SINUSOIDAL CODING OF SPEECH FOR VOICE OVER IP

by

Yannis Agiomyrgiannakis

A dissertation submitted to the faculty of

University of Crete

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

University of Crete

January 2007

UNIVERSITY OF CRETE
GRADUATE COMMITTEE APPROVAL
of a dissertation submitted by
Yannis Agiomyrgiannakis

This dissertation has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

Feb 20, 2007
Date

Yannis Stylianou, Associate Professor, University Of Crete, Chair

Feb. 20, 2007
Date

Apostolos Traganitis, Prof., University Of Crete

Feb 20, 2007
Date

Panos Tsakalidis, Assoc. Prof., University Of Crete

Feb 14, 2007
Date

Georgios Tziritas, Prof., University Of Crete

Feb 21, 2007
Date

Vassilios Digalakis, Prof., Technical University Of Crete

Feb 21, 2007
Date

Aleksandros Potamianos, Assoc. Prof., Technical University Of Crete

Feb 2, 2007
Date

Gernot Kubin, Prof., Technical University of Graz, Austria

ABSTRACT

SINUSOIDAL CODING OF SPEECH FOR VOICE OVER IP

Yannis Agiomyrgiannakis

University Of Crete

Doctor of Philosophy

It is widely accepted that Voice-over-Internet-Protocol (VoIP) will dominate wireless and wireline voice communications in the near future. Traditionally, a minimum level of Quality-of-Service is achieved by careful traffic monitoring and network fine-tuning. However, this solution is not feasible when there is no possibility of controlling/monitoring the parameters of the network. For example, when speech traffic is routed through Internet there are increased packet losses due to network delays and the strict end-to-end delay requirements for voice communication. Most of today's speech codecs were not initially designed to cope with such conditions. One solution is to introduce channel coding at the expense of end-to-end delay. Another solution is to perform joint source/channel coding of speech by designing speech codecs which are natively robust to increased packet losses.

This thesis proposes a framework for developing speech codecs which are

robust to packet losses. The thesis addresses the problem in two levels: at the basic source/channel coding level where novel methods are proposed for introducing controlled redundancy into the bitstream, and at the signal representation/coding level where a novel speech parameterization/modelling is presented that is amenable to efficient quantization using the proposed source coding methods. The speech codec is designed to facilitate high-quality Packet Loss Concealment (PLC). The speech signal is modeled with harmonically related sinusoids; a representation that enables fine time-frequency resolution which is vital for high-quality PLC. Furthermore, each packet is encoded independently of the previous packets in order to avoid a desynchronization between the encoder and the decoder upon a packet loss. This allows some redundancy to exist in the bit-stream.

A number of contributions are made to well-known harmonic speech models. A fast analysis/synthesis method is proposed and used in the construction of an Analysis-by-Synthesis (AbS) pitch detector. Harmonic Codecs tend to rely on phase models for the reconstruction of the harmonic phases, introducing artifacts that effect the quality of the reconstructed speech signal. For a high-quality speech reconstruction, the quantization of phase is required. Unfortunately, phase quantization is not a trivial problem because phases are circular variables. A novel phase-quantization algorithm is proposed to address this problem. Harmonics phases are properly aligned and modeled with a Wrapped Gaussian Mixture Model (WGMM) capable of handling parameters that belong to circular spaces. The WGMM is estimated with a suitable Expectation-Maximization (EM) algorithm. Phases are then quantized by extending the efficient GMM-based quantization techniques for linear spaces to WGMM and circular spaces.

When packet losses are increased, additional redundancy can be introduced using Multiple Description Coding (MDC). In MDC, each frame is encoded in two descriptions; receiving both descriptions provides a high-quality reconstruction while receiving one description provides a lower-quality reconstruction. With current GMM-based MDC schemes it is possible to quantize the amplitudes of the harmonics which represent an important portion of the information of the speech signal. A novel WGMM-based MDC scheme is proposed and used for MDC of the harmonic phases. It is shown that it is possible to construct high-quality MDC codecs based on harmonic models. Furthermore, it is shown that the redundancy between the MDC descriptions can be used to "correct" bit errors that may have occurred during transmission.

At the source coding level, a scheme for *Multiple Description Transform Coding* (MDTC) of multivariate Gaussians using Parseval Frame expansions and a source coding technique referred to as *Conditional Vector Quantization* (CVQ), are proposed. The MDTC algorithm is extended to generic sources that can be modeled with GMM. The proposed frame facilitates a computationally efficient *Optimal Consistent Reconstruction* algorithm (OCR) and *Cooperative Encoding* (CE). In CE, the two MDTC encoders cooperate in order to provide better central/side distortion tradeoffs. The proposed scheme provides scalability, low complexity and storage requirements, excellent performance in low redundancies and competitive performance in high redundancies. In CVQ, the focus is given in correcting the most frequent type of errors; single and double packet losses. Furthermore, CVQ finds application to *BandWidth Expansion* (BWE), the extension of the bandwidth of narrowband speech to wideband.

Concluding, two *proof-of-concept* harmonic codecs are constructed, a single

description and a multiple description codec. Both codecs are narrowband, variable rate, similar to quality with the state-of-the-art iLBC (internet Low Bit-Rate Codec) under perfect channel conditions and better than iLBC when packet losses occur. The single description codec requires 14 kbps and it is capable of accepting 20% packet losses with minimal quality degradation while the multiple description codec operates at 21 kbps while it is capable of accepting 40% packet losses without significant quality degradation.

ΠΕΡΙΛΗΨΗ

ΗΜΙΤΟΝΟΕΙΔΗΣ ΚΩΔΙΚΟΠΟΙΗΣΗ ΣΗΜΑΤΩΝ ΦΩΝΗΣ ΓΙΑ ΜΕΤΑΔΟΣΗ ΜΕΣΩ ΔΙΚΤΥΩΝ IP

ΙΩΑΝΝΗΣ ΑΓΙΟΜΥΡΓΙΑΝΝΑΚΗΣ

ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

Είναι ευρέως αποδεκτό ότι η μετάδοση φωνής μέσω δικτύων IP θα κυριαρχήσει στις ενσύρματες και ασύρματες τηλεπικοινωνίες στο προσεχές μέλλον. Παραδοσιακά, ένα ελάχιστο επίπεδο ποιότητας επικοινωνίας διασφαλίζεται με προσεκτική παρακολούθηση και ρύθμιση της κίνησης του δικτύου. Μια τέτοια προσέγγιση όμως δεν είναι εφικτή όταν δεν υπάρχει η δυνατότητα ελέγχου ή/και προσαρμογής των παραμέτρων του δικτύου. Για παράδειγμα, όταν τα δεδομένα φωνής δρομολογούνται μέσω του Διαδικτύου, οι καθυστερήσεις που εισάγονται από το δίκτυο σε συνδυασμό με τις αυστηρές προδιαγραφές μέγιστης καθυστέρησης προκαλούν συνθήκες αυξημένης απώλειας πακέτων φωνής. Οι περισσότεροι κωδικοποιητές φωνής όμως δεν έχουν σχεδιαστεί να λειτουργούν υπό αυτές τις συνθήκες. Μια λύση είναι να εισαχθεί κωδικοποίηση καναλιού, εις βάρος όμως της καθυστέρησης

μετάδοσης των πακέτων. Μια άλλη λύση είναι να γίνει συνδυασμένη κωδικοποίηση πηγής/καναλιού για το σήμα της φωνής με κατάλληλο σχεδιασμό κωδικοποιητών φωνής οι οποίοι είναι εγγενώς ευσταθείς στις απώλειες πακέτων.

Στην εργασία αυτή προτείνεται ένα πλαίσιο ανάπτυξης κωδικοποιητών φωνής οι οποίοι είναι ευσταθείς σε απώλειες πακέτων. Το θέμα αντιμετωπίζεται σε δύο επίπεδα: στο βασικό επίπεδο κωδικοποίησης πηγής/καναλιού όπου προτείνονται νέες μέθοδοι εισαγωγής πλεονάζουσας πληροφορίας στα μεταδιδόμενα πακέτα καθώς και στο επίπεδο της αναπαράστασης/κωδικοποίησης της φωνής όπου προτείνεται μια παραμετροποίηση/μοντελοποίηση η οποία επιτρέπει την χρήση των προαναφερθέντων μεθόδων κωδικοποίησης πηγής/καναλιού. Ο κωδικοποιητής φωνής έχει σχεδιαστεί με γνώμονα την επίτευξη υψηλής ποιότητας αλγορίθμων απόκρυψης απώλειας πακέτων (Packet Loss Concealment (PLC). Το σήμα της φωνής έχει μοντελοποιηθεί ως άθροισμα αρμονικά σχετισμένων συνημίτονων, μια παραμετροποίηση η οποία επιτρέπει λεπτομερή χειρισμό τόσο στο πεδίο των συχνοτήτων όσο και στο πεδίο του χρόνου, ιδιότητα η οποία είναι ουσιώδης για την ανάπτυξη υψηλής ποιότητας αλγορίθμων PLC. Κάθε πακέτο φωνής κωδικοποιείται ανεξάρτητα από τα προηγούμενα πακέτα προκειμένου να αποφευχθεί ο αποσυγχρονισμός του κωδικοποιητή από τον αποκωδικοποιητή σε μια ενδεχόμενη απώλεια πακέτου. Αυτός ο χειρισμός επιτρέπει ένα φυσικό επίπεδο πλεονασμού πληροφορίας στην ροή δεδομένων φωνής.

Προτείνονται διάφορες συνεισφορές στα ευρέως διαδεδομένα αρμονικά μοντέλα φωνής. Ειδικότερα, προτείνεται μια γρήγορη μέθοδος ανάλυσης/σύνθεσης η οποία χρησιμοποιείται σε έναν πρωτότυπο αλγόριθμο εκτίμησης τονικότητας. Οι αρμονικοί κωδικοποιητές φωνής συνήθως βασίζονται σε μοντέλα φάσης για την ανακατασκευή των αρμονικών αποτυγχάνοντας όμως έτσι να επιτύχουν υψηλής ποιότητας ανακατασκευή φωνής. Μια υψηλής ποιότητας κωδικοποίηση χρειάζεται

τον κβαντισμό της φάσης των αρμονικών. Ο κβαντισμός φάσης όμως δεν είναι ένα τετριμμένο πρόβλημα γιατί η φάση είναι μια κυκλική μεταβλητή με modulo-2π συμπεριφορά. Για να αντιμετωπιστεί αυτό το πρόβλημα προτείνεται ένας ειδικά σχεδιασμένος αλγόριθμος κωδικοποίησης φάσης. Οι φάσεις των αρμονικών, μετά από κατάλληλη επεξεργασία, μοντελοποιούνται με ένα Wrapped Gaussian Mixture Model (WGMM). Η εκτίμηση του WGMM γίνεται με έναν αλγόριθμο Εκτίμησης-Μεγιστοποίησης (Expectation-Maximization). Οι φάσεις των αρμονικών κβαντίζονται με μια μέθοδο η οποία αποτελεί επέκταση των μεθόδων κβαντισμού μέσω GMM για γραμμικούς χώρους σε μεθόδους κβαντισμού μέσω WGMM για κυκλικούς χώρους.

Σε συνθήκες αυξημένης απώλειας πακέτων απαιτείται η εισαγωγή περισσότερης πλεονάζουσας πληροφορίας και αυτό μπορεί να επιτευχθεί με Κωδικοποίηση Πολλαπλών Περιγραφών (ΚΠΠ). Στην ΚΠΠ, κάθε τεμάχιο φωνής κωδικοποιείται σε δυο περιγραφές. Η λήψη και των δύο επιτρέπει μια υψηλής ποιότητας ανακατασκευή των δεδομένων ενώ η λήψη μονάχα ενός εκ των δύο παρέχει μια χαμηλότερης ποιότητας ανακατασκευή. Με τις υπάρχουσες μεθόδους ΚΠΠ μέσω GMM είναι δυνατόν να κωδικοποιηθούν τα πλάτη των αρμονικών τα οποία συνιστούν ένα μεγάλο μέρος της πληροφορίας του σήματος της φωνής. Για τις φάσεις προτείνεται μια μέθοδος ΚΠΠ μέσω WGMM. Με την χρήση της προτεινόμενης μεθόδου είναι πλέον δυνατόν να κατασκευαστούν υψηλής ποιότητας αρμονικοί κωδικοποιητές φωνής πολλαπλών περιγραφών. Ακόμη, η πλεονάζουσα πληροφορία μεταξύ των περιγραφών μπορεί να χρησιμοποιηθεί για την 'διόρθωση' σφαλμάτων σε επίπεδο bit που τυχόν προέκυψαν κατά την μετάδοση μιας περιγραφής.

Στο επίπεδο της κωδικοποίησης πηγής/καναλιού, προτείνεται μια μέθοδος Κωδικοποίησης Πολλαπλών Περιγραφών μέσω Μετασχηματισμού (ΚΠΠΜ) καθώς και μια τεχνική κωδικοποίησης πηγής που ονομάζεται Διανυσματικός Κβαντισμός

υπό Συνθήκη (ΔΚΣ). Η μέθοδος ΚΠΠΜ βασίζεται στα λεγόμενα Parseval Frames και αρχικά προτείνεται για πολυδιάστατες Κανονικές κατανομές ενώ εν' συνέχεια επεκτείνεται σε πηγές που μπορούν να μοντελοποιηθούν με GMM. Η προτεινόμενη μέθοδος έχει την δυνατότητα να χειριστεί μεγάλο αριθμό διαστάσεων, υψηλούς ρυθμούς μετάδοσης bit σε συνδυασμό με χαμηλή πολυπλοκότητα και απαιτήσεις σε μνήμη. Έχει άριστη απόδοση σε συνθήκες χαμηλού πλεονασμού και ανταγωνιστική απόδοση σε συνθήκες υψηλού πλεονασμού. Στη μέθοδο ΔΚΣ η εστίαση γίνεται στην διόρθωση των πιο συχνών τύπων λάθους, όπως τις μονές και τις διπλές απώλειες πακέτων. Ακόμη, η ΔΚΣ βρίσκει εφαρμογή στην επέκταση του φάσματος της φωνής από τα 0-4 kHz στα 4-8 kHz με ελάχιστη μετάδοση πληροφορίας.

Συνοψίζοντας, παρουσιάζονται δύο πρωτότυποι αρμονικοί κωδικοποιητές φωνής, μονής και διπλής περιγραφής. Αμφότεροι κωδικοποιούν τα πρώτα 0-4 kHz του φάσματος, λειτουργούν με μεταβαλλόμενο ρυθμό μετάδοσης και έχουν ποιότητα αντίστοιχη της ποιότητας του iLBC (internet Low Bitrate Codec) δίχως απώλεια πακέτων ενώ υπερτερούν του iLBC υπό συνθήκες απώλειας πακέτων. Ο κωδικο-ποιητής μονής περιγραφής χρειάζεται 13 kbps ενώ δέχεται 20% απώλειες με ελάχιστη υποβάθμιση της ποιότητας, ενώ ο κωδικοποιητής διπλής περιγραφής χρειάζεται 21 kbps ενώ μπορεί να δεχθεί απώλειες 40% δίχως σημαντική υποβάθμιση της ποιότητας.

Contents

List of Figures

List of Tables

List Of Publications

CONFERENCES

- Yannis Agiomyrgiannakis and Yannis Stylianou, "Combined Estimation/Coding of Highband Spectral Envelopes for Speech Spectrum Expansion", ICASSP, May 2004.
- Yannis Agiomyrgiannakis and Yannis Stylianou, "Coding with Side Information Techniques for LSF Reconstruction in Voice-over-IP", ICASSP, April 2005
- Miltiadis Vasilakis, Yannis Agiomyrgiannakis and Yannis Stylianou, "Fast Analysis/Synthesis of Harmonic Signals", ICASSP, May 2006.
- Yannis Agiomyrgiannakis and Yannis Stylianou, "Stochastic Modeling and quantization of harmonic phases in speech using Wrapped Gaussian Mixture Models", accepted in ICASSP 2007
- Athanasios Mouchtaris, Yannis Agiomyrgiannakis and Yannis Stylianou, "Conditional Vector Quantization for Voice Conversion", accepted in ICASSP 2007

JOURNALS

- Yannis Agiomyrgiannakis and Yannis Stylianou, "Conditional Vector Quantization for Speech Coding", accepted at IEEE Trans. Speech & Audio Processing.
- Yannis Agiomyrgiannakis and Bastiaan W. Kleijn, "Multiple Description Transform Coding of Gaussians using Parseval Frame Expansions", submitted to IEEE Transactions in Communications
- Yannis Agiomyrgiannakis and Yannis Stylianou, "Wrapped Gaussian Mixture Models for modeling and quantization of phase data from speech", submitted to IEEE Trans. Speech & Audio Processing
- Yannis Agiomyrgiannakis and Yannis Stylianou "GMM-based Multiple Description Transform Coding of Speech Spectral Envelopes", submitted to IEEE Trans. Speech & Audio Processing

Abbreviations

ACR: Absolute Category Rating

ADPCM: Adaptive PCM

AR: Auto-Regressive

BWE: Band-Width Expansion

CELP: Code Excited Linear Prediction

CSI: Coding with Side Information

CVQ: Conditional Vector Quantization

DCR: Degradation Category Rating

DMOS: Degradation Mean Opinion Score

DOF: Degrees Of Freedom

ECQ: Entropy Constrained Quantization

EM: Expectation-Maximization (algorithm)

ETSI: European Telecommunication Standards Institute

FEC: Forward Error Correction

GMM: Gaussian Mixture Model

GMMCF: GMM Conversion Function

GMM-MDSQ$_{TC}$: GMM-based Transform Coding using MDSQ

GMM-MDTC: GMM-based MDTC

GMM-MDTC$_{CE}$: GMM-based MDTC with Cooperative Encoding

HB: High-Band (4-8 kHz spectrum)

HM: Harmonic Model

HMC: Harmonic Model Codec

HMC-MD: Harmonic Model Codec - Multiple Description

HMC-SD: Harmonic Model Codec - Single Description

HMM: Hidden Markov Models

iLBC: internet Low Bitrate Codec

IP: Internet Protocol

LE: Linear Estimation

CF: Conversion Function

LP: Linear Prediction

LP-AS: Linear Prediction Analysis/Synthesis

LPC: Linear Prediction Coefficients

LSF: Line Spectrum Frequencies

MBE: Multi-Band Excitation

MDC: Multiple Description Coding

MDSQ: Multiple Description Scalar Quantization

MDSQ$_{TC}$: Transform Coding using MDSQ

MDTC: Multiple Description Transform Coding

MDTC$_{CE}$: Multiple Description Transform Coding with Cooperative Encoding

MMSE: Mininum MSE (solution)

MOS: Mean Opinion Score

MPLPC: Multi-Pulse Linear Prediction Coding

MRWSE: Mean-Root-Wrapped Square Error

MSE: Mean Square Error

NB: Narrow-Band (speech signal)

NLIVQ: Non-Linear Interpolative Vector Quantization

OCR: Optimal Consistent Reconstruction

OLA: OverLap-Add

PCF: Polynomial CodeFunctions (or Polynomial Codepoint generator Functions)

PESQ: Perceptual Evaluation of Speech Quality

PLC: Packet Loss Concealment

PSTN: Public Swith Telephone Network

RCC: Real Cepstrum Coefficients

RCQ: Resolution Constrained Quantization

REW: Rapidly Evolving Waveform

RS: Reed-Solomon (code)

SEW: Slowly Evolving Waveform

SKL: Symmetric Kullback Leibler (distance)

SLB: Shannon Lower Bound

SNR: Signal-to-Noise Ratio

STC: Sinusoidal Transform Codec

TIMIT: A DARPA database of phonetically balanced 0-8 kHz speech signals

VoIP: Voice-over-IP

VQ: Vector Quantization

WB: Wide-Band (speech signal)

WGMM: Wrapped Gaussian Mixture Model

WI: Waveform Interpolation

WSE: Wrapped-Square-Error

Chapter 1

Introduction

It is widely accepted that Voice-over-Internet-Protocol (VoIP) will dominate the wireless and wireline voice communication market in the near future. A large percentage of voice traffic in conventional telephony networks is already routed through private IP networks. As the bandwidth cost of IP networks decreases, the cost savings become substantial when voice is routed through IP networks. Routing voice traffic through the Internet is a possibility that has the potential of changing the landscape of telephony today. The traditional *"Messenger"* software has evolved from a text-based chatting tool to a cost-free voice communication terminal that is able to link continents. There is an increasing number of voice communication messengers that provide completely free PC-to-PC phone-calls, like *Skype*[1], *Google Talk*[2], *Yahoo Messenger Voice Chat*[3], *VoIPBuster* [4], *MSN Messenger*[5] and others. Most of these messengers offer the possibility of calling conventional and mobile telephony networks at a cost that is comparable to the cost of local phone-calls. Hardware makers are introducing smart phones that directly link to these VoIP messengers through a broadband Internet connection without the necessity of a PC. It seems that there is a momentum towards VoIP telephony.

