_n = ST_6[ih_n] \cdot LUT_6[ih_n] \cdot y_{p_n} \quad (\text{lower subband}), \quad (7.3b)$$

$$dq_n = ST_2[ih_n] \cdot LUT_2[ih_n] \cdot y_{p_n} \quad (\text{higher subband}), \quad (7.3c)$$

with LUT referring to the corresponding lookup table and ST denoting the sign table.

In the lower subband, adding the truncated quantized difference signal $dqtl_n$ and the estimated signal se_n , the truncated reconstructed signal $srtl_n$ is obtained by $srtl_n = dqtl_n + se_n$. Both $dqtl_n$ and $srtl_n$ are used as the input to the adaptive predictor to generate se_n in a feedback loop. The lower subband reconstructed signal srl_n is the sum of the (non truncated) quantized difference signal dq_n and the estimated signal se_n , yielding $srl_n = dq_n + se_n$. Similarly, for the higher subband, we have $srh_n = dq_n + se_n$. Finally, both srl_n and srh_n are used to reconstruct the output signal sr_n by a QMF.

7.2 New Hard-Decision Decoding With Time-Variant Codebooks

The speech quality of the ADPCM system has been enhanced by two frequency-domain approaches: noise spectral shaping at the speech encoder [Makhoul and Berouti, 1979] and adaptive post-filtering at the decoder output [Chen and Gersho, 1995]. The combination of these two approaches is adopted in [Ramamoorthy et al., 1988, Lee, 1999], with both encoder and decoder being modified. The soft-decision (SD) decoding approach mentioned in Chapter 2 has been applied to the G.726 ADPCM decoder in [Fingscheidt, 2003] and to G.722 ADPCM decoder in [López-Oller et al., 2016], respectively. With this approach, however, the ADPCM performance has only been improved in error-prone transmission conditions.

In ADPCM, although the difference signal is scalar quantized, residual correlation of the difference signal is still observed. Replacing \tilde{u}_ℓ by the corresponding quantizer input signal in ADPCM (e.g., \tilde{d}_n in Fig. 7.1 in G.726 ADPCM) and using $\varkappa = 1$ in (2.2), the correlation coefficients² for G.726 ADPCM with 16 kbps and 32 kbps are 0.32 and 0.2, respectively. Similarly, the correlation coefficient for the higher subband G.722 ADPCM reaches 0.33.

Exploiting this moderate source correlation, the approach proposed in Chapter 6 can improve the quantization performance, especially for the lower quantization rates. As a result, the concept of the improved scalar quantization approach is applied to G.726 ADPCM operating at 16 kbps (with a 2-bit quantizer) and 32 kbps (with a 4-bit quantizer), and to the higher subband of G.722 (with a 2-bit quantizer) as well. Due to the similarity of G.726 ADPCM and the subband structure in G.722 ADPCM, the principle of both applications is very similar. For ease of description, the 16 kbps mode with a 4-level quantizer (i.e., $i_n \in \{0, 1, 2, 3\}$) as being adopted in G.726 ADPCM [ITU-T, 1990] or in the higher subband of G.722 ADPCM [ITU-T, 1988], is focused in the following description. However, our new improved hard-decision (HD) decoding approach is also applicable for the other bit rates. As already shown in Section 6.4, the improved scalar quantization approach can be used both with HD and SD decoding. The superiority of SD decoding has already been discussed in Chapters 3 and 6. Therefore, the HD decoding is focused in this chapter. In this section, first the ADPCM transmission over error-free

²These results are the mean values of the correlation coefficients from a number of 96 English speech files spoken by American native speakers with the NTT monaural speech database [NTT, 1994].

channel conditions is investigated, and the application to the error-prone transmission conditions is presented afterwards.

7.2.1 Error-Free Transmission Conditions

An Outline

A better reconstructed signal sr_n is needed for an improved ADPCM performance. As shown in Figs. 7.1 and 7.2, due to $sr_n = dq_n + se_n$, the signal can be reconstructed better if either the signal is better estimated or the difference signal is better quantized. As mentioned in Section 7.1.1, the estimated signal se_n is generated by an adaptive ARMA predictor with the quantized difference signal dq_n and the reconstructed signal sr_n being the input in a feedback loop. For the sake of keeping the ARMA predictor coefficients estimation synchronized in the encoder and decoder, the standard ADPCM decoder providing signals se_n and yl_n should be adopted. In other words, the goal is to calculate a new and better quantized difference signal \widehat{dq}_n which is further used to obtain a better reconstructed signal $\widehat{sr}_n = se_n + \widehat{dq}_n$ [Han and Fingscheidt, 2015b].

Moreover, as can be seen in Fig. 7.1, the normalized difference signal \widetilde{dl}_n is actually the encoder quantizer input. Comparing $|ds_n| = 2^{\widetilde{dl}_n + yl_n}$ in the encoder and $|dq_n| = 2^{dl_n + yl_n}$ in the decoder, it can be stated that based on the same given scaling factor yl_n , the new quantized difference signal \widehat{dq}_n will be closer to the original difference signal ds_n at the encoder side, if \widetilde{dl}_n is quantized with a smaller quantization error (i.e., dl_n is closer to \widetilde{dl}_n). A better quantized normalized difference signal \widehat{dl}_n can be obtained by applying the improved scalar quantization approach from Chapter 6 [Han and Fingscheidt, 2015b], which is outlined in the following.

Application to ADPCM

For the training process to obtain the probability density function (PDF) $p_{\widetilde{DL}}(\widetilde{dl})$ of the unquantized signal \widetilde{dl} , the active speech part (speech pause excluded) of 20 languages excluding English spoken by American native speakers from the NTT monaural speech database [NTT, 1994] with 8 kHz sampling rate is used. According to the PDF $p_{\widetilde{DL}}(\widetilde{dl})$ and the centroid condition [Gersho and Gray, 1992] in (6.2), an Lloyd-Max quantization (LMQ) codebook is obtained. The reconstruction level $dl^{(i)}$ is calculated by

$$dl^{(i)} = \frac{\int_{I^{(i)}} \widetilde{dl} \cdot p_{\widetilde{DL}}(\widetilde{dl}) d\widetilde{dl}}{\int_{I^{(i)}} p_{\widetilde{DL}}(\widetilde{dl}) d\widetilde{dl}}, \quad (7.4)$$

with $I^{(i)}$ being the quantization interval for quantization index i . As expected, using a number of 96 American English speech files, compared using the standard quantization codebook/lookup table from Recommendation G.726 to using the reconstruction levels from (7.4), the same perceptual evaluation of speech quality (PESQ) mean opinion score (MOS) of 2.48 is observed. As a consequence, it can be stated that the conditional centroid

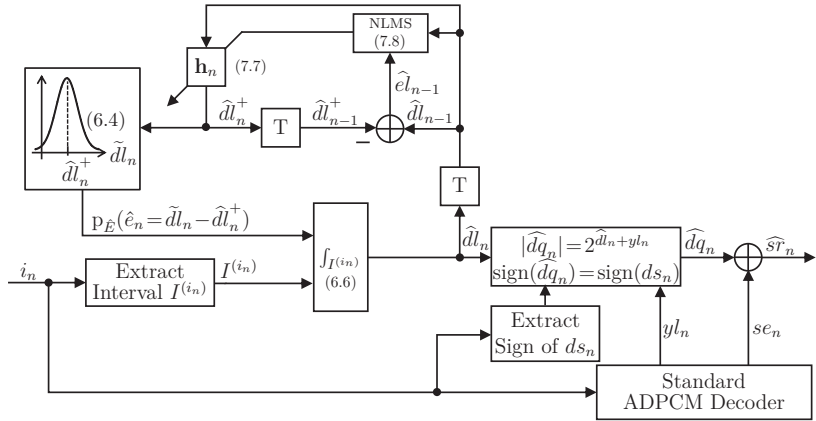


Figure 7.6: Block diagram of the proposed bit-stream compatible ADPCM decoder in error-free transmission conditions [Han and Fingscheidt, 2015b].

condition (6.6) can be further utilized and written as

$$dl_n^{(i_n)} = \frac{\int_{I^{(i_n)}} \tilde{d}l_n \cdot p_{\tilde{E}}(\hat{e}_n = \tilde{d}l_n - \hat{d}l_n^+) d\tilde{d}l_n}{\int_{I^{(i_n)}} p_{\tilde{E}}(\hat{e}_n = \tilde{d}l_n - \hat{d}l_n^+) d\tilde{d}l_n}, \quad (7.5)$$

with the receiver-sided prediction error PDF $p_{\text{DL}}(\tilde{d}l_n | \hat{d}l_n^+) = p_{\tilde{E}}(\hat{e}_n = \tilde{d}l_n - \hat{d}l_n^+) = f(\tilde{d}l_n)$ being a function of the quantizer input $\tilde{d}l_n$ [Han and Fingscheidt, 2015b].

The block diagram of the proposed ADPCM decoder is shown in Fig. 7.6. In error-free transmission conditions, the correct quantization index i_n is received by the decoder, with a two-bit representation: The first bit implies the sign and the second bit represents the magnitude. According to the value of $|ds_n|$, in fact one bit amplitude quantization is performed in the quantization domain (i.e., $\tilde{d}l_n$): $i_n=0$ or $i_n=1$ for $ds_n > 0$ and $i_n=2$ or $i_n=3$ for $ds_n < 0$. Therefore, the extraction of the actual 1-bit quantization interval (I_0, I_1) satisfies

$$I^{(i_n)} = \begin{cases} I_0, & \text{if } i_n=0 \text{ or } i_n=3, \\ I_1, & \text{else.} \end{cases} \quad (7.6)$$

Based on the previously received signal $\widehat{\mathbf{d}}_{n-N_p}^{n-1} = (\hat{d}_{n-1}, \hat{d}_{n-2}, \dots, \hat{d}_{n-N_p})^T$, the receiver-sided predicted signal $\hat{d}l_n^+$ can be calculated by a linear prediction with a predictor order of N_p :

$$\hat{d}l_n^+ = \mathbf{h}_n^T \cdot \widehat{\mathbf{d}}_{n-N_p}^{n-1}, \quad (7.7)$$

with $\mathbf{h}_n = (h_n(1), h_n(2), \dots, h_n(N_p))^T$ being the predictor coefficients in a transposed vector manner [Han and Fingscheidt, 2015b].

Consequently, the receiver-sided prediction error PDF is shifted by $\hat{d}l_n^+$. In error-free transmission conditions, the use of the standard ADPCM encoder implies that the correct quantization interval $I^{(i_n)}$ (where the original signal occurs) is known by the decoder. As

a result, the same quantization intervals as in the G.726 Recommendation [ITU-T, 1990] are adopted in (7.5) in the proposed decoder [Han and Fingscheidt, 2015b].

Thereafter, at each time index n , according to the standard quantization intervals and the shifted prediction error PDF $p_{\hat{E}}(\hat{e}_n = \tilde{d}_n - \hat{d}_n^+)$, \hat{d}_n is generated by the centroid condition from (7.5) with $\hat{d}_n = d_n^{(i_n)}$. Initially, \hat{d}_0^+ equals 0 for time index $n = 0$, which implies that the received signal is taken from the standard quantized value $\hat{d}_0 = d_0$.

As depicted in Fig. 7.6, extracting the sign of ds_n from i_n , based on the scaling factor yl_n from the standard ADPCM decoder, the new quantized difference signal $\hat{d}q_n$ can be computed by $|\hat{d}q_n| = 2^{\hat{d}_n + yl_n}$, with $\text{sign}(\hat{d}q_n) = \text{sign}(ds_n)$. As mentioned in the outline, the new reconstructed signal $\hat{s}r_n$ is finally obtained by $\hat{s}r_n = \hat{d}q_n + se_n$, with the estimated signal se_n being the same as in the standard decoder.

Normalized Least-Mean-Squares (NLMS) Algorithm

Because of the nonstationarity of speech, the predictor coefficients \mathbf{h}_n should be adaptively updated to minimize the mean square error $E\{(\hat{e}_n)^2\}$ with $\hat{e}_n = \hat{d}_n - \hat{d}_n^+$. This can be achieved by using the normalized least-mean-squares (NLMS) algorithm [Haykin, 2002]:

$$\mathbf{h}_n = \mathbf{h}_{n-1} + \frac{\hat{e}_{n-1}}{1 + \xi \cdot \|\widehat{\mathbf{d}}_{n-N_p}^{n-1}\|^2} \cdot \widehat{\mathbf{d}}_{n-N_p}^{n-1}, \quad (7.8)$$

with $\|\widehat{\mathbf{d}}_{n-N_p}^{n-1}\|$ being the Euclidean norm. Herein, instead of utilizing both step-size and regularization parameters [Haykin, 2002], only one tuning parameter ξ is adopted [Schuller et al., 2002, Pflug and Fingscheidt, 2011a]. In addition, for time index $n = 0$, the initialization of $\widehat{\mathbf{d}}\mathbf{l}$ and \mathbf{h} are $\widehat{\mathbf{d}}\mathbf{l}_{-N_p}^{-1} = (0, \dots, 0)^\top$ and $\mathbf{h}_{-1} = (1/N_p, \dots, 1/N_p)^\top$ [Han and Fingscheidt, 2015b].

7.2.2 Hard-Decision Decoding

For the transmission over erroneous channel conditions, replacing i_n by the received quantization index \hat{i}_n in Fig. 7.6 and (7.5), (7.6), the improved scalar HD decoding approach can be applied to ADPCM.

7.3 Simulation Setup and Results

In the following, the speech database and the instrumental measurements are introduced in Section 7.3.1. Section 7.3.2 shows the simulation results for G.726 and G.722 ADPCM using both standard and proposed improved HD decoding approaches.

7.3.1 Simulation Setup

Similar to Adaptive Multi-Rate Narrowband (AMR-NB) and AMR Wideband (AMR-WB) in Section 3.4.1, the NTT monaural speech database [NTT, 1994] with a sampling rate of 8 kHz and 16 kHz is used for G.726 and G.722 ADPCM, respectively. All signals are normalized to -26 dBov. The MOS of PESQ [ITU-T, 2001], WB PESQ [ITU-T, 2007], and perceptual objective listening quality assessment (POLQA) [ITU-T, 2011] is used as an instrumental measurement for the performance.

The core of the proposed ADPCM decoder is to generate a time-variant quantization codebook with a shifted prediction error PDF. Therefore, it is important to identify the optimal prediction error PDF $p_{\hat{E}}(\hat{e}_n = \tilde{d}l_n - \hat{d}l_n^+)$. For the development process, a number of 96 English speech files spoken by 4 female and 4 male American native speakers, each with a length of 8 s is exclusively used. A number of 20 different languages excluding American English, each language including 96 speech files with 4 female and 4 male speakers, is used for test. Varying E_b/N_0 from 0 dB to 10 dB, the additive white Gaussian noise (AWGN) channel with binary phase-shift keying (BPSK) modulation is adopted for the simulations of HD decoding, with a number of 96 English speech files spoken by British native speakers.

(1) G.726 ADPCM With 16 kbps

For G.726 ADPCM with 16 kbps, Laplacian and Gaussian PDFs are used as the prediction error PDF in two separate experiments. The Laplacian PDF is defined as

$$p_{\hat{E}}(\hat{e}_n = \tilde{d}l_n - \hat{d}l_n^+) = \frac{1}{\sqrt{2}\hat{\sigma}_{\hat{e}}} \exp\left(-\frac{\sqrt{2}|\tilde{d}l_n - \hat{d}l_n^+ - \hat{\mu}_{\hat{e}}|}{\hat{\sigma}_{\hat{e}}}\right). \quad (7.9)$$

In each experiment, varying the receiver-sided prediction error variance $\hat{\sigma}_{\hat{e}}^2$, a full numerical search is performed in advance. Moreover, it is found that for a searched optimal $\hat{\sigma}_{\hat{e}}$, varying the mean of the prediction error $\hat{\mu}_{\hat{e}}$, the decoder performance can be slightly improved further. Therefore, a two-dimensional full numerical search over the number range $0.5 < \hat{\sigma}_{\hat{e}} < 1.1$ in steps of 0.1, and in the range $-0.3 < \hat{\mu}_{\hat{e}} < 0.1$ in steps of 0.01 is performed. The optimal values $\hat{\sigma}_{\text{opt}}$ and $\hat{\mu}_{\text{opt}}$ are identified by the maximum MOS score of PESQ and made known to the decoder. $N_p = 10$ and $\xi = 10000$ are employed in the NLMS algorithm. It is found that using a Gaussian PDF as the shifted prediction error PDF can achieve a better improvement:

$$p_{\hat{E}}(\hat{e}_n = \tilde{d}l_n - \hat{d}l_n^+) = \frac{1}{\sqrt{2\pi}\hat{\sigma}_{\hat{e}}} \exp\left(-\frac{(\tilde{d}l_n - \hat{d}l_n^+ - \hat{\mu}_{\hat{e}})^2}{2\hat{\sigma}_{\hat{e}}^2}\right), \quad (7.10)$$

with $\hat{\sigma}_{\text{opt}} = 0.9$ and $\hat{\mu}_{\text{opt}} = -0.23$.

(2) G.726 ADPCM With 32 kbps

For the operation at 32 kbps, it is found that the optimal PDF is a combined version of Laplacian and Gaussian PDF, with the mean value $\hat{\mu}_{\hat{e}}$ being in the center, and half of

a Laplacian and a Gaussian PDF being on the left and right side, respectively (i.e., for $\hat{e}_n \leq \hat{\mu}_e$, the Laplacian PDF is defined as in (7.9), while the definition of the Gaussian PDF from (7.10) is valid for $\hat{e}_n > \hat{\mu}_e$). Using $N_p = 10$, $\xi = 80000$ in the NLMS algorithm, after a full numerical search, the optimal values $\hat{\mu}_{\text{opt}} = -0.5$, $\hat{\sigma}_{e,\text{Gaussian}} = 0.8$, $\hat{\sigma}_{e,\text{Laplacian}} = 1.7$ are found.

(3) *G.722 ADPCM*

The optimization process of G.722 ADPCM is performed in analogy to G.726 ADPCM. Employing $N_p = 10$, $\xi = 2$ in the NLMS algorithm, a Gaussian PDF with $\hat{\mu}_{\text{opt}} = -830$, $\hat{\sigma}_{\text{opt}} = 120$ is utilized for the shifted prediction error PDF.

In addition, in order to know how well the NLMS-based prediction worked, the prediction gain is measured by

$$G_p = 10 \cdot \log_{10} \left(\frac{E\{\hat{d}_n^2\}}{E\{\hat{e}_n^2\}} \right). \quad (7.11)$$

Note that for each language, $\frac{E\{\hat{d}_n^2\}}{E\{\hat{e}_n^2\}}$ is calculated considering all samples in each speech file. At the end, the mean of 96 values of $\frac{E\{\hat{d}_n^2\}}{E\{\hat{e}_n^2\}}$ in the linear domain is transformed to the dB domain and used as G_p .

7.3.2 Simulation Results

G.726

(1) *Error-Free Transmission Conditions*

(1a) *32 kbps*

The simulation results of the standard and proposed G.726 ADPCM operating at 32 kbps for 20 languages are shown in Tab. 7.1 (error-free transmission conditions). Using the proposed decoder, the MOS points basically remain the same, with a tiny increase from 0.002 to 0.009 for PESQ and from 0.006 to 0.021 for POLQA. In average, the G.726 ADPCM performance has been improved by 0.004 and 0.014 MOS points of PESQ and POLQA, respectively. These slight improvements are because of the small correlation coefficients as observed in Section 7.2. Moreover, as mentioned in Chapter 6, the improved scalar quantization approach is especially valid for a low quantization rate. Therefore, it is necessary to observe the performance for the G.726 ADPCM operating at 16 kbps.

(1b) *16 kbps*

In contrast to G.726 ADPCM operating at 32 kbps, a clear improvement can be observed in Tab. 7.2 with the 16 kbps mode. The MOS gain between the proposed and

Test language	PESQ MOS			POLQA MOS		
	Standard decoder	Proposed decoder	Improvement	Standard decoder	Proposed decoder	Improvement
Arabic	3.875	3.878	0.003	3.952	3.965	0.013
Chinese	3.926	3.935	0.009	4.001	4.022	0.021
Dutch	4.001	4.008	0.007	4.052	4.065	0.013
English (British)	4.004	4.011	0.007	4.059	4.078	0.019
Finnish	4.002	4.005	0.003	4.031	4.042	0.011
French	3.914	3.921	0.007	3.993	4.006	0.013
German	3.971	3.975	0.004	4.009	4.019	0.010
Greek	3.987	3.989	0.002	4.088	4.103	0.015
Hindi	3.935	3.939	0.004	4.026	4.040	0.014
Hungarian	3.980	3.984	0.004	3.981	3.997	0.016
Indonesian	3.955	3.960	0.005	4.032	4.044	0.012
Italian	4.016	4.021	0.005	4.111	4.128	0.017
Japanese	3.924	3.927	0.003	3.974	3.985	0.011
Korean	3.947	3.951	0.004	4.011	4.024	0.013
Polish	3.919	3.926	0.007	3.941	3.962	0.021
Portuguese	4.028	4.031	0.003	4.044	4.059	0.015
Russian	3.904	3.910	0.006	3.931	3.947	0.016
Spanish	4.027	4.031	0.004	4.080	4.098	0.018
Swedish	3.975	3.974	-0.001	3.932	3.938	0.006
Thai	3.981	3.983	0.002	4.045	4.057	0.012
Average	3.964	3.968	0.004	4.015	4.029	0.014

Table 7.1: MOS results of using the standard and proposed G.726 ADPCM decoders with 32 kbps mode in error-free transmission conditions.

Test language	PESQ MOS			POLQA MOS		
	Standard decoder	Proposed decoder	Improvement	Standard decoder	Proposed decoder	Improvement
Arabic	2.482	2.623	0.141	2.391	2.414	0.023
Chinese	2.441	2.614	0.173	2.374	2.420	0.046
Dutch	2.627	2.801	0.174	2.423	2.491	0.068
English (British)	2.507	2.673	0.166	2.355	2.401	0.046
Finnish	2.731	2.922	0.191	2.424	2.500	0.076
French	2.463	2.618	0.155	2.381	2.411	0.030
German	2.480	2.614	0.134	2.485	2.514	0.029
Greek	2.563	2.720	0.157	2.467	2.511	0.044
Hindi	2.526	2.667	0.141	2.437	2.472	0.035
Hungarian	2.684	2.843	0.159	2.455	2.505	0.050
Indonesian	2.534	2.676	0.142	2.403	2.449	0.046
Italian	2.499	2.607	0.108	2.476	2.509	0.033
Japanese	2.509	2.648	0.139	2.428	2.464	0.036
Korean	2.550	2.688	0.138	2.500	2.537	0.037
Polish	2.566	2.755	0.189	2.361	2.404	0.043
Portuguese	2.563	2.691	0.128	2.393	2.440	0.047
Russian	2.495	2.669	0.174	2.388	2.422	0.034
Spanish	2.583	2.727	0.144	2.430	2.460	0.030
Swedish	2.789	2.966	0.177	2.448	2.544	0.096
Thai	2.521	2.644	0.123	2.382	2.435	0.053
Average	2.556	2.708	0.153	2.420	2.465	0.045

Table 7.2: MOS results of using the standard and proposed G.726 ADPCM decoders with 16 kbps mode in error-free transmission conditions.