The main obstacle in VoIP telephony arises from the fact that voice communication has strict end-to-end delay requirements while Internet provides no widely adopted mechanism for real-time communications. In Internet, a voice packet can be lost or delayed beyond its playback time, rendering it useless. Therefore, under the spurious network congestions and drop-outs that occur in a packet's path, speech codecs have to deal with increased packet losses. Unfortunately, most of the dominant speech codecs today cannot cope with increased packet loss conditions because they were developed for private over-provisioned networks where the Quality-of-Service was guarantied by the owner of the network. This has recently re-spurred the interest

[1] http://www.skype.com
[2] http://www.google.com/talk/
[3] http://messenger.yahoo.com/
[4] http://www.voipbuster.com/en/splash.html
[5] http://get.live.com/messenger/overview

in speech coding; particularly in speech coding that is robust to packet losses. This problem usually identified as "Speech Coding for VoIP" can also be formally stated as "joint source/channel coding of speech" as opposed to the traditional "speech coding" problem.

1.1 Quality-Of-Service in VoIP

Before getting into the details of VoIP Quality-of-Service (QoS), it is important to clarify the notion of quality in speech communication and to provide a small overview of the factors that affect QoS.

1.1.1 Measuring Speech Quality

The quality of speech can be measured with subjective tests where listeners rate the quality of the communication. There are two categories of subjective tests: the *conversational tests* where two people talk to each other and rate the quality of the conversation, and the *listening tests* where a single person listens to speech signals processed by the transmission system and rates them for their quality. Conversational tests are expensive and time-consuming, but evaluate the effect of factors that listening tests cannot address, like the end-to-end delay and interactivity. Listening tests are easier to devise and they are widely used in speech quality assessment.

There are several types of listening tests, depending on the purpose of the experiment. The most common type of subjective test is the Absolute Category Rating (ACR) test, where the listeners are asked to evaluate the quality of speech stimuli according to the 5-point scale presented in Table 1.1. The sample average of the ACR test is often referred to as *Mean Opinion Score* (MOS). The perception of quality depends on many factors (for example, listening conditions, stimuli, persona) differs from listener to listener, and MOS ratings are usually not directly comparable. A comparison though can be made using reference stimuli which are speech utterances processed by standardized speech codecs or a Modulated Noise Reference Unit (MNRU) at 5, 10, 15, 25 dB [1] (pg. 484).

Another type of subjective testing, the Degradation Category Rating (DCR) test, is more suitable for coding purposes. In DCR, the listeners rate the degradation of a speech signal according to a reference signal, using the scale presented in Table 1.2. The DCR test is more sensitive to degradations introduced by the speech codec and doesn't need the evaluation of reference stimuli. The sample average of DCR is sometimes referred to as *Degradation Mean Opinion Score* DMOS [1] (pg. 477).

More systematic feedback to the codec designer is provided by other types of subjective tests. The most common are the Diagnostic Acceptability Measure (DAM) which measures the difference in quality between two signals [2], the Diagnostic Rhyme Test (DRT) test which evaluates the intelligibility of speech [3] and the MUltiple Stimuli with Hidden Reference and Anchor (MUSHRA) test that is suitable for

medium-rate speech and audio coding systems [4].

The subjective quality can also be evaluated using sophisticated algorithms that predict objectively the MOS score using psychoacoustic criteria. The most widely used algorithm is the *Perceptual Evaluation of Speech Quality* (PESQ) which has been standardized at the ITU Recommendation P.862 [5].

Description	Rating
Excellent	5
Good	4
Average	3
Poor	2
Bad	1

Table 1.1 ACR test scale.

Description	Rating
Degradation is not perceived	5
Degradation is perceived but not annoying	4
Degradation is slightly annoying	3
Degradation is annoying	2
Degradation is very annoying	1

Table 1.2 DCR test scale.

1.1.2 Factors that affect Quality-of-Service

The QoS in VoIP is determined by several factors which are related to the communication system and to the network conditions.

System-related Degradations

The speech codec is a vital part of a VoIP system because it defines an upper bound to the quality of speech when there are no other degradations. The robustness of the speech codec to packet losses, tandeming, transcoding, echo and other degradations has a major impact to the QoS in VoIP. Furthermore, the speech codec defines the bandwidth of the speech signal that is transmitted. The bandwidth of the speech signal extends up to 12 kHz and the corresponding frequency bands are not equally important. Most of the speech codecs encode only *narrowband speech* which corresponds to the first 4 kHz of the spectrum, but narrowband speech has a "muffled" character while the first 8 kHz of the spectrum are necessary for a high quality transmission of speech [6]. Therefore, the trend currently is towards wideband speech codecs which encode the first 8 kHz of the spectrum.

A speech codec is traditionally evaluated in terms of subjective quality (i.e. MOS score) and bitrate. Depending on the application, other criteria may also become important like the computational *complexity* of the codec which is vital for low-power-consumption portable devices, the *algorithmic* and the *encoding delay* that influences the total end-to-end delay and the quality of the conversation, the *sensitivity* of the codec to bit-errors which is important for wireless applications, the ability of the codec to transmit DTMF (Dual Tone Multi-Frequency) signaling tones [7], and the quality of audio encoding.

Tandeming and *transcoding* are two sources of quality degradation that occur when speech is encoded with one codec, decoded, and then recoded with the same or another codec, respectively. This situation is common when a phone-call is directed through two or more different networks that use different codecs, for example from a GSM mobile network [8] to a CDMA2000 mobile network [9].

Another source of degradation is caused by an *echo effect* where an attenuated version of the speech signal returns back to the original speaker after some time which is called *echo path delay*. One source of echo is the acoustic feedback from the speaker to the microphone, while another source of echo is often caused by a mismatched hybrid (2-to-4 wire) convertor on the analog part of the telephony connection [10]. Echo can cause a serious degradation to the perceived quality of speech and its suppression is the subject of specialized *echo cancelation* algorithms [11], [12].

Finally, the quality of speech communication is affected by the characteristics of the terminals, like the frequency response of the microphone and the speakers, the background noise of the environment and the size of the so-called *jitter buffer*. The jitter buffer is a buffer that is used to compensate the various delays and potential reorderings of the packets that may occur in the packet network. The size of the jitter buffer is usually adapted according to the network conditions. A large jitter buffer reduces the packet loss rate because most of the packet losses are caused by packets that are delayed beyond playback time, but it also increases the end-to-end delay. Advanced VoIP systems offer adaptive jitter buffer resizing by making time-scale modifications to the speech signal during playback [13], [14], [15]. Time-scaling allows rapid adaptation to changing channel conditions, therefore, a good tradeoff between delay impairments and packet losses [16].

Network-related Degradations

The network introduces delay to the transmission of the packets. ITU (International Telecommunication Union) studies recommend a maximum end-to-end delay of 150 ms [17] when there is no echo. Delays over 150 ms are perceived as an impairment for highly interactive conversations. For delays above 300 ms, there is a noticeable degradation to the quality of the conversation [17], [18], and the speakers tend to engage into double talking and mutual silence. But, in normal conversation talks, delays of 400-500 ms can be tolerated without significant degradation [19]. However, large delays make echo control harder and echo-related degradations more

evident [16].

In IP networks, a packet can be lost, damaged or delayed beyond its playback time. In either case, the packet is considered to be lost and the decoder uses a *Packet Loss Concealment* (PLC) algorithm to fill the gap of the lost speech samples. The PLC algorithm is usually tightly integrated with the speech codec. The impact of a packet loss and the effectiveness of the concealment depends heavily on the speech codec, because in some codecs the loss may cause a desynchronization between the encoder and the decoder, with catastrophic result to the quality of speech.

Measurements have shown that packet losses in IP networks have a bursty nature. Thus, it is more probable to loose a packet after a packet loss than after a packet arrival [20], [21], [22]. The statistics of the loss process can be captured by Markov Models [22], [21]. The second order Markov Model (also referred to as the *Gilbert-Elliot model*) provides a good compromise between simplicity and effectiveness, and it is frequently used for theoretical analysis of the tradeoffs that arise between quality, redundancy and loss rate [23], [24]. The channels where a packet is either lost or received are referred to as *Erasure Channels*.

Concluding, the QoS in VoIP depends on many factors which are interrelated. An attempt to model the relationship between subjective quality and all these parameters is made with the ITU E-model [25]. The E-model provides the means for a systematic approach to the study and the fine-tuning of VoIP systems [24].

1.2 Speech Coding

Speech coding is the process of reducing the bitrate of digital speech signals. The bitrate reduction is achieved by minimizing the transmission of redundant and irrelevant information that exists in the speech signal. Redundant information is the information that exists between correlated parameters of a representation of the speech signal. Irrelevant information is the information that is not perceptually important. The typical speech coding process splits the waveform of speech in small intervals of 10 ms to 30 ms called *frames* and encodes one or two consecutive frames into a single packet.

The continuous speech signal is digitized through the process of sampling and quantization. Sampling rates of 8 kHz and 16 kHz are commonly used. The speech signals with bandwidth less than 4 kHz are called *narrowband* (NB) while the speech signals with bandwidth around 7-8 kHz are called *wideband* (WB). The speech samples are digitized using 16-bit uniform quantization, a representation that is called *Pulse Code Modulation* (PCM). The rate can be reduced to 8 bits/sample (64 kbps for narrowband speech) with *companded PCM* which uses non-uniform quantization to reduce the expected error, (ITU Recommendation G.711) [26]. Further rate reduction can be obtained with *Adaptive Differential PCM* (ADPCM, ITU Recommendation G.726 [27]) which takes into account the correlations between the speech samples in time. ADPCM quantizes narrowband speech with rates between 16 kbps and 40 kbps.

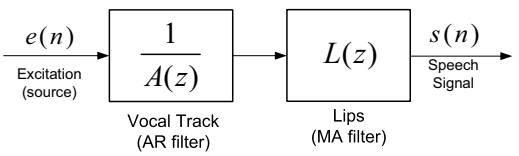


Figure 1.1 Source-Filter model of the speech signal

Speech codecs encode PCM quantized speech to more compact representations, suitable for transmission over wireline and wireless networks. Speech codecs can be classified into two broad categories: the *parametric codecs* and the *waveform approximating codecs*. Parametric codecs use a model of the speech production process and quantize its parametric representation. They are very efficient in lower bit-rates but cannot provide high-quality speech in higher bit-rates. Waveform approximating codecs produce a reconstructed signal that converges to the original waveform with increasing bit-rate. PCM, companded PCM and ADPCM encodings belong to the class of waveform approximating codecs, since they encode the speech waveform sample-by-sample.

1.2.1 Waveform-approximating codecs

Sophisticated waveform-approximating algorithms use a source-filter representation to decompose speech in two parts: an "*excitation signal*" (source) that roughly corresponds to the output of the vocal chords, an AR (Auto-Regressive) filter that roughly resembles the effect of the vocal tract to the excitation signal and a differentiator $L(z)$ that models the effect of the lips. The source-filter model of speech production is illustrated in Figure 1.1. An important class of waveform-approximating codecs is the *Linear-Prediction based Analysis-by-Synthesis* (LP-AS) algorithms which compute the AR filter using *Linear Prediction* (LP) methods [28] and quantize the excitation by minimizing a perceptually weighted mean square error between the original waveform and the reconstructed waveform [29].

LP-AS codecs have proven to be quite successful in providing medium to high-quality speech at rates between 5.3 kbps and 16 kbps for narrowband speech and 6 kbps to 23.85 kbps for wideband speech. LP-AS codecs differ mainly on the way that they encode the excitation. In *Multi-Pulse LPC* (MP-LPC) and *Code-Excited Linear Prediction* (CELP) the excitation is whitened with a filter that removes pitch-related inter-sample correlations and quantized using two codebooks; an *adaptive excitation* consisting of past-frame excitation signals and an *innovative excitation* consisting of pulse sequences. In *Multi-Pulse LPC* (MP-LPC), the innovative excitation is encoded with a series of pulses. The location and the gain of the pulses is explicitly quantized. The ITU G.723.1 codec [30] (narrowband, 5.3 kbps to 6.3 kbps) and the ETSI (European Telecommunication Standards Institute) GSM FR (GSM Full-Rate) codec are two standardized MP-LPC codecs. In CELP codecs, the innovative excitation is constructed using a specially designed codebook with pulses. CELP

technology has proven to be quite successful and it is used in most standardized and widely adopted speech codecs, like the ITU G.729 [31] codec, the ETSI GSM EFR (Enhanced Full Rate) codec [32], the AMR [33] codec and the wideband AMR-WB codec [34].

The dominating CELP codecs are also used for VoIP applications. However, in VoIP, the nature of the IP networks and the strict end-to-end delay requirements for communication result to increased packet losses. These codecs were not initially designed to cope with such conditions. For example, G.729 degrades rapidly from a 3.9 MOS score at 0% packet losses to a 2.75 MOS score at 5% packet losses [35]. AMR falls from a PESQ-MOS score of 3.98 at 0% packet losses to 3.31 at 2.3% packet losses [36]. In general, the speech quality of CELP codecs is unacceptable for packet losses higher than 3%. CELP codecs were designed to operate in a circuit-switched manner mainly for wireless communications where each phone-call occupies pre-allocated channel bandwidth. Therefore, the focus was given to reduce the bit-rate and to introduce channel coding capable of correcting bit-errors that may occur in a wireless transmission. Packet losses occurred only due to excessive bit-errors. The voice stream was transmitted through a private over-provisioned network that assured the QoS; the fact that the end-to-end delay requirements and the bit-error-rate specifications were met. However, strong control of the network conditions is not always possible in IP networks.

The rapid quality degradation that CELP codecs suffer upon packet losses can be attributed to the desynchronization that occurs between the encoder and the decoder. The desynchronization of the decoder buffers damages the adaptive excitation codebook for several subsequent frames. A number of enhancements have been proposed within the CELP framework. In [37], [38], for example, the AMR-WB encoder is modified in order to reduce the contribution of the adaptive excitation codebook to the decoded excitation, biasing the decoder towards a faster recovery at the expense of coding efficiency. In [39], late frames (frames arriving after playback time) are used for faster resynchronization of the AMR decoder. However, a single packet loss can still cause severe degradation. This has led some researchers to develop iLBC (*internet Low Bitrate Codec*), an LP-AS codec that encodes each packet independently of the other (previous) packets [40]. iLBC completely avoids the desynchronization at the cost of a higher bitrate: 13.33 kbps for the 30 ms version and 15.2 kbps for the 20 ms version for quality equivalent to the 8 kbps G.729.

iLBC is more robust than CELP codecs because it does not remove inter-packet redundancy. The excessive redundancy can also be introduced with channel coding techniques over the highly efficient CELP encodings. In fact, experimental evaluations in [25], as well as theoretical evaluations based on the E-model in [35], report that G.729 encodings with redundancy introduced via *Forward Error Correction* (FEC) techniques outperform iLBC. FEC techniques introduce redundancy to the bitstream in order to compensate packet losses. The simplest form of FEC is to repeat the information of a packet to the next packet. The comparison however does not take into account the effect of an improved PLC algorithm. Packet independent coding

facilitates the design of highly efficient PLC algorithms which are fine-tuned to the specifics of the speech signal. Furthermore, redundant side-information bitstreams can provide a substantial improvement to PLC, as *Global IP Sound* (GIPS) claims that it has achieved with the GIPS RCU (Redundant Coding Unit) [41]. The combination of iLBC 13.33 kbps with the 1.33 kbps RCU bitstream provides a relatively high MOS score of 3.4 for 15% packet losses. Clearly, the dilemma between efficient source coding of speech in combination with channel coding and redundant source coding with fine-tuned PLC is not an easy one to answer and further investigation is required.

1.2.2 Parametric codecs

Parametric codecs use a model of the speech signal and quantize the parameters of the model. The perceptual quality of parametric codecs is bounded by the intrinsic quality of the model. Parametric codecs outperform CELP codecs at low bit rates below 4.8 kbps, but cannot match the quality of CELP technology at higher rates above 8 kbps. Therefore, the application of parametric codecs is usually limited to satellite and military communications where a minimal payload is protected with strong channel coding. Two important classes of codecs can be identified as parametric: the LP-based codecs where a sequence of pulses resembling the glottal excitation is fed into an LP filter and the sinusoidal codecs. The military 2.4 kbps FS1015 codec [42] and its successor, the 2.4 kbps MELP (*Mixed Excitation Linear Prediction*) [43] are two standardized codecs of the first class. These codecs provide intelligible speech (useful in battlefields) but their perceptual quality is low for commercial applications.

Sinusoidal codecs model the speech signal $x(n)$ with a series of harmonically related oscillators of fundamental frequency ω_0, according to formula

$$\hat{x}(n) = \sum_{k=1}^{K} A_k \cos(k\omega_0 n + \phi_k), \tag{1.1}$$

where K is the number of harmonics, A_k and ϕ_k are the amplitudes and the phases of the harmonically related sinusoids. The most notable technologies of sinusoidal codecs are the *Multi-Band Excitation* (MBE) family of codecs, the *Sinusoidal Transform Codec* (STC) and the *Waveform Interpolation* (WI) codecs. MBE codecs group the sinusoids in spectral bands and classify each band as voiced or unvoiced. Sinusoids that belong to voiced bands are synthesized with an impulse like (zero-phase) excitation, while sinusoids of unvoiced bands are synthesized with random phases. The MBE family of codecs [43] has many standards in satellite telecommunications, like the IMBE (Improved MBE) and the AMBE (Advanced MBE) codecs which are employed in satellite telecommunication systems like Inmarsat, Iridium and others [44].

STC-like codecs use a strategy that is similar to MBE and split the narrowband speech spectrum in two bands [45], [1]. The lower band is considered to be voiced while the upper band is considered to be unvoiced. As in MBE, voiced harmonics are

synthesized using a zero-phase excitation, while unvoiced harmonics are synthesized using random phases. WI codecs use a different approach. The speech signal is considered to be a process generated by evolving waveforms that describe a single pitch-cycle. The evolving waveforms which are referred to as *characteristic waveforms* (and in early versions of the WI concept as *Prototype Waveforms*) are decomposed to a slowly evolving harmonic part called *Slowly Evolving Waveform* (SEW) and a fast evolving stochastic part called *Rapidly Evolving Waveform* (REW). The SEW component is modeled using a sinusoidal representation while the REW is synthesized using colored noise [1]. WI codecs require longer algorithmic delays for the extraction of the characteristic waveforms but provide good perceptual quality at bit-rates below 4 kbps as reported in [46], [47], [48].

Sinusoidal codecs have not found their way to VoIP applications. This can be partially attributed to the fact that the rival CELP technology was already widely adopted in commercial cellular telephony networks when VoIP emerged, and partially to the fact that sinusoidal codecs cannot reach the perceptual quality of CELP codecs at higher bitrates because of their poor waveform-approximating capability. The expectations of end-users in VoIP are much higher than in satellite telephony systems. However, sinusoidal codecs are well posed for VoIP, namely, for the following factors: first, the sinusoidal representation facilitates high-quality PLC because it is well suited for interpolation and extrapolation of speech [49]. Second, the sinusoidal representation allows high-quality time-scaling [50], [15], useful for on-the-fly resizing of the jitter buffer. Adaptive jitter buffer mechanisms are a vital part of VoIP systems [16].

There are a few experimental sinusoidal codecs proposed for VoIP in the literature [51], [52]. The quality of sinusoidal codecs is bounded by the fact that the phases of the harmonically related sinusoids are reconstructed according to a phase model rather than quantized. Many authors argue that for higher perceptual quality, phase information must be incorporated to sinusoidal codecs [53], [51], [54]. In [54], a variable rate wideband sinusoidal codec that explicitly encodes phases is proposed and evaluated for VoIP. The codec encodes each frame independently of the others (like iLBC) and is capable of accepting packet losses of 10% with slight degradation while it provides acceptable quality even at 20%-30% packet losses. However, the average bitrate of the codec is relatively high, around 21 kbps, which brings up the aforementioned dilemma between efficient source coding with channel coding and redundant source coding with fine-tuned PLC.

1.3 Source/Channel Coding

This section discusses the VoIP problem using arguments driven from information theory. Lets assume that we seek to transmit an encoded representation of the speech signal through a communication channel with fixed capacity. The typical approach, motivated by the *Shannon Separation Theorem*, is to remove the redundancy of the

source using source coding techniques and then to apply channel coding techniques to increase the robustness to channel errors.

Source coding techniques are optimized to minimize the average quantization distortion for a fixed rate. Techniques that encode each source vector $\mathbf{x}$ with a fixed number of bits are referred to as *resolution-constrained (or level-constrained) quantization* (RCQ) [55]. Techniques that encode a sequence of source vectors with variable number of bits per vector but a fixed rate on average are referred to as *entropy-constrained quantization* (ECQ) [55] (pg. 295). In ECQ, the source is quantized with a uniform quantizer and the resulting indices are coded with an entropy-coder like a *Huffman code* or an *arithmetic code* [55]. ECQ is widely used in image, video and audio compression but rarely used in speech coding, mainly because ECQ reduces the resilience of the code to bit-errors, secondarily due to the increased complexity of ECQ and thirdly because ECQ of speech would result in variable-size speech packets. Therefore, speech coding is typically made using RCQ methods, but exceptions also exist [56].

Channel coding techniques introduce redundancy to counteract the losses introduced by the network. Since we consider only erasure channels, a packet can either be lost or received. The possibility of losing the contents of a packet can be reduced using FEC (*Forward Error Correction*) schemes which can be implemented with error correcting block codes. The *Reed-Solomon* (RS) codes are typical block codes used for erasure correction in VoIP. An (N, K) RS code stores K packets of speech in N packets ($N > K$). If any K of these N packets are received, the payload is perfectly reconstructed. If less than K packets are received, the payload is lost [57]. Typical Reed-Solomon codes used in VoIP are the $(3, 2)$ RS and the $(4, 2)$ RS which encode 2 packets in 3 and 4 packets, respectively. The main disadvantage of such FEC schemes is that they introduce additional delay [24], [58], [35].

A common approach to VoIP is to use efficient CELP codecs with FEC [24]. These schemes are similar to the source/channel separation strategy where source-coding and channel-coding are optimized separately and operate in tandem. This approach, which is motivated by the Shannon Separation Theorem (also known as the Channel Coding Theorem) [59] pg. 198, is justified only when we encode sufficiently long sequences of data, thus, when we can afford to have large encoding buffers. An alternative is to jointly optimize source-coding and channel-coding with respect to the reconstruction distortion at the decoder. Intuition can be obtained from the examination of simple paradigms. For example, Lim in [60] shows that joint source/channel coding outperforms separated source/channel optimization in transform coding when the acceptable coding delay is below a threshold. Since speech coding has strict delay requirements, we could expect that joint source/channel coding of speech should outperform cascaded source/channel coding. However, due to the complexity of the subject this is merely an educated guess rather than a fact.

Multiple Description Coding (MDC) is a plausible framework for joint source/channel coding in erasure channels [61]. A schematic representation of MDC is shown in Figure 1.2. MDC encodes each data vector in two descriptions I_1, I_2. Each of

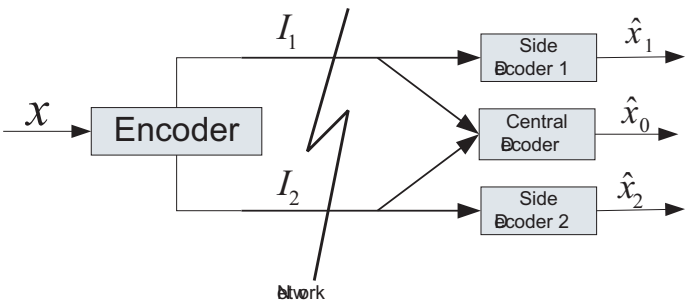


Figure 1.2 Multiple Description Coding

the descriptions is routed through a different channel. When both descriptions are received, the *central decoder* is used to provide a high quality reconstruction of the source. When only one description is received, an intermediate quality reconstruction is made by the corresponding *side decoder*. The quantization distortion related to the central decoder is called *central distortion* while the quantization distortion related to the side decoders is referred to as *side distortion*. If the two channels are independent, the probability of loosing all information regarding a speech frame is substantially reduced compared to the single channel case. The MDC framework provides a formal mechanism to trade central distortion for side distortions. A comparison between FEC and MDC for memoryless Gaussian sources in terms of Shannon's Rate-Distortion theory shows that MDC is better than FEC even when the two descriptions are routed through the same channel [23].

An insightful description of the MDC and its history is provided by Goyal in [61]. Although the first appearance of the MDC concept was made for speech coding applications in 1970s at Bell Labs, even now, three decades later, efficient MDC of speech remains an open problem with unresolved aspects. Most publications on MDC of speech encode the waveform in a sample-by-sample manner, namely in three ways: scalar MDC of PCM samples, scalar MDC in a predictive coding scheme and even-odd separation schemes. The first and the second approach use scalar MDC quantizers to encode the PCM samples [62] or a prediction residual, for example the DPCM (Differential PCM) or the ADPCM residual, [63], [64]. Even-odd separation schemes split the waveform in two descriptions consisting of the even and odd indexed samples, respectively. When a description is lost, the lost samples are either interpolated [61] or reconstructed using side information carried in the received description [65], [66].

Sample-based schemes are inefficient in terms of bitrate. Some researchers propose to modify CELP codecs to operate in a multiple description mode. A simple solution is to use dithering in waveform-approximating codecs [67], [68]. In this approach, each description encodes a dithered version of the signal. When more than one descriptions are received, the decoded signals are averaged to a reconstruction with less quantization noise than the reconstruction made using a single description.

Other researchers construct the two descriptions by distributing the bits of a single CELP description in two descriptions [69], [70], [71]. Some bits are replicated in both descriptions and consist a "base" layer of information. The aforementioned methods are based on heuristics that do not guarantee that the selected excitation vectors are optimal for specific central/side description tradeoffs. The main obstacle in CELP-based MDC is the construction and the optimization of the excitation vectors in a way that is optimal for predefined central/side distortion tradeoffs.

Another approach is to use the inherent redundancy that exists between the auditory channels in order to improve the resilience of the transmission. In [72], [73] the speech signal is analyzed into several frequency bands that resemble the auditory channels of the human hearing process. The analysis process is viewed as a frame expansion operation and it is implemented using a filterbank. The output of each channel is independently encoded and transmitted through the network. If some channels are lost through the transmission, an appropriate reconstruction is still possible. The overall system operates at a fixed central/side distortion tradeoff point using a fixed amount of redundancy. Furthermore, several aspects remain in blind; the employed over-complete transform is justified from an auditory point of view but little justification is provided from a joint source/channel coding aspect. Note, however, that an optimization with respect to the auditory principles as well as the source/channel considerations is not a trivial task.

1.4 Summary of Contributions

This thesis proposes a framework for developing speech codecs that are specially designed for the VoIP conditions. The thesis attacks the problem in two levels: at the basic source coding level where novel methods, that introduce controlled redundancy into the bitstream, are proposed, and at the speech modelling/coding level where a novel speech parameterization, that is amenable to efficient quantization using these methods, is proposed. Figure 1.3 has a diagram of the contributions of the thesis.

The speech signal is modeled with a high-quality sinusoidal Harmonic Model (HM). The parameters of the Harmonic Model, namely, the amplitudes and the phases of the harmonically related sinusoids, are quantized using single description quantization and multiple description quantization. A single description and a multiple description Harmonic Coder are then constructed, along with a high-quality sinusoidal Packet Loss Concealment (PLC) algorithm. The designed codecs are scalable, variable rate and robust to packet losses. Compared to the well known iLBC codec, that is also designed for VoIP applications, the 14.8 kbps variable rate single description codec has similar perceptual quality at similar bitrates while it is more robust to packet losses. The 20.8 kbps variable rate multiple description codec (with channel diversity) can accept 30% packet losses with a perceptual degradation that is not annoying and 40% packet losses with a slight perceptual degradation.

Chapter 2 presents the Harmonic Model of speech which approximates the speech

signal with harmonically related sinusoids. A very fast analysis method for the estimation of the amplitudes and the phases of the sinusoids is proposed along with a fast harmonic synthesis algorithm. A robust analysis-by-synthesis pitch detector that is based on these methods is then proposed. The chapter also reviews an algorithm that fits the harmonic amplitudes to a real cepstrum spectral envelope. This spectral envelope provides a fixed dimension parameterization of the harmonic amplitudes.

Chapter 3 reviews an efficient high-rate quantization algorithm that uses Gaussian Mixture Models (GMM) to model the source statistics. The theoretical background and the algorithm is thoroughly discussed because it forms the basis of the quantization methods that are proposed in this thesis.

Chapter 4 is about the quantization of the phases of the harmonically related sinusoids. The harmonic phases are decomposed into a scalar translation term and a vector of *"dispersion phases"*. The dispersion phases are treated using circular statistics which are suitable for variables with modulo-2π behavior. A mixture model referred to as *"Wrapped Gaussian Mixture Model"* (WGMM) consisting of multivariate wrapped Gaussian pdfs is then presented and the corresponding estimation algorithm is provided. The algorithm uses Expectation-Maximization to maximize the likelihood of the model. WGMM is then used to construct a quantizer for phases. The WGMM-based quantizer uses scalar quantization of wrapped Gaussian random variables. However, the design of such a quantizer is not trivial and two solutions with different complexity/performance tradeoffs are proposed. Finally, the designed WGMM-based quantizer is used to quantize the dispersion phases of speech.

Chapter 5 proposes a novel Packet Loss Concealment algorithm for Harmonic Models of speech. The algorithm uses the jitter buffer to interpolate speech when a future frame is available while it extrapolates speech when the jitter buffer is empty. The sinusoids within a frame are classified as "voiced" or "unvoiced" and a different procedure is used for each case. Sinusoids classified as voiced are interpolated and extrapolated with respect to phase coherence (continuity over time) while the unvoiced sinusoids are synthesized with random phases. The algorithm proves to be effective in concealing packet losses.

Chapter 6 reviews and proposes multiple description coding techniques. Focus is given to GMM-MDSQ$_{TC}$, a GMM-based MDC quantizer suitable for the quantization of spectral envelopes. GMM-MDSQ$_{TC}$ uses transform coding based on the *Multiple Description Scalar Quantization* (MDSQ) quantizers proposed by Vaishampayan [74]. Then, a novel WGMM-based MDC quantizer for phase data, that combines ideas from GMM-MDSQ$_{TC}$ and Chapter 4, is provided. Finally, a novel bit-erasure channel decoder for GMM-based MDC quantizers is proposed. The decoder uses the correlations that exist between the descriptions to reduce the impact of bit-errors that occur in one description. Experimental results show that the proposed decoder battles effectively single and double bit-errors, but the complexity increases rapidly for more than two bit-errors.

Chapter 7 contributes to the "Transform Coding" family of MDC quantizers. A novel resolution-constrained Multiple Description Transform Coding (MDTC) algo-

rithm for multivariate Gaussians is proposed and evaluated. The scheme is based on a specially designed frame expansion. Then, a GMM-based MDTC (GMM-MDTC) algorithm is proposed. GMM-MDTC is better than GMM-MDSQ$_{TC}$ in lower packet losses, slightly inferior in higher packet losses and features scalability, low complexity and low storage requirements. Finally, the performance of MDTC (and GMM-based MDTC) in higher packet losses is improved by a modification of the traditional MDTC encoding procedure. The Gaussian encoders in the new scheme are cooperating in order to minimize a total distortion measure that takes into account the central distortion as well as the side distortions. The new encoding provides central/side distortion tradeoff points similar to the tradeoffs provided by the more complicated MDSQ-based quantizers.

Chapter 8 examines another way of introducing redundancy to the bitstream. When the encoder does not remove the inter-packet redundancy, *Coding with Side Information* (CSI) can be used to introduce redundancy that correct specific types of errors, for example single and double packet losses. A codebook-based CSI framework, referred to as *Conditional Vector Quantization* (CVQ) , is proposed and evaluated for the purpose of recovering the lost spectral envelopes in single and double packet losses. Then, CVQ is used to expand the bandwidth of narrowband speech to wideband. The narrow-band spectral envelopes are used to reconstruct the 4-8 kHz spectral envelopes. A high quality extension of narrowband speech to wideband can be obtained with a minimal of 134 bps for the spectral envelopes using 33.3 Hz frame refresh rate. Full quantization of the 4-8 kHz speech requires 1 kbps for 100 Hz frame refresh rate.

Chapter 9 combines some of the results of the previous chapters in two proof-of-concept speech codecs referred to as *"Harmonic Model Codecs"* (HMC), a single description version of HMC and a multiple description version. The codecs are narrowband, variable rate, similar to quality with iLBC (yielding a PESQ-MOS score of 3.88 under perfect channel conditions) and robust to packet losses.

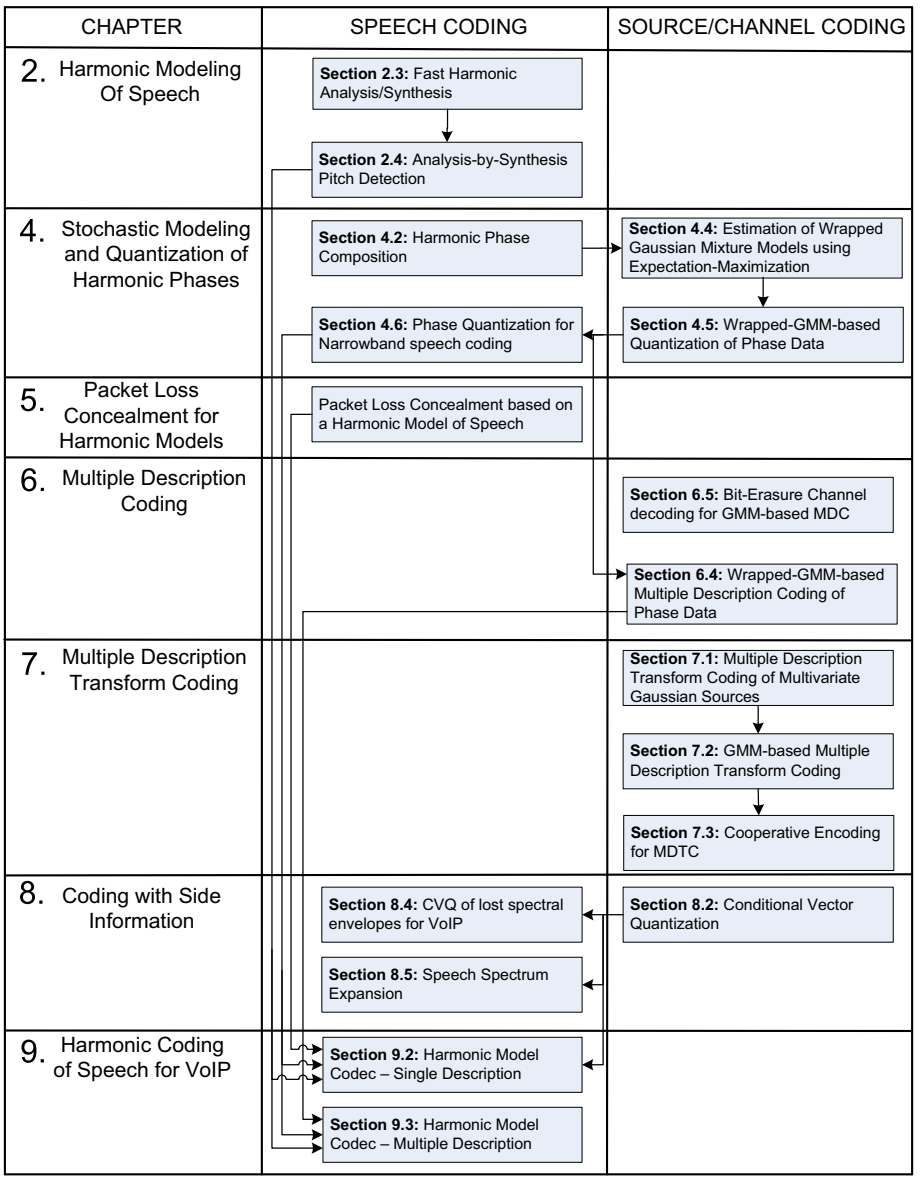


Figure 1.3 Schematic overview of the contributions of the thesis. An arrow that links two boxes $A \rightarrow B$ indicates that "A was used in B".

Chapter 2

Harmonic Modeling of Speech

2.1 Overview

Some parts of the speech signal exhibit a quasi-harmonic behavior that is easily observed in the time-frequency plane. Modeling these parts with a harmonic model is a common practice that has led to high quality analysis/synthesis of speech [75], [76] and to competitive speech coding algorithms [29], [1], [45].

Sinusoidal coders use a harmonic sinusoidal model for voiced speech and have shown superior performance over the rival CELP codecs in lower bit rates (less than 4 Kbps) [29], [1], [45]. Harmonic speech coders rely on a magnitude spectral envelope model and a phase model for the reconstruction of the amplitudes and the phases of the harmonically related sinusoids, respectively.

The spectral envelope is usually obtained using auto-regressive filters or cepstral envelopes [45]. In Sinusoidal Transform Coder (STC) [45], the phase model is based on a voicing decision that effectively splits the speech spectrum into two bands; a lower voiced band and a higher unvoiced band. The phases of the voiced band are constructed by sampling the minimum-phase spectral envelope, while the phases of the unvoiced band are considered random. In Multi-Band Excitation (MBE) [29], the spectrum is split into a number of bands, and a voicing decision is made for each band. Each band is encoded according to its voicing state and the phases are determined by the minimum-phase spectral envelope and/or random noise. In Waveform Interpolative (WI) coders [1], the voiced speech is decomposed into two parts, a slowly evolving component that exhibits a harmonic structure and a fast evolving component that consists of colored noise.

Harmonic speech coders are well posed for VoIP because it is easy to use the harmonic model to interpolate and extrapolate speech for packet loss concealment [77], [54], [51]. Furthermore, the harmonic representation allows efficient jitter buffer resizing via on-the-fly time-scaling of the speech signals. However, sinusoidal coders do not provide speech of high-quality at higher bitrates, mainly due to the fact the phases are estimated and not encoded [45]. On the other hand, speech coders that explicitly encode phases, tend to require increased bitrates [54].

This chapter presents the harmonic sinusoidal model of speech and proposes a very fast analysis/synthesis method. The harmonic amplitudes will be represented by a spectral envelope based on cepstral coefficients and a suitable post-filtering enhancement method will be discussed. A novel algorithm will be proposed that uses the cepstral envelope to select the pitch among a list of possible pitch candidates. The latter method will be used to construct an Analysis-by-Synthesis pitch detector.

2.2 Analysis/Synthesis using a Harmonic Model

The accurate measurement of phases is not of primary interest for sinusoidal coders based on phase models. However, a high quality sinusoidal representation requires accurate measurements of both harmonic amplitudes and phases [76]. The Harmonic Model is a high quality parametric model used for signal analysis/synthesis. The signal is represented as a weighted sum of harmonically related cosines and sines:

$$\hat{x}(n) = \sum_{k=1}^{K} [c_k \cos(k\omega_0(n - n_0)) + s_k \sin(k\omega_0(n - n_0))] \tag{2.1}$$

where N is the duration of the analysis frame in samples, $n_0 = \frac{N-1}{2}$ is the center of the analysis frame, ω_0 is the fundamental frequency, K is the number of harmonics, c_k and s_k are the cosine and sine coefficients describing the even and odd part of the k-th harmonic sinusoid, respectively, and n is the time index. Equivalently, the harmonic model can be expressed in terms of K harmonic amplitudes A_k and K harmonic phases ϕ_k:

$$\hat{x}(n) = \sum_{k=1}^{K} A_k \cos(k\omega_0(n - n_0) + \phi_k),\ n = 0, \ldots, N-1, \tag{2.2}$$

The unknown parameters c_k and s_k are evaluated using a weighted least-squares method that minimizes the square error criterion with respect to c_k and s_k:

$$\epsilon = \sum_{n=0}^{N-1} w^2(n)(x(n) - \hat{x}(n))^2, \tag{2.3}$$

where $x(n)$ is the original signal, $\hat{x}(n)$ is its harmonic representation and $w(n)$ is the analysis window. Using matrix formulation, we may rewrite (2.1) as

$$\hat{\mathbf{x}} = \mathbf{B} \begin{bmatrix} \mathbf{c} \\ \mathbf{s} \end{bmatrix} \tag{2.4}$$

where $\mathbf{B}$ is the $N - by - 2K$ cosine/sine basis matrix

$$\mathbf{B} = [\mathbf{C}\ \mathbf{S}] \tag{2.5}$$

and where $\mathbf{C}$ and $\mathbf{S}$ are the cosine and sine bases matrices, respectively, with size $N - by - K$ and elements that are defined by:

$$C_{n,k} = \cos(k\omega_0(n - n_0)) \tag{2.6}$$

$$S_{n,k} = \sin(k\omega_0(n - n_0)) \tag{2.7}$$

for $n = 0, \ldots, N-1$ and $k = 1, \ldots, K$, while vectors $\mathbf{c}$, $\mathbf{s}$ hold the parameters to be computed:

$$\mathbf{c} = [c_1 c_2 \ldots c_K]^T \tag{2.8}$$

$$\mathbf{s} = [s_1 s_2 \ldots s_K]^T \tag{2.9}$$

The solution to the least-squares problem (2.3) is then given by the normal equations [78]:

$$(\mathbf{B}^T\mathbf{W}^T\mathbf{W}\mathbf{B}) \begin{bmatrix} \mathbf{c} \\ \mathbf{s} \end{bmatrix} = \mathbf{B}^T\mathbf{W}^T\mathbf{W}\mathbf{x} \tag{2.10}$$

where $\mathbf{W} = diag\,(w(0), w(1), \ldots, w(N-1))$ is a diagonal matrix with the symmetric window $\mathbf{w}$ for diagonal and $\mathbf{x}$ is a $N - by - 1$ vector that holds the original signal $\mathbf{x} = [x(0),\ x(1), \ldots x(N-1)]^T$. Note that $(\mathbf{B}^T\mathbf{W}^T\mathbf{W}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{W}^T$ is the pseudoinverse matrix that projects the (weighted) signal into the subspace of the weighted harmonic sines and cosines.