E_b/N_0 (dB)	PESQ MOS			POLQA MOS		
	Standard decoder	Proposed decoder	Improve- ment	Standard decoder	Proposed decoder	Improve- ment
0	1.233	1.232	-0.001	1.033	1.032	-0.001
1	1.247	1.246	-0.001	1.085	1.087	0.002
2	1.306	1.306	0	1.203	1.202	-0.001
3	1.420	1.421	0.001	1.370	1.367	-0.003
4	1.631	1.634	0.003	1.606	1.603	-0.003
5	1.925	1.928	0.003	1.897	1.896	-0.001
6	2.319	2.321	0.002	2.240	2.250	0.010
7	2.861	2.860	-0.001	2.682	2.684	0.002
8	3.349	3.348	-0.001	3.168	3.168	0
9	3.824	3.828	0.004	3.789	3.801	0.012
10	3.989	3.995	0.006	4.031	4.049	0.018

(a) 32 kbps mode

E_b/N_0 (dB)	PESQ MOS			POLQA MOS		
	Standard decoder	Proposed decoder	Improve- ment	Standard decoder	Proposed decoder	Improve- ment
0	1.276	1.280	0.004	1.078	1.082	0.004
1	1.339	1.344	0.005	1.167	1.177	0.010
2	1.443	1.454	0.011	1.307	1.320	0.013
3	1.602	1.626	0.024	1.477	1.494	0.017
4	1.804	1.842	0.038	1.693	1.707	0.014
5	2.006	2.065	0.059	1.896	1.914	0.018
6	2.210	2.290	0.080	2.094	2.109	0.015
7	2.384	2.492	0.108	2.240	2.265	0.025
8	2.466	2.604	0.138	2.315	2.353	0.038
9	2.500	2.659	0.159	2.349	2.394	0.045
10	2.507	2.672	0.165	2.354	2.400	0.046

(b) 16 kbps mode

Table 7.3: MOS results of the standard and proposed G.726 ADPCM transmission over an AWGN channel with HD decoding.

Test language	G_p (dB)		
	G.726 16 kbps	G.726 32 kbps	G.722
Arabic	22.32	3.35	3.28
Chinese	22.80	2.93	3.26
Dutch	22.74	3.38	3.14
English (British)	21.97	2.83	3.23
Finnish	23.75	4.32	3.06
French	21.82	2.74	3.28
German	20.89	2.51	3.41
Greek	22.67	3.18	3.24
Hindi	23.11	2.84	3.23
Hungarian	21.68	3.39	3.21
Indonesian	22.87	3.03	3.22
Italian	19.71	1.89	3.46
Japanese	23.37	2.96	3.30
Korean	21.93	2.89	3.33
Polish	24.65	3.46	3.13
Portuguese	21.62	2.57	3.30
Russian	23.34	3.36	3.16
Spanish	22.26	2.69	3.25
Swedish	27.22	4.17	3.01
Thai	22.33	3.29	3.28
Average	22.65	3.09	3.24

Table 7.4: Prediction gains (in dB) using the NLMS algorithm to predict the quantizer output signals in the G.726 and G.722 ADPCM decoders.

the standard decoding approach ranges from 0.108 to 0.191 points of WB PESQ, while from 0.023 to 0.096 points of POLQA. Correspondingly, the average MOS gain of 20 languages reaches 0.153 MOS points of WB PESQ and 0.045 MOS points of POLQA, respectively. Informal listening tests show that the difference between the standard and proposed approach is distinguishable for G.726 ADPCM operating at 16 kbps, but not for the operation at 32 kbps.

(2) Error-Prone Transmission Conditions

The MOS results for the G.726 ADPCM transmission over an AWGN channel are shown in Tab. 7.3, with the operation at 32 kbps and 16 kbps shown in Tab. 7.3(a) and Tab. 7.3(b), respectively. Similar to the error-free transmission conditions, the MOS gain between the proposed and the standard decoder is very tiny (up to 0.018 POLQA MOS) in Tab. 7.3(a). For the 16 kbps mode in Tab. 7.3(b), generally the MOS improvement increases along with the rise of E_b/N_0 , with up to 0.165 PESQ MOS gain. This implies that especially good channel conditions take profit from the proposed decoder.

In addition, as the results of the prediction gain (7.11) depicted in Tab. 7.4, even

**7. Improving G.726 and G.722 ADPCM by Hard-Decision Decoding With
128 Time-Variant Codebooks**

Test language	WB PESQ MOS				POLQA MOS		
	Standard decoder	Proposed decoder	Improvement	Ideal	Standard decoder	Proposed decoder	Ideal
Arabic	3.747	3.794	0.047	3.980	4.662	4.656	4.732
Chinese	3.925	3.967	0.042	4.127	4.715	4.712	4.736
Dutch	3.856	3.911	0.055	4.053	4.711	4.712	4.743
English (British)	3.919	3.961	0.042	4.130	4.733	4.737	4.747
Finnish	3.709	3.781	0.072	3.975	4.711	4.710	4.736
French	3.883	3.931	0.048	4.092	4.738	4.738	4.750
German	3.937	3.968	0.031	4.149	4.664	4.661	4.697
Greek	3.840	3.896	0.056	4.043	4.711	4.710	4.739
Hindi	3.937	3.977	0.040	4.156	4.693	4.689	4.746
Hungarian	3.834	3.887	0.053	4.085	4.707	4.705	4.746
Indonesian	3.844	3.890	0.046	4.051	4.705	4.707	4.741
Italian	4.071	4.091	0.020	4.271	4.681	4.676	4.738
Japanese	3.958	3.994	0.036	4.164	4.690	4.686	4.731
Korean	3.893	3.932	0.039	4.123	4.673	4.665	4.727
Polish	3.809	3.864	0.055	4.021	4.668	4.664	4.715
Portuguese	4.039	4.073	0.034	4.243	4.714	4.714	4.740
Russian	3.765	3.826	0.061	3.999	4.674	4.668	4.732
Spanish	3.971	4.018	0.047	4.170	4.675	4.673	4.728
Swedish	3.603	3.667	0.064	3.878	4.681	4.686	4.741
Thai	3.904	3.953	0.049	4.116	4.721	4.719	4.749
Average	3.872	3.919	0.047	4.091	4.696	4.694	4.736

Table 7.5: MOS results of using the standard and proposed G.722 ADPCM decoders in error-free transmission conditions.

though an ARMA predictor is already employed in ADPCM, more than 20 dB prediction gain is still observed for ADPCM operating at 16 kbps. Therefore, it can be stated that the 10th-order NLMS-based predictor in the proposed system works reasonably well. As expected, the prediction gain for the 32 kbps mode is significantly lower.

Note that the American English database is used for the optimization of the prediction error PDF, but the optimized values are also applicable to other languages, which implies that the optimization is independent of language.

G.722

(1) Error-Free Transmission Conditions

The simulation results of the G.722 ADPCM for error-free transmission conditions are depicted in Tab.7.5. Since the improved scalar quantization approach has only been employed in the higher subband, the lower subband remains using the standard quantization approach, the speech quality has been slightly improved by 0.047 MOS of WB

E_b/N_0 (dB)	WB PESQ MOS			POLQA MOS		
	Standard decoder	Proposed decoder	Improve- ment	Standard decoder	Proposed decoder	Improve- ment
0	1.429	1.440	0.011	1.107	1.107	0
1	1.527	1.540	0.013	1.228	1.228	0
2	1.619	1.638	0.019	1.347	1.347	0
3	1.736	1.751	0.015	1.494	1.493	-0.001
4	2.015	2.028	0.013	1.825	1.824	-0.001
5	2.375	2.391	0.016	2.435	2.435	0
6	2.770	2.790	0.020	3.242	3.249	0.007
7	3.218	3.245	0.027	4.113	4.116	0.003
8	3.616	3.651	0.035	4.571	4.576	0.005
9	3.870	3.910	0.040	4.712	4.717	0.005
10	3.916	3.958	0.042	4.733	4.737	0.004

(a) Distorted higher subband

E_b/N_0 (dB)	WB PESQ MOS			POLQA MOS		
	Standard decoder	Proposed decoder	Improve- ment	Standard decoder	Proposed decoder	Improve- ment
0	1.076	1.076	0	1.178	1.180	0.002
1	1.101	1.101	0	1.211	1.211	0
2	1.149	1.148	-0.001	1.193	1.192	-0.001
3	1.252	1.252	0	1.213	1.212	-0.001
4	1.407	1.406	-0.001	1.343	1.341	-0.002
5	1.711	1.707	-0.004	1.712	1.711	-0.001
6	2.110	2.105	-0.005	2.429	2.432	0.003
7	2.630	2.632	0.002	3.380	3.386	0.006
8	3.257	3.275	0.018	4.265	4.268	0.003
9	3.710	3.745	0.035	4.640	4.644	0.004
10	3.892	3.934	0.042	4.706	4.710	0.004

(b) Distorted both subbands

Table 7.6: MOS results of the standard and proposed G.722 ADPCM transmission over an AWGN channel with HD decoding.

PESQ and even decreased by 0.002 MOS of POLQA on average in 20 languages. The column depicted as “ideal” implies that the quantized difference signals dq_n in Fig. 7.5 are the unquantized values directly taken from the transmitter side. It shows the maximum potential for a MOS improvement of 0.219 and 0.04 points of WB PESQ and POLQA, respectively. Note that only the dq_n in the higher subband are used as the values from the transmitter side, the dq_n in the lower subband remains in a standard ADPCM decoding fashion.

Moreover, a prediction gain of only 3.24 dB for G.722 ADPCM decoder is observed in Tab. 7.4 (a similarly poor values as in 32 kbps G.726 ADPCM).

(2) Error-Prone Transmission Conditions

The simulation results of the G.722 ADPCM transmission over an AWGN channel is shown in Tab. 7.6. In Tab. 7.6(a), only the bits of the higher subband are distorted, while the bits of both lower and higher subbands are corrupted in Tab. 7.6(b). Due to the effect of the lower subband, the speech quality has only been improved for $E_b/N_0 \geq 7$ dB with WB PESQ in Tab. 7.6(b). Distorting the bits of only the higher subband, the MOS results of WB PESQ show an improved performance for all the channel conditions. The MOS improvement of WB PESQ again increases along with the increase of E_b/N_0 in general, with up to 0.042 WB PESQ MOS gain.

To sum up, using the improved scalar quantization approach generating a time-variant quantization codebook, the G.726 and G.722 ADPCM performance can insignificantly be improved in error-free and error-prone transmission conditions. Some improvement, however, is observed with application to G.726 ADPCM in the 16 kbps mode in good channel conditions.

7.4 Summary

In Adaptive Differential Pulse Code Modulation (ADPCM), although the difference signal is quantized, some residual correlation of the quantizer inputs signals is still observed. In this chapter, after introducing the standard G.726 and G.722 ADPCM encoders and decoders, the new hard-decision (HD) decoding approach proposed in Chapter 6 is applied to G.726 and G.722 ADPCM, both for error-free and error-prone transmission conditions. In error-free transmission conditions, using the proposed ADPCM decoder, the simulation results show an average of 0.153 mean opinion score (MOS) of perceptual evaluation of speech quality (PESQ) improvement for G.726 operating at 16 kbps, while only 0.014 MOS of POLQA improvement for 32 kbps. About 0.047 MOS gain of WB PESQ is observed for G.722 ADPCM. Moreover, for error-prone transmission conditions, the simulations over an additive white Gaussian noise (AWGN) channel show an increase of up to 0.165, 0.006, and 0.042 MOS gain for G.726 16 kbps, 32 kbps, and G.722, respectively. It can be summarized that only the G.726 ADPCM in the 16 kbps mode takes some profit from the proposed ADPCM decoder, especially for good and error-free transmission conditions.

Chapter 8

Conclusions

In this thesis, the decoding performance of speech and audio codecs has been improved, in a fashion compliant with the corresponding standard encoders. The contributions in this thesis mainly include three parts:

- (1) Robust AMR-NB and AMR-WB speech decoders have been built for erroneous transmission conditions, by employing the FL/SD decoding algorithm.
- (2) Utilizing both FL/SD and VL/SD decoding methods, an enhanced HE-AAC decoder has been proposed for adverse transmission conditions.
- (3) A decoding approach is proposed to improve the scalar quantization performance for correlated processes, both for error-free and error-prone transmission conditions, both with HD and SD decoding. This proposed approach has been applied to the G.726 and G.722 ADPCM decoders aiming to improve the speech quality for both error-free and error-prone transmission conditions, with a focus on HD decoding.

The detailed results are listed as follows.

The FL/SD decoding approach has been applied to the AMR-NB and AMR-WB speech codecs. After calculating the APPs for each relevant codec parameter, the corresponding parameter can be estimated. In adverse transmission conditions, compared to the traditional HD decoding methods or the standard error concealment techniques in AMR-NB and AMR-WB, the speech quality has been significantly improved by using the FL/SD decoding algorithm. Moreover, for the LSF parameters in AMR-NB, based on the approach only using interframe redundancy, two approaches utilizing both interframe and intraframe redundancy have been proposed. These two approaches are distinguished by either exploiting the redundancy of only adjacent LSF submatrices or of both adjacent and non-adjacent LSF submatrices. Similarly, the approach exploiting both interframe and intraframe redundancy of ISF subvectors is proposed and adopted in AMR-WB. The performance of both AMR-NB and AMR-WB have been further improved. The proposed scheme is applicable in a standard-compliant fashion to replace the non-mandatory GSM/UMTS/LTE error concealment of the respective speech decoders.

In the VL/SD decoding scheme, based on the trellis representation considering all the possible states at each time index, the APPs including forward and backward recursions

can be calculated. As in the FL/SD decoding approach, the parameter can be estimated based on the APPs. The definitions of the stage boundaries and state intervals in the trellis representation are newly explained in detail in this thesis. The links between the FL/SD and VL/SD decoding algorithms have been elaborated, which show that the VL/SD decoding applied to FLCs in fact turns out to be the FL/SD decoding approach. In addition, the performance tradeoffs between FL/SD and VL/SD decoding schemes for different block lengths, quantization bit rates, and source correlation have been discussed. Two FL/SD decoding approaches with either only forward recursion or both forward and backward recursions are adopted. On the one hand, the VL/SD decoding approach exceeds the FL/SD decoding for uncorrelated sources and short block lengths with medium to good channel qualities, especially for a lower quantization bit rate. On the other hand, in general the FL/SD decoding scheme is a better choice than the VL/SD scheme for correlated sources.

For the purpose of obtaining a robust HE-AAC audio codec for adverse transmission conditions, the FL/SD decoding approach has been further applied to the parameter global gain in the HE-AAC audio decoder, and the VL/SD decoding method has been applied to the scale factors and quantized spectral coefficients. Since additional sign bits are required for the quantized spectral coefficients with unsigned Huffman codebooks, a new trellis representation considering the sign bits has been proposed. A corresponding codebook is also set up for the sign bits. With the support of subjective listening tests, a clearly enhanced audio quality has been observed employing SD decoding approaches, especially when only the global gain and scale factors are distorted. Compared to the standard error concealment in HE-AAC, no extra delay is introduced. The proposed decoder can be applied in mobile music streaming or digital audio broadcasting.

In order to improve the scalar quantization performance for correlated source processes, a new decoding approach with the standard encoder and employing a predictor only at the receiver side is proposed in this thesis. In the receiver, the parameter is predicted according to the past received information. Thereafter, based on the shifted receiver-sided prediction error PDF, a new time-variant codebook with time-variant codebook entries can be generated. In both error-free and error-prone transmission conditions, in comparison to the standard quantization approach, an improvement is observed using the proposed approach, especially for low quantization bit rates and highly correlated source processes. This approach has been advantageously applied to both HD and SD decoding schemes. In addition, an analytical solution of the prediction error variance is also proposed, which provides a reasonable performance, compared to the full numerical search for the optimal prediction error variance. Finally, the improved scalar quantization approach has been applied to the G.726 and G.722 ADPCM decoders, both for error-free and erroneous transmission conditions with a focus on HD decoding in this work. An enhanced speech quality is observed by using the proposed decoder included in 16 kbps G.726 ADPCM in error-free and good channel conditions. The proposed approach is applicable in a standard-compatible fashion.

Appendix A

APP Calculation of Exploiting Both Interframe and Intraframe Redundancy in AMR-NB

The two approaches ((2a),(2b)) exploiting both interframe and intraframe redundancy of LSF submatrices in Section 3.3.1 is derived in this appendix (see also [Tian, 2015]).

(2a) *Approach 1: Exploiting Interframe Redundancy and Intraframe Redundancy of Only Adjacent LSF Submatrices*

The APPs of a possibly transmitted bit combination $\mathbf{x}_{\ell,\eta}^{(i)}$ given the past received bit combinations from frame 1 to ℓ of three LSF submatrices (the current calculated η and two adjacent $\eta - 1, \eta + 1$) satisfy

$$\begin{aligned} & P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell) \\ &= \sum_{f=0}^{2^{M_\eta-1}} \sum_{j=0}^{2^{M_{\eta-1}-1}} \sum_{h=0}^{2^{M_{\eta+1}-1}} P(\mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta+1}^{(h)} | \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell). \end{aligned} \quad (\text{A.1})$$

Herein, f, j, h denote the M_η , $M_{\eta-1}$, and $M_{\eta+1}$ -bit quantization codebook indices at LSF submatrix $\eta, \eta - 1$, and $\eta + 1$, with $M_\eta \in \{7, 8, 9, 8, 6\}$ being assigned for LSF submatrix $\eta \in \{1, 2, 3, 4, 5\}$.

Based on the chain rule [A. Papoulis, 2002], we can write

$$\begin{aligned} & P(\mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta+1}^{(h)} | \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell) \\ &= \frac{P(\mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta+1}^{(h)}, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell)}{P(\hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell)}, \end{aligned} \quad (\text{A.2})$$

with the joint probability $P(\mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta-1}^{(h)}, \hat{\mathbf{x}}_{1,\eta}^{\ell}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-1})$ being computed by

$$\begin{aligned}
 & P(\mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta-1}^{(h)}, \hat{\mathbf{x}}_{1,\eta}^{\ell}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-1}) \\
 &= P(\hat{\mathbf{x}}_{\ell,\eta}^{\ell}, \hat{\mathbf{x}}_{\ell,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{\ell,\eta+1}^{\ell-1} \mid \mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta-1}^{(h)}, \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &\quad \cdot P(\mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta+1}^{(h)} \mid \mathbf{x}_{\ell-1,\eta}^{(f)}, \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &\quad \cdot P(\mathbf{x}_{\ell-1,\eta}^{(f)} \mid \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &\quad \cdot P(\hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}).
 \end{aligned} \tag{A.3}$$

Applying (A.3) and (A.2) to (A.1), the calculation of the APPs turns out to be [Adrat et al., 2000]

$$\begin{aligned}
 & P(\mathbf{x}_{\ell,\eta}^{(i)} \mid \hat{\mathbf{x}}_{1,\eta}^{\ell}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-1}) \\
 &= \frac{1}{C} \cdot \sum_{f=0}^{2^{M_{\eta-1}}} \sum_{j=0}^{2^{M_{\eta-1}-1}} \sum_{h=0}^{2^{M_{\eta+1}-1}} P(\hat{\mathbf{x}}_{\ell,\eta}^{\ell}, \hat{\mathbf{x}}_{\ell,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{\ell,\eta+1}^{\ell-1} \mid \mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \\
 &\quad \mathbf{x}_{\ell,\eta+1}^{(h)}, \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &\quad \cdot P(\mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta+1}^{(h)} \mid \mathbf{x}_{\ell-1,\eta}^{(f)}, \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &\quad \cdot P(\mathbf{x}_{\ell-1,\eta}^{(f)} \mid \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}),
 \end{aligned} \tag{A.4}$$

with the constant $\frac{1}{C} = \frac{P(\hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2})}{P(\hat{\mathbf{x}}_{1,\eta}^{\ell}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-1})}$.

Assuming a memoryless channel and the independence of LSF submatrices neighboring in frame (ℓ) or submatrix index (η), the first two terms in the right-hand-side of (A.4) can be respectively obtained by

$$\begin{aligned}
 & P(\hat{\mathbf{x}}_{\ell,\eta}^{\ell}, \hat{\mathbf{x}}_{\ell,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{\ell,\eta+1}^{\ell-1} \mid \mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta+1}^{(h)}, \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &= P(\hat{\mathbf{x}}_{\ell,\eta}^{\ell} \mid \mathbf{x}_{\ell,\eta}^{(i)}) \cdot P(\hat{\mathbf{x}}_{\ell,\eta-1}^{\ell-1} \mid \mathbf{x}_{\ell,\eta-1}^{(j)}) \cdot P(\hat{\mathbf{x}}_{\ell,\eta+1}^{\ell-1} \mid \mathbf{x}_{\ell,\eta+1}^{(h)}),
 \end{aligned} \tag{A.5}$$

and [Adrat et al., 2000]

$$\begin{aligned}
 & P(\mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell,\eta-1}^{(j)}, \mathbf{x}_{\ell,\eta+1}^{(h)} \mid \mathbf{x}_{\ell-1,\eta}^{(f)}, \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &= P(\mathbf{x}_{\ell,\eta-1}^{(j)} \mid \mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell,\eta+1}^{(h)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &\quad \cdot P(\mathbf{x}_{\ell,\eta+1}^{(h)} \mid \mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell-1,\eta}^{(f)}, \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &\quad \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} \mid \mathbf{x}_{\ell-1,\eta}^{(f)}, \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-2}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-2}) \\
 &\approx P(\mathbf{x}_{\ell,\eta-1}^{(j)} \mid \mathbf{x}_{\ell,\eta}^{(i)}) \cdot P(\mathbf{x}_{\ell,\eta+1}^{(h)} \mid \mathbf{x}_{\ell,\eta}^{(i)}) \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} \mid \mathbf{x}_{\ell-1,\eta}^{(f)}).
 \end{aligned} \tag{A.6}$$

As a result, applying (A.5) and (A.6) to (A.4), the APPs can finally be calculated by

$$\begin{aligned}
P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell) &= \frac{1}{C} \cdot P(\hat{\mathbf{x}}_{\ell,\eta} | \mathbf{x}_{\ell,\eta}^{(i)}) \\
&\cdot \sum_{f=0}^{2^{M_\eta-1}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell-1,\eta}^{(f)}) \cdot P(\mathbf{x}_{\ell-1,\eta}^{(f)} | \hat{\mathbf{x}}_{1,\eta}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta-1}^{\ell-1}, \hat{\mathbf{x}}_{1,\eta+1}^{\ell-1}) \\
&\cdot \sum_{j=0}^{2^{M_{\eta-1}-1}} P(\hat{\mathbf{x}}_{\ell,\eta-1} | \mathbf{x}_{\ell,\eta-1}^{(j)}) \cdot P(\mathbf{x}_{\ell,\eta-1}^{(j)} | \mathbf{x}_{\ell,\eta}^{(i)}) \\
&\cdot \sum_{h=0}^{2^{M_{\eta+1}-1}} P(\hat{\mathbf{x}}_{\ell,\eta+1} | \mathbf{x}_{\ell,\eta+1}^{(h)}) \cdot P(\mathbf{x}_{\ell,\eta+1}^{(h)} | \mathbf{x}_{\ell,\eta}^{(i)}),
\end{aligned} \tag{A.7}$$

(2b) *Approach 2: Exploiting Interframe Redundancy and Intraframe Redundancy of Both Adjacent and Non-Adjacent LSF Submatrices*

According to the chain rule [A. Papoulis, 2002], the APPs representing a possible transmitted bit combination $\mathbf{x}_{\ell,\eta}^{(i)}$ given the past received bit combinations from frame 1 to ℓ of all the five LSF submatrices can be calculated by

$$P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell) = \frac{P(\mathbf{x}_{\ell,\eta}^{(i)}, \hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell)}{P(\hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell)}. \tag{A.8}$$

Again using the chain rule, the term $P(\mathbf{x}_{\ell,\eta}^{(i)}, \hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell)$ in (A.8) can be expressed as

$$\begin{aligned}
P(\mathbf{x}_{\ell,\eta}^{(i)}, \hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell) &= P(\hat{\mathbf{x}}_{1,\eta-2}^\ell | \mathbf{x}_{\ell,\eta}^{(i)}, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell) \\
&\cdot P(\hat{\mathbf{x}}_{1,\eta-1}^\ell | \mathbf{x}_{\ell,\eta}^{(i)}, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell) \\
&\cdot P(\hat{\mathbf{x}}_{1,\eta}^\ell | \mathbf{x}_{\ell,\eta}^{(i)}, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell) \\
&\cdot P(\hat{\mathbf{x}}_{1,\eta+1}^\ell | \mathbf{x}_{\ell,\eta}^{(i)}, \hat{\mathbf{x}}_{1,\eta}^\ell) \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell) \cdot P(\hat{\mathbf{x}}_{1,\eta}^\ell).
\end{aligned} \tag{A.9}$$

Considering the independence of $\hat{\mathbf{x}}_{1,1}^\ell, \hat{\mathbf{x}}_{1,2}^\ell, \dots, \hat{\mathbf{x}}_{1,5}^\ell$ (i.e., $\hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \dots, \hat{\mathbf{x}}_{1,\eta+2}^\ell$), and applying the Bayes rule, (A.9) turns out to be

$$\begin{aligned}
&P(\mathbf{x}_{\ell,\eta}^{(i)}, \hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell) \\
&= P(\hat{\mathbf{x}}_{1,\eta-2}^\ell | \mathbf{x}_{\ell,\eta}^{(i)}) \cdot P(\hat{\mathbf{x}}_{1,\eta-1}^\ell | \mathbf{x}_{\ell,\eta}^{(i)}) \cdot P(\hat{\mathbf{x}}_{1,\eta+1}^\ell | \mathbf{x}_{\ell,\eta}^{(i)}) \\
&\cdot P(\hat{\mathbf{x}}_{1,\eta+2}^\ell | \mathbf{x}_{\ell,\eta}^{(i)}) \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell) \cdot P(\hat{\mathbf{x}}_{1,\eta}^\ell) \\
&= \frac{P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell) \cdot P(\hat{\mathbf{x}}_{1,\eta-2}^\ell)}{P(\mathbf{x}_{\ell,\eta}^{(i)})} \cdot \frac{P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-1}^\ell) \cdot P(\hat{\mathbf{x}}_{1,\eta-1}^\ell)}{P(\mathbf{x}_{\ell,\eta}^{(i)})} \\
&\cdot \frac{P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta+1}^\ell) \cdot P(\hat{\mathbf{x}}_{1,\eta+1}^\ell)}{P(\mathbf{x}_{\ell,\eta}^{(i)})} \cdot \frac{P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta+2}^\ell) \cdot P(\hat{\mathbf{x}}_{1,\eta+2}^\ell)}{P(\mathbf{x}_{\ell,\eta}^{(i)})} \\
&\cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell) \cdot P(\hat{\mathbf{x}}_{1,\eta}^\ell).
\end{aligned} \tag{A.10}$$

Applying (A.10) to (A.8), the calculation of the APPs can be written as

$$\begin{aligned} P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell) &= \frac{1}{P(\mathbf{x}_{\ell,\eta}^{(i)})^4} \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell) \\ &\cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell) \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-1}^\ell) \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta+1}^\ell) \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta+2}^\ell). \end{aligned} \quad (\text{A.11})$$

According to the chain rule, the term $P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell)$ in (A.11) is actually

$$\begin{aligned} P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell) &= \sum_{j=0}^{2^{M_{\eta-2}}} P(\mathbf{x}_{\ell,\eta}^{(i)}, \mathbf{x}_{\ell,\eta-2}^{(j)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell) \\ &= \sum_{j=0}^{2^{M_{\eta-2}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta-2}^{(j)}, \hat{\mathbf{x}}_{1,\eta-2}^\ell) \cdot P(\mathbf{x}_{\ell,\eta-2}^{(j)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell). \end{aligned} \quad (\text{A.12})$$

Note that if $\mathbf{x}_{\ell,\eta-2}^{(j)}$ is given, $\hat{\mathbf{x}}_{1,\eta-2}^\ell$ and $\mathbf{x}_{\ell,\eta}^{(i)}$ are conditionally independent, therefore (A.12) can be further written as

$$P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell) = \sum_{j=0}^{2^{M_{\eta-2}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta-2}^{(j)}) \cdot P(\mathbf{x}_{\ell,\eta-2}^{(j)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell). \quad (\text{A.13})$$

Similarly, replacing $\hat{\mathbf{x}}_{1,\eta-2}^\ell$ by $\hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell$ and applying (A.13) to the corresponding terms in (A.11), the APP calculation can finally be written as

$$\begin{aligned} P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell, \hat{\mathbf{x}}_{1,\eta-1}^\ell, \hat{\mathbf{x}}_{1,\eta}^\ell, \hat{\mathbf{x}}_{1,\eta+1}^\ell, \hat{\mathbf{x}}_{1,\eta+2}^\ell) &= \frac{1}{P(\mathbf{x}_{\ell,\eta}^{(i)})^4} \cdot P(\mathbf{x}_{\ell,\eta}^{(i)} | \hat{\mathbf{x}}_{1,\eta}^\ell) \\ &\cdot \sum_{j=0}^{2^{M_{\eta-2}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta-2}^{(j)}) \cdot P(\mathbf{x}_{\ell,\eta-2}^{(j)} | \hat{\mathbf{x}}_{1,\eta-2}^\ell) \\ &\cdot \sum_{f=0}^{2^{M_{\eta-1}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta-1}^{(f)}) \cdot P(\mathbf{x}_{\ell,\eta-1}^{(f)} | \hat{\mathbf{x}}_{1,\eta-1}^\ell) \\ &\cdot \sum_{h=0}^{2^{M_{\eta+1}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta+1}^{(h)}) \cdot P(\mathbf{x}_{\ell,\eta+1}^{(h)} | \hat{\mathbf{x}}_{1,\eta+1}^\ell) \\ &\cdot \sum_{r=0}^{2^{M_{\eta+2}}} P(\mathbf{x}_{\ell,\eta}^{(i)} | \mathbf{x}_{\ell,\eta+2}^{(r)}) \cdot P(\mathbf{x}_{\ell,\eta+2}^{(r)} | \hat{\mathbf{x}}_{1,\eta+2}^\ell). \end{aligned} \quad (\text{A.14})$$

Appendix B

Time-Variant Quantization Codebook

The derivation of the proposed time-variant quantization codebook in Chapter 6 is shown in this Appendix. Section B.1 describes the derivation of the new reconstruction level calculation (6.8) in Section 6.2. The derivation of the receiver-sided prediction error variance $\hat{\sigma}_\varepsilon^2$ (6.14) in Section 6.3 is presented in Section B.2.

B.1 New Reconstruction Levels

As mentioned in Section 6.2, applying (6.7) to (6.6), the new reconstruction level can be calculated as

$$u_n^{(i)} = \frac{\int_{d_i}^{d_{i+1}} \tilde{u}_n \cdot \frac{1}{\sqrt{2\pi}\hat{\sigma}_\varepsilon} \exp\left(-\frac{(\tilde{u}_n - \hat{u}_n^+)^2}{2\hat{\sigma}_\varepsilon^2}\right) d\tilde{u}_n}{\int_{d_i}^{d_{i+1}} \frac{1}{\sqrt{2\pi}\hat{\sigma}_\varepsilon} \exp\left(-\frac{(\tilde{u}_n - \hat{u}_n^+)^2}{2\hat{\sigma}_\varepsilon^2}\right) d\tilde{u}_n}. \quad (\text{B.1})$$

Due to the fact that $\frac{1}{\sqrt{2\pi}\hat{\sigma}_\varepsilon}$ is a constant, (B.1) becomes

$$\begin{aligned} u_n^{(i)} &= \frac{\int_{d_i}^{d_{i+1}} \tilde{u}_n \cdot \exp\left(-\frac{(\tilde{u}_n - \hat{u}_n^+)^2}{2\hat{\sigma}_\varepsilon^2}\right) d\tilde{u}_n}{\int_{d_i}^{d_{i+1}} \exp\left(-\frac{(\tilde{u}_n - \hat{u}_n^+)^2}{2\hat{\sigma}_\varepsilon^2}\right) d\tilde{u}_n} \\ &= \frac{\int_{d_i}^{d_{i+1}} (\tilde{u}_n - \hat{u}_n^+) \cdot \exp\left(-\frac{(\tilde{u}_n - \hat{u}_n^+)^2}{2\hat{\sigma}_\varepsilon^2}\right) d\tilde{u}_n + \hat{u}_n^+ \cdot \exp\left(-\frac{(\tilde{u}_n - \hat{u}_n^+)^2}{2\hat{\sigma}_\varepsilon^2}\right) d\tilde{u}_n}{\int_{d_i}^{d_{i+1}} \exp\left(-\frac{(\tilde{u}_n - \hat{u}_n^+)^2}{2\hat{\sigma}_\varepsilon^2}\right) d\tilde{u}_n}. \end{aligned} \quad (\text{B.2})$$

Replacing $\tilde{u}_n - \hat{u}_n^+$ by x and considering that \hat{u}_n^+ is a constant, we have $d\tilde{u}_n = dx$. As a consequence, (B.2) turns out to be

$$u_n^{(i)} = \hat{u}_n^+ + \frac{\int_{d_i - \hat{u}_n^+}^{d_{i+1} - \hat{u}_n^+} x \cdot \exp\left(-\frac{x^2}{2\hat{\sigma}_e^2}\right) dx}{\int_{d_i - \hat{u}_n^+}^{d_{i+1} - \hat{u}_n^+} \exp\left(-\frac{x^2}{2\hat{\sigma}_e^2}\right) dx}. \quad (\text{B.3})$$

Defining the numerator of the second summand in (B.3) as A and the denominator as B ($u_n^{(i)} = \hat{u}_n^+ + \frac{A}{B}$), the numerator and denominator will be derived separately.

For the numerator A , assuming $y = -\frac{x^2}{2\hat{\sigma}_e^2}$ leading to $x dx = -\hat{\sigma}_e^2 dy$, the numerator A can be calculated by

$$\begin{aligned} A &= -\hat{\sigma}_e^2 \cdot \int_{\frac{-(d_i - \hat{u}_n^+)^2}{2\hat{\sigma}_e^2}}^{\frac{-(d_{i+1} - \hat{u}_n^+)^2}{2\hat{\sigma}_e^2}} \exp(y) dy \\ &= -\hat{\sigma}_e^2 \cdot \left(\exp\left(\frac{-(d_{i+1} - \hat{u}_n^+)^2}{2\hat{\sigma}_e^2}\right) - \exp\left(\frac{-(d_i - \hat{u}_n^+)^2}{2\hat{\sigma}_e^2}\right) \right). \end{aligned} \quad (\text{B.4})$$

For the denominator, replacing x by $\sqrt{2}\hat{\sigma}_e \cdot z$, which results in $dx = \sqrt{2}\hat{\sigma}_e \cdot dz$, the denominator B therefore becomes

$$B = \sqrt{2}\hat{\sigma}_e \cdot \int_{\frac{d_i - \hat{u}_n^+}{\sqrt{2}\hat{\sigma}_e}}^{\frac{d_{i+1} - \hat{u}_n^+}{\sqrt{2}\hat{\sigma}_e}} \exp(-z^2) dz. \quad (\text{B.5})$$

With the help of the error function $\int_a^b \exp(-z^2) dz = \frac{\sqrt{\pi}}{2} \cdot (\text{erf}(b) - \text{erf}(a))$, (B.5) is easily solved numerically according to

$$B = \sqrt{\frac{\pi}{2}} \hat{\sigma}_e \cdot \left(\text{erf}\left(\frac{(d_{i+1} - \hat{u}_n^+)}{\sqrt{2}\hat{\sigma}_e}\right) - \text{erf}\left(\frac{(d_i - \hat{u}_n^+)}{\sqrt{2}\hat{\sigma}_e}\right) \right). \quad (\text{B.6})$$

Finally, applying (B.4) and (B.6) to (B.3), the new receiver-sided time-variant codebook reconstruction value $u_n^{(i)}$ can be computed by

$$u_n^{(i)} = \hat{u}_n^+ - \sqrt{\frac{2}{\pi}} \hat{\sigma}_e \cdot \frac{\exp\left(\frac{-(d_{i+1} - \hat{u}_n^+)^2}{2\hat{\sigma}_e^2}\right) - \exp\left(\frac{-(d_i - \hat{u}_n^+)^2}{2\hat{\sigma}_e^2}\right)}{\text{erf}\left(\frac{(d_{i+1} - \hat{u}_n^+)}{\sqrt{2}\hat{\sigma}_e}\right) - \text{erf}\left(\frac{(d_i - \hat{u}_n^+)}{\sqrt{2}\hat{\sigma}_e}\right)}. \quad (\text{B.7})$$

B.2 Receiver-Sided Prediction Error Variance

The calculation of each term in (6.13) is derived in this section. With the LMQ property [Gersho and Gray, 1992], we know that the quantizer output is uncorrelated with the quantization error: $E\{u_{n-1} \cdot e_{n-1}^{\text{LMQ}}\} = 0$, therefore,

$$E\{\tilde{u}_{n-1} \cdot e_{n-1}^{\text{LMQ}}\} = E\{(u_{n-1} - e_{n-1}^{\text{LMQ}}) \cdot e_{n-1}^{\text{LMQ}}\} = -\sigma_{e^{\text{LMQ}}}^2, \quad (\text{B.8})$$

with $\sigma_{e^{\text{LMQ}}}^2$ being the variance of the Lloyd-Max quantization error. According to the definition of the normalized auto-correlation coefficient ρ , we have

$$E\{\tilde{u}_n \cdot \tilde{u}_{n-1}\} = \varphi_{\tilde{u}\tilde{u}}(1) = \rho \cdot \varphi_{\tilde{u}\tilde{u}}(0) = \rho \cdot \sigma_{\tilde{u}}^2, \quad (\text{B.9})$$

with $\sigma_{\tilde{u}}^2$ being the variance of the unquantized samples \tilde{u}_n . Adopting (B.8), the last term in (6.13) can be computed by

$$\begin{aligned} E\{\tilde{u}_n \cdot e_{n-1}^{\text{LMQ}}\} &= E\{(\rho \cdot \tilde{u}_{n-1} + \tilde{\epsilon}_n) \cdot e_{n-1}^{\text{LMQ}}\} \\ &= \rho \cdot E\{\tilde{u}_{n-1} \cdot e_{n-1}^{\text{LMQ}}\} + E\{\tilde{\epsilon}_n \cdot e_{n-1}^{\text{LMQ}}\} \\ &= -\rho \cdot \sigma_{e^{\text{LMQ}}}^2, \end{aligned} \quad (\text{B.10})$$

with $E\{\tilde{\epsilon}_n \cdot e_{n-1}^{\text{LMQ}}\} = E\{\tilde{\epsilon}_n\} \cdot E\{e_{n-1}^{\text{LMQ}}\} = 0$, because the innovation $\tilde{\epsilon}_n$ is independent from the previous Lloyd-Max quantization error e_{n-1}^{LMQ} .

Combining (B.8), (B.9), and (B.10), equation (6.13) can be written as

$$\begin{aligned} \hat{\sigma}_e^2 &= \sigma_{\tilde{u}}^2 + \hat{a}^2 \sigma_{\tilde{u}}^2 + \hat{a}^2 \sigma_{e^{\text{LMQ}}}^2 - 2\hat{a}^2 \sigma_{e^{\text{LMQ}}}^2 - 2\hat{a}\rho \sigma_{\tilde{u}}^2 + 2\hat{a}\rho \sigma_{e^{\text{LMQ}}}^2 \\ &= (1 + \hat{a}^2 - 2\hat{a}\rho) \cdot \sigma_{\tilde{u}}^2 - (\hat{a}^2 - 2\hat{a}\rho) \cdot \sigma_{e^{\text{LMQ}}}^2. \end{aligned} \quad (\text{B.11})$$

Furthermore, according to the property of a Gaussian AR(1) process, we know that $\frac{\sigma_{\tilde{\epsilon}}^2}{\sigma_{\tilde{u}}^2} = 1 - \rho^2$, which results in $\sigma_{\tilde{u}}^2 = \frac{\sigma_{\tilde{\epsilon}}^2}{1 - \rho^2}$. Also, the signal-to-noise ratio of the Lloyd-Max quantizer is given as a function of rate M [bits/sample]: $\text{SNR}(M) = \frac{\sigma_{\tilde{u}}^2}{\sigma_{e^{\text{LMQ}}}^2}$, which results in $\sigma_{e^{\text{LMQ}}}^2 = \frac{\sigma_{\tilde{\epsilon}}^2}{\text{SNR}(M) \cdot (1 - \rho^2)}$.

Finally, substituting $\sigma_{\tilde{u}}^2$ and $\sigma_{e^{\text{LMQ}}}^2$ in (B.11), the receiver-sided prediction error variance can be calculated by

$$\begin{aligned} \hat{\sigma}_e^2 &= \frac{(1 + \hat{a}^2 - 2\hat{a}\rho) \cdot \sigma_{\tilde{\epsilon}}^2}{1 - \rho^2} - \frac{(\hat{a}^2 - 2\hat{a}\rho) \cdot \sigma_{\tilde{\epsilon}}^2}{\text{SNR}(M) \cdot (1 - \rho^2)} \\ &= \frac{((1 + \hat{a}^2 - 2\hat{a}\rho) \cdot \text{SNR}(M) - (\hat{a}^2 - 2\hat{a}\rho)) \cdot \sigma_{\tilde{\epsilon}}^2}{\text{SNR}(M) \cdot (1 - \rho^2)}. \end{aligned} \quad (\text{B.12})$$

Appendix C

Audio Database

This section describes the audio database used in Chapter 5. Note that 15 training audio files are from Tab.C.1, while 10 audio files are taken from the music pieces given in Tab. C.2 for test.

Instruments	Composer/Title
Piano	Beethoven: Piano Sonata No. 32, Movt. 1: <i>Maestoso—Allegro con brio ed appassionato</i>
String	Beethoven: String Quartet No. 10, Op. 74 (harp quartet) Movt. 1: <i>Poco Adagio. Allegro</i>
Orchestra	Britten: Peter Grimes, Act 1: <i>Dawn: Lento tranquillo</i>
Piano	Hisaishi: Kaze no Oka (Windy Hill) of the Japanese animated fantasy film “Kiki’s Delivery Service”
Synthesizer, Rhythm machine	Dee3: A Definition of House (4 Melissa)
String	Dvořák: Violin Concerto in A minor, Op. 53, Movt. 3: <i>Finale: Allegro giocoso ma non troppo</i>
Speech, Sound effect	Soundtrack of the short film “Elephant’s Dream”
Orchestra	Giménez: El baile de Luis Alonso, Movt. 4: <i>Intermedio</i>
Piano	Gjeilo: North Country II
Organ	Kleive: Bridge Over Troubled Water
String	Mozart: Violin Concerto No. 4 in D major, K. 218, Movt. 1: <i>Allegro</i>
Orchestra	Respighi: Rossiniana P. 148, Movt. 3 : <i>Intermezzo</i>
Orchestra	Stravinsky: Jeu de cartes, Movt. 1: <i>Première donne (Alla breve — Moderato assai — Tranquillo)</i>
Choir, Orchestra	Williams: A Sea Symphony, Movt. 3: <i>The Waves: Scherzo (Allegro brillante)</i>

Table C.1: Description of the audio signals used for training.

Instruments	Composer/Title
String	Vivaldi: Cantata RV 679: <i>Che giova II sospirar povero core</i> - <i>Aria: Cupido, tu vedi</i>
String	Haydn: Quartett No. 64 in D-Dur, Op. 76, No. 5: <i>Finale (Presto)</i>
Church organ	Kåre Nordstoga: Orgelimpromvisata over Deilig er Jorden (Sandvold)
String, Percussion instrument, Sound effect	Soundtrack of the short film "Big Buck Bunny"
Percussion instrument, Wind instrument	Stravinsky: L'histoire du soldat: <i>Marche royale</i>
Orchestra	Schumann: Symphony No. 1, Movt. 3: <i>Scherzo (Molto vivace)</i>

Table C.2: Description of the audio signals used for test.

Appendix D

PEAQ ODG Results

This appendix shows corresponding PEAQ ODG results in Section 5.5.2.

Method	E_b/N_0 (dB)										
	0	1	2	3	4	5	6	7	8	9	10
SD	-3.02	-3.14	-3.23	-3.19	-3.04	-2.89	-2.81	-2.79	-2.77	-2.77	-2.77
HD	-3.13	-3.10	-3.07	-3.02	-2.96	-2.88	-2.83	-2.79	-2.77	-2.77	-2.77

(a) Distorting a number of 4 signed codebooks

Method	E_b/N_0 (dB)										
	0	1	2	3	4	5	6	7	8	9	10
SD_10CBs	-3.45	-3.25	-3.06	-3.10	-3.18	-3.13	-3.02	-1.72	-2.80	-2.77	-2.77
SD_4CBs	-2.73	-2.91	-3.10	-3.24	-3.24	-3.14	-2.99	-2.88	-2.80	-2.77	-2.77
HD	-3.28	-3.29	-3.27	-3.27	-3.22	-3.14	-3.00	-2.88	-2.80	-2.77	-2.77

(b) Distorting a number of 10 signed and unsigned codebooks (except codebook 11)

Table D.1: Global SNR (dB) results for HE-AAC transmission over an AWGN channel, with corrupted quantized spectral coefficients (cf. Tab. 5.8).

Method	E_b/N_0 (dB)										
	0	1	2	3	4	5	6	7	8	9	10
g_AK1/dsf_AK1	-3.21	-3.19	-3.27	-3.27	-3.32	-3.21	-3.01	-2.85	-2.78	-2.77	-2.77
g_AK1/dsf_AK0	-3.19	-3.15	-3.20	-3.19	-3.29	-3.20	-3.01	-2.85	-2.78	-2.77	-2.77
g_AK0/dsf_AK0	-2.86	-2.87	-2.98	-3.06	-3.25	-3.19	-3.03	-2.85	-2.79	-2.77	-2.77
HD	-1.37	-1.39	-1.43	-1.57	-1.80	-2.58	-2.94	-2.95	-2.86	-2.78	-2.77
EC	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54	-1.54

(a) Transmission over an AWGN channel

GE parameters	Method	$E_b/N_{0 bad}$ (dB)										
		5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0	9.5	10.0
$P=0.9$ $Q=0.1$ $P_{bad}=82\%$	g_AK1/dsf_AK1	-3.18	-3.06	-2.97	-2.90	-2.84	-2.80	-2.78	-2.77	-2.77	-2.77	-2.77
	g_AK1/dsf_AK0	-3.17	-3.06	-2.97	-2.91	-2.84	-2.80	-2.78	-2.77	-2.77	-2.77	-2.77
	g_AK0/dsf_AK0	-3.16	-3.08	-2.99	-2.92	-2.85	-2.80	-2.78	-2.77	-2.77	-2.77	-2.77
$P=0.62$ $Q=0.38$ $P_{bad}=50\%$	HD	-2.71	-2.88	-2.96	-3.04	-2.94	-2.90	-2.85	-2.79	-2.79	-2.77	-2.77
	EC	-3.38	-3.49	-3.42	-3.42	-3.44	-3.46	-3.50	-3.54	-3.45	-3.41	-3.46
	g_AK1/dsf_AK1	-3.07	-2.97	-2.89	-2.86	-2.81	-2.78	-2.77	-2.77	-2.77	-2.77	-2.77
$P=0.62$ $Q=0.38$ $P_{bad}=50\%$	g_AK1/dsf_AK0	-3.06	-2.97	-2.89	-2.86	-2.81	-2.78	-2.78	-2.77	-2.77	-2.77	-2.77
	g_AK0/dsf_AK0	-3.07	-2.99	-2.91	-2.87	-2.82	-2.79	-2.78	-2.78	-2.77	-2.77	-2.77
	HD	-2.93	-3.03	-3.03	-2.95	-2.92	-2.84	-2.83	-2.78	-2.78	-2.77	-2.77
$P_{bad}=50\%$	EC	-3.43	-3.39	-3.38	-3.42	-3.41	-3.41	-3.37	-3.43	-3.40	-3.42	-3.40

(b) Transmission over a GE channel with $E_b/N_{0|good} = 10$ dB

Table D.2: Global SNR (dB) results for HE-AAC transmission over an erroneous channel. Only the global gain and scale factors are being corrupted (cf. Tab. 5.9).