Using simple trigonometric algebra and the fact that the window $\mathbf{w}$ is a symmetric one, we have that

$$\begin{aligned} \mathbf{B}^T\mathbf{W}^T\mathbf{W}\mathbf{B} &= \begin{bmatrix} \mathbf{C}^T\mathbf{W}^T\mathbf{W}\mathbf{C} & \mathbf{C}^T\mathbf{W}^T\mathbf{W}\mathbf{S} \\ \mathbf{S}^T\mathbf{W}^T\mathbf{W}\mathbf{C} & \mathbf{S}^T\mathbf{W}^T\mathbf{W}\mathbf{S} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A}_c & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_s \end{bmatrix} \end{aligned} \tag{2.11}$$

and

$$\mathbf{B}^T\mathbf{W}^T\mathbf{W}\mathbf{x} = \begin{bmatrix} \mathbf{C}^T\mathbf{W}^T\mathbf{W}\mathbf{x} \\ \mathbf{S}^T\mathbf{W}^T\mathbf{W}\mathbf{x} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_c \\ \mathbf{b}_s \end{bmatrix} \tag{2.12}$$

where $\mathbf{A}_c$ and $\mathbf{A}_s$ are the $K - by - K$ cosine and sine, respectively, weighted correlation matrices, $\mathbf{b}_c$ and $\mathbf{b}_s$ are the cosine and sine $K - by - 1$ projection vectors, and $\mathbf{0}$ is the $K - by - K$ zero matrix. Therefore, from (2.10) we get the following two systems to solve:

$$\mathbf{A}_c\mathbf{c} = \mathbf{b}_c \tag{2.13}$$

$$\mathbf{A}_s\mathbf{s} = \mathbf{b}_s. \tag{2.14}$$

Matrices $\mathbf{A}_c = [a^c_{i,j}]$ and $\mathbf{A}_s = [a^s_{i,j}],\ i,j \in \{1,..,K\}$ can be restated as [50]:

$$a^c_{i,j} = \sum_{n=0}^{N-1} w(n)^2 \cos(i\omega_0(n-n_0)) \cos(j\omega_0(n-n_0)) = \tau_{i-j} + h_{i+j} \tag{2.15}$$

$$a^s_{i,j} = \sum_{n=0}^{N-1} w(n)^2 \sin(i\omega_0(n-n_0)) \sin(j\omega_0(n-n_0)) = \tau_{i-j} - h_{i+j} \tag{2.16}$$

where $\mathbf{T} = [\tau_{i-j}]$ is a Toeplitz matrix and $\mathbf{H} = [h_{i+j}]$ is a Hankel matrix:

$$\tau_{i-j} = \frac{1}{2} \sum_{n=0}^{N-1} w(n)^2 \cos((i-j)\omega_0(n-n_0)) \tag{2.17}$$

$$h_{i+j} = \frac{1}{2} \sum_{n=0}^{N-1} w(n)^2 \cos((i+j)\omega_0(n-n_0)), \tag{2.18}$$

therefore, $\mathbf{A}_c = \mathbf{T} + \mathbf{H}$ and $\mathbf{A}_s = \mathbf{T} - \mathbf{H}$ are Toeplitz-plus-Hankel matrices. Linear systems with Toeplitz-plus-Hankel coefficient matrices can be efficiently solved with $O(K^2)$ complexity using Levinson-type and Schur-type algorithms [79]. An interesting property of the Toeplitz-plus-Hankel matrices is that their inverses are the so called *Toeplitz+Hankel-Bezoutians*: matrices that can be accurately reconstructed with $O(K^2)$ complexity using only $O(K)$ parameters [79].

2.3 A method for Fast Harmonic Analysis/Synthesis

Approximate solutions of the linear systems in equations (2.13) and (2.14) can reduce the complexity of the analysis to $O(K)$ with minor degradation to the quality of the solution. The fact that $\mathbf{A}_c$ and $\mathbf{A}_s$ are band-diagonal is used in [80] (ch. 6) to reduce the complexity to $O(K)$. The latter method is using the observation that the correlations between the sinusoids are influenced mostly by the main lobe of the analysis window. Therefore, only the sinusoids that are close in frequency, in terms of the effective bandwidth of the main lobe, are considered for the inversion. This results to the inversion of a highly sparse system, and to a complexity of $O(K)$ calculations.

This section proposes a novel method that also leads to a complexity of $O(K)$, but involves no inversion at all. The method is motivated by the observation that the inverses of $\mathbf{A}_c$ and $\mathbf{A}_s$ are *Toeplitz+Hankel-Bezoutians* and can be accurately stored using only $O(K)$ parameters [79]. Furthermore, for the range of values that is interesting for speech analysis and coding (analysis frame with 20ms or 30ms duration and pitch frequency f_0 between 60 and 400 Hz), the inverses $\mathbf{A}_c^{-1}$ and $\mathbf{A}_s^{-1}$ are approximately equal $\mathbf{A}_c^{-1} \approx \mathbf{A}_s^{-1}$ and share a structure that can easily be encoded with a minimal set of parameters. The proposed method is not as accurate or generic as the inverse reconstruction theorems in [79] that reconstruct the inverse matrix with

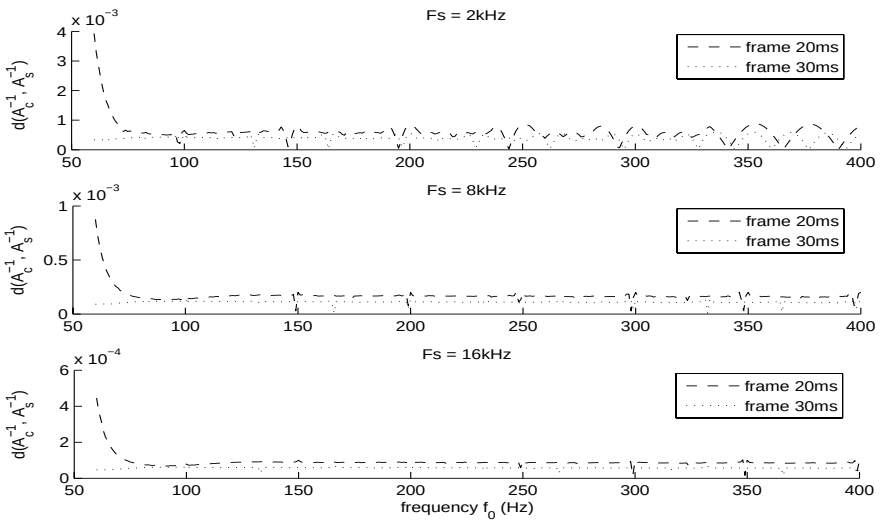


Figure 2.1 Similarity measurement using Hamming window.

$O(K^2)$ complexity, but it offers significant computational advantages at the cost of a minimal SNR (Signal-to-Noise Ratio) degradation.

Initially, it will be shown that $\mathbf{A}_c^{-1} \approx \mathbf{A}_s^{-1}$ for many interesting frame durations (20ms and 30ms), sampling rates $F_s = \{$ 2 kHz, 8 kHz, 16 kHz $\}$ and for all integer fundamental frequencies f_0 between 60 Hz and 400 Hz. Then we will define $\mathbf{A} \equiv \mathbf{A}_c^{-1}$ and present an encoding algorithm $Q(\cdot)$ that encodes $\mathbf{A}$ to $\hat{\mathbf{A}} = Q^{-1}(Q(\mathbf{A}))$. Finally, the quality of the approximation made by using the encoded matrix $\hat{\mathbf{A}}$, for the cases under examination, will be addressed.

2.3.1 Similarity between $\mathbf{A}_c^{-1}$ and $\mathbf{A}_s^{-1}$

The similarity between the inverse matrices $\mathbf{A}_c^{-1}$ and $\mathbf{A}_s^{-1}$ was measured for all examined cases of sampling rates, frame durations and ω_0, using the *Hamming* window. As a distance measure we used the element-wise mean value of the absolute difference between the product of the first inverse matrix with the second matrix and the identity matrix $\mathbf{I}$:

$$d(\mathbf{A}_1^{-1}, \mathbf{A}_2^{-1}) = mean\|\mathbf{A}_1^{-1}\mathbf{A}_2 - \mathbf{I}\| \tag{2.19}$$

The results of the two measurements: $d(\mathbf{A}_c^{-1}, \mathbf{A}_s^{-1})$ and $d(\mathbf{A}_s^{-1}, \mathbf{A}_c^{-1})$ are similar, so we chose to present only the former for clarity.

As shown in Figure 2.1, the distance $d(\mathbf{A}_c^{-1}, \mathbf{A}_s^{-1})$ is well below 10^{-3} for most of the examined cases. Note that this corresponds to a mean relative error of 0.1% because the elements of the product $\mathbf{A}_c^{-1}\mathbf{A}_s$ should approximate the identity matrix $\mathbf{I}$. Similar results were obtained for other commonly used windows like *Hanning*, *Rectangular*, and *Blackman*. Therefore, for the rest of this section, the matrix $\mathbf{A} \equiv \mathbf{A}_c^{-1} \approx \mathbf{A}_s^{-1}$

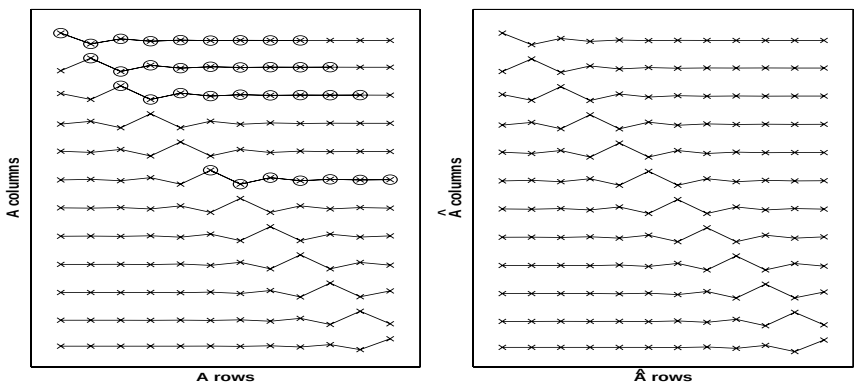


Figure 2.2 **A** and $\hat{\mathbf{A}}$ columns for $F_s = 2kHz$, $f_0 = 70Hz$ and $20ms$ frame. Circled are the selected representative patterns.

will serve as an approximation of the inverse of both matrices $\mathbf{A}_c$ and $\mathbf{A}_s$.

2.3.2 Encoding Algorithm

It is unrealistic to store one inverse matrix for every possible f_0. The storage requirements for **A** can be reduced if we can exploit the structure of the matrix. The elements of **A** have a specific pattern of similarity in the columns of **A**. Each column is similar to the other columns in accordance to a shift in row sense. Additionally, the main pattern is symmetric. We developed an encoding algorithm that selects representative patterns from the columns of **A**. These representative patterns are used to create the decoded matrix $\hat{\mathbf{A}}$.

The selection of the representative patterns is made using an energy criterion so that the decoded columns retain 99.999% of the energy of the original columns. Since the pattern of the center column is approximately symmetric, only half of it needs to be stored. A variable number of patterns (1 to 4) may be extracted for each matrix **A**. The patterns are extracted from columns 1,2,3 and $\lfloor K/2 \rfloor$. The main representative pattern, p_0, is created from the $\lfloor K/2 \rfloor$-th (center) column and it is always kept. Let p_1, p_2 and p_3 be the representative patterns from the 1^{st}, the 2^{nd} and the 3^{rd} column, respectively. From these three patterns, only those representing columns for which p_0 does not satisfy the energy criterion are kept. Finally, all kept representative patterns are extended to include as many elements as the longest one, and may be zero-padded if they do not have enough elements. The encoded representation of **A** consists of these patterns and requires only a few parameters per matrix.

An example of a matrix A and its compressed version $\hat{A}$ is shown in Figure 2.2. The figure plots the columns of A and the columns of the corresponding matrix $\hat{A}$. The representative patterns in the leftmost matrix A are circled. The rightmost matrix $\hat{A}$ is constructed using only the circled representative patterns. It is evident that $\hat{A}$

captures the coarse structure of A. The number of parameters required to encode *all matrices* $A(f_0)$ for $f_0 = 60, 61, ..., 400Hz$ and the corresponding compression ratios are shown in Table 2.1. Clearly, the memory requirements for the representative patterns p_i,$i = 0, ..., 3$ are quite low.

$F_s(kHz)$	Frame(ms)	Parameters	Compression Ratio
2	20	1387	7.2
2	30	506	19.8
8	20	1833	103.2
8	30	504	375.4
16	20	2069	392.8
16	30	503	1616.0

Table 2.1 Number of parameters needed to encode all matrices for $f_0 = 60, 61, ..., 400Hz$. A Hamming window was used.

The matrix $\hat{\mathbf{A}}$ is decoded from the stored representative patterns. The center column representative pattern is mirrored and concatenated to itself, to approximate the original center column and is copied to all columns using the appropriate shift. The representative patterns of the first, second and third column -whichever are kept- are copied to their respective column, with the appropriate mirroring and concatenation for the second and third pattern. These are also flipped and copied to the last, second to last and third to last column accordingly.

The representative patterns p_i evolve smoothly with respect to f_0. Therefore, we can interpolate the representative patterns for values of f_0 that were not used at the encoding stage. For example, the patterns p_i for $f_0 = 100.5\ Hz$ can be taken from the linear interpolation of the nearest patterns at $f_0 = 100\ Hz$ and $f_0 = 101\ Hz$: $p_i(100.5) = 0.5(p_i(100) + p_i(101))$.

The patterned structure of A is broken when the harmonic analysis includes sinusoids that are near the Nyquist frequency, $F_s/2$. Therefore, all the experiments in the chapter were conducted with a maximum harmonic frequency that is lower than $F_s/2$. In particular, the cutoff frequencies for $F_s = 2\ kHz$, $F_s = 8\ kHz$ and $F_s = 16\ kHz$ were 0.9 kHz, 3.7 kHz and 7.6 kHz, respectively.

The complexity of the decoding process depends on the number of harmonics K and the size of the patterns. Since the size of patterns is bounded to a few coefficients, the complexity of the proposed algorithm is linear, i.e. $O(n)$. In fact, the decoding process is just a multiplication of b_c (or b_s) with a sparse matrix generated by the representative patterns.

2.3.3 Quality of the Approximation

The approximation made by the proposed encoding algorithm is evaluated with the distance measure in (2.19). The distances $d(\hat{\mathbf{A}}, \mathbf{A}_c^{-1})$ and $d(\hat{\mathbf{A}}, \mathbf{A}_s^{-1})$ were measured for

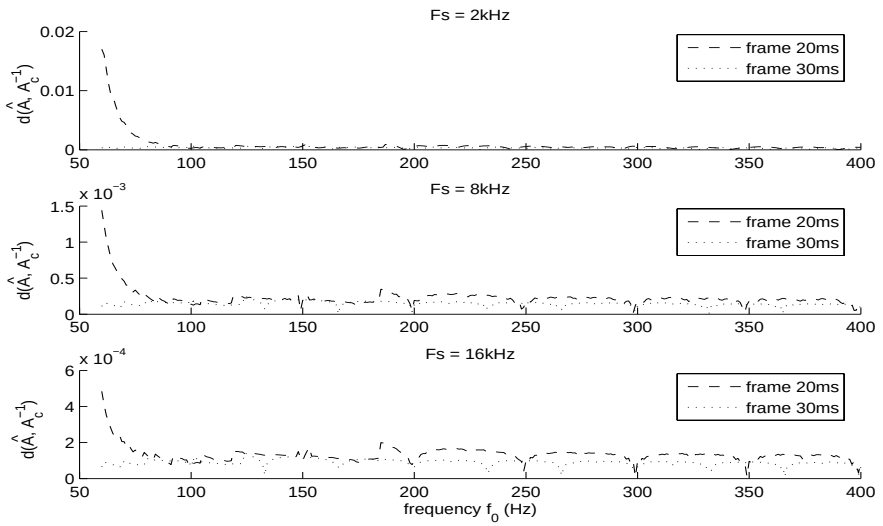


Figure 2.3 Encoding quality measurement using Hamming window.

$f_0 = 60, ..., 400Hz$ and several sampling rates, windows, frame durations. However, the corresponding distances $d(\hat{\mathbf{A}}, \mathbf{A}_c^{-1})$ and $d(\hat{\mathbf{A}}, \mathbf{A}_s^{-1})$ are very close in a numerical sense, therefore for clarity, we will present results regarding only $d(\hat{\mathbf{A}}, \mathbf{A}_c^{-1})$. The results are depicted in Figure 2.3 where it is shown that distance $d(\hat{\mathbf{A}}, \mathbf{A}_c^{-1})$ is below 0.02 for the case of $F_s = 2kHz$ and well below 0.01 for the rest of the examined cases. Note that the measurements were made using a Hamming window and that similar results were obtained for other windows.

2.3.4 Fast Harmonic Synthesis

The computation of the cosine and sine bases, used in (2.13) and (2.14), requires a considerable portion of the complexity of the HM analysis. However, the cosine and sine functions can be computed iteratively over time using the equations [81]:

$$\cos(k\omega_0(n+1)) \simeq (1-\alpha)\cos(k\omega_0 n) - \beta \sin(k\omega_0 n)$$
$$\sin(k\omega_0(n+1)) \simeq (1-\alpha)\sin(k\omega_0 n) + \beta \cos(k\omega_0 n)$$

where $\alpha = 2\sin^2(0.5k\omega_0)$ and $\beta = \sin(k\omega_0)$. These computations need 2 MAC (Multiply-Accumulate) operations for each cosine or sine evaluation. The complexity can be further reduced if the recurrence is taken over the harmonic frequencies:

$$\cos(k\omega_0 n) = 2\cos((k-1)\omega_0 n)\cos(\omega_0 n) - \cos((k-2)\omega_0 n)$$
$$\sin(k\omega_0 n) = 2\sin((k-1)\omega_0 n)\cos(\omega_0 n) - \sin((k-2)\omega_0 n)$$

With proper implementation, these computations need only 1 MAC operation each. Furthermore, the symmetry and antisymmetry of $\mathbf{C}$ and $\mathbf{S}$ over the rows can be

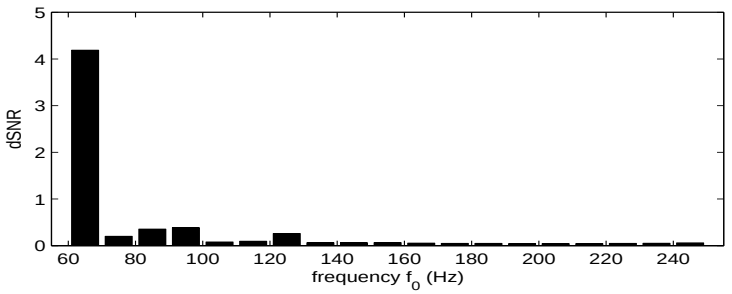


Figure 2.4 Average segmental SNR degradation for several f_0 intervals.

exploited to reduce the cost for the computation of the corresponding matrices. Note that when N is odd, the central rows of $\mathbf{C}$ and $\mathbf{S}$ have constant values of 1 and 0, respectively. We use the following combination of the presented recurrence relations. Initially, we use the recurrence over the rows of $\mathbf{C}$, $\mathbf{S}$ (time index n) to compute the sine/cosines of the first harmonic. Then we use the recurrence over the frequencies to compute the rest of the harmonics.

2.3.5 Experimental Evaluation

The performance of the proposed algorithm was evaluated in terms of segmental SNR. Tests of analysis/synthesis of narrowband speech signals ($F_s = 4$ kHz) using 20ms frames weighted by a Hanning window were conducted. Estimation of pitch was used for the voiced frames, while for the unvoiced frames, a constant fundamental frequency $f_0 = 100\ Hz$ was used. The comparison was made using 512 narrowband utterances (256 males, 256 females). Let SNR_{inv} be the segmental SNR provided when $\mathbf{A}_c$ and $\mathbf{A}_s$ are typically inverted and SNR_{fast} be the segmental SNR provided by the proposed algorithm, as described in Section 2.3.2. The difference $dSNR = SNR_{inv} - SNR_{fast}$ was taken on a frame-by-frame basis. Figure 2.4, depicts the average $dSNR$ for several f_0 intervals. Note that the SNR degradation is negligible for most frequencies. However, in lower frequencies ($f_0 < 70$ Hz) the degradation is more evident. That is to be expected, since as we showed in Figures 2.1 and 2.3 our algorithm doesn't perform well for this range. This is not a significant problem because such pitch values are very rare and the SNR is already very high due to the dense frequency sampling. In order to reduce the SNR loss, the number of representative patterns must be increased.

2.4 Modeling Harmonic Amplitudes using Cepstral Envelopes

The amplitudes of the harmonically related sinusoids of voiced speech evolve slowly over frequency and form a *spectral envelope*. The notion of the "spectral envelope" in

speech signals is justified if the speech production is seen as a source-filter operation. In this perspective, the spectral envelope resembles the effect of the vocal tract to the excitation signal that is emitted from the vocal folds [45]. From the source coding point of view, the spectral envelope provides a way to exploit the dependencies between the harmonic amplitudes, as well as a way to remove an amount of perceptual irrelevancy that exists in this source.

There is considerable work on the extraction of spectral envelopes from speech signals [45], [1]. The methods can be roughly categorized to those that represent the spectral envelope using an all-pole filter and to those that describe the spectral envelope in terms of *cepstral* coefficients. The latter spectral envelopes resulting are also called *cepstral envelopes*.

2.4.1 Mel-Scale Cepstral Envelope

Let $H_s(f)$ be the spectral envelope of a speech frame. A parametric cepstral envelope of order P is provided by formula:

$$\log|H_s(f)| = c_0 + 2\sum_{p=1}^{P} c_p \cos(2\pi f p), \tag{2.20}$$

where $\mathbf{c} = [c_p]$, $p = 1, \ldots, P$ are the P real cepstrum coefficients plus c_0 which states the energy of the signal. The cepstral envelope is a minimum phase spectral envelope. In fact, the log-amplitude and the (minimum) phase spectrum have a Hilbert transform relationship [1] (pg. 144):

$$\angle H_s(f) = -2\sum_{p=1}^{P} c_p \sin(2\pi f p). \tag{2.21}$$

The cepstral envelope approximates the log-spectrum harmonic amplitudes $\log(A_k)$ at the corresponding frequencies by minimizing a least-squares error at the log-spectrum domain:

$$\epsilon = \sum_{k=1}^{K} \| \ \log(A_k) - \log|H_s(kf_0)| \ \|^2 \tag{2.22}$$

In other words, the harmonic log-amplitudes are projected to a subspace generated by the columns of $\mathbf{M} = [\mu_{i+1,j+1}]$, $i = 0, \ldots, K$, $j = 0, \ldots, P$, where

$$\mu_{i,j} = 2\cos(2\pi i f_0 j) - \delta(j). \tag{2.23}$$

where $\delta(.)$ is the discrete delta function. The least-squares error solution is provided by the pseudo-inverse of $\mathbf{M}$:

$$\mathbf{c} = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{a} \tag{2.24}$$

where $\mathbf{a} = [\log(A_1)\log(A_2)\ldots\log(A_K)]^T$. However, this solution does not guarantee a smooth cepstral envelope $C(f)$. The rapid variations of the cepstral envelope can be penalized by adding to the distortion criterion ϵ a term that expresses the energy of the derivative of $\frac{d}{df}\log|H_s(f)|$; thus by minimizing a regularized criterion $\acute{\epsilon}$:

$$\acute{\epsilon} = \epsilon + \lambda \int_{-\frac{F_s}{2}}^{\frac{F_s}{2}} (\frac{d}{df}\log|H_s(f)|)^2 df \tag{2.25}$$

where λ is a regularization parameter that controls the smoothness of the derived cepstral envelope. The least-squares error solution in the regularized case can be shown to be [82]:

$$\mathbf{c} = [\mathbf{M}^T\mathbf{M} + \lambda\mathbf{R}]^{-1}\mathbf{M}^T\mathbf{a} \tag{2.26}$$

where $\mathbf{R} = diag([0\ 8\pi^2 1^2\ 8\pi^2 2^2\ \ldots\ 8\pi^2 P^2])$.

The distortion criterion ϵ does not take into account the perceptual importance of the lower frequencies. The modeling error at the lower harmonics can be reduced if the cepstral envelope is computed using the Bark frequency scale instead of the linear frequency scale [50]. There is considerable work in closed form expressions that link Bark-scale to linear frequency scale. We used the Traunmüller formula with low frequency and high frequency corrections [83]:

$$b = \frac{26.81f}{1960 + f} - 0.53 \tag{2.27}$$

$$\text{low frequency correction : if b} < 2, \text{b} \leftarrow \text{b} + 0.15(2 - \text{b}) \tag{2.28}$$

$$\text{high frequency correction : if b} > 20, \text{b} \leftarrow \text{b} + 0.22(\text{b} - 20.1), \tag{2.29}$$

where b is in Barks and f is in Hz. When the cepstral envelope is computed in Bark scale, a value of $\lambda = 0.002$ seems to provide an acceptable balance between smoothness and modeling quality.

2.4.2 Adaptive Post-Filtering

The speech that is reconstructed using original phases and harmonic amplitudes sampled from the cepstral envelope is usually of high quality when the model order is adequate (i.e. $P \geq 20$ for narrowband speech). However, for low-pitched speakers, there is a slight *loss-of-presence* effect. This is usually perceived as a alternation of the speech signal rather than a degradation. This type of degradation is closely related to the muffling effect observed in sinusoidal coders which can be attributed to the reduction of the dynamic range of a formant peak-to-null distance. Adaptive post-filtering techniques can then be used to deepen the formant nulls. The technique described in [1] (pg. 148) will briefly be reviewed.

Let $\hat{A}_k$ be the K harmonic amplitudes which are sampled from the cepstral envelope. The energy R_0 and the correlation coefficient R_1 can be expressed in terms of

$\hat{A}_k$:

$$R_0 = \sum_{k=1}^{K} \hat{A}_k^2 \tag{2.30}$$

$$R_1 = \sum_{k=1}^{K} \hat{A}_k^2 \cos(k\omega_0). \tag{2.31}$$

The post-filter weight W_k associated with the k-th harmonic is provided by formula:

$$W_k = \hat{A}_k^{\gamma} \left[\frac{K\,(R_0^2 - 2R_1R_0cos(k\omega_0) + R_1^2)}{R_0(R_0^2 - R_1^2)} \right]^{\frac{\gamma}{2}}. \tag{2.32}$$

where the factor in the brackets removes the tilt from the spectral envelope, and the exponentiation by $\gamma \in [0, 1]$ applies a root-γ compression rule to the tilt-removed log-spectrum. In this thesis a value of $gamma = \frac{1}{2}$ is used. A clipping rule is then applied to the weights W_k in order to avoid excessive spectral shaping:

$$W_k' = \begin{cases} 1.2, \text{ if } W_k > 1.2 \\ 0.8, \text{ if } W_k < 0.8 \\ W_k, \text{ otherwise} \end{cases} \tag{2.33}$$

Concluding, the post-filtered harmonic amplitudes are $\hat{A}_k' = W_k'\hat{A}_k$.

The post-filter reduces the loss-of-presence effect associated with the cepstral envelope and increases the PESQ-MOS [5] score about 0.1 units, on average.

2.5 A novel Pitch Detection algorithm

Pitch is the perceived tonality of an audio signal that is composed by one or more tones. The pitch estimation problem has troubled speech researchers and engineers from the beginning of the speech processing discipline. Pitch detection is also stated as a fundamental frequency (f_0) estimation problem. However, strictly speaking, the fundamental frequency is a characteristic of the behavior of a signal, while pitch is a characteristic of the human sound perception system. In this thesis, pitch detection will be addressed as a fundamental frequency estimation problem. Therefore, a pitch detector tries to find the frequency f_0 that best describes the harmonic or quasi-harmonic behavior of the signal.

The methods that perform pitch detection can be classified to those that use time-domain criteria like the YIN estimator [84], the RAPT estimator [1] (ch. 14) and the MBE pitch estimator [29] (pg. 242), and those that use frequency domain criteria like the Harmonic Sinewave pitch estimator [45] (pg. 510), [50]. A detailed review of pitch detection is beyond the scope of this thesis. An early guide to pitch detection can be found in [85].

Pitch halving and pitch doubling are common errors in pitch detection. The following subsection presents a novel algorithm that corrects pitch halving/doubling errors. Then, the fast analysis/synthesis techniques presented in section 2.3 will be used to develop a novel pitch estimator that directly minimizes the harmonic modeling error with high level of accuracy and affordable complexity.

2.5.1 Pitch Ambiguity Correction using Envelope Continuity Constrains

Let $F_0 = \{f_{0,i} : i = 1, \ldots, L_{cand}\}$ be a set of L_{cand} candidate pitch values from the output of a pitch detector. Such a set can be constructed either directly by the pitch detector, or from a single pitch estimation f_0', by including pitch doubling/halving candidates, for example $F_0 = \{\ 0.5f_0',\ f_0',\ 2f_0'\ \}$. The aim of this method is to resolve the ambiguity and choose the "best" $f_{0,i}$ with a memoryless approach. The selection criterion is the segmental SNR between the original signal $x(n)$ and the reconstructed signal $\hat{x}(n; f_{0,i})$ defined as:

$$SNR(x, \hat{x}) = 10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} x(n)^2}{\sum_{n=0}^{N-1} (x(n) - \hat{x}(n))^2} \right) \tag{2.34}$$

Signal $\hat{x}(n; f_{0,i})$ is created using the original phases and harmonic amplitudes $\hat{A}_k$ which are sampled from a P-th order Bark-scale regularized ($\lambda = 0.002$) cepstral envelope. For narrowband speech, the order P is set in the range of 16-20 coefficients.

This simple scheme is quite effective in resolving the pitch halving/doubling ambiguities. Let f_0 be the *true* pitch, and $SNR(f)$ be the SNR measured by the proposed method using f as the fundamental frequency. Then, $SNR(2f_0)$ will be significantly lower than $SNR(f_0)$ because half of the harmonics will be missing from the reconstructed signal. Using a spectral envelope, $SNR(0.5f_0)$ will be lower than $SNR(f_0)$ because an additional error will be introduced from fitting both *higher amplitude harmonics* and *lower amplitude interharmonics* to the same cepstral envelope. The interharmonics will drug down and ripple the cepstral envelope resulting in a poor reconstruction of both harmonics and interharmonics in $\hat{A}_k$.

An example of the penalty associated with bad fitting of harmonics and interharmonics to a fixed 20-th order cepstral envelope is depicted in Figure 2.5. The original amplitudes A_k were used to generate a series of harmonic/interharmonic amplitudes $B_k = \left(1 - \alpha \frac{1+(-1)^{k+1}}{2}\right) A_k$, with α ranging between 0 and 0.9. SNR is measured between the signal synthesized using the B_k amplitudes and the signal synthesized using the $\hat{B}_k$ amplitudes sampled from the cepstral envelope. The rightmost plot shows the SNR as a function of α. The SNR falls as the interharmonics become weaker. The leftmost plot shows A_k, B_k and the corresponding spectral envelopes for $\alpha = 0.5$. The lower interharmonics cause the cepstral envelope to become rippled and lower; a worse fit to both harmonics and interharmonics.

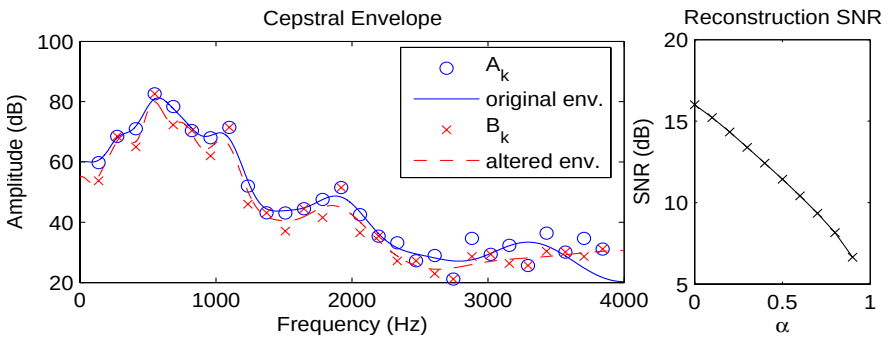


Figure 2.5 Penalty from using interharmonics to fit a cepstral envelope

Table 2.2 MOS scores from PESQ evaluation

Method	Males	Females
Without pitch correction	3.954	3.659
With pitch correction	4.055	3.879

We evaluated the proposed pitch correction algorithm using PESQ-MOS [5] and sinusoidal analysis/synthesis of a database consisting of 256 female and 256 male utterances from TIMIT. In all cases, speech was reconstructed using OLA (OverLap-Add) techniques and sinusoidal amplitudes sampled from the cepstral envelope. The experiment was conducted twice, once for the output of a reference pitch detector and once for the output of the proposed method. The reference pitch detector is based on MBE [29] (pg. 242), [50].

The PESQ MOS results are shown in Table 2.2. It can be clearly seen that the proposed method increases the PESQ MOS score by 0.1 and 0.2 for males and females, respectively.

Further insight into the obtained results is provided by Figure 2.6, which shows the

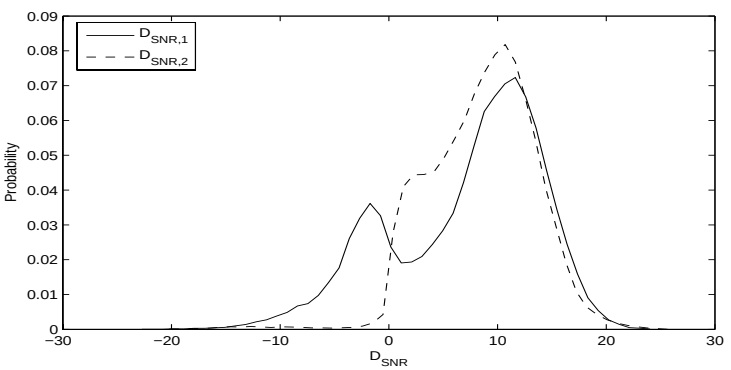


Figure 2.6 Distribution of differential SNR

histogram of two differences. The first difference (*solid lines*) is $D_{SNR,1} = SNR(f_0') - SNR(0.5f_0')$ and shows when f_0' is better than $0.5f_0'$, while the second difference (*dashed lines*) is $D_{SNR,2} = SNR(f_0') - SNR(2f_0')$ and shows when f_0' is better than $2f_0'$. Only voiced frames were used for this evaluation. The histogram states that the proposed post-processing method frequently reduces the estimated pitch f_0' to $0.5f_0'$, which corresponds to a pitch doubling error if the correction is true. This is consistent with our observation of frequent pitch doubling errors of the employed pitch detector. Note however, that pitch halving is sometimes natural at the end of a phrase or a word due to a phenomenon called "vocal fry" [45]. In vocal fry the interharmonics are almost equally strong with the harmonics and the proposed method correctly states that the pitch is halved (from a perception point of view).

2.5.2 Analysis-by-Synthesis Pitch Detection

The fast analysis/synthesis technique proposed in Section 2.3 enables an exhaustive search for the f_0 that provides the best SNR with tractable complexity. In fact, this corresponds to an Analysis-by-Synthesis approach to the pitch detection problem.

Analysis-by-Synthesis (AbS) techniques have proven to be more robust than open-loop techniques in CELP codecs [29] (ch. 6) because they take into account a wide range of factors that influence the quality of the resulting speech. Open-loop pitch detection is based on assumptions regarding the behavior of the signal which are much weaker than the assumptions made by AbS pitch detection. Analysis-by-Synthesis techniques are not well suited for Sinusoidal Coding when the harmonic phases are derived from a voiced/unvoiced phase model, in the sense that the AbS distortion criterion has to match two inevitably different waveforms. However, some researchers reported an improvement when AbS techniques were used in minimum phase models [86]. In [87] AbS is made with measured phases only at the analysis stage of the pitch detector while the synthesis stage is conducted with an STC-like phase model [45] (pg. 523). On the other hand, AbS is well suited for high-quality/high-rate sinusoidal codecs that directly encode the phases [54].

It is a common practice in sinusoidal AbS pitch detectors to estimate the harmonic amplitudes by pick-peaking the spectrum [54], [87], in order to avoid the increased complexity of solving the corresponding linear systems (2.13) and (2.14). Such an approach is optimal only under rather idealized conditions [45] (pg. 437) that decorrelate the harmonically related sinusoids.

The proposed AbS pitch detection algorithm consists of the following steps:

Step 1: Coarse Search

An exhaustive search is made for a wide range of pitch values using a frame of 30 ms. Let $\mathbf{x} = \{x(n) : n = 0, \ldots, N-1\}$ be the signal with sampling rate $F_s = 8\ KHz$ and F_{search} be the set of f_0 between $f_{0,min} = 70$ Hz and $f_{0,max} = 400$ Hz with 1 Hz step. The search is made using a downsampled signal $\mathbf{x}_{low} = \{x_{low}(n) : n = 0, \ldots, N_{low} - 1\}$ which is obtained from the narrowband speech signal after removing the DC component with a notch filter, lowpass filtering with a cutoff frequency of

800 Hz and downsampling with a factor of 4 (sampling rate = 2000 Hz). A Hamming window is applied to each of the 30 ms ($N_{low} = 60$ samples) speech frames and fast analysis/synthesis is made with the method proposed in Section 2.3. The evaluation of each possible f_0 is made using the SNR criterion (or equivalently the Mean-Square-Error criterion). The result is a sampling of the function $SNR(f; \mathbf{x}_{low})$ that links SNR to f_0 when signal $\mathbf{x}_{low}$ is analyzed.

For a fixed frame size, the number of sinusoids that describe the spectrum increases as f_0 decreases. Therefore, the function $SNR(f; \mathbf{x}_{low})$ is biased to favor lower fundamental frequencies. This bias is independent of the characteristics of the signal and an intuitive way to compute it would be to take an expectation of the SNR over all possible (energy normalized) signals $\mathbf{x}_{low}$:

$$SNR_{bias}(f) = E_{\mathbf{x}_{low}}\{SNR(f; \mathbf{x}_{low})\}, \tag{2.35}$$

where $E_{\mathbf{x}_{low}}\{\cdot\}$ denotes the expectation over the stochastic signal $\mathbf{x}_{low}$. However, the distribution of $\mathbf{x}_{low}$ is unknown and only the energy of $\mathbf{x}_{low}$ is known (i.e. it is normalized to unit energy). In this case, a plausible choice is to use the distribution that maximizes the entropy; to assume that $\mathbf{x}_{low}$ is a stochastic vector with $x_{low}(n) \sim N(0, \frac{1}{N_{low}})$. Therefore, the bias can be computed with stochastic integration of $SNR(f; \mathbf{x}_{low})$ using a white, gaussian noise model for $\mathbf{x}_{low}$:

$$SNR_{bias}(f; \mathbf{x}_{low}) = \frac{1}{L} \sum_{l=1}^{L} SNR(f; \mathbf{x}_{low,\, l}) \tag{2.36}$$

where L is the number of random realizations $\mathbf{x}_{low,\, l}$ that were used. Note that the expectation is computed on the logarithmic domain. The *coarse* pitch detection is made by peak-picking the *normalized* SNR function which is defined as:

$$SNR_{norm}(f; \mathbf{x}_{low}) = SNR(f; \mathbf{x}_{low}) - SNR_{bias}(f; \mathbf{x}_{low}) \tag{2.37}$$

The evaluation of the SNR bias is precomputed using $L = 10000$ realizations and stored in a table for fast access.

An example of the SNR, the SNR bias and the normalized SNR for a voiced speech frame lowpass filtered to 800 Hz and sampled at 2000 Hz is depicted in Figure 2.7. It is evident that $SNR(f; \mathbf{x}_{low})$ has a tilt which favors lower frequencies. This tilt is well captured by formula (2.36) so that the normalized SNR reveals a structure with a single dominant peak. The step-like structure of SNR_{bias} is associated with by the changing number of sinusoids that describe the 1000 Hz spectrum. The insertion/removal of a single sinusoid is quite evident because the total number of sinusoids that fit into a 1000 Hz spectrum is small. Sampling a narrowband spectrum or a wideband spectrum leads to much smoother $SNR_{bias}(f)$ functions.