GE parameters	Method	$E_b/N_{0 bad}$ (dB)										
		5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0	9.5	10.0
$P=0.9$ $Q=0.1$ $P_{bad}=82\%$	$g_AK1/dsf_AK0/X_k-AK0$	-3.17	-3.26	-3.20	-3.09	-3.03	-2.90	-2.85	-2.80	-2.82	-2.79	-2.78
	$g_AK1/dsf_AK0/X_k-HD$	-3.19	-3.26	-3.19	-3.10	-3.03	-2.90	-2.85	-2.80	-2.82	-2.79	-2.78
	HD	-2.55	-2.84	-2.95	-3.10	-3.06	-2.98	-2.91	-2.81	-2.83	-2.79	-2.78
$P=0.62$ $Q=0.38$ $P_{bad}=50\%$	EC	-3.38	-3.49	-3.42	-3.42	-3.44	-3.46	-3.50	-3.54	-3.45	-3.41	-3.46
	$g_AK1/dsf_AK0/X_k-AK0$	-3.19	-3.17	-3.08	-2.97	-2.94	-2.00	-1.98	-2.79	-2.80	-2.79	-2.78
	$g_AK1/dsf_AK0/X_k-HD$	-3.19	-3.16	-3.08	-2.97	-2.94	-2.86	-2.81	-2.79	-2.79	-2.78	-2.78
$P_{bad}=50\%$	HD	-2.84	-3.04	-3.07	-3.00	-3.01	-2.91	-2.86	-2.80	-2.80	-2.79	-2.78
	EC	-3.44	-3.39	-3.38	-3.41	-3.41	-3.41	-3.37	-3.43	-3.39	-3.42	-3.40

Table D.3: Global SNR (dB) results for HE-AAC transmission over a Gilbert-Elliott channel with $E_b/N_{0|good} = 10$ dB. The global gain, scale factors, and quantized spectral coefficients are being corrupted, with spectral side information (cf. Tab. 5.11).

GE parameters	Method	$E_b/N_{0 bad}$ (dB)										
		5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0	9.5	10.0
$P=0.9$ $Q=0.1$ $P_{bad}=82\%$	$g_AK1/dsf_AK0/X_k-AK0$	-3.20	-3.16	-3.09	-2.99	-2.92	-2.85	-2.81	-2.78	-2.78	-2.77	-2.77
	$g_AK1/dsf_AK0/X_k-HD$	-3.22	-3.16	-3.08	-3.00	-2.92	-2.85	-2.81	-2.78	-2.78	-2.77	-2.77
	HD	-2.60	-2.83	-2.94	-3.06	-2.99	-2.94	-2.87	-2.80	-2.79	-2.77	-2.77
$P_{bad}=82\%$	EC	-3.38	-3.49	-3.42	-3.42	-3.44	-3.46	-3.50	-3.54	-3.45	-3.41	-3.46

Table D.4: Global SNR (dB) results for HE-AAC transmission over a Gilbert-Elliott channel with $E_b/N_{0|good} = 10$ dB. The global gain, scale factors, and the bits of quantized spectral coefficients with Huffman codebook 1 to 10 are being corrupted, with spectral side information (cf. Tab. 5.12).

Appendix E

Simulation Results in Tabular Forms

In order to see the minor difference between the simulation results shown with figures, which are not always recognizable based on graphical curves, this appendix presents the corresponding simulation results in tabular forms.

Chapter 3: Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

Parameter	Method	E_b/N_0 (dB)															
		-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
Adaptive codebook index	AK1	3.91	3.93	3.93	3.98	4.00	4.02	4.02	4.05	4.07	4.08	4.09	4.09	4.09	4.09	4.09	4.09
	AK0	3.79	3.81	3.84	3.89	3.92	3.95	3.96	4.02	4.05	4.06	4.08	4.09	4.09	4.09	4.09	4.09
	EC_BFI=0 (HD)	3.76	3.79	3.84	3.86	3.91	3.94	3.96	4.01	4.05	4.06	4.08	4.08	4.09	4.09	4.09	4.09
	EC_BFI=1	3.62	3.66	3.66	3.65	3.61	3.62	3.63	3.63	3.63	3.63	3.64	3.65	3.62	3.63	3.65	3.63
Fixed codebook index	AK1	3.81	3.83	3.86	3.90	3.93	3.97	4.00	4.04	4.05	4.06	4.08	4.09	4.09	4.09	4.09	4.09
	AK0	3.73	3.76	3.79	3.84	3.88	3.94	3.98	4.02	4.03	4.06	4.08	4.09	4.09	4.09	4.09	4.09
	EC_BFI=0 (HD)	2.67	2.77	2.89	3.06	3.17	3.39	3.50	3.72	3.84	3.95	4.01	4.07	4.09	4.09	4.09	4.09
	EC_BFI=1	3.35	3.41	3.38	3.39	3.39	3.35	3.37	3.36	3.41	3.37	3.45	3.37	3.40	3.38	3.40	3.42
Pitch delay	AK1	3.35	3.42	3.45	3.54	3.61	3.67	3.75	3.87	3.94	4.00	4.05	4.07	4.08	4.08	4.09	4.09
	AK0	2.92	2.99	3.07	3.12	3.22	3.31	3.41	3.59	3.76	3.87	3.98	4.03	4.07	4.08	4.09	4.09
	EC_BFI=0 (HD)	2.90	2.98	3.03	3.07	3.15	3.27	3.37	3.55	3.71	3.83	3.95	4.04	4.06	4.08	4.09	4.09
	EC_BFI=1	3.46	3.44	3.46	3.44	3.41	3.46	3.43	3.45	3.46	3.43	3.43	3.42	3.44	3.42	3.44	3.39
ISFs	AK1_Inter+Intra (Apr.2)	3.58	3.68	3.69	3.77	3.79	3.85	3.90	3.96	4.01	4.04	4.06	4.08	4.08	4.09	4.09	4.09
	AK1_Inter+Intra (Apr.1)	3.53	3.64	3.64	3.73	3.77	3.84	3.89	3.96	4.02	4.04	4.07	4.08	4.09	4.09	4.09	4.09
	AK1	3.48	3.57	3.56	3.64	3.65	3.73	3.79	3.86	3.96	4.01	4.05	4.07	4.09	4.09	4.09	4.09
	EC_BFI=0 (HD)	3.32	3.42	3.43	3.51	3.54	3.64	3.72	3.83	3.92	3.99	4.05	4.07	4.08	4.09	4.09	4.09
EC_BFI=0 (HD)	EC_BFI=0 (HD)	2.61	2.77	2.83	2.98	3.04	3.18	3.34	3.58	3.70	3.88	4.00	4.05	4.07	4.08	4.09	4.09
	EC_BFI=1	3.45	3.50	3.48	3.48	3.44	3.45	3.43	3.47	3.48	3.48	3.49	3.45	3.45	3.45	3.51	3.46

Table E.1: PESQ MOS results for AMR-NB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} \rightarrow \infty$, $P = 0.11$, $Q = 0.89$. Single parameter is individually corrupted (Fig. 3.6).

Chapter 3: Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

Parameter	Method	E_b/N_0 (dB)															
		-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
Adaptive codebook index	AK1	4.06	4.07	4.09	4.16	4.17	4.18	4.18	4.24	4.26	4.27	4.29	4.29	4.29	4.29	4.29	4.29
	AK0	3.87	3.90	3.95	4.01	4.06	4.09	4.10	4.20	4.24	4.24	4.28	4.29	4.28	4.29	4.29	4.29
	EC_BFI=0 (HD)	3.80	3.85	3.95	3.95	4.04	4.07	4.08	4.16	4.23	4.23	4.27	4.28	4.29	4.29	4.29	4.29
	EC_BFI=1	3.56	3.61	3.64	3.61	3.54	3.56	3.56	3.56	3.56	3.57	3.55	3.54	3.62	3.58	3.59	3.59
Fixed codebook index	AK1	3.99	4.02	4.06	4.08	4.12	4.14	4.19	4.21	4.22	4.26	4.28	4.29	4.29	4.29	4.29	4.29
	AK0	3.91	3.95	3.98	4.02	4.07	4.11	4.15	4.20	4.22	4.26	4.28	4.29	4.28	4.29	4.29	4.29
	EC_BFI=0 (HD)	2.66	2.79	2.93	3.07	3.18	3.39	3.50	3.80	3.95	4.10	4.19	4.26	4.28	4.29	4.29	4.29
	EC_BFI=1	3.57	3.64	3.61	3.62	3.60	3.57	3.58	3.57	3.62	3.62	3.63	3.59	3.63	3.59	3.63	3.63
Pitch delay	AK1	3.25	3.32	3.42	3.49	3.57	3.67	3.77	3.97	4.03	4.16	4.22	4.26	4.27	4.27	4.29	4.29
	AK0	2.71	2.74	2.88	2.95	3.03	3.17	3.28	3.53	3.79	3.97	4.10	4.19	4.26	4.27	4.29	4.29
	EC_BFI=0 (HD)	2.64	2.72	2.80	2.90	2.93	3.12	3.24	3.49	3.71	3.91	4.05	4.22	4.24	4.26	4.29	4.29
	EC_BFI=1	3.47	3.38	3.42	3.42	3.38	3.47	3.40	3.41	3.45	3.38	3.41	3.41	3.42	3.41	3.41	3.41
ISFs	AK1_Inter+Intra (Apr.2)	3.67	3.79	3.84	3.95	3.96	4.02	4.08	4.14	4.21	4.23	4.24	4.28	4.28	4.29	4.29	4.29
	AK1_Inter+Intra (Apr.1)	3.63	3.74	3.80	3.90	3.92	4.01	4.07	4.16	4.21	4.24	4.26	4.27	4.28	4.28	4.29	4.29
	AK1	3.58	3.67	3.72	3.82	3.78	3.89	3.98	4.04	4.15	4.22	4.24	4.28	4.29	4.29	4.29	4.29
	AK0	3.40	3.47	3.55	3.64	3.67	3.78	3.88	4.01	4.11	4.19	4.24	4.27	4.27	4.29	4.29	4.29
EC_BFI=0 (HD)	EC_BFI=0 (HD)	2.50	2.63	2.80	3.01	3.03	3.17	3.36	3.65	3.79	4.02	4.15	4.25	4.27	4.28	4.29	4.29
	EC_BFI=1	3.50	3.58	3.56	3.55	3.50	3.50	3.49	3.56	3.56	3.59	3.52	3.54	3.51	3.51	3.51	3.55

Table E.2: POLQA MOS results for AMR-NB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} \rightarrow \infty$, $P = 0.11$, $Q = 0.89$. Single parameter is individually corrupted (Fig. 3.6).

Chapter 3: Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

Parameter	Method	E_b/N_0 (dB)															
		-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
$P=0.9$ $Q=0.1$	AK1_Inter+Intra (Apr.2)	1.34	1.38	1.44	1.51	1.60	1.73	1.93	2.16	2.49	2.92	3.36	3.71	3.95	4.05	4.08	4.09
	AK1_Inter+Intra (Apr.1)	1.33	1.36	1.41	1.48	1.58	1.70	1.91	2.15	2.48	2.93	3.38	3.73	3.96	4.06	4.08	4.09
	AK1	1.31	1.34	1.39	1.45	1.53	1.63	1.81	2.03	2.36	2.82	3.31	3.69	3.94	4.05	4.08	4.09
	AK0	1.24	1.26	1.29	1.32	1.38	1.45	1.57	1.75	2.04	2.48	3.03	3.51	3.90	4.03	4.08	4.09
$P=0.62$ $Q=0.38$	EC_BFI=0 (HD)	1.12	1.13	1.15	1.17	1.21	1.27	1.35	1.48	1.69	2.08	2.61	3.29	3.72	3.99	4.08	4.09
	EC_BFI=1	1.17	1.18	1.20	1.18	1.19	1.16	1.23	1.17	1.16	1.23	1.16	1.16	1.15	1.20	1.18	1.18
	AK1_Inter+Intra (Apr.2)	1.57	1.63	1.69	1.78	1.91	2.06	2.26	2.52	2.87	3.25	3.61	3.82	4.00	4.06	4.08	4.09
	AK1_Inter+Intra (Apr.1)	1.55	1.61	1.67	1.76	1.89	2.05	2.24	2.52	2.87	3.26	3.62	3.83	4.01	4.07	4.08	4.09
$P=0.11$ $Q=0.89$	AK1	1.53	1.58	1.63	1.71	1.82	1.96	2.15	2.40	2.74	3.17	3.58	3.80	3.99	4.06	4.08	4.09
	AK0	1.40	1.43	1.46	1.54	1.60	1.71	1.86	2.08	2.40	2.85	3.35	3.69	3.97	4.05	4.08	4.09
	EC_BFI=0 (HD)	1.18	1.20	1.23	1.27	1.33	1.41	1.54	1.73	1.99	2.42	3.00	3.54	3.84	4.03	4.08	4.09
	EC_BFI=1	1.28	1.28	1.27	1.28	1.29	1.28	1.29	1.28	1.28	1.28	1.27	1.28	1.27	1.27	1.27	1.28
$P=0.11$ $Q=0.89$	AK1_Inter+Intra (Apr.2)	2.69	2.82	2.88	2.97	3.03	3.20	3.31	3.52	3.72	3.85	3.99	4.04	4.07	4.08	4.09	4.09
	AK1_Inter+Intra (Apr.1)	2.68	2.80	2.85	2.95	3.02	3.19	3.31	3.52	3.72	3.86	4.00	4.04	4.07	4.08	4.09	4.09
	AK1	2.65	2.77	2.81	2.90	2.96	3.12	3.24	3.44	3.68	3.83	3.98	4.03	4.07	4.08	4.09	4.09
	AK0	2.38	2.50	2.55	2.61	2.69	2.82	2.98	3.21	3.47	3.68	3.94	4.00	4.06	4.08	4.09	4.09
$P=0.11$ $Q=0.89$	EC_BFI=0 (HD)	1.70	1.85	1.92	2.02	2.15	2.32	2.53	2.86	3.14	3.50	3.81	3.95	4.03	4.07	4.08	4.09
	EC_BFI=1	2.48	2.58	2.54	2.54	2.50	2.51	2.52	2.55	2.58	2.56	2.61	2.56	2.59	2.56	2.60	2.57

Table E.3: PESQ MOS results for AMR-NB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB (Fig. 3.7).

Chapter 3: Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

Parameter	Method	E_b/N_0 (dB)															
		-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
$P=0.9$ $Q=0.1$	AK1_Inter+Intra (Apr.2)	1.25	1.33	1.41	1.49	1.58	1.69	1.82	2.02	2.32	2.83	3.35	3.82	4.09	4.25	4.28	4.29
	AK1_Inter+Intra (Apr.1)	1.22	1.30	1.37	1.46	1.55	1.67	1.82	2.01	2.32	2.83	3.38	3.82	4.11	4.25	4.28	4.29
	AK1	1.19	1.24	1.33	1.40	1.48	1.61	1.74	1.92	2.22	2.74	3.32	3.79	4.09	4.24	4.28	4.29
	AK0	1.07	1.09	1.12	1.15	1.23	1.36	1.47	1.67	1.91	2.31	2.92	3.52	3.99	4.20	4.28	4.29
	EC_BFI=0 (HD)	1.07	1.08	1.09	1.10	1.14	1.17	1.29	1.41	1.61	1.98	2.48	3.26	3.75	4.14	4.27	4.28
$P=0.62$ $Q=0.38$	AK1_Inter+Intra (Apr.2)	1.57	1.64	1.69	1.77	1.82	1.93	2.12	2.38	2.76	3.22	3.68	3.93	4.17	4.26	4.28	4.29
	AK1_Inter+Intra (Apr.1)	1.56	1.61	1.66	1.74	1.81	1.92	2.10	2.38	2.76	3.23	3.67	3.94	4.18	4.26	4.28	4.29
	AK1	1.56	1.58	1.64	1.71	1.77	1.87	2.03	2.27	2.65	3.15	3.63	3.91	4.15	4.25	4.28	4.29
	AK0	1.32	1.34	1.40	1.48	1.51	1.63	1.76	1.94	2.26	2.73	3.32	3.75	4.09	4.22	4.28	4.29
	EC_BFI=0 (HD)	1.11	1.13	1.15	1.19	1.26	1.35	1.49	1.64	1.88	2.31	2.91	3.58	3.89	4.18	4.27	4.28
	EC_BFI=1	1.31	1.32	1.31	1.34	1.36	1.32	1.35	1.36	1.36	1.35	1.35	1.35	1.33	1.36	1.33	1.36
$P=0.11$ $Q=0.89$	AK1_Inter+Intra (Apr.2)	2.57	2.67	2.78	2.87	2.98	3.13	3.29	3.54	3.80	3.99	4.16	4.25	4.28	4.29	4.29	4.29
	AK1_Inter+Intra (Apr.1)	2.54	2.65	2.75	2.84	2.95	3.12	3.28	3.54	3.82	3.98	4.17	4.25	4.25	4.27	4.29	4.29
	AK1	2.53	2.62	2.70	2.79	2.88	3.05	3.22	3.46	3.77	3.97	4.16	4.24	4.25	4.28	4.29	4.29
	AK0	2.20	2.29	2.38	2.42	2.52	2.65	2.86	3.12	3.45	3.74	4.08	4.19	4.23	4.27	4.28	4.29
	EC_BFI=0 (HD)	1.65	1.74	1.85	1.93	2.04	2.18	2.42	2.75	3.04	3.49	3.88	4.08	4.20	4.26	4.28	4.28
	EC_BFI=1	2.49	2.58	2.57	2.57	2.51	2.53	2.54	2.59	2.61	2.61	2.62	2.61	2.63	2.62	2.66	2.62

Table E.4: POLQA MOS results for AMR-NB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB (Fig. 3.7).

Chapter 3: Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

Parameter	Method	E_b/N_0 (dB)															
		-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
ISFs	AK1_Intra+Intra	2.94	2.99	3.04	3.07	3.14	3.22	3.19	3.34	3.37	3.41	3.45	3.46	3.47	3.47	3.47	3.47
	AK1	2.90	2.95	2.99	3.02	3.06	3.14	3.18	3.30	3.34	3.40	3.44	3.46	3.47	3.47	3.47	3.47
	AK0	2.58	2.64	2.72	2.77	2.85	2.96	3.01	3.20	3.29	3.36	3.41	3.45	3.46	3.47	3.47	3.47
	EC_BFI=0 (HD)	2.16	2.29	2.46	2.48	2.58	2.73	2.84	3.11	3.21	3.34	3.41	3.44	3.46	3.47	3.47	3.47
Pitch delay	EC_BFI=1	2.62	2.63	2.58	2.62	2.64	2.60	2.63	2.65	2.64	2.62	2.65	2.67	2.68	2.66	2.66	2.66
	AK1	2.57	2.64	2.71	2.79	2.81	2.92	2.99	3.14	3.24	3.34	3.38	3.44	3.46	3.47	3.47	3.47
	AK0	2.09	2.15	2.23	2.30	2.38	2.47	2.63	2.84	3.01	3.22	3.33	3.41	3.44	3.47	3.47	3.47
	EC_BFI=0 (HD)	2.08	2.13	2.21	2.26	2.35	2.43	2.61	2.81	3.00	3.20	3.29	3.41	3.44	3.46	3.47	3.44
VQ gain	EC_BFI=1	2.09	2.15	2.12	2.16	2.17	2.18	2.21	2.31	2.30	2.34	2.37	2.41	2.42	2.40	2.39	2.44
	AK1	2.84	2.80	2.92	3.02	3.05	3.13	3.23	3.29	3.34	3.42	3.43	3.45	3.46	3.47	3.47	3.47
	AK0	2.66	2.68	2.78	2.85	2.90	2.98	3.10	3.20	3.27	3.38	3.42	3.45	3.46	3.47	3.47	3.47
	EC_BFI=0 (HD)	1.70	1.78	1.84	2.00	2.13	2.31	2.52	2.73	3.01	3.22	3.36	3.40	3.46	3.47	3.47	3.47
	EC_BFI=1	2.13	2.16	2.15	2.17	2.16	2.15	2.14	2.19	2.13	2.18	2.19	2.17	2.17	2.14	2.15	2.20

Table E.5: PESQ MOS results for AMR-WB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} \rightarrow \infty$, $P = 0.11$, $Q = 0.89$. Single parameter is individually corrupted (Fig. 3.8).

Chapter 3: Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

Parameter	Method	E_b/N_0 (dB)															
		-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
ISFs	AK1_Intra+Intra	3.57	3.60	3.69	3.71	3.77	3.85	3.89	4.02	4.03	4.09	4.12	4.11	4.12	4.16	4.13	4.16
	AK1	3.52	3.57	3.67	3.66	3.71	3.80	3.85	4.00	4.02	4.09	4.13	4.14	4.16	4.16	4.16	4.16
	AK0	3.02	3.12	3.26	3.29	3.40	3.53	3.62	3.86	3.96	4.05	4.10	4.13	4.15	4.16	4.16	4.16
	EC_BFI=0 (HD)	2.47	2.68	2.97	2.95	3.05	3.27	3.41	3.75	3.87	4.01	4.08	4.11	4.15	4.16	4.16	4.16
Pitch delay	EC_BFI=1	3.26	3.25	3.26	3.27	3.29	3.22	3.25	3.31	3.27	3.32	3.32	3.25	3.30	3.27	3.28	3.34
	AK1	2.96	3.03	3.22	3.25	3.31	3.48	3.54	3.75	3.85	3.99	4.04	4.12	4.15	4.16	4.16	4.16
	AK0	2.13	2.26	2.41	2.45	2.53	2.75	2.89	3.29	3.48	3.80	3.96	4.08	4.12	4.16	4.16	4.16
	EC_BFI=0 (HD)	2.12	2.23	2.35	2.41	2.46	2.70	2.85	3.24	3.46	3.79	3.91	4.07	4.11	4.15	4.16	4.13
VQ gain	EC_BFI=1	2.38	2.42	2.47	2.48	2.49	2.50	2.59	2.73	2.72	2.90	2.90	2.92	2.93	2.92	2.91	3.05
	AK1	3.50	3.42	3.57	3.70	3.76	3.84	3.92	4.01	4.05	4.11	4.13	4.14	4.16	4.16	4.16	4.16
	AK0	3.24	3.28	3.42	3.46	3.56	3.63	3.78	3.88	3.97	4.05	4.10	4.14	4.16	4.16	4.16	4.16
	EC_BFI=0 (HD)	2.21	2.29	2.44	2.58	2.78	2.97	3.19	3.45	3.70	3.91	4.03	4.10	4.16	4.16	4.16	4.16
	EC_BFI=1	2.64	2.63	2.70	2.68	2.62	2.62	2.63	2.69	2.58	2.70	2.69	2.66	2.67	2.61	2.64	2.74

Table E.6: POLQA MOS results for AMR-WB transmission over a Gilbert-Elliott channel with $E_b/N_{0|\text{good}} \rightarrow \infty$, $P=0.11$, $Q=0.89$. Single parameter is individually corrupted (Fig. 3.8).