After the computation of the normalized SNR, pick-peaking is used to get a number of pitch candidate values. The following rules are then used to select up to 3 pitch candidates:

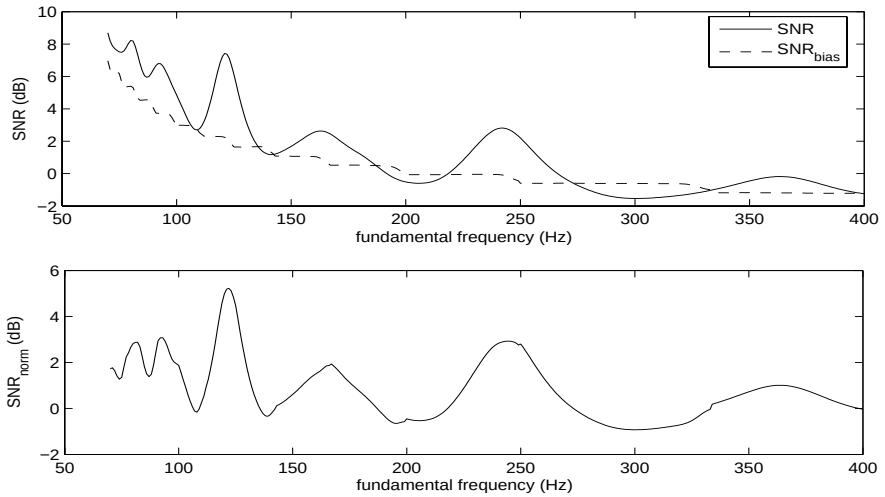


Figure 2.7 SNR, SNR bias and normalized SNR for a 30ms speech frame (sampling rate = 2000 Hz).

- remove weak peaks which are closer than 30 Hz to a stronger peak
- remove peaks which are more than 15 dB weaker than the strongest peak
- select at most 3 strongest peaks

Step 2: Fine Search

Let $F_0 = \{f_{0,i} : i = 1, ..., L_{cand}\}$ be the set of candidate pitch values estimated from the previous step. The purpose of this step is to refine the estimations by searching the full narrowband signal ± 10 Hz around each pitch candidate $f_{0,i}$, with a step of 1 Hz. The analysis/synthesis is made using a 20 ms analysis frame and a Hamming window. The "best" SNR peaks for the narrowband (4000 Hz) speech signal are usually within 10 Hz of the corresponding peaks for the 800 Hz lowpass speech signal. Therefore, a considerable complexity reduction is made when these peaks are exhaustively located at the 800 Hz lowpass signal and then refined at the full narrowband speech signal. Step 2 requires 21 analysis/synthesis operations.

Step 3: Super Fine Search

Let $F'_0 = \{f'_{0,i} : i = 1, ..., L_{cand}\}$ be the set of refined candidate pitch values estimated from step 2. The $f'_{0,i}$ has an accuracy of 1 Hz, corresponding to an 40 Hz error at the 40-th harmonic. The purpose of this step is to reduce this error by searching ± 0.9 Hz around each pitch candidate $f'_{0,i}$ with a step of 0.1 Hz. Step 3 requires 19 analysis/synthesis operations per candidate.

Step 4: Ambiguity Correction

Let $F''_0 = \{f''_{0,i} : i = 1, ..., L_{cand}\}$ be the set of refined candidate pitch values estimated from the step 3. This step selects the most probable pitch candidate using the method described in Section 2.5.1. The order of the cepstral envelope is set to P=16

and the smoothness Lagrangian to λ=0.004. In fact, until this step, the estimation of the fundamental frequency was treated as a generic signal analysis/synthesis problem. This step introduces speech-specific knowledge to the pitch estimator by penalizing non-smooth spectral envelopes per candidate.

Step 5: Pitch Halving Detection & Correction

The proposed pitch estimator is based on the assumption of "stationarity", which is fairly accurate in most voiced speech frames for an analysis interval of 20 ms. However, some transitional voiced speech frames violate this assumption; for example, when the time envelope of the signal rapidly ramps up or ramps down. The outcome of this deficiency is a single pitch halving error that can be detected and corrected with simple control logic using the past pitch values and the assumption that pitch evolves smoothly.

Performance and Complexity

The proposed pitch detection algorithm performs an exhaustive search to find the fundamental frequency that best describes the signal with a Harmonic Model. The exhaustive search is made on a subsampled version of the signal for computational efficiency. Then, a set of (maximum 3) peaks is refined in steps 2 and 3. A total of 40 full narrowband analysis/synthesis operations per peak. The cost of these operations is significantly reduced with the analysis/synthesis method proposed in Section 2.3 and the overall complexity of the pitch detector is affordable for modern DSP chips.

On the other hand, the proposed algorithm is accurate and suitable for harmonic models where it provides speech of high quality. Furthermore, we speculate that the proposed AbS pitch detector can be made much faster without losing robustness. For example, an autocorrelation criterion can be used to reduce the search space in step 1. Steps 2 and 3 can also benefit from a polynomial interpolation to reduce the search space. Finally, step 4 can benefit from heuristics that reduce the number of candidate pitch values.

2.5.3 Voicing Detection

The pitch detector allows the classification of speech frames in voiced frames and unvoiced frames. A speech frame is considered unvoiced when one of the following conditions hold for the 20 ms (160 samples) narrowband speech frame:

- the reconstruction SNR, when the harmonic amplitudes are sampled from the cepstral envelope, is lower than 2 dB
- the normalized SNR, the number of zero-crossings ζ and the fundamental frequency f_0 satisfy one of the following conditions: ($\zeta > 70$ and $SNR_{norm}(f_0; \mathbf{x}) <$ 3 dB) or ($\zeta > 80$ and $SNR_{norm}(f_0; \mathbf{x}) < 5$ dB) or ($\zeta > 90$ and $SNR_{norm}(f_0; \mathbf{x}) <$ 6 dB) or ($\zeta \geq 100$ and $SNR_{norm}(f_0; \mathbf{x}) < 8$ dB and $f_0 < 80$).

The zero-crossings ζ is an integer that counts the number of times two consecutive samples $x[n]$ and $x[n-1]$ have different signs ($x[n]x[n-1] < 0$) within the frame.

The voicing decision is biased towards labeling frames as voiced. The biased decision ensures that transitional frames and plosives are treated as voiced frames and it is important for the robustness of the sinusoidal speech codecs described in Chapter 9.

Chapter 3

High-Rate Quantization based on Gaussian Mixture Models

Efficient vector quantization at high rates is a difficult problem that has troubled source coding engineers for years. The main obstacle is that the complexity of the quantizer increases rapidly with the number of dimensions. Reduced complexity solutions are obtained by constraining the structure of the codevectors. For example, by setting constraint to trained codebooks [55] (ch. 12) or by constructing codebooks with a highly regular structure using lattice vector quantization [55] (ch. 10). Transform coding is a popular way to quantize multivariate data with very low complexity [55] (ch. 8). However, it is not efficient because it is a product code technique [55] (pg. 430) and the resulting codepoints fill empty regions of the P-dimensional space.

An extension to transform coding is made with GMM-based quantization in [88]. The basic idea is to assign a *different* transform quantizer to *local* regions of the P-dimensional space. The *local* transform quantizer is operating according to the local statistics of the source, thus resulting in a versatile quantization scheme that combines the computational efficiency of transform coding with near state-of-the-art performance.

This chapter provides the necessary background for the GMM-based quantization schemes that will be used in the next chapters. The focus is given on *resolution constrained quantization*, where each vector x is quantized using a predefined number of bits R. This type of quantization is typically used in speech coding, mainly due to network and end-to-end delay constraint. Section 3.1 reviews resolution constrained quantization of multivariate Gaussians, focusing on transform coding. Section 3.2 reviews GMM-based quantization techniques. The material in this chapter is largely adapted from [55] and [88] with insights and comments from other sources, when necessary.

3.1 Quantization for Multivariate Gaussians

This section presents a high-rate theory analysis for resolution constrained quantization of a scalar Gaussian variable, discusses bit allocation in transform coding of multivariate Gaussians and shows how companding can be used to quantize a $N(0,1)$ Gaussian variable without precomputed codebooks.

3.1.1 High-Rate Quantization of a Scalar Gaussian

Let $x \sim f_x(x)$ be a scalar random variable and $Q(.)$ a scalar quantizer. The Mean-Square-Error (MSE) distortion D_x is provided by the expectation:

$$D_x = \int_{-\infty}^{\infty} f_x(x)(x - Q(x))^2 dx = \sum_{i=1}^{N} \int_{x \in Q_i} (x - \hat{x}_i)^2 f_x(x) dx \tag{3.1}$$

where $\hat{x}_i$ is the i-th codeword, N is the total number of codewords, $f_x(.)$ is the source pdf and Q_i is the i-th quantization cell associated with $\hat{x}_i$. If the quantization rate is high, then the quantization cell Q_i is small enough to assume that the pdf $f_x(x)$ is constant in Q_i: $f_x(x) \approx \frac{p_x(\hat{x}_i)}{\Delta_i}$ when $x \in Q_i$, where $p_x(\hat{x}_i)$ is the probability of having the i-th cell and Δ_i is the length of Q_i. The average distortion is then approximated by:

$$D_x \approx \sum_{i=1}^{N} \frac{p_x(\hat{x}_i)}{\Delta_i} \int_{x \in Q_i} (x - \hat{x}_i)^2 dx = \frac{1}{12} \sum_{i=1}^{N} p_x(\hat{x}_i) \Delta_i^2 \tag{3.2}$$

The average distortion can be rewritten as:

$$D_x \approx \frac{1}{12} \sum_{i=1}^{N} p_x(\hat{x}_i) \left(N g(\hat{x}_i)\right)^{-2} \tag{3.3}$$

where $g(\hat{x}_i) = \frac{1}{N\Delta_i}$. As $N \to \infty$, $\Delta_i \to 0$, $g(x)$ becomes the so called *point density function* and represents the probability density function that describes the distribution of the codepoints $\hat{x}_i$. Therefore, at high rates, the average distortion can be approximated by the following integral [55] (pg. 163):

$$D_x \approx \frac{1}{12N^2} \int_{-\infty}^{\infty} f_x(x) g(x)^{-2} dx \tag{3.4}$$

The *optimal* point density function $g_{opt}(x)$ for resolution-constrained vector quantization can be obtained if the average distortion D_x is minimized with respect to the constraint that $g(x)$ is a pdf: $\int_{-\infty}^{\infty} g(x) = 1$. The solution can be obtained by means of variational calculus:

$$g(x)_{opt} = \frac{f_x(x)^{\frac{1}{3}}}{\int_{-\infty}^{\infty} f_x(x)^{\frac{1}{3}} dx} \tag{3.5}$$

Note that the optimal distribution of the codepoints is different from the distribution of the samples. If we combine equations (3.5) and (3.4) for the $N(0, \sigma^2)$ Gaussian case, it is straightforward to show that the optimal average distortion is provided by formula [55] (pg. 228):

$$D_{x,opt} = Q_c \sigma^2 2^{-2R} \tag{3.6}$$

where $Q_c = \frac{\sqrt{3}\pi}{2}$ is the quantization constant and $R = \log_2(N)$ is the rate in bits.

3.1.2 Bit Allocation for Transform Coding

Let $\mathbf{x} \in \Re^P$ be a zero mean multivariate Gaussian random variable with covariance matrix Σ_x and R be the total rate for $\mathbf{x}$. The total rate is the sum of the individual rates r_p of each variable:

$$R = \sum_{p=1}^{P} r_p. \tag{3.7}$$

If $\Sigma_{\mathbf{x}}$ is not diagonal, the Karhunen-Loeve Transform (KLT) can be used to decorrelate the variables. It can be shown that the KLT is the optimal transform for high-rate quantization of a multivariate Gaussian vector [55] (pg. 242). Without loss of generality, we can assume that $\Sigma_x = diag(\sigma_1^2,\ \sigma_2^2,\ \ldots,\ \sigma_P^2)$. Let $Q(.)$ be the quantizer of $\mathbf{x}$ and $\hat{\mathbf{x}} = Q(\mathbf{x})$ be the quantized version of x. The quality of the quantization is addressed using a square-error distortion measure:

$$d(\mathbf{x}, \hat{\mathbf{x}}) = \sum_{p=1}^{P} (x_p - \hat{x}_p)^2 \tag{3.8}$$

where x_p and $\hat{x}_p$ are the p-th variables of $\mathbf{x}$ and $\hat{\mathbf{x}}$ respectively. The average distortion can be provided using the high rate theory presented in the previous subsection:

$$D = \sum_{p=1}^{P} Q_c \sigma_p^2 2^{-2r_p} \tag{3.9}$$

Finding the optimal bit allocation is a well-known optimization problem and near-optimal solutions are typically used. If we relax the constraints and allow r_p to take non-integer or even negative values, we can use Lagrangian methods to analytically minimize the average distortion D for the optimal r_p under the constraint (3.7), then the following bit allocation [55] (pg. 229) is obtained:

$$r_p = \frac{R}{P} + \frac{1}{2}\log_2\left(\frac{\sigma_p^2}{c}\right),\ p = \{1, ..., P\}, \tag{3.10}$$

where

$$c = \left(\prod_{p=1}^{P} \sigma_p^2\right)^{1/P}, \tag{3.11}$$

is the geometric mean of the variances. A convenient expression for the average distortion D can then be obtained from equations (3.9) and (3.10):

$$D = Q_c P c 2^{-2R/P} \tag{3.12}$$

Note however that formula (3.10) provides continuous rates not necessarily corresponding to integer sized codebooks. The resulting rates can even be negative when the variables have small variances. A typical suboptimal solution to the problem of non-integer rates is to use the codebook sizes $\lfloor 2^{r_p} \rfloor$ as an initial guess and distribute the remaining bits with a greedy approach as in [55] (pg. 234).

3.1.3 Companding and Lattices

Companding can be used to avoid storing precomputed codebooks for the quantization of scalar Gaussians. The idea behind companding is to introduce an one-to-one invertible mapping $G(.) : \Omega_x \rightarrow \Omega_u$ which maps the support Ω_x of x-space onto a finite support Ω_u of a random variable $u = G(x)$ with approximately uniform distribution over Ω_u. The new variable u is easily quantized to $\hat{u}$ using a lattice quantizer (for example a simple uniform quantizer) and the quantized value of x is obtained with the inverse mapping $\hat{x} = G^{-1}(\hat{u})$.

The optimal compander for a scalar Gaussian random variable x and a mean square error distortion measure under high rate assumptions can be found in closed form [89]:

$$G(x) = \frac{1}{2}\left(1 + erf(\frac{x}{\sqrt{6}})\right) \tag{3.13}$$

where $erf(a) = \frac{2}{\pi}\int_0^a e^{-x^2}dx$ is the error function. However, since the derivation is based on high rate assumptions, companding is not effective when the Gaussians are encoded in low rates. In our experiments, we found that it is better to use trained codebooks for rates lower than 5 bits and companding for higher rates. The total size of these codebooks is low: $\frac{32(32+1)}{2} = 528$ codewords. Furthermore, depending on the complexity/accuracy of the implementation of the $erf(.)$ function and its inverse $erf^{-1}(.)$, it is faster to use codebooks for the lower rates.

Transform coding based on cartesian companding (that uses a scalar compander like $G(.)$ to each of the variables in vector $\mathbf{x}$) and scalar quantizers is suboptimal because of the so-called *space filling loss*; the fact that the shape of the rectangular quantization cell is not optimal for more than one dimension [90], [91]. Therefore, some researchers have proposed the use of lattice vector quantizers [55] that reduce the space filling loss [88] at the expense of relatively increased computational complexity. However, the use of cartesian companding is not justified in the multi-dimensional case. It is merely a practical choice associated with the difficulties arising upon the design of optimal multi-dimensional companders [92]; namely, the interactions between the compander and the lattice [93].

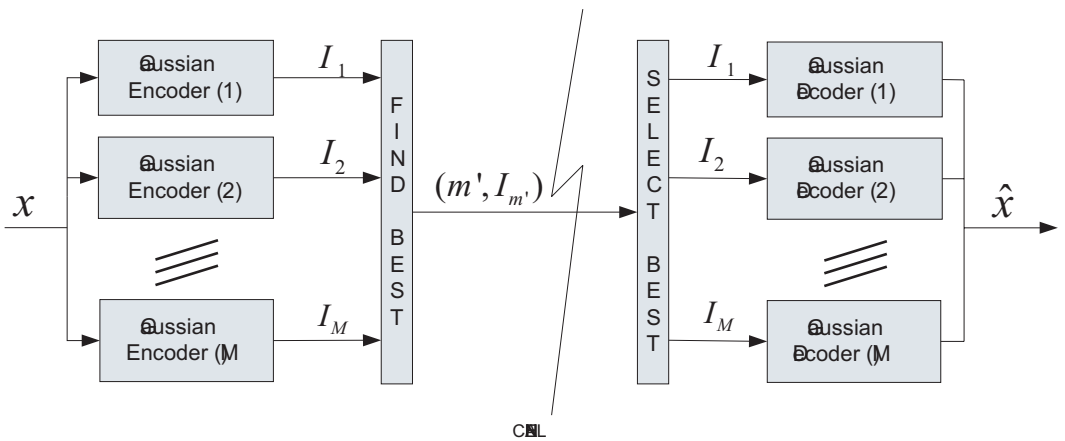


Figure 3.1 Basic GMM-based Vector Quantization scheme.

3.2 Quantization based on Gaussian Mixture Models

The design of an optimal unconstrained vector quantizer (for example an LBG-like quantizer [55]) can be viewed as a procedure which optimizes the positioning of the codevectors according to the point density function of the codevectors. Therefore, an unconstrained vector quantizer has the dual task of capturing the statistics of the point density function (or equivalently the pdf of the source), as well as the optimal local arrangement of the quantization cells. GMM-based quantizers [88], [94] effectively decouple the estimation of the source statistics from the optimal allocation of the codepoints. The statistics of the source are modeled with a Gaussian Mixture Model, while the optimal allocation of the codepoints is computed using high rate theory assumptions. The result is a versatile quantizer that enables variable/high rate operation, and state-of-the-art tradeoff between complexity and performance.

3.2.1 Encoding/Decoding Process

The encoding/decoding process is depicted in Figure 3.1. The basic idea is to encode the data vector $\mathbf{x}$ with each of the M Gaussian encoders and to transmit the indices $I_{m'}$ of the "best" encoding together with the index m' of the corresponding Gaussian quantizer. The Gaussian encoder/decoder can be constructed according to the discussion in Section 3.1. The decoder receives the two indices $(m', I_{m'})$ and selects the m'-th Gaussian decoder to decode $I_{m'}$ in order to obtain the quantized vector $\mathbf{x}' = Q_{m'}(\mathbf{x})$, where $Q_{m'}(.)$ is the corresponding Gaussian quantizer.

Let $\mathbf{x} \sim GMM(\alpha_{x,m}, \mu_{x,m}, \Sigma_{x,m})$, $m = \{1, 2, \ldots, M\}$ be a P-dimensional source

that is modeled using a GMM with M Gaussians, where $\alpha_{x,m}$ is the prior probability, $\mu_{x,m}$ is the mean and $\Sigma_{x,m}$ is the covariance matrix of the m-th Gaussian component. Each Gaussian is encoded using the KLT transform provided by the eigenvalue decomposition of the corresponding covariance matrix $\Sigma_{x,m}$:

$$\Sigma_{x,m} = V_{x,m}\Lambda_{x,m}V_{x,m}^T \tag{3.14}$$

where the columns of $V_{x,m}$ are the eigenvectors of $\Sigma_{x,m}$ and

$$\Lambda_{x,m} = diag(\sigma_{m,1}^2,\ \sigma_{m,2}^2,\ \ldots,\ \sigma_{m,P}^2)$$

is a diagonal matrix with the eigenvalues (variances) $\sigma_{m,p}^2$ on it's diagonal.