Chapter 3: Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

Parameter	Method	E_b/N_0 (dB)															
		-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
$P=0.9$ $Q=0.1$	AK1_Inter+Intra	1.13	1.14	1.16	1.18	1.22	1.27	1.34	1.48	1.71	2.04	2.48	2.88	3.25	3.42	3.47	3.47
	AK1	1.12	1.13	1.14	1.16	1.20	1.25	1.32	1.44	1.68	2.00	2.45	2.89	3.24	3.43	3.47	3.47
	AK0	1.09	1.10	1.10	1.11	1.13	1.16	1.21	1.29	1.46	1.72	2.16	2.69	3.17	3.40	3.46	3.46
$P=0.62$ $Q=0.38$	EC_BFI=0 (HD)	1.06	1.06	1.07	1.08	1.10	1.11	1.16	1.21	1.35	1.55	1.98	2.48	3.07	3.37	3.46	3.46
	EC_BFI=1	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.10	1.09	1.09	1.09	1.09	1.09	1.08	1.08	1.09
	AK1_Inter+Intra	1.23	1.25	1.27	1.32	1.36	1.46	1.54	1.74	2.03	2.37	2.76	3.06	3.33	3.44	3.47	3.47
$P=0.62$ $Q=0.38$	AK1	1.22	1.24	1.26	1.30	1.34	1.43	1.51	1.71	1.99	2.34	2.73	3.05	3.33	3.44	3.47	3.47
	AK0	1.16	1.16	1.18	1.20	1.24	1.29	1.36	1.47	1.70	2.03	2.46	2.91	3.29	3.42	3.47	3.46
	EC_BFI=0 (HD)	1.08	1.09	1.10	1.11	1.13	1.17	1.22	1.33	1.52	1.80	2.31	2.81	3.22	3.39	3.46	3.46
$P=0.11$ $Q=0.89$	EC_BFI=1	1.19	1.20	1.20	1.19	1.20	1.21	1.20	1.20	1.21	1.22	1.22	1.23	1.23	1.22	1.22	1.23
	AK1_Inter+Intra	1.95	2.00	2.08	2.16	2.24	2.38	2.53	2.72	2.92	3.14	3.30	3.38	3.43	3.47	3.47	3.47
	AK1	1.95	1.99	2.06	2.14	2.21	2.35	2.51	2.69	2.91	3.14	3.29	3.38	3.43	3.47	3.47	3.47
$P=0.89$	AK0	1.69	1.76	1.80	1.86	1.94	2.06	2.19	2.42	2.65	2.98	3.19	3.35	3.42	3.46	3.47	3.46
	EC_BFI=0 (HD)	1.34	1.37	1.42	1.47	1.57	1.67	1.85	2.11	2.48	2.77	3.10	3.28	3.40	3.46	3.47	3.46
	EC_BFI=1	1.88	1.90	1.94	1.99	1.99	1.95	1.97	2.04	2.08	2.12	2.14	2.14	2.13	2.19	2.13	2.17

Table E.7: PESQ MOS results for AMR-WB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB (Fig. 3.9).

Chapter 3: Improving AMR Narrowband and Wideband by Fixed-Length Soft-Decision Decoding

Parameter	Method	E_b/N_0 (dB)															
		-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
$P=0.9$ $Q=0.1$	AK1_Inter+Intra	1.05	1.06	1.09	1.12	1.17	1.28	1.39	1.59	1.92	2.38	2.97	3.50	3.91	4.11	4.16	4.16
	AK1	1.05	1.05	1.08	1.11	1.16	1.25	1.36	1.54	1.88	2.32	2.91	3.50	3.91	4.11	4.16	4.16
	AK0	1.03	1.03	1.03	1.04	1.04	1.08	1.13	1.22	1.45	1.82	2.44	3.17	3.74	4.06	4.15	4.16
	EC_BFI=0 (HD)	1.01	1.01	1.01	1.01	1.01	1.03	1.05	1.13	1.34	1.68	2.23	2.99	3.69	4.04	4.14	4.15
	EC_BFI=1	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.02	1.01	1.01	1.01	1.01	1.01
$P=0.62$ $Q=0.38$	AK1_Inter+Intra	1.21	1.24	1.27	1.33	1.39	1.57	1.67	1.96	2.32	2.80	3.31	3.70	4.01	4.10	4.12	4.16
	AK1	1.20	1.23	1.25	1.30	1.37	1.50	1.63	1.91	2.28	2.78	3.29	3.69	4.01	4.13	4.16	4.16
	AK0	1.06	1.07	1.14	1.10	1.14	1.21	1.29	1.45	1.75	2.24	2.82	3.46	3.92	4.11	4.15	4.16
	EC_BFI=0 (HD)	1.01	1.02	1.02	1.02	1.05	1.11	1.15	1.31	1.60	2.03	2.64	3.38	3.85	4.06	4.15	4.15
	EC_BFI=1	1.15	1.13	1.14	1.14	1.17	1.17	1.16	1.17	1.20	1.20	1.21	1.22	1.22	1.23	1.23	1.23
$P=0.11$ $Q=0.89$	AK1_Inter+Intra	2.20	2.27	2.43	2.45	2.60	2.81	3.02	3.28	3.49	3.79	3.94	4.07	4.12	4.16	4.16	4.16
	AK1	2.19	2.26	2.42	2.43	2.58	2.78	3.00	3.25	3.48	3.79	3.96	4.07	4.12	4.16	4.16	4.16
	AK0	1.77	1.86	1.96	2.00	2.08	2.27	2.42	2.80	3.07	3.57	3.82	4.02	4.10	4.15	4.16	4.16
	EC_BFI=0 (HD)	1.44	1.44	1.52	1.65	1.75	1.86	2.14	2.50	2.96	3.35	3.75	3.95	4.09	4.14	4.16	4.15
	EC_BFI=1	2.16	2.21	2.19	2.31	2.26	2.25	2.32	2.40	2.53	2.59	2.58	2.61	2.61	2.69	2.58	2.70

Table E.8: POLQA MOS results for AMR-WB transmission over a Gilbert-Elliott channel with $E_b/N_0|_{\text{good}} = 10$ dB (Fig. 3.9).

Chapter 4: Variable-Length Soft-Decision Decoding

Block length	Method	-2	-1	0	1	2	3	4	5	6	7	8	9	10	11	12	13
$B=1$	FLC (FBA)/AK1	1.58	1.94	2.39	2.93	3.58	4.32	5.20	6.12	7.08	7.93	8.57	8.97	9.19	9.26	9.28	9.28
	FLC (FA)/AK1	1.58	1.94	2.39	2.93	3.58	4.32	5.20	6.12	7.08	7.93	8.57	8.97	9.19	9.26	9.28	9.28
	VLC/AK1	2.57	2.98	3.43	4.04	4.66	5.40	6.17	6.99	7.74	8.32	8.83	9.08	9.23	9.27	9.28	9.28
	FLC/AK0	1.58	1.94	2.39	2.93	3.58	4.32	5.20	6.12	7.08	7.93	8.57	8.97	9.19	9.26	9.28	9.28
	VLC/AK0	2.57	2.98	3.43	4.04	4.66	5.40	6.17	6.99	7.74	8.32	8.83	9.08	9.23	9.27	9.28	9.28
	FLC/HD	-0.49	-0.01	0.58	1.26	2.07	2.98	4.05	5.17	6.36	7.46	8.29	8.85	9.15	9.25	9.28	9.28
VLC/HD	-0.86	-0.46	0.02	0.67	1.39	2.28	3.35	4.55	5.80	6.98	8.06	8.75	9.11	9.24	9.28	9.28	
$B=2$	FLC (FBA)/AK1	2.59	3.13	3.80	4.53	5.35	6.25	7.13	7.87	8.47	8.86	9.08	9.22	9.26	9.28	9.28	9.28
	FLC (FA)/AK1	2.06	2.48	3.05	3.64	4.37	5.17	6.06	6.90	7.72	8.35	8.81	9.12	9.22	9.26	9.28	9.28
	VLC/AK1	2.73	3.24	3.86	4.56	5.33	6.09	6.94	7.67	8.28	8.75	9.03	9.18	9.25	9.28	9.28	9.28
	FLC/AK0	1.58	1.93	2.40	2.91	3.57	4.31	5.19	6.11	7.07	7.90	8.56	8.99	9.18	9.25	9.28	9.28
	VLC/AK0	2.15	2.56	3.08	3.69	4.39	5.15	6.03	6.86	7.68	8.37	8.82	9.09	9.22	9.27	9.28	9.28
	FLC/HD	-0.49	-0.02	0.58	1.24	2.05	2.95	4.03	5.18	6.33	7.42	8.28	8.86	9.15	9.24	9.28	9.28
VLC/HD	-1.21	-0.87	-0.42	0.13	0.84	1.66	2.74	3.95	5.27	6.66	7.78	8.63	9.07	9.22	9.27	9.28	
$B=4$	FLC (FBA)/AK1	3.76	4.44	5.21	5.96	6.71	7.42	8.01	8.47	8.78	9.05	9.15	9.24	9.27	9.28	9.28	9.29
	FLC (FA)/AK1	2.65	3.19	3.79	4.47	5.20	6.00	6.77	7.49	8.12	8.65	8.94	9.16	9.25	9.27	9.28	9.28
	VLC/AK1	3.13	3.70	4.43	5.19	5.97	6.82	7.55	8.20	8.66	8.95	9.15	9.23	9.27	9.28	9.28	9.29
	FLC/AK0	1.57	1.96	2.39	2.93	3.58	4.33	5.18	6.11	7.06	7.94	8.55	8.99	9.19	9.25	9.28	9.28
	VLC/AK0	1.75	2.12	2.66	3.26	4.00	4.84	5.75	6.72	7.57	8.29	8.79	9.09	9.22	9.27	9.28	9.28
	FLC/HD	-0.51	-0.00	0.57	1.26	2.06	2.97	4.04	5.16	6.34	7.48	8.26	8.88	9.15	9.24	9.28	9.28
VLC/HD	-1.52	-1.21	-0.84	-0.34	0.26	1.06	2.08	3.30	4.67	6.13	7.46	8.46	9.03	9.20	9.27	9.28	
$B=8$	FLC (FBA)/AK1	4.62	5.29	6.00	6.68	7.31	7.87	8.32	8.66	8.91	9.10	9.19	9.25	9.27	9.28	9.28	9.29
	FLC (FA)/AK1	3.21	3.76	4.39	5.06	5.78	6.50	7.19	7.83	8.35	8.78	9.03	9.18	9.26	9.27	9.28	9.28
	VLC/AK1	3.16	3.81	4.56	5.38	6.23	7.06	7.80	8.43	8.81	9.07	9.19	9.25	9.27	9.28	9.28	9.29
	FLC/AK0	1.58	1.95	2.38	2.92	3.57	4.33	5.18	6.13	7.06	7.92	8.57	8.98	9.19	9.26	9.28	9.28
	VLC/AK0	1.27	1.64	2.10	2.70	3.43	4.32	5.30	6.42	7.39	8.22	8.77	9.06	9.22	9.27	9.28	9.29
	FLC/HD	-0.50	0.00	0.56	1.25	2.05	2.97	4.01	5.18	6.35	7.46	8.30	8.86	9.15	9.25	9.28	9.28
VLC/HD	-1.73	-1.49	-1.17	-0.77	-0.21	0.49	1.41	2.59	3.96	5.53	7.06	8.21	8.93	9.19	9.27	9.28	

Table E.9: Simulation results for an $M=2$ bit quantized Gaussian AR(1) process with different block lengths B (part 1) (Fig. 4.5).

Chapter 4: Variable-Length Soft-Decision Decoding

Block length	Method	E_s/N_0 (dB)															
		-2	-1	0	1	2	3	4	5	6	7	8	9	10	11	12	13
$B = 16$	FLC (FBA)/AK1	5.09	5.72	6.38	7.02	7.58	8.06	8.45	8.76	8.99	9.13	9.21	9.26	9.28	9.28	9.28	9.29
	FLC (FA)/AK1	3.54	4.10	4.74	5.40	6.10	6.78	7.43	8.00	8.48	8.85	9.07	9.20	9.26	9.28	9.28	9.29
	VLC/AK1	2.86	3.50	4.19	4.96	5.83	6.75	7.58	8.32	8.80	9.09	9.21	9.26	9.28	9.28	9.28	9.29
	FLC/AK0	1.58	1.94	2.39	2.92	3.57	4.33	5.19	6.11	7.08	7.91	8.57	8.98	9.18	9.26	9.28	9.28
	VLC/AK0	0.92	1.25	1.63	2.15	2.81	3.59	4.61	5.80	7.00	8.05	8.67	9.05	9.21	9.27	9.28	9.29
	FLC/HD	-0.49	-0.02	0.57	1.25	2.04	2.98	4.08	5.16	6.37	7.43	8.30	8.87	9.14	9.25	9.28	9.28
VLC/HD	-1.83	-1.63	-1.36	-1.01	-0.55	0.07	0.88	1.93	3.33	4.90	6.54	7.96	8.83	9.17	9.27	9.28	
$B = 32$	FLC (FBA)/AK1	5.32	5.91	6.59	7.18	7.74	8.17	8.55	8.81	9.02	9.13	9.21	9.26	9.28	9.28	9.29	9.29
	FLC (FA)/AK1	3.72	4.26	4.93	5.59	6.29	6.93	7.57	8.09	8.55	8.86	9.09	9.21	9.26	9.28	9.28	9.29
	VLC/AK1	2.52	3.10	3.77	4.51	5.30	6.16	7.04	7.88	8.56	8.99	9.19	9.25	9.28	9.28	9.28	9.29
	FLC/AK0	1.58	1.94	2.38	2.92	3.59	4.33	5.18	6.11	7.07	7.89	8.56	8.99	9.18	9.26	9.28	9.28
	VLC/AK0	0.71	1.01	1.35	1.80	2.36	3.02	3.84	4.93	6.22	7.55	8.51	9.00	9.20	9.26	9.28	9.28
	FLC/HD	-0.49	-0.01	0.56	1.24	2.06	2.98	4.02	5.15	6.36	7.41	8.29	8.88	9.14	9.25	9.28	9.28
VLC/HD	-1.90	-1.73	-1.50	-1.20	-0.79	-0.26	0.48	1.41	2.66	4.26	6.02	7.62	8.64	9.15	9.26	9.28	
$B = 64$	FLC (FBA)/AK1	5.41	6.06	6.68	7.27	7.80	8.23	8.58	8.85	9.04	9.15	9.22	9.26	9.28	9.28	9.28	9.29
	FLC (FA)/AK1	3.78	4.37	5.02	5.69	6.36	7.02	7.60	8.16	8.59	8.89	9.09	9.20	9.26	9.28	9.28	9.28
	VLC/AK1	2.31	2.83	3.45	4.15	4.86	5.59	6.43	7.32	8.17	8.78	9.11	9.24	9.28	9.28	9.29	9.29
	FLC/AK0	1.57	1.95	2.39	2.93	3.58	4.33	5.19	6.11	7.08	7.92	8.58	8.98	9.19	9.26	9.28	9.28
	VLC/AK0	0.58	0.83	1.15	1.56	2.06	2.67	3.39	4.30	5.40	6.88	8.21	8.89	9.20	9.26	9.28	9.28
	FLC/HD	-0.50	-0.00	0.58	1.26	2.05	2.98	4.02	5.16	6.36	7.43	8.31	8.87	9.14	9.25	9.28	9.28
VLC/HD	-1.97	-1.82	-1.63	-1.36	-1.02	-0.56	0.05	0.90	2.02	3.52	5.30	7.10	8.46	9.04	9.26	9.28	
$B = 128$	FLC (FBA)/AK1	5.51	6.11	6.73	7.31	7.82	8.26	8.61	8.86	9.05	9.16	9.23	9.26	9.28	9.28	9.28	9.29
	FLC (FA)/AK1	3.86	4.42	5.06	5.73	6.39	7.06	7.66	8.18	8.62	8.91	9.10	9.21	9.26	9.28	9.28	9.28
	VLC/AK1	2.10	2.61	3.15	3.77	4.45	5.17	5.93	6.78	7.66	8.44	9.01	9.22	9.27	9.28	9.29	9.29
	FLC/AK0	1.58	1.93	2.37	2.94	3.57	4.33	5.19	6.13	7.08	7.92	8.57	8.98	9.18	9.26	9.28	9.28
	VLC/AK0	0.47	0.70	0.98	1.35	1.81	2.39	3.10	3.95	4.92	6.22	7.70	8.76	9.15	9.26	9.28	9.28
	FLC/HD	-0.50	-0.02	0.56	1.27	2.04	2.99	4.04	5.19	6.36	7.44	8.31	8.86	9.14	9.24	9.28	9.28
VLC/HD	-2.05	-1.94	-1.77	-1.55	-1.27	-0.85	-0.34	0.40	1.30	2.62	4.45	6.42	7.98	8.93	9.22	9.28	

Table E.10: Simulation results for an $M = 2$ bit quantized Gaussian AR(1) process with different block lengths B (part 2) (Fig. 4.5).

Chapter 4: Variable-Length Soft-Decision Decoding

Block length	Method	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
$B=1$	FLC (FBA)/AK1	1.54	1.92	2.40	3.00	3.76	4.71	5.93	7.42	9.30	11.52	14.04	16.61	18.75	19.74	20.13	20.22	
	FLC (FA)/AK1	1.54	1.92	2.40	3.00	3.76	4.71	5.93	7.42	9.30	11.52	14.04	16.61	18.75	19.74	20.13	20.22	
	VLC/AK1	2.65	3.16	3.82	4.64	5.61	6.75	8.22	9.92	11.83	14.07	16.26	18.31	19.55	20.06	20.18	20.22	20.22
	FLC/AK0	1.54	1.92	2.40	3.00	3.76	4.71	5.93	7.42	9.30	11.52	14.04	16.61	18.75	19.74	20.13	20.22	20.22
	VLC/AK0	2.65	3.16	3.82	4.64	5.61	6.75	8.22	9.92	11.83	14.07	16.26	18.31	19.55	20.06	20.18	20.22	20.22
	FLC/HD	-2.14	-1.61	-0.98	-0.20	0.72	1.87	3.26	4.93	6.96	9.41	12.17	15.21	17.97	19.47	20.08	20.22	20.22
$B=2$	VLC/HD	-1.29	-0.93	-0.44	0.18	0.94	1.96	3.27	4.86	6.86	9.40	12.27	15.45	18.16	19.53	20.12	20.22	20.22
	FLC (FBA)/AK1	2.63	3.27	4.12	5.11	6.36	7.90	9.78	11.79	14.13	16.33	18.14	19.32	19.91	20.13	20.21	20.22	20.22
	FLC (FA)/AK1	2.06	2.53	3.18	3.91	4.87	6.01	7.45	9.06	11.09	13.25	15.54	17.96	19.25	19.86	20.19	20.22	20.22
	VLC/AK1	1.82	2.47	3.23	4.24	5.43	6.90	8.66	10.63	12.78	15.07	17.18	18.85	19.74	20.10	20.22	20.22	20.22
	FLC/AK0	1.54	1.91	2.40	2.99	3.75	4.70	5.93	7.41	9.30	11.48	13.97	16.77	18.73	19.68	20.16	20.22	20.22
	VLC/AK0	1.26	1.79	2.45	3.31	4.33	5.65	7.26	9.11	11.18	13.58	16.07	18.16	19.49	20.01	20.20	20.22	20.22
$B=4$	FLC/HD	-2.14	-1.63	-0.98	-0.21	0.73	1.84	3.25	4.96	6.93	9.38	12.08	15.26	17.96	19.34	20.11	20.22	20.22
	VLC/HD	-1.67	-1.37	-0.97	-0.45	0.21	1.16	2.39	3.92	5.87	8.34	11.37	14.59	17.70	19.29	20.10	20.21	20.22
	FLC (FBA)/AK1	4.17	5.10	6.22	7.56	9.06	10.74	12.50	14.23	16.00	17.57	18.71	19.60	19.98	20.17	20.22	20.22	20.22
	FLC (FA)/AK1	2.79	3.42	4.18	5.12	6.19	7.51	9.00	10.66	12.61	14.78	16.54	18.57	19.68	19.98	20.20	20.22	20.22
	VLC/AK1	1.76	2.48	3.42	4.55	5.96	7.65	9.55	11.53	13.71	15.88	17.77	19.11	19.83	20.14	20.21	20.22	20.22
	FLC/AK0	1.54	1.92	2.39	3.00	3.76	4.72	5.91	7.43	9.27	11.53	13.98	16.66	18.73	19.71	20.15	20.21	20.22
$B=8$	VLC/AK0	0.63	1.00	1.53	2.28	3.32	4.58	6.20	8.09	10.36	12.90	15.46	17.71	19.27	20.01	20.20	20.22	20.22
	FLC/HD	-2.13	-1.61	-0.99	-0.21	0.73	1.85	3.24	4.94	6.93	9.40	12.12	15.27	17.99	19.35	20.09	20.20	20.22
	VLC/HD	-1.98	-1.76	-1.43	-0.99	-0.40	0.38	1.49	2.92	4.82	7.27	10.23	13.74	16.95	19.23	20.02	20.21	20.22
	FLC (FBA)/AK1	5.46	6.47	7.71	8.98	10.37	11.90	13.43	15.01	16.51	17.96	18.98	19.69	20.02	20.17	20.22	20.22	20.22
	FLC (FA)/AK1	3.53	4.25	5.14	6.12	7.30	8.66	10.14	11.82	13.68	15.75	17.48	18.91	19.80	20.07	20.21	20.22	20.22
	VLC/AK1	1.84	2.69	3.72	4.99	6.51	8.22	10.19	12.32	14.37	16.43	18.11	19.36	19.95	20.20	20.22	20.22	20.22
FLC/AK0	1.54	1.92	2.39	3.01	3.75	4.71	5.91	7.40	9.27	11.55	14.06	16.66	18.81	19.77	20.14	20.22	20.22	
VLC/AK0	0.20	0.51	0.95	1.61	2.53	3.78	5.34	7.27	9.48	12.01	14.90	17.40	19.09	19.91	20.18	20.22	20.22	
FLC/HD	-2.14	-1.62	-0.98	-0.22	0.71	1.84	3.22	4.89	6.91	9.47	12.16	15.36	18.14	19.46	20.05	20.21	20.22	
VLC/HD	-2.18	-1.98	-1.72	-1.33	-0.85	-0.18	0.83	2.17	3.94	6.24	9.26	12.72	16.49	18.94	20.02	20.21	20.22	

Table E.1.1: Simulation results for an $M=4$ bit quantized Gaussian AR(1) process with different block lengths B (part 1) (Fig. 4.6).

Chapter 4: Variable-Length Soft-Decision Decoding

Block length	Method	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
$B=16$	FLC (FBA)/AK1	6.19	7.20	8.35	9.62	10.99	12.38	13.95	15.39	16.84	18.12	19.13	19.76	20.07	20.18	20.22	20.22	
	FLC (FA)/AK1	4.06	4.84	5.73	6.81	8.01	9.36	10.92	12.56	14.37	16.24	17.87	19.14	19.86	20.10	20.21	20.22	
	VLC/AK1	1.94	2.74	3.74	5.02	6.52	8.23	10.23	12.36	14.60	16.69	18.35	19.44	19.98	20.15	20.21	20.22	
	FLC/AK0	1.54	1.92	2.39	3.01	3.76	4.72	5.90	7.38	9.26	11.52	14.06	16.65	18.74	19.75	20.16	20.22	
	VLC/AK0	0.11	0.33	0.69	1.24	1.99	3.05	4.45	6.27	8.52	11.21	14.00	16.94	18.96	19.83	20.16	20.22	
	FLC/HD	-2.13	-1.60	-0.99	-0.21	0.72	1.84	3.21	4.86	6.93	9.43	12.19	15.26	18.03	19.48	20.12	20.21	
VLC/HD	-2.29	-2.14	-1.92	-1.60	-1.17	-0.53	0.35	1.60	3.28	5.58	8.50	12.32	15.96	18.63	19.87	20.20		
$B=32$	FLC (FBA)/AK1	6.55	7.58	8.73	9.97	11.34	12.73	14.16	15.61	17.00	18.24	19.18	19.79	20.07	20.19	20.22	20.22	
	FLC (FA)/AK1	4.34	5.15	6.10	7.18	8.44	9.76	11.35	13.00	14.68	16.57	18.07	19.27	19.87	20.14	20.21	20.22	
	VLC/AK1	1.89	2.65	3.62	4.76	6.16	7.78	9.75	11.95	14.37	16.54	18.27	19.48	19.98	20.16	20.22	20.22	
	FLC/AK0	1.54	1.92	2.40	3.00	3.77	4.70	5.89	7.40	9.27	11.48	13.99	16.63	18.64	19.80	20.16	20.21	
	VLC/AK0	0.15	0.34	0.65	1.07	1.71	2.52	3.63	5.17	7.23	10.05	13.40	16.50	18.75	19.86	20.16	20.22	
	FLC/HD	-2.14	-1.61	-0.97	-0.21	0.74	1.85	3.20	4.90	6.94	9.39	12.16	15.27	17.89	19.55	20.13	20.21	
VLC/HD	-2.37	-2.22	-2.02	-1.76	-1.33	-0.77	0.03	1.20	2.76	4.96	7.92	11.56	15.23	18.68	19.99	20.20		
$B=64$	FLC (FBA)/AK1	6.70	7.81	8.93	10.14	11.51	12.86	14.33	15.74	17.08	18.29	19.22	19.80	20.09	20.19	20.22	20.22	
	FLC (FA)/AK1	4.49	5.33	6.28	7.36	8.60	10.02	11.58	13.23	14.94	16.67	18.24	19.28	19.92	20.13	20.22	20.22	
	VLC/AK1	1.83	2.55	3.43	4.46	5.68	7.28	9.01	11.27	13.71	16.09	18.16	19.40	19.98	20.17	20.22	20.22	
	FLC/AK0	1.54	1.92	2.41	3.00	3.74	4.71	5.92	7.42	9.23	11.50	14.02	16.59	18.65	19.75	20.16	20.21	
	VLC/AK0	0.19	0.36	0.62	0.97	1.49	2.19	3.12	4.40	6.18	8.73	12.24	15.73	18.72	19.84	20.15	20.21	
	FLC/HD	-2.14	-1.61	-0.96	-0.20	0.70	1.86	3.24	4.90	6.90	9.36	12.20	15.26	17.83	19.45	20.11	20.20	
VLC/HD	-2.43	-2.30	-2.11	-1.87	-1.50	-0.99	-0.26	0.79	2.34	4.43	7.21	10.90	14.69	18.06	19.88	20.21		
$B=128$	FLC (FBA)/AK1	6.83	7.89	9.01	10.27	11.59	12.93	14.37	15.86	17.20	18.34	19.21	19.81	20.07	20.19	20.22	20.22	
	FLC (FA)/AK1	4.55	5.38	6.35	7.47	8.73	10.12	11.69	13.40	15.14	16.78	18.23	19.32	19.90	20.13	20.21	20.22	
	VLC/AK1	1.79	2.46	3.25	4.18	5.31	6.73	8.46	10.41	12.92	15.51	17.74	19.35	19.94	20.17	20.21	20.22	
	FLC/AK0	1.53	1.90	2.40	3.00	3.75	4.72	5.90	7.39	9.29	11.49	14.04	16.65	18.70	19.76	20.13	20.21	
	VLC/AK0	0.22	0.36	0.58	0.88	1.32	1.95	2.79	3.94	5.52	7.73	10.88	15.18	18.36	19.78	20.16	20.22	
	FLC/HD	-2.15	-1.62	-0.98	-0.21	0.71	1.86	3.23	4.88	6.96	9.39	12.23	15.17	17.97	19.51	20.08	20.20	
VLC/HD	-2.48	-2.37	-2.19	-1.98	-1.67	-1.18	-0.52	0.44	1.85	3.78	6.44	10.11	14.12	17.58	19.92	20.14		

Table E.12: Simulation results for an $M=4$ bit quantized Gaussian AR(1) process with different block lengths B (part 2) (Fig. 4.6).

Chapter 4: Variable-Length Soft-Decision Decoding

Block length	Method	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
$B=1$	FLC (FBA)/AK1	1.71	2.12	2.65	3.30	4.14	5.18	6.51	8.15	10.26	12.85	16.03	19.94	24.61	28.51	30.88	31.64	
	FLC (FA)/AK1	1.71	2.12	2.65	3.30	4.14	5.18	6.51	8.15	10.26	12.85	16.03	19.94	24.61	28.51	30.88	31.64	
	VLC/AK1	1.37	1.77	2.27	2.94	3.80	4.89	6.28	8.02	10.17	12.92	16.42	20.28	24.68	29.12	31.13	31.62	
	FLC/AK0	1.71	2.12	2.65	3.30	4.14	5.18	6.51	8.15	10.26	12.85	16.03	19.94	24.61	28.51	30.88	31.64	
	VLC/AK0	1.37	1.77	2.27	2.94	3.80	4.89	6.28	8.02	10.17	12.92	16.42	20.28	24.68	29.12	31.13	31.62	
	FLC/HD	-2.08	-1.52	-0.84	-0.01	0.97	2.20	3.68	5.49	7.71	10.43	13.74	17.75	22.65	27.01	30.40	31.67	
$B=2$	VLC/HD	-2.26	-1.97	-1.56	-0.98	-0.27	0.72	1.99	3.66	5.73	8.43	11.80	15.80	21.19	26.39	30.33	31.49	
	FLC (FBA)/AK1	2.90	3.56	4.52	5.52	6.91	8.59	10.57	13.00	15.83	18.90	22.39	26.01	29.06	30.90	31.52	31.65	
	FLC (FA)/AK1	2.26	2.77	3.49	4.26	5.30	6.54	8.08	9.91	12.20	14.86	18.00	22.45	26.21	29.10	31.36	31.65	
	VLC/AK1	0.90	1.41	2.07	2.92	4.01	5.38	7.04	9.04	11.37	14.38	17.79	21.74	26.41	29.55	31.09	31.65	
	FLC/AK0	1.70	2.10	2.66	3.28	4.13	5.16	6.52	8.11	10.24	12.82	15.93	20.25	24.35	28.30	31.26	31.63	
	VLC/AK0	0.60	0.98	1.51	2.17	3.07	4.22	5.63	7.44	9.64	12.42	15.75	19.73	24.82	28.43	31.04	31.64	
$B=4$	FLC/HD	-2.07	-1.54	-0.84	-0.03	0.97	2.18	3.68	5.46	7.68	10.44	13.73	17.99	22.31	26.76	30.90	31.60	
	VLC/HD	-2.39	-2.13	-1.77	-1.27	-0.60	0.31	1.55	3.11	5.17	7.85	11.13	15.39	20.21	25.42	29.49	31.37	
	FLC (FBA)/AK1	4.46	5.47	6.63	8.03	9.45	10.96	12.66	14.55	16.64	18.09	20.96	23.90	27.14	29.60	31.10	31.60	31.66
	FLC (FA)/AK1	3.02	3.71	4.51	5.50	6.65	8.08	9.67	11.61	13.89	16.79	19.55	23.80	27.92	29.69	31.45	31.66	
	VLC/AK1	0.96	1.54	2.32	3.37	4.59	6.24	8.16	10.36	12.90	15.87	19.33	23.04	27.32	30.04	31.26	31.63	
	FLC/AK0	1.70	2.13	2.65	3.31	4.14	5.18	6.46	8.16	10.23	12.85	15.90	20.04	24.74	28.37	31.20	31.60	
$B=8$	VLC/AK0	0.13	0.40	0.80	1.43	2.23	3.42	4.90	6.71	8.96	11.79	15.14	19.06	24.02	28.12	30.95	31.61	
	FLC/HD	-2.07	-1.51	-0.85	-0.01	0.99	2.17	3.64	5.52	7.68	10.49	13.67	17.81	22.84	26.81	30.99	31.58	
	VLC/HD	-2.58	-2.37	-2.06	-1.65	-1.08	-0.27	0.82	2.25	4.22	6.82	10.02	14.22	19.48	24.86	29.72	31.63	
	FLC (FBA)/AK1	5.75	6.82	8.03	9.45	10.96	12.66	14.55	16.64	19.03	21.70	24.63	27.62	29.92	31.28	31.56	31.66	
	FLC (FA)/AK1	3.78	4.53	5.44	6.52	7.76	9.21	10.90	12.86	15.17	18.03	21.13	24.66	28.56	30.35	31.45	31.66	
	VLC/AK1	1.10	1.76	2.67	3.76	5.22	6.96	8.96	11.32	13.91	16.99	20.40	24.13	27.88	30.68	31.37	31.66	
$B=16$	FLC/AK0	1.71	2.12	2.64	3.31	4.12	5.16	6.49	8.17	10.20	12.85	16.05	19.96	24.55	28.42	30.96	31.60	
	VLC/AK0	-0.03	0.15	0.45	0.92	1.69	2.72	4.12	5.89	8.20	10.95	14.41	18.54	23.15	28.66	30.77	31.63	
	FLC/HD	-2.07	-1.50	-0.85	-0.04	0.97	2.17	3.65	5.49	7.60	10.44	13.79	17.88	22.62	26.92	30.40	31.55	
	VLC/HD	-2.73	-2.57	-2.34	-2.04	-1.54	-0.88	0.07	1.37	3.19	5.70	8.83	12.79	17.96	23.46	29.38	31.12	

Table E.13: Simulation results for an $M=6$ bit quantized Gaussian AR(1) process with different block lengths B (part 1) (Fig. 4.7).

Chapter 4: Variable-Length Soft-Decision Decoding

Block length	Method	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
$B=16$	FLC (FBA)/AKI	6.48	7.50	8.74	10.10	11.59	13.26	15.11	17.21	19.56	22.14	25.13	27.92	30.20	31.23	31.59	31.65	
	FLC (FA)/AKI	4.32	5.10	6.06	7.16	8.47	9.92	11.63	13.63	15.99	18.72	21.90	25.33	29.00	30.58	31.50	31.65	
	VLC/AKI	1.12	1.78	2.67	3.81	5.26	7.17	9.27	11.63	14.47	17.40	21.04	25.33	29.00	30.58	31.40	31.64	31.64
	FLC/AK0	1.71	2.13	2.64	3.30	4.13	5.18	6.45	8.14	10.24	12.78	16.03	19.96	24.39	28.38	31.12	31.57	31.57
	VLC/AK0	-0.05	0.09	0.31	0.65	1.24	2.09	3.30	4.98	7.23	10.01	13.49	17.60	22.71	27.99	30.61	31.54	31.54
	FLC/HD	-2.07	-1.52	-0.86	-0.04	0.98	2.18	3.64	5.47	7.70	10.45	13.71	17.78	22.41	27.04	30.76	31.54	31.54
VLC/HD	-2.83	-2.70	-2.54	-2.29	-1.93	-1.38	-0.60	0.58	2.25	4.53	7.71	11.78	17.12	22.52	28.45	31.01	31.01	
$B=32$	FLC (FBA)/AKI	6.81	7.85	9.12	10.43	11.94	13.65	15.40	17.60	19.83	22.43	25.17	28.04	30.17	31.27	31.60	31.66	
	FLC (FA)/AKI	4.59	5.41	6.40	7.54	8.89	10.34	12.09	14.23	16.45	19.16	22.32	25.73	28.98	30.79	31.51	31.65	
	VLC/AKI	1.02	1.60	2.42	3.59	5.06	6.90	9.10	11.62	14.38	17.55	21.17	24.67	28.40	30.71	31.55	31.67	
	FLC/AK0	1.71	2.11	2.66	3.30	4.15	5.18	6.48	8.14	10.23	12.78	16.12	19.99	24.44	28.56	31.11	31.59	31.59
	VLC/AK0	-0.01	0.10	0.26	0.52	0.91	1.51	2.47	3.89	5.96	8.84	12.31	16.57	21.28	26.69	30.35	31.17	31.17
	FLC/HD	-2.08	-1.54	-0.84	-0.03	0.97	2.17	3.66	5.47	7.72	10.34	13.76	17.81	22.66	27.45	31.00	31.56	31.56
VLC/HD	-2.86	-2.79	-2.64	-2.47	-2.16	-1.72	-1.07	-0.03	1.47	3.54	6.70	10.66	15.51	21.90	29.02	31.58	31.58	
$B=64$	FLC (FBA)/AKI	6.99	8.06	9.31	10.66	12.20	13.75	15.64	17.68	19.99	22.59	25.39	28.11	30.18	31.25	31.63	31.67	
	FLC (FA)/AKI	4.70	5.57	6.58	7.73	9.05	10.59	12.36	14.44	16.76	19.39	22.54	25.79	28.91	31.01	31.57	31.66	
	VLC/AKI	0.92	1.47	2.24	3.29	4.76	6.66	8.88	11.41	14.29	17.38	21.09	24.91	28.56	30.77	31.45	31.66	
	FLC/AK0	1.71	2.12	2.65	3.32	4.13	5.17	6.52	8.17	10.22	12.84	16.08	19.83	24.36	28.67	31.17	31.53	31.53
	VLC/AK0	0.03	0.11	0.23	0.43	0.73	1.21	1.92	3.08	4.88	7.49	11.10	15.59	20.92	26.21	30.39	31.56	31.56
	FLC/HD	-2.09	-1.52	-0.85	-0.02	0.98	2.16	3.70	5.49	7.70	10.37	13.82	17.65	22.48	27.26	30.72	31.39	31.39
VLC/HD	-2.90	-2.83	-2.71	-2.54	-2.30	-1.90	-1.33	-0.41	0.91	2.99	5.77	9.83	14.75	20.41	27.82	31.51	31.51	
$B=128$	FLC (FBA)/AKI	7.06	8.19	9.44	10.76	12.28	13.88	15.71	17.79	20.12	22.65	25.38	28.21	30.26	31.28	31.61	31.66	
	FLC (FA)/AKI	4.77	5.65	6.68	7.84	9.19	10.70	12.46	14.51	16.97	19.48	22.77	26.06	29.07	30.93	31.59	31.66	
	VLC/AKI	0.88	1.37	2.16	3.16	4.59	6.40	8.60	11.16	14.03	17.26	20.93	25.00	28.49	30.86	31.40	31.65	
	FLC/AK0	1.70	2.12	2.65	3.32	4.13	5.20	6.48	8.18	10.24	12.87	16.05	19.92	24.40	28.61	31.22	31.57	
	VLC/AK0	0.06	0.12	0.23	0.40	0.65	1.07	1.68	2.64	4.10	6.41	10.00	14.69	20.38	26.26	30.22	31.64	31.64
	FLC/HD	-2.09	-1.53	-0.86	-0.01	0.96	2.20	3.69	5.50	7.69	10.43	13.72	17.86	22.41	27.25	31.11	31.54	31.54
VLC/HD	-2.91	-2.85	-2.75	-2.59	-2.38	-2.05	-1.52	-0.73	0.50	2.42	5.28	8.89	14.80	19.95	27.29	30.07	30.07	

Table E.14: Simulation results for an $M=6$ bit quantized Gaussian AR(1) process with different block lengths B (part 2) (Fig. 4.7).

Chapter 5: Improving HE-AAC by Soft-Decision Decoding

Parameters	Method	E_b/N_0 (dB)										
		0	1	2	3	4	5	6	7	8	9	10
global gain	AK1	9.12	10.32	11.93	13.53	14.41	16.22	17.09	17.56	17.70	17.73	17.74
	AK0	7.57	8.63	10.28	11.90	13.26	15.27	16.38	17.48	17.69	17.72	17.74
	HD	-18.32	-16.78	-15.16	-7.97	-7.70	5.22	10.37	15.58	17.47	17.60	17.74
	EC	0	0	0	0	0	0	0	0	0	0	0
scale factors	AK1	7.32	7.79	8.76	10.01	12.21	14.40	16.76	17.36	17.72	17.72	17.74
	AK0	6.96	7.30	8.23	9.37	11.81	14.16	16.61	17.31	17.71	17.72	17.74
	HD	-5.95	-4.20	-1.63	1.55	5.84	8.86	13.64	16.05	17.20	17.51	17.74
	EC	0	0	0	0	0	0	0	0	0	0	0

Table E.15: Global SNR (dB) results for HE-AAC transmission over an AWGN channel, with the individual parameter global gain or scale factors being distorted (Fig. 5.10).

Test Conditions			file										
			1	2	3	4	5	6	7	8	9	10	all
reference			100.00	98.69	99.50	99.63	99.88	99.63	98.25	100.00	95.31	98.94	98.98
	mid anchor		59.31	62.81	85.88	65.00	62.88	68.63	57.25	68.81	65.44	67.06	66.31
	low anchor		45.13	47.06	53.50	43.19	44.44	46.38	43.88	49.13	43.94	45.50	46.21
two parameters	$E_b/N_0 _{\text{bad}} = 6.5$ dB	SD	73.06	94.88	63.06	88.50	90.06	67.75	84.56	64.44	91.38	81.88	79.96
		HD	45.69	88.81	14.81	71.44	56.25	27.94	49.00	30.69	51.25	58.13	49.40
corrupted	$E_b/N_0 _{\text{bad}} = 5$ dB	SD	67.44	46.13	41.13	54.44	53.31	40.88	56.94	40.63	48.75	58.81	50.84
		HD	22.38	18.56	14.31	23.25	8.88	7.75	21.25	14.69	11.13	24.31	16.65
three parameters	$E_b/N_0 _{\text{bad}} = 6.5$ dB	SD	37.56	31.69	37.94	22.38	53.94	36.56	70.75	28.50	47.44	58.50	42.52
		HD	22.19	29.38	12.81	22.94	37.13	25.38	43.81	13.63	39.81	48.75	29.58
corrupted	$E_b/N_0 _{\text{bad}} = 5$ dB	SD	18.19	13.69	11.69	11.88	12.56	9.25	19.88	11.75	11.94	21.31	14.21
		HD	7.31	9.06	7.75	9.63	6.25	3.06	11.25	7.25	6.69	9.81	7.81

Table E.16: MUSHRA results of the mean score (Fig. 5.11).

Test Conditions			file										
			1	2	3	4	5	6	7	8	9	10	all
reference			0	1.27	0.76	0.53	0.25	0.73	2.01	0	4.20	1.43	0.54
	mid anchor		7.15	6.59	9.96	7.66	8.25	9.77	8.29	9.56	8.81	5.62	2.79
	low anchor		8.28	8.95	9.29	7.84	7.76	7.92	8.18	9.72	9.22	7.32	2.65
two parameters	$E_b/N_0 _{\text{bad}} = 6.5$ dB	SD	11.57	2.57	12.81	6.52	5.51	10.38	9.52	11.59	7.05	9.81	3.34
		HD	11.34	6.09	7.24	12.76	13.85	8.83	13.95	11.79	11.56	11.41	4.68
corrupted	$E_b/N_0 _{\text{bad}} = 5$ dB	SD	12.89	9.19	11.47	11.67	11.47	12.95	12.09	11.37	12.29	10.92	3.82
		HD	12.86	6.42	5.46	8.14	3.74	3.25	8.75	5.42	12.70	11.49	2.52
three parameters	$E_b/N_0 _{\text{bad}} = 6.5$ dB	SD	13.01	10.36	12.45	11.77	11.02	12.23	9.97	11.01	13.03	8.85	4.15
		HD	6.96	10.39	5.85	11.04	9.49	8.62	14.06	7.00	11.86	10.04	3.53
corrupted	$E_b/N_0 _{\text{bad}} = 5$ dB	SD	6.27	4.70	3.84	4.41	4.18	4.81	7.86	4.38	4.05	7.70	1.77
		HD	3.65	4.22	4.09	4.21	3.43	2.07	5.32	3.26	3.66	5.15	1.28

Table E.17: MUSHRA results of the associate 95% confidence interval (δ_{IF} and δ in Section 5.5.1) (Fig. 5.11).

**Chapter 6: Low-Rate Fixed-Length Hard- and
Soft-Decision Decoding With Time-Variant Codebooks**

Parameters		Method	E_b/N_0										
			-3	-2	-1	0	1	2	3	4	5	6	7
$M = 1$ bit $\hat{\sigma}_\epsilon^2 = 1$	AK1	proposed	2.20	2.48	2.77	3.05	3.34	3.61	3.86	4.07	4.24	4.36	4.45
		standard	2.52	2.80	3.08	3.32	3.57	3.79	3.98	4.12	4.23	4.30	4.35
	AK0	proposed	1.34	1.60	1.90	2.22	2.58	2.95	3.33	3.67	3.99	4.21	4.38
		standard	1.32	1.59	1.88	2.19	2.55	2.92	3.29	3.61	3.90	4.11	4.26
	HD	proposed	1.60	1.88	2.18	2.49	2.84	3.18	3.53	3.82	4.09	4.27	4.41
		standard	0.42	0.77	1.16	1.56	2.01	2.48	2.96	3.37	3.75	4.02	4.22
$M = 1$ bit $\hat{\sigma}_\epsilon^2 = 2.02$	AK1	proposed	2.76	3.10	3.44	3.76	4.10	4.41	4.68	4.90	5.09	5.21	5.29
		standard	2.52	2.80	3.08	3.32	3.57	3.79	3.98	4.12	4.23	4.30	4.35
	AK0	proposed	1.73	2.07	2.45	2.84	3.28	3.73	4.17	4.54	4.86	5.08	5.23
		standard	1.32	1.59	1.88	2.19	2.55	2.92	3.29	3.61	3.90	4.11	4.26
	HD	proposed	1.59	1.94	2.33	2.72	3.16	3.61	4.05	4.44	4.79	5.03	5.21
		standard	0.42	0.77	1.16	1.56	2.01	2.48	2.96	3.37	3.75	4.02	4.22

Table E.18: Simulation results for $M = 1$ bit quantized Gaussian AR(1) samples with correlation coefficient $\rho = 0.9$ and $\hat{a} = \rho$ (Fig. 6.6).

Parameters		Method	E_b/N_0										
			0	1	2	3	4	5	6	7	8	9	10
$M = 2$ bit $\hat{\sigma}_{\text{opt}}^2 = 1.57$	AK1	proposed	3.99	4.60	5.28	5.99	6.72	7.48	8.15	8.79	9.32	9.72	9.96
		standard	3.92	4.50	5.13	5.78	6.44	7.11	7.68	8.20	8.62	8.92	9.10
	AK0	proposed	1.89	2.32	2.83	3.45	4.18	5.02	5.96	6.96	8.01	8.86	9.51
		standard	1.58	1.95	2.40	2.94	3.59	4.35	5.20	6.12	7.11	7.93	8.58
	HD	proposed	0.58	1.10	1.70	2.42	3.22	4.16	5.20	6.32	7.50	8.51	9.32
		standard	-0.47	0.02	0.60	1.29	2.08	3.01	4.04	5.18	6.39	7.45	8.32
$M = 3$ bit $\hat{\sigma}_{\text{opt}}^2 = 1.35$	AK1	proposed	4.51	5.27	6.18	7.18	8.24	9.39	10.64	11.76	12.88	13.78	14.43
		standard	4.50	5.26	6.16	7.14	8.17	9.30	10.50	11.56	12.59	13.40	13.98
	AK0	proposed	1.72	2.13	2.65	3.28	4.07	5.04	6.26	7.66	9.26	11.02	12.65
		standard	1.57	1.95	2.43	3.03	3.77	4.70	5.85	7.20	8.76	10.48	12.09
	HD	proposed	-0.75	-0.22	0.43	1.20	2.12	3.22	4.56	6.12	7.90	9.89	11.86
		standard	-1.43	-0.91	-0.27	0.48	1.39	2.48	3.81	5.37	7.15	9.16	11.17

Table E.19: Simulation results for $M = 2, 3$ bit quantized Gaussian AR(1) samples with correlation coefficient $\rho = 0.9$ and $\hat{a} = \rho$ (Fig. 6.6).