The m-th Gaussian encoder assumes that the statistics of $\mathbf{x}$ follow the statistics of the m-th Gaussian component of the GMM, namely $N(\mu_{x,m}, \Sigma_{x,m})$. Vector $\mathbf{x}$ is translated and rotated in order to obtain a zero mean vector $\mathbf{x}'_m$ with diagonal covariance matrix $\Lambda_{x,m}$:

$$\mathbf{x}'_m = V_{x,m}^T(\mathbf{x} - \mu_{x,m}). \tag{3.15}$$

The *uncorrelated* vector $\mathbf{x}'_m$ is then quantized with a series of scalar quantizers to obtain $\hat{\mathbf{x}}'_m = Q(\mathbf{x}'_m)$. The "best" encoding m' is selected according to a square-error criterion:

$$m' = \arg\min_m \|x'_m - \hat{x}'_m\|_2^2 . \tag{3.16}$$

Note that no rotation is needed since the distortion is not affected by $V_{x,m}^T$ which is a unitary transform. The m'-th Gaussian decoder performs the inverse operation. The transmitted indices $I_{m'}$ are used to decode the corresponding value $\hat{\mathbf{x}}_{m'}$, which is then rotated and translated to obtain the resulting codevector:

$$\hat{\mathbf{x}} = \mu_{m'} + V_{x,m'}\hat{\mathbf{x}}_{m'}. \tag{3.17}$$

3.2.2 Quantizer Bit Allocation

Each of the M multivariate Gaussian quantizers $Q_m(.)$ may operate at a different rate R_m. Let R be the total encoding rate:

$$R = \sum_{m=1}^{M} R_m. \tag{3.18}$$

High-rate theory assumptions can be used to find the *optimal* rate. We will examine the case where $Q_m(.)$ are transform-based Gaussian quantizers. In that case, the average distortion from each quantizer is provided by formula (3.12). If we assume that the Gaussians of the GMM are well separated, the average distortion from the GMM-based quantizer can be approximated by the summation of the individual distortions:

$$D_{GMM} \approx \sum_{m=1}^{M} \alpha_m Q_c P c_m 2^{-2R_m/P}, \tag{3.19}$$

where c_m is the geometric mean of the variances $\sigma^2_{m,p}$:

$$c_m = \left(\prod_{p=1}^{P} \sigma^2_{m,p} \right)^{1/P} . \tag{3.20}$$

Lagrangian optimization can then be used to minimize D_{GMM} for R_m under the rate constraint (3.18). The *optimal* quantizer bit allocation can be shown to be [88]:

$$R_m = R + \log_2 \frac{(\alpha_m c_m)^{\frac{P}{P+2}}}{\sum_{m'=1}^{M} (\alpha_{m'} c_{m'})^{\frac{P}{P+2}}} \tag{3.21}$$

3.3 Example: Quantization of Cepstral Envelopes

An experiment was conducted using GMM-based quantization in order to evaluate the quantization rate for the cepstral envelopes. The training set of the TIMIT database was analyzed using the harmonic analysis described in Chapter 2. Speech was analyzed/synthesized using 20 ms frames, 10 ms overlapping and a Hanning window. Pitch was estimated according to Section 2.5. A 20-th order Bark-scale regularized cepstral envelope was computed for each frame, according to Section 2.4. The Expectation Maximization (EM) algorithm [95] was used to estimate a GMM with 16 components from 400.000 training samples. The evaluation was made in terms of PESQ-MOS [5] with a subset of TIMIT test set consisting of 256 male utterances and 256 female utterances. The test set utterances were synthesized using the quantized cepstral envelopes for quantization rates between 20 bits/frame and 70 bits/frame. All other parameters (pitch, voicing and phases of the harmonics) were not quantized. The experiments were made separately for males and females since these groups behave differently in terms of PESQ-MOS score. Note that no post-filtering is applied to the cepstral envelopes. The corresponding PESQ-MOS scores for unquantized cepstral envelopes are depicted in Figure 3.2 with dashed horizontal lines. The following observations can be made:

- The PESQ-MOS saturates for rates above 60 bits.
- There is a graceful degradation of PESQ-MOS with decreasing rate.

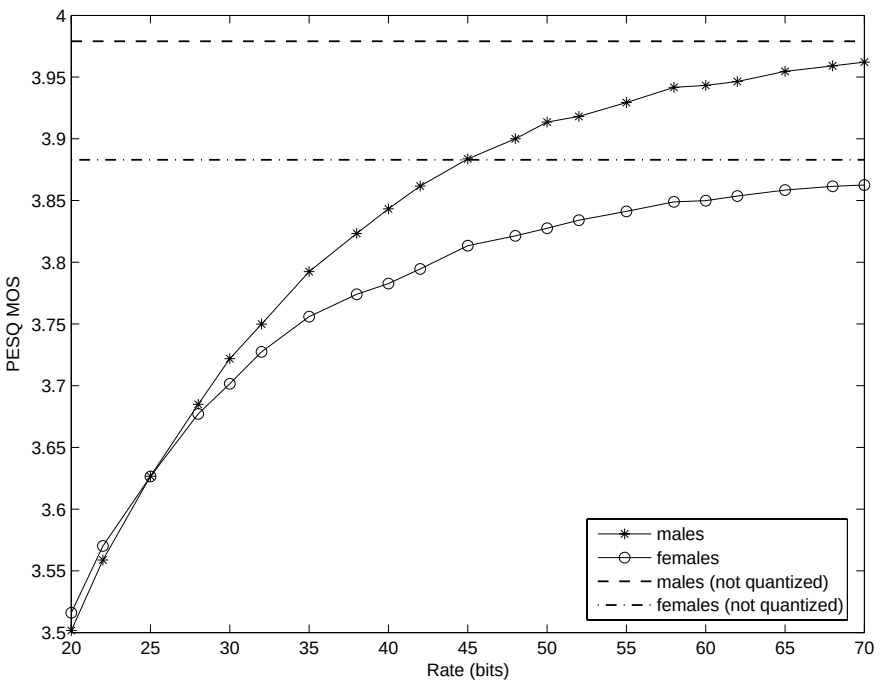


Figure 3.2 Relationship between rate and PESQ-MOS for the quantization of cepstral envelopes using a GMM-based quantizer.

Chapter 4

Stochastic Modeling and Quantization of Harmonic Phases

4.1 Overview

The spectrum of a voiced speech sound is typically treated as if it consists of two parts: an *amplitude spectrum* and a *phase spectrum*. The statistical behavior and the properties of amplitude spectra are well known and have many applications in Speech Processing. On the contrary, phase spectra are usually disregarded due to the intrinsic difficulties associated with the accurate and robust modeling of the phases in voiced speech. However, there are several studies that indicate the importance of phase in speech perception [96], [97], [98].

There is a considerable literature on phase modeling and related problems like the estimation of group delay spectra (also referred to as *group delay processing*) and glottal flow estimation. Understanding and modeling phase is very important for speech coding in the sense that a robust and computationally efficient phase modeling algorithm can also be used for coding. Furthermore, they can provide insight to the designer of a speech codec. It is beyond the scope of this thesis to review these methods and focus will be given to the application of speech coding. The interested reader can find a detailed review of glottal flow estimation techniques in [99], [100] and of group delay processing in [101].

In CELP coding, the innovative excitation is typically encoded with a closed-loop codebook search. Therefore, phases and fine-spectral details which are not captured by the AR spectral envelope are blindly encoded together [29].

On the other hand, sinusoidal coders rely on a phase model to reconstruct speech. For example, in codecs based on STC (Sinusoidal Transform Coding) and MBE (Multi-Band Excitation), the harmonics are classified as voiced or unvoiced, and a phase model is used to construct phases that provide pleasant speech. The voiced harmonics are constructed using the assumption that the excitation is a zero phase signal, while the unvoiced harmonics are constructed using random phases [1], [29], [45]. The zero phase model is not an accurate assumption because the excitation corresponds

to the glottal flow derivate (according to the source-filter model of speech production) and the glottal flow is well modeled by maximum phase systems [45] (pg. 151). This poses an upper bound to the quality of encoded speech at higher bit-rates but, in practice, it works well at low bit-rate coders (below 4 kbps). As a consequence, many researchers argue that high-quality sinusoidal speech coding requires the encoding of phases.

A *model-based* approach is to fit a deterministic model to the excitation or directly to the sinusoidal phases ϕ_k. In [102], the excitation is constructed using a Rosenberg glottal pulse model [103]. Another idea is to use all-pass filters to correct the phase response of the minimum phase AR spectral envelope [104], [105]. A drawback of the latter methods is that the resulting all-pass filters may be unstable. The parameters of the all-pass filter can also be computed in the frequency domain [106] by minimizing a squared-error criterion that is used directly on the phases, but this distortion measure is prone to errors due to the modulo-2π behavior of the phases.

The harmonic phases ϕ_k can also be quantized without the requirement of a deterministic model. In [107], [108], the *phase residual*, the difference between the phase of the current frame and its prediction from the previous frame is quantized. Vector quantization of phases was proposed in [48] for the quantization of the harmonic phases of the SEW (Slowly Evolving Waveform) in the context of WI (Waveform Interpolation) coders. An important contribution of the latter work is the introduction of a distortion measure that takes into account the modulo-2π behavior of phases, and the derivation of the corresponding k-means algorithm. However, codebook-based phase quantizers cannot operate at increased bit-rates. A GMM-based phase quantization algorithm capable of operating at high rates was provided in [54], but the quantizer restricts the GMM to $(0, 2\pi]$ and does not take into account the modulo-2π behavior of the phase data.

A comparative evaluation of these algorithms is not always possible due to the lack of a widely accepted phase distortion measure and to the strong coupling between the phase quantizer and the analysis/synthesis procedure of the sinusoidal coder. An important limitation is that the typically used squared-error distortion measure between the original and the reconstructed waveform does not correlate well with the perceived distortion at low/medium rates [109]. Another option is to compute the distortion directly on the phases ϕ_k. A psychoacoustic study of a simple difference phase distortion measure is made in [97], [110] to facilitate perceptual weighting of the harmonic phases.

This chapter proposes a novel phase modeling and quantization method. Phases are not modeled in a deterministic manner (i.e., through a glottal flow model or an all-pass filter), but in a statistical manner as multivariate circular random variables. Raw phase data have an approximately uniform distribution. Section 4.2 describes a procedure to determine the translation that aligns the waveforms according to a reference point within the glottal cycle. Processing of the phases of the aligned waveforms reveal the presence of dependencies between the harmonic phases. This motivates the construction of a vector quantizer for phases. Since phase data exhibit a circular

behavior, Section 4.3 provides the necessary background for *circular statistics*, giving emphasis to the *wrapped Gaussian distributions*. Section 4.4 presents a GMM suitable for circular spaces; the so-called *Wrapped Gaussian Mixture Model* (WGMM). A detailed derivation of an Expectation-Maximization algorithm for training is provided and focus is given to the case where the Gaussian components have diagonal covariance matrices. Section 4.5 discusses the construction of a quantizer that is based on WGMM, using a distortion measure that is suitable for circular spaces. Subsection 4.5.1 proposes the construction of scalar quantizers for wrapped Gaussian variables by wrapping codebooks made for linear Gaussian variables. Two WGMM bit-allocation algorithms for these quantizers are then proposed. However, wrapping linear Gaussian codebooks is sub-optimal when the linear Gaussian pdf does not approximate well the wrapped Gaussian pdf. A better quantizer for wrapped Gaussian random variables is proposed in Section 4.5.2 by introducing the concept of Polynomial CodeFunctions (PCF). In PCF-based quantization, the construction of a codebook for a specific variance σ^2 is made by sampling a set of polynomial functions. A k-means-like training algorithm is provided along with a bit-allocation procedure for WGMM. Finally, Section 4.6 evaluates the proposed quantizers for phase quantization of narrowband speech.

4.2 Harmonic Phase Decomposition

Let ϕ_k, $k = 1, \ldots, K$ denote the harmonic phases. Phase can be decomposed to a minimum phase term $\angle H_s(\omega)$, a *linear phase term* $k\omega_0\tau$ and a *dispersion term* ψ_k:

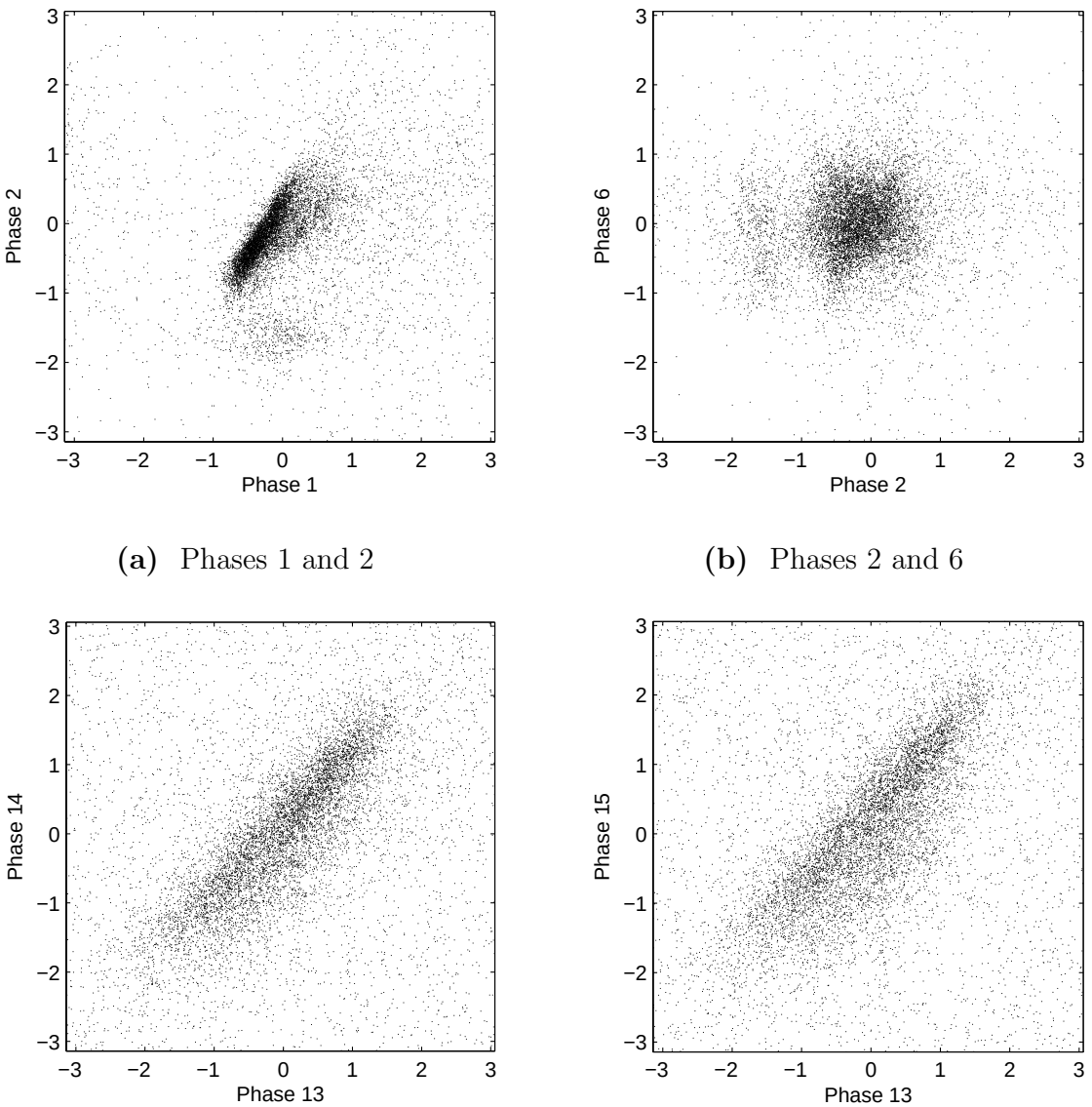
$$\phi_k = k\omega_0\tau + \angle H_s(k\omega_0) + \psi_k \tag{4.1}$$

The dispersion phase term ψ_k corresponds to the phase of the excitation signal since the subtraction of the minimum phase term corresponds to inverse filtering with the linear system $H(\omega)$. The excitation signal $e(n)$ can be reconstructed according to the formula:

$$e(n) = \sum_{k=1}^{K} \cos(k\omega_o(n - n_0) + k\omega_0\tau + \psi_k),\ n = 0, \ldots, N-1, \tag{4.2}$$

where $n_0 = \frac{N-1}{2}$ is the center of the analysis frame. The linear phase term $k\omega_0\tau$ corresponds to a τ-sample translation of the excitation with respect to a reference point inside the pitch period. As a reference point, we used the maximum peak of the excitation $e(n)$ within a single pitch period. The peak-picking is performed on a uniformly sampled version of the excitation $e(n)$ using 128 samples (7 bits). We found that this procedure provided robust reference points within the glottal cycle.

Some insight regarding the distribution of ψ_k can be obtained from the two-dimensional marginal distributions between phases. The underlying marginal pdf can be visualized with a scatter plot of the corresponding samples. Figure 4.1 plots



(a) Phases 1 and 2

(b) Phases 2 and 6

(c) Phases 13 and 14

(d) Phases 13 and 15

Figure 4.1 Scatter plots of two harmonic phases. The phases were extracted from the excitation of narrowband speech frames with pitch between 95 Hz and 115 Hz. The mean phase of the dataset was removed.

the samples that correspond to phases ψ_1 and ψ_2, ψ_2 and ψ_6, ψ_{13} and ψ_{14}, ψ_{13} and ψ_{15}. The samples were extracted from speech frames with pitch between 95 Hz and 115 Hz, and the (circular) mean phase of the dataset was removed. The marginal distributions of ψ_k reveal the presence of structure and covariation within the phase parameters ψ_k. This is an important observation that provides justification for vector quantization of phases.

4.3 Circular Statistics

Let $\vec{\psi} = [\psi_1, \ , \psi_2, \ \ldots, \ \psi_K \ ]^T$ be a vector that contains the phases ψ_k, $k = 1, 2, \ldots, K$. Phases exhibit a modulo-2π periodic behavior in the sense that the excitation signal $e(n; \vec{\psi}) = e(n; \vec{\psi} + \mathbf{u}2\pi)$, $\mathbf{u} \in \mathbb{Z}^K$. Thus, for each time instant n, the excitation signal $e(n; \vec{\psi})$ is a function on the surface of an "*n-Torus*" manifold defined as $\mathbb{T}^K = \mathbb{R}^K / 2\pi\mathbb{Z}^K$. The $\mathbb{T}^1$ n-Torus is the unit circle, while $\mathbb{T}^K = \mathbb{T}^1 \times \mathbb{T}^1 \times \ldots \times \mathbb{T}^1$ is the K times product of $\mathbb{T}^1$. The corresponding statistics are called *circular (or directional) statistics* and the random variables $\vec{\psi}$ are called *circular (or directional) random variables*. The material in this section is largely adapted from [111].

4.3.1 Circular Mean and Circular Variance

Let $\theta \in \mathbb{T}^1$ be a circular random variable distributed according to the periodic probability density function $f(\theta) = f(\theta + w2\pi)$, $w \in \mathbb{Z}$, and let θ_n, $n = 1, \ldots, N$ be N samples drawn from $f(.)$. Since $f(\theta)$ is a pdf, $f(\theta) \geq 0$, $\forall\ \theta\ \in \mathbb{R}$ and $\int_0^{2\pi} f(\theta) d\theta = 1$. The *circular mean* $\mu_{\theta,c}$ and the *circular variance* $\sigma^2_{\theta,c}$ of θ are defined as [111] (pg. 20):

$$\text{Circular Mean} : \mu_{\theta,c} = \arg\left(\mathrm{E}\{e^{j\theta_n}\}\right) \tag{4.3}$$

$$\text{Circular Variance} : \sigma^2_{\theta,c} = 1 - \left|\mathrm{E}\{e^{j\theta_n}\}\right|, \tag{4.4}$$

where $\mathrm{E}\{.\}$ denotes the expectation operator and j is the imaginary unit ($j^2 = -1$). The circular mean $\mu_{\theta,c}$ measures the mean direction of the data and $\sigma^2_{\theta,c} \in [0, 1]$.

4.3.2 Wrapped Univariate Gaussian Distribution

Let $g(\theta)$, $\theta \in \mathbb{R}$, be the pdf of a distribution defined on a line. A *circular distribution* that is defined on $\mathbb{T}^1$ can be obtained by wrapping $g(.)$ around the circumference of the unit circle. The random variable θ_w of the wrapped pdf $g_w(.)$ is given by:

$$\theta_w = \langle \theta \rangle_{2\pi}, \tag{4.5}$$

where $\langle \theta \rangle_{2\pi} \equiv \theta \ mod\ 2\pi$ denotes the modulo-2π operation. The wrapped pdf is then obtained from infinite repetitions of $g(.)$ at regular 2π intervals:

$$g_w(\theta_w) = \sum_{w=-\infty}^{\infty} g(\theta_w + w2\pi), \ \theta_w \in (0, 2\pi] \tag{4.6}$$

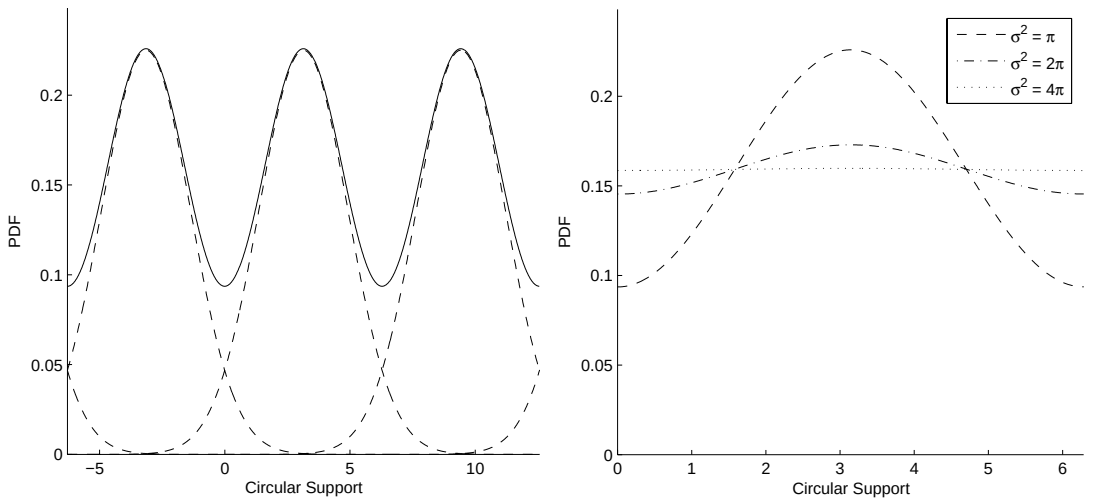


Figure 4.2 Examples of scalar wrapped Gaussian pdf with $\mu_{\theta,w} = \pi$. The left plot shows the Gaussian component (dashed line) that is wrapped and the corresponding wrapped pdf (with $\sigma^2_{\theta,w} = \pi$). The right plot shows three examples of wrapped pdfs with variances $\sigma^2_{\theta,w} = \{\pi, 2\pi, 4\pi\}$.