List of Symbols

$()^T$	Transpose
α	Forward probability
\hat{a}	Receiver-sided predictor coefficient
a	Predictor coefficient
β	Backward probability
BER	Bit error probability
$\overline{\text{BER}}$	Average bit error probability
B	Number of scale factor bands/block length for scale factors (VLCs)
b	Scale factor band index ($b \in \mathcal{B} = \{0, 1, \dots, B - 1\}$)
B'	Block length for FLCs
B_c	Block length for quantized spectral coefficients
B_{\max}	Maximum number of scale factor bands
\hat{c}	Received (real-valued) symbol
c	Transmitted modulated symbol
C, C_1, C_2, C_3, C'_1	Normalization constant
ΔR	Residual redundancy
ΔR_c	Utilizable, correlation-dependent residual redundancy
ΔR_d	Utilizable, distribution-dependent residual redundancy
δ	Confidence interval for subjective listening test
Δ_{\min}	Half the quantizer step size of the 14 bit analogue-to-digital converter
$\hat{d}l$	Received normalized difference signal
$\hat{d}l^+$	Receiver-sided predicted normalized difference signal
$\hat{d}q$	New quantized difference signal
$\hat{d}sf$	Received differential scale factor
$\tilde{d}l$	Unquantized normalized difference signal
D	Distortion
$D_b _{\min}$	Minimum distortion
dq	Quantized difference signal
$dqtl$	Truncated quantized difference signal
ds	Difference signal
dsf	Differential scale factor
$[d_i, d_{i+1}]$	Quantization interval
η	LSF submatrix index ($\eta \in \{1, 2, \dots, 5\}$)
$\bar{\eta}$	LSF submatrix indices excluding η
η'	ISF subvector index ($\eta' \in \{1, 2, \dots, 7\}$)

$\bar{\eta}'$	ISF subvector indices excluding η'
\hat{e}	Receiver-sided prediction error
\hat{e}_l	Error signal between \hat{d}_l and \hat{d}_l^+
\tilde{e}	Gaussian innovation (transmitter-sided prediction error)
e^{LMQ}	LMQ quantization error
en	Scale factor band energy
$\text{erf}(\cdot)$	Error function
$\exp(\cdot)$	Exponential function
$E(\cdot)$	Expectation value
E_b	Signal energy per bit
E_s	Signal energy per symbol
f	Quantization codebook index
f_s	Sampling rate of SBR
\hat{g}	Received global gain
g	Global gain
G_p	Prediction gain
\mathbf{h}	Predictor coefficients in a transposed vector manner
h	Quantization codebook index
$H(\cdot)$	Entropy
\hat{i}	Received quantization codebook index
\hat{i}_c	Received Huffman codebook index for quantized spectral coefficients
\hat{i}_s	Received Huffman codebook index for differential scale factors
i	Quantization codebook index
i'	Quantization codebook index
$I^{(i)}$	Quantization interval for quantization index i
i_F	Test file index ($i_F = \{1, 2, \dots, N_F\}$)
i_L	Listener index ($i_L = \{1, 2, \dots, N_L\}$)
i_c	Huffman codebook index for quantized spectral coefficients
i_g	Huffman codebook index for global gain
i_s	Huffman codebook index for differential scale factors
ih	Quantization codebook index for higher subband signal
il	Quantization codebook index for lower subband signal
itl	Quantization codebook index for truncated lower subband signal
j	Quantization codebook index
j'	Quantization codebook index
j_c	Column index of the VL/HD decoder-sided lookup table
j_r	Row index of the VL/HD decoder-sided lookup table
k	Spectral coefficient index
K_s	Total number of spectral coefficients in one section
$k_{\max,b}$	Last spectral coefficient index in scale factor band b
$k_{\min,b}$	First spectral coefficient index in scale factor band b
ℓ	Frame index
Λ	Number of different-length codewords
λ	Global subframe index
l	Global symbol index in new trellis representation ($l = \{1, 2, \dots, B_c\}$)

l''	Intermediate global symbol index
$L(\cdot)$	Log-likelihood ratios
l_c	Local symbol index in new trellis representation ($l_c = \{1, 2, \dots, 2 \cdot B_c\}$)
l_{ds}	Symbol index boundary between the diverging and the stationary stage for quantized spectral coefficients
l_{sc}	Symbol index boundary between the stationary and the converging stage for quantized spectral coefficients
LUT	Lookup table
$\hat{\mu}_{\hat{\epsilon}}$	Mean value of the prediction error
μ	State at previous symbol time $n - 1$ ($q_{n-1} = \mu \in \mathcal{Q}_{n-1}$)
\bar{M}	Average Huffman codeword length
M	Quantization bit rate
m	Bit index in bit combination
ν	State at current symbol time n ($q_n = \nu \in \mathcal{Q}_n$)
ν_{end}	Ending state at current symbol time n
ν_{start}	Starting state at current symbol time n
N	Markov process order
N'	FL codeword length
N'_{max}	Maximum length of the Huffman codewords used in the trellis representation
N'_{min}	Minimum length of the Huffman codewords used in the trellis representation
n, n'	Time index
$N^{(i)}$	VL codeword length for quantization codebook index i
N_0	Noise power spectral density
n_{ds}	Time index boundary between the diverging and the stationary stage
N_F	Number of test files
N_L	Number of listeners
n_{sc}	Time index boundary between the stationary and converging stage
n_p	Number of bits for <code>escape_prefix</code> in the bit sequence <code>escape_sequence</code>
N_{max}	Maximum length of all the Huffman codewords
N_{min}	Minimum length of all the Huffman codewords
N_p	Predictor order/autoregressive process order
nb	Bit demand
nl	Estimated number of spectral lines
ω	State at next symbol time $n + 1$ ($q_{n+1} = \omega \in \mathcal{Q}_{n+1}$)
o	White Gaussian noise signal
$P(\cdot)$	Probabilities
$p(\cdot)$	Probability density function (PDF)
P_B	State probabilities for bad state in a GE channel
P_G	State probabilities for good state in a GE channel
P	Transition probability from good to bad channel in a GE channel
Q	Transition probability from bad to good channel in a GE channel
q	State in trellis representation
q_{max}	Maximum number of states in a trellis representation

ρ	Correlation coefficient
R	Number of all bits in the block for VLCs
r	Quantization codebook index
R'	Number of all bits in the block for FLCs
R''	Number of bits for the intermediate global symbol index l''
R_c	Number of all bits in one section for quantized spectral coefficients
R_s	Number of all bits in one frame for scale factors
σ^2	Variance
$\widehat{sf}g$	Received gain value
\widehat{sf}	Received scale factor
\widehat{sr}	New reconstructed signal
s	Speech signal/Windowed input audio sequence
se	Estimated speech signal
sf	Scale factor
sfg	Gain value
sh	Higher subband signal
sl	Lower subband signal
sr	Reconstructed signal
srh	Higher subband reconstructed signal
srl	Lower subband reconstructed signal
$srtl$	Truncated lower subband reconstructed signal
ST	Sign table
$\bar{\tau}$	Mean MUSHRA score
τ	MUSHRA score
θ	Threshold
t	Tuple size
T_1	Nearest integer to the pitch delay from the previous odd subframe
\hat{u}	Received unquantized sample/symbol/parameter
\hat{u}^+	Predicted sample at the receiver side
\tilde{u}	Unquantized sample/symbol/parameter
\tilde{u}^+	Predicted sample at the transmitter side
u	Quantized sample/symbol/parameter
$\varphi_{\tilde{u}\tilde{u}}$	Auto-correlation function of \tilde{u}
W	Logarithmic scaling factors multipliers
$\hat{\mathbf{x}}$	Received bit combination
\hat{X}'_k	Received inversely quantized spectral coefficients
$\hat{x}(m)$	Received hard-decided m th bit of bit combination $\hat{\mathbf{x}}$
\mathbf{x}	Transmitted bit combination
$\mathbf{x}^{(i)}$	Transmitted bit combination for quantization index i
\hat{X}_k	Unquantized spectral coefficients
ξ	Tuning parameter
X'_k	Inversely quantized spectral coefficients
$x(m)$	Transmitted hard-decided m th bit of bit combination $\hat{\mathbf{x}}$
X_k	Quantized spectral coefficients
$\bar{\mathbf{y}}$	Bipolar representation of bit combination \mathbf{y}

\bar{y}_1^R	A stream of bipolar modulation symbols
\hat{y}_1^R	Received hard-decided bipolar bit combination
\hat{y}	Received bit combination
y	Transmitted bit combination
y_1^R	Transmitted unipolar bit stream
yl	Scaling factor in log domain
yp	Scaling factor in linear domain
z	VL/HD decoder lookup table entry

List of Abbreviations

AAC	Advanced audio coding
ACELP	Algebraic code excited linear prediction
ACR	Absolute category rating
ADPCM	Adaptive Differential Pulse Code Modulation
AK0	Zeroth-order <i>a priori</i> knowledge
AK1	First-order <i>a priori</i> knowledge
AMR-NB	Adaptive Multi-Rate Narrowband
AMR-WB	Adaptive Multi-Rate Wideband
APB	Adaptive prediction backwards
APPs	<i>A posteriori</i> probabilities
AQB	Adaptive quantization backwards
AR	Autoregressive
ARMA	Autoregressive moving-average
AWGN	Additive white Gaussian noise
BFI	Bad frame indicator
BM	Bit mapping
BM ⁻¹	Inverse bit mapping
BPSK	Binary phase-shift keying
CAT-1q	Cordless advanced technology-internet and quality
CCR	Comparison category rating
CQS	Continuous quality scale
CRC	Cyclic redundancy check
DCR	Degradation category rating
DECT	Digital enhanced cordless telephony
EFR	Enhanced Fullrate
EVS	Enhanced Voice Services
FEC	Forward error correction
FIR	Finite impulse response
FLCs	Fixed-length codes
GE	Gilbert-Elliott
GSM	Global System for Mobile Communications
HD	Hard-decision
HE-AAC	High-Efficiency Advanced Audio Coding
ICC	Inter channel coherence
ICS	Individual channel stream

IID	Inter intensity difference
IMDCT	Inverse modified discrete cosine transform
ISFs	Immittance spectral frequencies
IP	Internet protocol
LLRs	Log-likelihood ratios
LMQ	Lloyd-Max quantization
LP	Linear prediction
LSFs	Line spectral frequencies
LTE	Long-Term Evolution
MA	Moving-average
MAP	Maximum <i>a posteriori</i>
MDCT	Modified discrete cosine transform
MMSE	Minimum mean-square error
MOPS	Million operations per second
MOS	Mean opinion score
MPEG	Moving Picture Experts Group
MSVQ	Multistage vector quantization
MUSHRA	MUlti Stimulus test with Hidden Reference and Anchor
NBC	Natural binary code
NLMS	Normalized least-mean-squares
ODG	Objective difference grade
PDF	Probability density function
PEAQ	Perceptual evaluation of audio quality
PESQ	Perceptual evaluation of speech quality
POLQA	Perceptual objective listening quality assessment
PS	Parametric stereo
QMFs	Quadrature mirror filters
RAM	Random-access memory
ROM	Read-only memory
SBR	Spectral band replication
SD	Soft-decision
SMQ	Split matrix quantization
SNR	Signal-to-noise ratio
SVQ	Split vector quantization
TNS	Temporal noise shaping
UEP	Unequal error protection
UMTS	Universal Mobile Telecommunications System
USAC	Unified Speech and Audio Coding
VAD	voice activity detector
VLCs	Variable-length codes
VoIP	Voice over IP
VQ gain	Vector-quantized codebook gain

Bibliography

- [3GPP, 2004a] 3GPP, “3GPP TS 26.401, General Audio Codec Audio Processing Functions; Enhanced aacPlus General Audio Codec; General Description,” 3GPP, May 2004.
- [3GPP, 1999a] —, “Mandatory Speech Codec Speech Processing Functions: AMR Speech Codec; Transcoding Functions (3GPP TS 26.090),” 3GPP; TSG-SA, Dec. 1999.
- [3GPP, 1999b] —, “Mandatory Speech Codec Speech Processing Functions: AMR Speech Codec; Error Concealment of Lost Frames (3GPP TS 26.091),” 3GPP; TSG-SA, Dec. 1999.
- [3GPP, 2001b] —, “Speech Codec Speech Processing Functions: Adaptive Multi-Rate-Wideband (AMR-WB) Speech Codec; Transcoding Functions (3GPP TS 26.190),” 3GPP; TSG-SA, Dec. 2001.
- [3GPP, 2001a] —, “Adaptive Multi-Rate - Wideband (AMR-WB) Speech Codec; Error Concealment of Erroneous or Lost Frames; (3GPP TS 26.191),” 3GPP; TSG-SA, Mar. 2001.
- [3GPP, 2004d] —, “3GPP TS 26.402, General Audio Codec Audio Processing Functions; Enhanced aacPlus General Audio Codec; Additional Decoder Tools,” 3GPP, May 2004.
- [3GPP, 2004e] —, “3GPP TS 26.403, General Audio Codec Audio Processing Functions; Enhanced aacPlus General Audio Codec; Encoder Specification; Advanced Audio Coding (AAC) Part,” 3GPP, May 2004.
- [3GPP, 2004b] —, “3GPP TS 26.404, General Audio Codec Audio Processing Functions; Enhanced aacPlus General Audio Codec; Enhanced aacPlus Encoder Spectral Band Replication (SBR),” 3GPP, Sep. 2004.
- [3GPP, 2004c] —, “3GPP TS 26.405, General Audio Codec Audio Processing Functions; Enhanced aacPlus General Audio Codec; Encoder Specification Parametric Stereo Part,” 3GPP, Sep. 2004.
- [3GPP, 2008] —, “3GPP TS 26.410, General Audio Codec Audio Processing Functions; Enhanced aacPlus General Audio Codec; Floating-Point ANSI-C Code,” 3GPP, Dec. 2008.

- [3GPP, 2014a] —, “3GPP TS 26.445, Codec for Enhanced Voice Services (EVS); Detailed Algorithmic Description,” 3GPP, Sep. 2014.
- [3GPP, 2014b] —, “3GPP TS 26.447, Codec for Enhanced Voice Services (EVS); Error Concealment of Lost Packets,” 3GPP, Sep. 2014.
- [A. Papoulis, 2002] S. P. A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 4th ed. New York, NY, USA: McGraw-Hill, 2002.
- [Adrat et al., 2000] M. Adrat, J. Spittka, S. Heinen, and P. Vary, “Error Concealment by Near Optimum MMSE-Estimation of Source Codec Parameters,” in *Proc. of IEEE Workshop on Speech Coding*, 2000, pp. 84–86.
- [Arnstein, 1975] D. Arnstein, “Quantization Error in Predictive Coders,” *IEEE Transactions on Communications*, vol. 23, no. 4, pp. 423–429, Apr. 1975.
- [Bahl et al., 1974] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate,” *IEEE Transactions on Information Theory*, vol. 20, pp. 284–287, Mar. 1974.
- [Balakirsky, 1997] V. Balakirsky, “Joint Source-Channel Coding with Variable Length Codes,” in *Proc. of IEEE International Symposium on Information Theory*, Ulm, Germany, Jun. 1997, p. 419.
- [Bauer and Hagenauer, 2001] R. Bauer and J. Hagenauer, “On Variable Length Codes for Iterative Source/Channel Decoding,” in *Proc. of Data Compression Conference*, Snowbird, UT, USA, Mar. 2001, pp. 273–282.
- [Bergeron and Lamy-Bergot, 2004] C. Bergeron and C. Lamy-Bergot, “Soft-Input Decoding of Variable-Length Codes Applied to the H.264 Standard,” in *Proc. of IEEE Workshop on Multimedia Signal Processing*, Siena, Italy, Sep. 2004, pp. 87–90.
- [Bossert, 1999] M. Bossert, *Channel Coding for Telecommunications*. New York, NY, USA: John Wiley & Sons, Inc., 1999.
- [Chang and Gray, 1986] P.-C. Chang and R. Gray, “Gradient Algorithms for Designing Predictive Vector Quantizers,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 4, pp. 679–690, Aug. 1986.
- [Chen and Gersho, 1995] J. Chen and A. Gersho, “Adaptive Postfiltering for Quality Enhancement of Coded Speech,” *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 59–71, Jan. 1995.
- [Cheng et al., 2002] S. Cheng, H. Yu, and Z. Xiong, “Error Concealment of MPEG-2 AAC Audio Using Modulo Watermarks,” in *Proc. of IEEE International Symposium on Circuits and Systems*, vol. 2, Scottsdale, AZ, USA, May 2002, pp. II-261–II-264.
- [Cummiskey et al., 1973] P. Cummiskey, N. S. Jayant, and J. L. Flanagan, “Adaptive Quantization in Differential PCM Coding of Speech,” *Bell System Technical Journal*, vol. 52, no. 7, pp. 1105–1118, Sep. 1973.

- [Cuperman et al., 1994] V. Cuperman, F.-H. Liu, and P. Ho, “Robust Vector Quantization for Noisy Channels Using Soft Decision and Sequential Decoding,” *European Transactions on Telecommunications*, vol. 5, no. 5, pp. 7–18, Sep. 1994.
- [Demir and Sayood, 1998] N. Demir and K. Sayood, “Joint Source/Channel Coding for Variable Length Codes,” in *Proc. of Data Compression Conference*, Snowbird, UT, USA, Mar. 1998, pp. 139–148.
- [Derrien et al., 2008] O. Derrien, M. Kieffer, and P. Duhamel, “Joint Source/Channel Decoding of Scalefactors in MPEG-AAC Encoded Bitstreams,” in *Proc. of EUSIPCO*, Lausanne, Switzerland, Aug. 2008, pp. 1–5.
- [EBU, 2003] EBU, “EBU Subjective Listening Tests on Low-Bitrate Audio Codecs,” EBU, Jun. 2003.
- [Elias, 1955] P. Elias, “Predictive Coding–I,” *IRE Transactions on Information Theory*, vol. 1, no. 1, pp. 16–24, Mar. 1955.
- [Ericsson, 2014] Ericsson, “Ericsson White Paper: Evolved HD Voice for LTE,” Oct. 2014.
- [ETSI, 1992b] ETSI, “ETSI EN 300 175, Digital Enhanced Cordless Telecommunications Common Interface,” ETSI, 1992.
- [ETSI, 1992a] —, “Substitution and Muting of Lost Frames for Full Rate Speech Traffic Channels (GSM 06.11),” ETSI TC-SMG, Feb. 1992.
- [ETSI, 1995] —, “European Digital Cellular Telecommunications System Half Rate Speech Part 3: Substitution and Muting of Lost Frames for Half Rate Speech Traffic Channels (GSM 06.21),” ETSI TM/TM5/TCH-HS, Jan. 1995.
- [ETSI, 1996] —, “Digital Cellular Telecommunications System: Substitution and Muting of Lost Frames for Enhanced Full Rate (EFR) Speech Traffic Channels (GSM 06.61),” ETSI TC-SMG, Feb. 1996.
- [ETSI, 2007a] —, “ETSI TS 102 527, Digital Enhanced Cordless Telecommunications New Generation DECT,” ETSI, 2007.
- [ETSI, 2007b] —, “ETSI TS 102 570, Digital Enhanced Cordless Telecommunications New Generation DECT, Overview and Requirements,” ETSI, 2007.
- [Eureka-147, 1997] Eureka-147, “Eureka-147-Digital Audio Broadcasting,” <http://www.teilar.gr/dbData/ProfAnn/profann-1a29fba4.pdf>, Aug. 1997.
- [Fazel and Fuja, 2003] T. Fazel and T. Fuja, “Robust Transmission of MELP-Compressed Speech: An Illustrative Example of Joint Source-Channel Decoding,” *IEEE Transactions on Communications*, vol. 51, no. 6, pp. 973–982, Jun. 2003.
- [Fingscheidt, 2008] T. Fingscheidt, “Parameter Models and Estimators in Soft Decision Source Decoding,” in *Advances in Digital Speech Transmission*, R. Martin, U. Heute, and C. Antweiler, Eds. West Sussex, England: John Wiley & Sons, Ltd, 2008, pp. 281–310.

- [Fingscheidt, 1998] —, “Softbit-Sprachdecodierung in digitalen Mobilfunksystemen,” Ph.D. dissertation, Aachener Beiträge zu digitalen Nachrichtensystemen, Hrsg. P. Vary, Band 9, 1998.
- [Fingscheidt, 2003] —, “Graceful Degradation in ADPCM Speech Transmission,” in *Proc. of DAGA*, Aachen, Germany, Mar. 2003, pp. 748–749.
- [Fingscheidt and Vary, 1996] T. Fingscheidt and P. Vary, “Error Concealment by Softbit Speech Decoding,” in *Proc. of ITG-Fachtagung “Sprachkommunikation”*, Frankfurt a.M., Germany, Sep. 1996, pp. 7–10.
- [Fingscheidt and Vary, 2001] —, “Softbit Speech Decoding: A New Approach to Error Concealment,” *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 3, pp. 240–251, Mar. 2001.
- [Fraunhofer-Gesellschaft, 2013] Fraunhofer-Gesellschaft, “White Paper: The AAC Audio Coding Family for Broadcast and Cable TV,” Fraunhofer Institute for Integrated Circuits IIS, Jul. 2013.
- [Fresia and Caire, 2010] M. Fresia and G. Caire, “A Linear Encoding Approach to Index Assignment in Lossy Source-Channel Coding,” *IEEE Transactions on Information Theory*, vol. 56, no. 3, pp. 1322–1344, Mar. 2010.
- [Gersho and Gray, 1992] A. Gersho and R. Gray, *Vector Quantization and Signal Compression*. Springer Science + Business Media, 1992.
- [Gilbert, 1960] E. N. Gilbert, “Capacity of a Burst-Noise Channel,” *Bell System Technical Journal*, vol. 39, no. 5, pp. 1253–1265, Sep. 1960.
- [Gray and Neuhoff, 1998] R. Gray and D. Neuhoff, “Quantization,” *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2325–2383, Oct. 1998.
- [Hagenauer, 1995] J. Hagenauer, “Source-Controlled Channel Decoding,” *IEEE Transactions on Communications*, vol. 43, no. 9, pp. 2449–2457, Sep. 1995.
- [Hagenauer, 1980] —, “Viterbi Decoding of Convolutional Codes for Fading- and Burst-Channels,” in *Proc. of 1980 Zurich Seminar on Digital Communications*, Zürich, Switzerland, 1980, pp. G2.1–G2.7.
- [Hagenauer and Hoehner, 1989] J. Hagenauer and P. Hoehner, “A Viterbi Algorithm with Soft-Decision Outputs and its Applications,” in *Proc. of GLOBECOM*, Dallas, TX, USA, Nov. 1989, pp. 1680–1686.
- [Han and Fingscheidt, 2015a] S. Han and T. Fingscheidt, “Robust MPEG-4 High-Efficiency AAC With Fixed- and Variable-Length Soft-Decision Decoding,” in *Proc. of AES 2015*, New York, USA, May 2015, pp. 1–10.
- [Han and Fingscheidt, 2014b] —, “Improving Scalar Quantization for Correlated Processes Using Adaptive Codebooks Only At the Receiver,” in *Proc. of EUSIPCO*, Lisbon, Portugal, Sep. 2014, pp. 386–390.

- [Han and Fingscheidt, 2014c] —, “Scalar Quantization With Optimized Receiver-Sided Adaptive Codebook Reconstruction Levels Controlled by a Predictor,” in *Proc. of 11th ITG Conference on Speech Communication*, Erlangen, Germany, Sep. 2014, pp. 47–50.
- [Han and Fingscheidt, 2015c] —, “Lloyd-Max Quantization of Correlated Processes: How to Obtain Gains by Receiver-Sided Time-Variant Codebooks,” in *Proc. of 10th International ITG Conference on Systems, Communications and Coding*, Hamburg, Germany, Feb. 2015, pp. 1–6.
- [Han and Fingscheidt, 2015b] —, “An Improved ADPCM Decoder by Adaptively Controlled Quantization Interval Centroids,” in *Proc. of EUSIPCO*, Nice, France, Sep. 2015, pp. 335–339.
- [Han and Fingscheidt, 2014a] —, “Variable-Length Versus Fixed-Length Coding: On Tradeoffs for Soft-Decision Decoding,” in *Proc. of ICASSP*, Florence, Italy, May 2014, pp. 4269–4273.
- [Han et al., 2013] S. Han, F. Pflug, and T. Fingscheidt, “Improved AMR Wideband Error Concealment for Mobile Communications,” in *Proc. of EUSIPCO*, Marrakech, Morocco, Sep. 2013, pp. 1–5.
- [Haykin, 2002] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 2002.
- [HEAD acoustics GmbH, 2012] HEAD acoustics GmbH, “POLQA Application Guide,” HEAD acoustics documentation, https://www.head-acoustics.de/downloads/eng/application_notes/telecom/Appl_Guide.POLQA_e0.pdf, Mar. 2012.
- [Herre and Dietz, 2008] J. Herre and M. Dietz, “MPEG-4 High-Efficiency AAC Coding [Standards in a Nutshell],” *IEEE Signal Processing Magazine*, vol. 25, no. 3, pp. 137–142, May 2008.
- [Hoeg and Lauterbach, 2009] W. Hoeg and T. Lauterbach, *Digital Audio Broadcasting, Principles and Applications of DAB, DAB+ and DMB*. West Sussex, United Kingdom: Wiley, 2009.
- [Huang and Schultheiss, 1963] J. Huang and P. Schultheiss, “Block Quantization of Correlated Gaussian Random Variables,” *IEEE Transactions on Communications Systems*, vol. 11, no. 3, pp. 289–296, Sep. 1963.
- [Huber and Ruppel, 1990] J. Huber and A. Ruppel, “Zuverlässigkeitsschätzung für die Ausgangssymbole von Trellis-Decodern,” *AEÜ* (in German), vol. 44, no. 1, pp. 8–21, Jan. 1990.
- [Huffman, 1952] D. Huffman, “A Method for the Construction of Minimum-Redundancy Codes,” *Proceedings IRE*, vol. 40, no. 9, pp. 1098–1101, Sep. 1952.

- [Hui and Neuhoff, 2001] D. Hui and D. Neuhoff, “Asymptotic Analysis of Optimal Fixed-Rate Uniform Scalar Quantization,” *IEEE Transactions on Information Theory*, vol. 47, no. 3, pp. 957–977, Mar. 2001.
- [Ion and Hüb-Umbach, 2006] V. Ion and R. Hüb-Umbach, “Uncertainty Decoding for Distributed Speech Recognition over Error-Prone Networks,” *Speech Communication*, vol. 48, no. 11, pp. 1435–1446, Nov. 2006.
- [ISO/IEC, 2004] ISO/IEC, “ISO/IEC 13818-7: Information Technology – Generic Coding of Moving Pictures and Associated Audio Information – Part 7: Advanced Audio Coding (AAC),” ISO/IEC, Oct. 2004.
- [ISO/IEC, 2005] —, “ISO/IEC 14496-3:2005, Information Technology – Coding of Audio-Visual Objects – Part 3: Audio,” ISO/IEC, Dec. 2005.
- [ISO/IEC, 2012] —, “ISO/IEC 23003-3:2012, Information Technology – MPEG Audio Technologies – Part 3: Unified Speech and Audio Coding,” ISO/IEC, Apr. 2012.
- [ITU-R, 2001] ITU-R, “Recommendation ITU-R BS.1387-1, Method for Objective Measurements of Perceived Audio Quality,” ITU, Nov. 2001.
- [ITU-R, 2014] —, “Recommendation ITU-R BS.1534-2, Method for the Subjective Assessment of Intermediate Quality Level of Audio Systems,” ITU, Jun. 2014.
- [ITU-R, 2003] —, “Recommendation ITU-R BS.1284-1, General Methods for the Subjective Assessment of Sound Quality,” ITU, Dec. 2003.
- [ITU-T, 2005] ITU-T, “ITU-T Recommendation G.191, Software Tools for Speech and Audio Coding Standardization,” ITU, Sep. 2005.
- [ITU-T, 1999] —, “ITU-T Recommendation G.711 Appendix I, A High Quality Low-Complexity Algorithm for Packet Loss Concealment with G.711,” ITU-T, Sep. 1999.
- [ITU-T, 1972] —, “ITU-T Recommendation G.711, Pulse Code Modulation (PCM) of Voice Frequencies,” ITU-T, 1972.
- [ITU-T, 2008] —, “ITU-T Recommendation G.718, Frame Error Robust Narrow-Band and Wideband Embedded Variable Bit-Rate Coding of Speech and Audio From 8-32 kbit/s,” ITU-T, Jun. 2008.
- [ITU-T, 2006] —, “ITU-T Recommendation G.722 Appendix III, A High-Quality Packet Loss Concealment Algorithm for G.722,” ITU-T, Nov. 2006.
- [ITU-T, 1988] —, “ITU-T Recommendation G.722, 7 kHz Audio-Coding Within 64 kbit/s,” ITU-T, Nov. 1988.
- [ITU-T, 1990] —, “ITU-T Recommendation G.726, 40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM),” ITU-T, Aug. 1990.
- [ITU-T, 1996] —, “ITU-T Recommendation P.800, Methods for Subjective Determination of Transmission Quality,” ITU, Aug. 1996.

- [ITU-T, 2001] —, “ITU-T Recommendation P.862, Perceptual Evaluation of Speech Quality (PESQ),” ITU, Feb. 2001.
- [ITU-T, 2007] —, “ITU-T Recommendation P.862.2, Wideband Extension to Recommendation P.862 for the Assessment of Wideband Telephone Networks and Speech Codecs,” ITU-T, Nov. 2007.
- [ITU-T, 2011] —, “ITU-T Recommendation P.863, Perceptual Objective Listening Quality Assessment,” ITU, Jan. 2011.
- [Jayant, 1974] N. Jayant, “Digital Coding of Speech Waveforms: PCM, DPCM, and DM Quantizers,” *Proc. of the IEEE*, vol. 62, no. 5, pp. 611–632, May 1974.
- [Jayant and Noll, 1984] N. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1984.
- [Jelinek et al., 2009] M. Jelinek, T. Vaillancourt, and J. Gibbs, “G.718: A New Embedded Speech and Audio Coding Standard With High Resilience to Error-Prone Transmission Channels,” *IEEE Communications Magazine*, vol. 47, no. 10, pp. 117–123, October 2009.
- [Kleijn and Hagen, 1996] W. Kleijn and R. Hagen, “On Memoryless Quantization in Speech Coding,” *IEEE Signal Processing Letters*, vol. 3, no. 8, pp. 228–230, Aug. 1996.
- [Kliewer and Thobaben, 2005] J. Kliewer and R. Thobaben, “Iterative Joint Source-Channel Decoding of Variable-Length Codes Using Residual Source Redundancy,” *IEEE Transactions on Wireless Communications*, vol. 4, no. 3, pp. 919–929, Sep. 2005.
- [Kozamernik, 1995] F. Kozamernik, “Digital Audio Broadcasting—Radio Now and for the Future,” *EBU Technical Review*, vol. 265, 1995.
- [Lahouti and Khandani, 2007] F. Lahouti and A. K. Khandani, “Soft Reconstruction of Speech in the Presence of Noise and Packet Loss,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 44–56, Jan. 2007.
- [Lecomte et al., 2015a] J. Lecomte, A. Tomasek, G. Markovic, M. Schnabel, K. Tsutsumi, and K. Kikuri, “Enhanced Time Domain Packet Loss Concealment in Switched Speech/Audio Codec,” in *Proc. of ICASSP*, Brisbane, Australia, Apr. 2015, pp. 5922–5926.
- [Lecomte et al., 2015b] J. Lecomte, T. Vaillancourt, S. Bruhn, H. Sung, K. Peng, K. Kikuri, B. Wang, S. Subasingha, and J. Faure, “Packet-Loss Concealment Technology Advances in EVS,” in *Proc. of ICASSP*, Brisbane, Australia, Apr. 2015, pp. 5708–5712.
- [Lee, 1999] C. Lee, “An Enhanced ADPCM Coder for Voice Over Packet Networks,” *International Journal of Speech Technology*, vol. 2, no. 4, pp. 343–357, May 1999.

- [Lee et al., 2005] C. Lee, M. Kieffer, and P. Duhamel, “Soft Decoding of VLC Encoded Data for Robust Transmission of Packetized Video,” in *Proc. of ICASSP*, vol. 3, Philadelphia, PA, USA, Mar. 2005, pp. III/737–III/740.
- [Linde et al., 1980] Y. Linde, A. Buzo, and R. Gray, “An Algorithm for Vector Quantizer Design,” *IEEE Transactions on Communications*, vol. 28, no. 1, pp. 84–95, Jan. 1980.
- [Lloyd, 1982] S. Lloyd, “Least Squares Quantization in PCM,” *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–136, Mar. 1982.
- [Lookabaugh et al., 1993] T. Lookabaugh, E. Riskin, P. Chou, and R. Gray, “Variable Rate Vector Quantization for Speech, Image, and Video Compression,” *IEEE Transactions on Communications*, vol. 41, no. 1, pp. 186–199, Jan. 1993.
- [López-Oller et al., 2016] D. López-Oller, S. Han, A. M. Gomez, J. L. Pérez-Córdoba, and T. Fingscheidt, “System-Compatible Robustness Improvement for New Generation DECT Decoders by G.722 Soft-Decision Decoding,” in *Proc. of ICASSP*, Shanghai, China, Mar. 2016, pp. 5945–5949.
- [Makhoul and Berouti, 1979] J. Makhoul and M. Berouti, “Adaptive Noise Spectral Shaping and Entropy Coding in Predictive Coding of Speech,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 27, no. 1, pp. 63–73, Feb. 1979.
- [Max, 1960] J. Max, “Quantizing for Minimum Distortion,” *IRE Transactions on Information Theory*, vol. 6, no. 1, pp. 7–12, Mar. 1960.
- [McDonald, 1966] R. A. McDonald, “Signal-to-Noise and Idle Channel Performance of Differential Pulse Code Modulation Systems — Particular Applications to Voice Signals,” *Bell System Technical Journal*, vol. 45, no. 7, pp. 1123–1151, Sep. 1966.
- [Melsa and Cohn, 1978] J. L. Melsa and D. L. Cohn, *Decision and Estimation Theory*. Tokyo, Japan: McGraw-Hill Kogakusha, 1978.
- [Murad and Fuja, 1998] A. Murad and T. Fuja, “Joint Source-Channel Decoding of Variable-Length Encoded Sources,” in *Proc. of Information Theory Workshop*, Killybeg, Ireland, Jun. 1998, pp. 94–95.
- [Nasruminallah and Hanzo, 2009] M. E. Nasruminallah and L. Hanzo, “Robust Transmission of H.264 Coded Video Using Three-Stage Iterative Joint Source and Channel Decoding,” in *Proc. of GLOBECOM*, Honolulu, HI, USA, Dec. 2009, pp. 1–6.
- [Neuendorf et al., 2013] M. Neuendorf, M. Multrus, N. Rettelbach, G. Fuchs, J. Robilliard, J. Lecomte, S. Wilde, S. Bayer, S. Disch, C. Helmrich, R. Lefebvre, P. Gournay, B. Besette, J. Lapierre, K. Kjörling, H. Purnhagen, L. Villemoes, W. Oomen, E. Schuijers, K. Kikuri, T. Chinen, T. Norimatsu, K. S. Chong, E. Oh, M. Kim, S. Quackenbush, and B. Grill, “The ISO/MPEG Unified Speech and Audio Coding Standard-Consistent High Quality for all Content Types and at all Bit Rates,” *Journal of the Audio Engineering Society*, vol. 61, no. 12, pp. 956–977, 2013.

- [Neyman, 1937] J. Neyman, “Outline of a Theory of Statistical Estimation Based on the Classical Theory of Probability,” *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 236, no. 767, pp. 333–380, Jun. 1937.
- [Nokia, 2015] Nokia, “Nokia Networks White Paper: The 3GPP Enhanced Voice Services (EVS) Codec,” networks.nokia.com, 2015.
- [NTT, 1994] NTT, “Multi-Lingual Speech Database for Telephonometry,” NTT-AT, 1994.
- [Oliver et al., 1948] B. Oliver, J. Pierce, and C. Shannon, “The Philosophy of PCM,” *Proceedings IRE*, vol. 36, no. 11, pp. 1324–1331, Nov. 1948.
- [Othman et al., 2005] N. S. Othman, S. X. Ng, and L. Hanzo, “Turbo-Detected Unequal Protection Audio and Speech Transceivers Using Serially Concatenated Convolutional Codes, Trellis Coded Modulation and Space-Time Trellis Coding,” in *IEEE VTC’05 (Fall)*, Dallas, Texas, USA, Sep. 2005, pp. 1044–1048.
- [Othman et al., 2007] N. S. Othman, M. El-Hajjar, O. Alamri, and L. Hanzo, “Soft-Bit Assisted Iterative AMR-WB Source-Decoding and Turbo-Detection of Channel-Coded Differential Space-Time Spreading Using Sphere Packing Modulation,” in *Proc. of IEEE Vehicular Technology Conference*, Dublin, Ireland, Apr. 2007, pp. 2010–2014.
- [Othman et al., 2009] N. S. Othman, M. El-Hajjar, O. Alamri, S. X. Ng, and L. Hanzo, “Iterative AMR-WB Source and Channel Decoding Using Differential Space-Time Spreading-Assisted Sphere-Packing Modulation,” *IEEE Transactions on Vehicular Technology*, vol. 58, no. 1, pp. 484–490, Jan. 2009.
- [P. Vary, 2005] R. M. P. Vary, *Digital Speech Transmission: Enhancement, Coding and Error Concealment*. West Sussex, England: Wiley, 2005.
- [Park and Miller, 2000] M. Park and D. Miller, “Joint Source-Channel Decoding for Variable-Length Encoded Data by Exact and Approximate MAP Sequence Estimation,” *IEEE Transactions on Communications*, vol. 48, no. 1, pp. 1–6, 2000.
- [Peinado et al., 2005] A. M. Peinado, V. Sánchez, J. L. Pérez-Córdoba, and A. J. Rubio, “Efficient MMSE-Based Channel Error Mitigation Techniques. Application to Distributed Speech Recognition Over Wireless Channels,” *IEEE Transactions on Wireless Communications*, vol. 4, no. 1, pp. 14–19, Jan. 2005.
- [Pflug, 2013] F. Pflug, “Funkübertragung von Audiosignalen mit prädiktiver Soft-Decision-Dekodierung,” Ph.D. dissertation, Mitteilung aus dem Institut für Nachrichtentechnik der Technischen Universität Braunschweig, Hrsg. T. Fingscheidt, Band 31, 2013.
- [Pflug and Fingscheidt, 2011a] F. Pflug and T. Fingscheidt, “Delayless Soft-Decision Decoding of High-Quality Audio Transmitted Over AWGN Channels,” in *Proc. of ICASSP*, Prague, Czech Republic, May 2011, pp. 489–492.

- [Pflug and Fingscheidt, 2013b] —, “Delayless Robust DPCM Audio Transmission for Digital Wireless Microphones,” in *Proc. of AES 2013*, Rome, Italy, May 2013, pp. 1–8.
- [Pflug and Fingscheidt, 2011b] —, “Delayless Soft-Decision Decoding of High-Quality Audio With Adaptively Shaped Priors,” in *Proc. of EUSIPCO*, Barcelona, Spain, Aug. 2011, pp. 2099–2103.
- [Pflug and Fingscheidt, 2012] —, “NLMS-Supported Decoding of High-Quality Speech for Burst Channels,” in *Proc. of 10th ITG Conference on Speech Communication*. Braunschweig, Germany: VDE-Verlag, Sep. 2012, pp. 211–214.
- [Pflug and Fingscheidt, 2013a] —, “Robust Ultra-Low Latency Soft-Decision Decoding of Linear PCM Audio,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 11, pp. 2324–2336, Nov 2013.
- [Proakis and Salehi, 2001] J. G. Proakis and M. Salehi, *Communication Systems Engineering*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2001.
- [Proakis, 2001] J. Proakis, *Digital Communications*. New York, NY, USA: McGraw-Hill, 2001.
- [Rainer and Hagenauer, 2000] B. Rainer and J. Hagenauer, “Symbol-by-Symbol MAP Decoding of Variable Length Codes,” in *Proc. of 3rd ITG Conference “Source and Channel Coding”*. Munich, Germany: VDE-Verlag, Jan. 2000, pp. 111–116.
- [Ramamoorthy et al., 1988] V. Ramamoorthy, N. Jayant, R. Cox, and M. Sondhi, “Enhancement of ADPCM Speech Coding with Backward-Adaptive Algorithms for Postfiltering and Noise Feedback,” *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 2, pp. 364–382, Feb. 1988.
- [Rawat et al., 2011] D. Rawat, S. Kumar, S. Nagaraj, and J. D. Matyjas, “Adaptive Modulation Approach for Robust MPEG-4 AAC Encoded Audio Transmission,” in *Military Communications Conference*, Nov 2011, pp. 561–565.
- [Roucos and Wilgus, 1985] S. Roucos and A. Wilgus, “High Quality Time-Scale Modification for Speech,” in *Proc. of ICASSP*, vol. 10, Tampa, FL, USA, Apr. 1985, pp. 493–496.
- [Salomon and Motta, 2010] D. Salomon and G. Motta, *Handbook of Data Compression*, 5th ed. New York, NY, USA: Springer, 2010.
- [Schuller et al., 2002] G. Schuller, B. Yu, D. Huang, and B. Edler, “Perceptual Audio Coding Using Adaptive Pre- and Post-filters and Lossless Compression,” *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 6, pp. 379–390, Sep. 2002.

- [Serizawa and Nozawa, 2002] M. Serizawa and Y. Nozawa, “A Packet Loss Concealment Method Using Pitch Waveform Repetition and Internal State Update on the Decoded Speech for the Sub-Band ADPCM Wideband Speech Codec,” in *Proc. of IEEE Workshop Speech Coding*, Tsukuba City, Japan, Oct. 2002, pp. 68–70.
- [Shannon, 1948] C. Shannon, “A Mathematical Theory of Communication,” *Bell Systems Technical Journal*, vol. 27, pp. 623–656, Oct. 1948.
- [Sperschneider, 2000] R. Sperschneider, “Error Resilient Source Coding with Variable Length Codes and Its Application to MPEG Advanced Audio Coding,” in *109th Audio Engineering Society Convention*. Los Angeles, CA, USA: Audio Engineering Society, Sep. 2000, pp. 5271–5288.
- [Thobaben and Kliever, 2005] R. Thobaben and J. Kliever, “Low-Complexity Iterative Joint Source-Channel Decoding for Variable-Length Encoded Markov Sources,” *IEEE Transactions on Communications*, vol. 53, no. 12, pp. 2054–2064, 2005.
- [Tian, 2015] C. Tian, “Concepts of Soft-Decision Decoding in the Adaptive Multirate (AMR) Speech Coder,” Master’s thesis, Technische Universität Braunschweig, Institute for Communications Technology, Nov. 2015.
- [Turin, 1998] W. Turin, *Digital Transmission Systems: Performance Analysis and Modeling*. McGraw-Hill, 1998.
- [Vincent, 2005] E. Vincent, “MUSHRAM: A MATLAB interface for MUSHRA listening tests,” <http://c4dm.eecs.qmul.ac.uk/downloads/#mushram>, Nov. 2005.
- [Wen and Villasenor, 1999] J. Wen and J. Villasenor, “Utilizing Soft Information in Decoding of Variable Length Codes,” in *Proc. of Data Compression Conference*, Snowbird, UT, USA, Mar. 1999, pp. 131–139.
- [Wohnort, 2013] Wohnort, “DAB Ensembles Worldwide,” <http://www.wohnort.org/DAB/australia.html#Melbourne>, Oct. 2013.
- [Xiang et al., 2003] W. Xiang, S. Pietrobon, and S. Barbulescu, “Iterative Decoding of JPEG Coded Images with Channel Coding,” in *Proc. of International Conference on Telecommunications*, vol. 2, Anchorage, AK, USA, May 2003, pp. 1356–1360.

Own Publications

S. Han, F. Pflug, and T. Fingscheidt, “Improved AMR Wideband Error Concealment for Mobile Communications,” in *Proc. of EUSIPCO*, Marrakech, Morocco, Sep. 2013, pp. 1–5.

S. Han and T. Fingscheidt, “Variable-Length Versus Fixed-Length Coding: On Tradeoffs for Soft-Decision Decoding,” in *Proc. of ICASSP*, Florence, Italy, May 2014, pp. 4269–4273.

S. Han and T. Fingscheidt, “Improving Scalar Quantization for Correlated Processes Using Adaptive Codebooks Only At the Receiver,” in *Proc. of EUSIPCO*, Lisbon, Portugal, Sep. 2014, pp. 386–390.

S. Han and T. Fingscheidt, “Scalar Quantization With Optimized Receiver-Sided Adaptive Codebook Reconstruction Levels Controlled by a Predictor,” in *Proc. of 11th ITG Conference on Speech Communication*, Erlangen, Germany, Sep. 2014, pp. 47–50.

S. Han and T. Fingscheidt, “Lloyd-Max Quantization of Correlated Processes: How to Obtain Gains by Receiver-Sided Time-Variant Codebooks,” in *Proc. of 10th International ITG Conference on Systems, Communications and Coding*, Hamburg, Germany, Feb. 2015, pp. 1–6.

S. Han and T. Fingscheidt, “An Improved ADPCM Decoder by Adaptively Controlled Quantization Interval Centroids,” in *Proc. of EUSIPCO*, Nice, France, Sep. 2015, pp. 335–339.

S. Han and T. Fingscheidt, “Robust MPEG-4 High-Efficiency AAC With Fixed- and Variable-Length Soft-Decision Decoding,” in *Proc. of AES 2015*, New York, USA, May 2015, pp. 1–10.

D. López-Oller, S. Han, A. M. Gomez, J. L. Pérez-Córdoba, and T. Fingscheidt, “System-Compatible Robustness Improvement for New Generation DECT Decoders by G.722 Soft-Decision Decoding,” in *Proc. of ICASSP*, Shanghai, China, Mar. 2016, pp. 5945–5949.

Z. Zhao, S. Han, and T. Fingscheidt, “Improving Vector Quantization-Based Decoders for Correlated Processes in Error-Free Transmission,” in *Proc. of 12th ITG Conference on Speech Communication*, Paderborn, Germany, Oct. 2016, pp. 70–74.