.

If $g(.)$ is a univariate Gaussian distribution then the *wrapped univariate Gaussian distribution* is given by [111] (pg. 55):

$$N_w(\theta_w; \mu_{\theta,w}, \sigma^2_{\theta,w}) = \frac{1}{\sqrt{2\pi\sigma^2_{\theta,w}}} \sum_{w=-\infty}^{\infty} \exp\left(-\frac{1}{2\sigma^2_{\theta,w}}\left(\theta_w - \mu_{\theta,w} - w2\pi\right)^2\right), \qquad (4.7)$$

where $\mu_{\theta,w}$ and $\sigma^2_{\theta,w}$ is the mean and the variance of the wrapped Gaussian, respectively. The mean $\mu_{\theta,w}$ and the variance $\sigma^2_{\theta,w}$ of the wrapped Gaussian is related to the circular mean $\mu_{\theta,c}$ and the circular variance $\sigma^2_{\theta,c}$ by:

$$\mu_{\theta,w} = \langle \mu_{\theta,c} \rangle_{2\pi} \qquad (4.8)$$

$$\sigma^2_{\theta,w} = -2\log\left(1 - \sigma^2_{\theta,c}\right). \qquad (4.9)$$

Several useful properties that hold for Gaussian distributions also hold for the wrapped Gaussian distributions. For example, the distribution is *unimodal* and *symmetric* around $\mu_{\theta,w}$. It possesses the *additive* property (a sum of wrapped Gaussian random variables is also a wrapped Gaussian random variable) [111] (pg. 56) and it appears in the *Central Limit theorem* on the circle [111] (pg. 90). An interesting note is that $N_w(\theta_w; \mu_{\theta,w}, \sigma^2_{\theta,w})$ tends to the uniform distribution when $\sigma^2_{\theta,c} \to 1$ or equivalently when $\sigma^2_{\theta,w} \to \infty$. Figure 4.2 depicts an example of a wrapped Gaussian with variance $\sigma^2_{\theta,w} = \pi$. Larger variances lead to more uniform distributions. Finally, the wrapped

Gaussian can be approximated by the linear Gaussian when the variance $\sigma^2_{\theta,w} \leq 1$, as shown in [112].

4.3.3 Wrapped Multivariate Gaussian Distribution

The *wrapped multivariate Gaussian distribution* can be obtained by wrapping a multivariate Gaussian to the surface of the n-Torus $\mathbb{T}^K$. This corresponds to an infinite tiling of the multivariate Gaussian on a K-dimensional grid with 2π intervals. Let $\vec{\theta} \in \mathbb{R}^K$ be the (unwrapped) phase vector and $p(\vec{\theta})$ the corresponding pdf. The following must hold:

- $p(\vec{\theta}) \geq 0$,
- $\underbrace{\int \int \dots \int_0^{2\pi}}_{K \text{ times}} p(\vec{\theta}) d\vec{\theta} = 1$,
- $p(\vec{\theta}) = p(\vec{\theta} + \vec{w}2\pi), \vec{w} \in \mathbb{Z}^K$.

Therefore, the wrapped multivariate Gaussian distribution can be defined as:

$$p(\vec{\theta}_w; \vec{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^K |\Sigma|}} \sum_{\vec{w} \in \mathbb{Z}^K} \exp\left(-\frac{1}{2}(\vec{\theta}_w - \vec{\mu} - \vec{w}2\pi)^T \Sigma^{-1} (\vec{\theta}_w - \vec{\mu} - \vec{w}2\pi)\right) \tag{4.10}$$

where $\vec{\theta}_w = \langle \vec{\theta} \rangle_{2\pi} \in (0, 2\pi]^K$, $\vec{\mu}$ and Σ are the mean and the covariance matrix of the multivariate Gaussian, respectively. For notational simplicity, in the following text we will assume that all circular random variables are confined to their principal value in $(0, 2\pi]$.

An application of wrapped Gaussian models can be found in [112], where wrapped multivariate Gaussians and *semi-wrapped* multivariate Gaussians (which model sources with circular and non-circular data) are proposed for handwriting recognition.

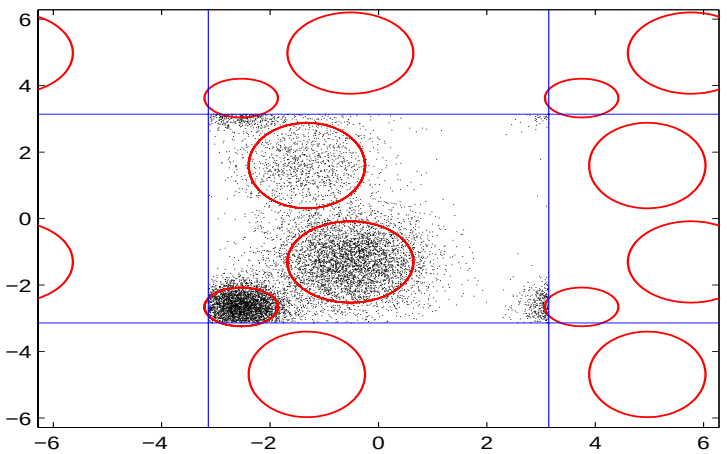


Figure 4.3 An example of a two-dimensional WGMM with diagonal covariance matrices. The ellipses correspond to iso-contours of the Gaussian kernel.

4.4 Wrapped Gaussian Mixture Model estimation using Expectation-Maximization

We propose to model the harmonic phases using a *mixture of wrapped multivariate Gaussian distributions*. Wrapped Gaussian Mixture Models (WGMM) can model a wide range of variables that exhibit a modulo-2π behavior. However, up to our knowledge, only a few recent publications utilize wrapped mixture models to model circular (or directional) data: In [113], *wrapped Hidden Markov Models* (HMM) are used to track the trajectories of sound sources inside a room. In [114], wrapped (*Normal*, *Cauchy*) mixture models are used to study time series with linear and circular variables. An Expectation-Maximization (EM) algorithm for wrapped multivariate Gaussians and an extension to HMM is presented in [113] for the case of Gaussian components with diagonal covariance matrices. However, the EM algorithm provided in [113] estimates the parameters by performing the EM steps one dimension at a time. This restriction is not necessary as it will be shown. This section presents an EM algorithm for a WGMM with full covariance matrices and then focuses to the more tractable case where the Gaussian components have diagonal covariance matrices.

Let $p(\vec{\theta};\Omega)$, $\vec{\theta} \in (0, 2\pi]^K$ be the pdf of a WGMM with M wrapped Gaussians and a set of parameters $\Omega = \{\alpha_m, \vec{\mu}_m, \Sigma_m : m = 1, \ldots, M\}$, with α_m, $\vec{\mu}_m$ and Σ_m being the m-th Gaussian weight, mean and covariance matrix, respectively:

$$p(\vec{\theta};\Omega) = \sum_{\vec{w}\in\mathbb{Z}^K} \sum_{m=1}^{M} p(m, \vec{w}, \vec{\theta};\Omega), \tag{4.11}$$

where

$$p(m, \vec{w}, \vec{\theta}; \Omega) = \alpha_m p(\vec{w}, \vec{\theta}|m; \Omega), \tag{4.12}$$

$$\alpha_m = p(m; \Omega), \tag{4.13}$$

$$p(\vec{w}, \vec{\theta}|m; \Omega) = \frac{1}{\sqrt{(2\pi)^K |\Sigma_m|}} \exp\left(-\frac{1}{2}(\vec{\theta} - \vec{\mu}_m - \vec{w}2\pi)^T \Sigma_m^{-1} (\vec{\theta} - \vec{\mu}_m - \vec{w}2\pi)\right), \tag{4.14}$$

is the $\vec{w}$-th tiling of the m-th translated Gaussian component. Note that m is the discrete random variable that states the Gaussian component and that it is interchangeably used as an index for notational simplicity. An example of a two-dimensional WGMM with diagonal covariance matrices is depicted in Figure 4.3. The ellipses are iso-contours of the Gaussian kernel and the dots correspond to random samples generated according to the WGMM pdf. For visualization purposes, the support is translated to $(-\pi, \pi]$.

The estimation of the model parameters Ω from N data samples $\vec{\theta}_n \in (0, 2\pi]^K$, $n = 1, \ldots, N$ can be made using a *Maximum Likelihood* (ML) criterion. The corresponding log-likelihood of WGMM is:

$$L(\Omega) = \sum_{n=1}^{N} \ln\left(p(\vec{\theta}_n; \Omega)\right). \tag{4.15}$$

The maximization of $L(\Omega)$ over all Ω is a difficult optimization task. However, it can easily be addressed with a two-step algorithm that belongs to the class of *Expectation-Maximization* algorithms [95], [115]. This treatment is suitable for mixture models and may lead to closed-form solutions. The *Expectation-Maximization* algorithm can be viewed as an iterative *bound optimization* algorithm in the sense that at each iteration the log-likelihood $L(\Omega)$ is lower bounded with another function $Q(\Omega; \Omega_0)$: $L(\Omega) \geq Q(\Omega; \Omega_0)$ which is an optimal (*potentially tight*) bound in Ω_0 (an estimation of Ω made in the previous iteration) that is easier to optimize than $L(\Omega)$ [116].

We will provide an EM algorithm for WGMM. First we will bound the log-likelihood using a set of parameters $q_n(m, \vec{w})$ with the following properties:

$$q_n(m, \vec{w}) > 0, \ \forall \ m, \ \vec{w} \tag{4.16}$$

$$\sum_{m=1}^{M} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w}) = 1 \tag{4.17}$$

Each component probability inside the log-likelihood is multiplied and divided by

$q_n(m, \vec{w})$:

$$\begin{aligned} L(\Omega) &= \sum_{n=1}^{N} \ln \left(\sum_{\vec{w} \in \mathbb{Z}^K} \sum_{m=1}^{M} p(m, \vec{w}, \vec{\theta}_n; \Omega) \right) \\ &= \sum_{n=1}^{N} \ln \left(\sum_{\vec{w} \in \mathbb{Z}^K} \sum_{m=1}^{M} p(m, \vec{w}, \vec{\theta}_n; \Omega) \frac{q_n(m, \vec{w})}{q_n(m, \vec{w})} \right) \\ &\geq \sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} \sum_{m=1}^{M} q_n(m, \vec{w}) \ln \left(\frac{p(m, \vec{w}, \vec{\theta}_n; \Omega)}{q_n(m, \vec{w})} \right) \\ &\equiv Q(\Omega, q_n(m, \vec{w})), \end{aligned} \tag{4.18}$$

where the lower bound of the log-likelihood $Q(\Omega, q_n(m, \vec{w})) = Q_1(\Omega, q_n(m, \vec{w})) + Q_2(q_n(m, \vec{w}))$ consists of two parts:

$$Q_1(\Omega, q_n(m, \vec{w})) = \sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} \sum_{m=1}^{M} q_n(m, \vec{w}) \ln \left(p(m, \vec{w}, \vec{\theta}_n; \Omega) \right) \tag{4.19}$$

$$Q_2(q_n(m, \vec{w})) = - \sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} \sum_{m=1}^{M} q_n(m, \vec{w}) \ln \left(q_n(m, \vec{w}) \right). \tag{4.20}$$

Note that in equation (4.18) we have used Jensen's inequality (Appendix A.1) to lower bound the log-likelihood $L(\Omega)$.

At the *expectation step*, the algorithm maximizes the bound $Q(\Omega, q_n(m, \vec{w}))$ at $\Omega = \Omega_0$ for the optimal parameters $q_n(m, \vec{w})$, while at the *maximization step* the algorithm maximizes the bound $Q(\Omega, q_n(m, \vec{w}))$ for the optimal model parameters Ω. The EM algorithm repeats these steps until the log-likelihood converges. The procedure will be discussed in the following subsections and summarized in Table 4.1.

4.4.1 Expectation Step

The lower bound $Q(\Omega_0, q_n(m, \vec{w}))$ is optimized with respect to the parameters $q_n(m, \vec{w})$, under the constraints posed by equations (4.16) and (4.17). We formulate the Lagrangian function:

$$F = Q(\Omega_0, q_n(m, \vec{w})) + \sum_{n=1}^{N} \lambda_n \left(\sum_{m=1}^{M} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w}) - 1 \right) \tag{4.21}$$

and maximize it to obtain:

$$q_n(m, \vec{w}) = \frac{p(m, \vec{w}, \vec{\theta}_n; \Omega_0)}{p(\vec{\theta}_n; \Omega_0)}. \tag{4.22}$$

The solution is always positive, so the first constraint (4.16) is always satisfied. The derivation can be found in Appendix A.2. Note that although we have shown that the bound $Q(\Omega, q_n(m, \vec{w}))$ is optimal with respect to $q_n(m, \vec{w})$, we have not been able to prove the *tightness* of the bound at $\Omega = \Omega_0$, which remains an open question.

4.4.2 Maximization Step

The optimal $q_n(m, \vec{w})$ given $\Omega = \Omega_0$ are now used to compute the optimal Ω for which the bound $Q(\Omega, q_n(m, \vec{w}))$ is optimized. Since $Q_2(q_n(m, \vec{w}))$ is independent of Ω, only $Q_1(\Omega, q_n(m, \vec{w}))$ has to be optimized under the constraint $\sum_{m=1}^{M} \alpha_m = 1$. The optimization is quite straight-forward and can be found in Appendix A.3. The model parameters Ω are updated according to the following equations:

$$\alpha_m \leftarrow \frac{1}{N} \sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w}) \tag{4.23}$$

$$\vec{\mu}_m \leftarrow \frac{\sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w}) \left(\vec{\theta}_n - \vec{w} 2\pi \right)}{\sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w})} \tag{4.24}$$

$$\Sigma_m \leftarrow \frac{\sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w}) (\vec{\theta}_n - \vec{\mu}_m - \vec{w} 2\pi)(\vec{\theta}_n - \vec{\mu}_m - \vec{w} 2\pi)^T}{\sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w})} \tag{4.25}$$

The resulting EM algorithm is summarized in Table 4.1.

Initialization:	Set Ω_0
1: Expectation Step:	$q_n(m,\vec{w}) \leftarrow \underset{q_n(m,\vec{w})}{\arg\max}\{Q(\Omega_0, q_n(m,\vec{w}))\} \Rightarrow$
	$q_n(m,\vec{w}) \leftarrow \frac{p(m,\vec{w},\vec{\theta}_n;\Omega_0)}{p(\vec{\theta}_n;\Omega_0)}$
2: Maximization Step:	$\Omega_0 = \underset{\Omega}{\arg\max}\, Q(\Omega, q_n(m,\vec{w})) \Rightarrow$
	$\alpha_m \leftarrow \frac{1}{N}\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})$
	$\vec{\mu}_m \leftarrow \frac{\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})\left(\vec{\theta}_n - \vec{w}2\pi\right)}{\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})}$
	$\Sigma_m \leftarrow \frac{\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})(\vec{\theta}_n - \vec{\mu}_m - \vec{w}2\pi)(\vec{\theta}_n - \vec{\mu}_m - \vec{w}2\pi)^T}{\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})}$
Check Convergence:	Repeat steps 1 and 2 until convergence.

Table 4.1 An overview of the EM algorithm.

4.4.3 Diagonal Covariance Model

The complete computation of the WGMM pdf in equation 4.11, as well as the update equations (4.23), (4.23), (4.23) require a summation over an infinite number of terms. In practice, an adequate approximation can be made if the summation is restricted to the first ± 2 terms at each dimension. This approximation is justified if the variances of $\vec{\theta}$ are small compared to 2π. However, even in this case, the number of computed terms increases exponentially with the number of dimensions K and becomes infeasible for more than 2 dimensions. A solution to this problem is to restrict the covariance matrices Σ_m to be diagonal $\Sigma_m = \mathrm{diag}\left(\sigma_{\mathrm{m}}^2(1), \sigma_{\mathrm{m}}^2(2), \ldots, \sigma_{\mathrm{m}}^2(\mathrm{K})\right)$. The corresponding wrapped pdf can then be computed according to the following equation:

$$p(\vec{\theta}|\Omega) = \sum_{m=1}^{M}\alpha_m \sum_{\vec{w}\in\mathbb{Z}^K}\prod_{k=1}^{K}\frac{1}{\sqrt{2\pi\sigma_m^2(k)}}\exp\left(-\frac{\left(\vec{\theta}(k) - \vec{\mu}_m(k) - \vec{w}(k)2\pi\right)^2}{2\sigma_m^2(k)}\right) \Rightarrow$$

$$p(\vec{\theta}|\Omega) = \sum_{m=1}^{M}\alpha_m \prod_{k=1}^{K}\sum_{w\in\mathbb{Z}}\frac{1}{\sqrt{2\pi\sigma_m^2(k)}}\exp\left(-\frac{\left(\vec{\theta}(k) - \vec{\mu}_m(k) - w2\pi\right)^2}{2\sigma_m^2(k)}\right) \tag{4.26}$$

where $\vec{\theta}(k)$, $\vec{\mu}_m(k)$ are the k-th element of $\vec{\theta}$ and $\vec{\mu}_m$, respectively. The interchange between the product over the dimensions and the summation over the wrappings allows a significant complexity reduction. The update equations (4.23), (4.24) and (4.25) can also benefit from interchanging the product and the summation, leading to an EM algorithm of tractable complexity. The corresponding update formulae are provided in Appendix A.4.

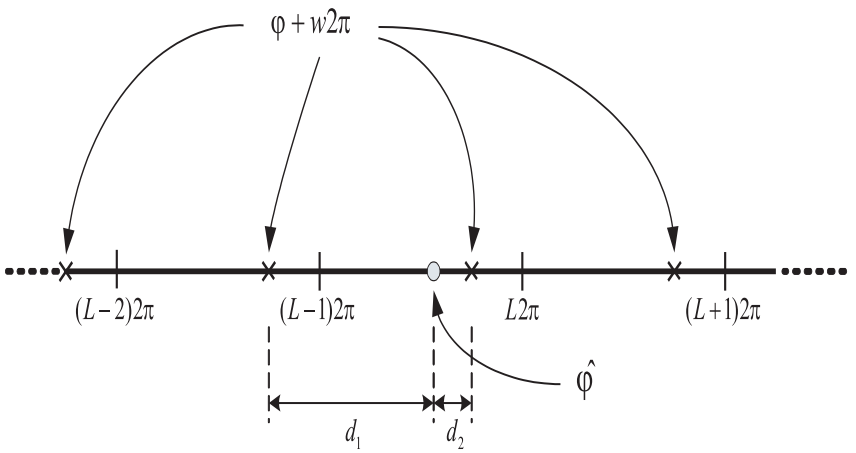


Figure 4.4 An illustration of the computation of the wrapped scalar Mean-Square-Error distortion measure. The distortion in this case is $d_w(\varphi, \hat{\varphi}) = min(d_1^2, d_2^2)$

4.5 Wrapped-GMM-based Quantization of Phase data

The WGMM can be used for efficient quantization of the phase data. The linear distortion measures are not suitable for circular variables, therefore, we define a *scalar Wrapped-Squared-Error* (WSE) distortion criterion according to equation:

$$d_w(\varphi, \hat{\varphi}) = \min_{w \in \mathbb{Z}} \left\{ (\varphi - \hat{\varphi} - w2\pi)^2 \right\}. \tag{4.27}$$

The distortion criterion as stated, requires a search over an infinite number of linear squared errors. However, if φ is confined to its principal value (in $(0, 2\pi]$) and if $\hat{\varphi}$ extends over at most $\pm L$ wrappings (thus $\hat{\varphi} \in (-L2\pi, (L+1)2\pi]$), the WSE needs to be computed only for $\pm(L+1)$ wrappings.

Figure 4.4 displays this in a schematic manner. The wrapped MSE distance (4.27) corresponds to a codebook search of a codebook generated from the tiling $\varphi + w2\pi$, $w \in \mathbb{Z}$, of φ for the point that is nearest to $\hat{\varphi}$. Figure 4.4 illustrates the codebook $\varphi + w2\pi$, $w \in \mathbb{Z}$ and the fixed point $\hat{\varphi}$, which belongs to the L-th wrapping of 2π. It is easy to see that the best codepoint is at most ± 1 wrappings away from $\hat{\varphi}$. Thus, a search over $L+1$ wrappings is adequate. In other words, if the quantization of the phases is made using a codebook of phases $\hat{\varphi}$ restricted to $\pm L$ wrappings, the WSE can be computed using only $\pm(L+1)$ wrappings.

A *vector Wrapped-Squared-Error* (WSE) criterion $d_w(\vec{\theta}, \hat{\vec{\theta}})$ can be formulated ac-

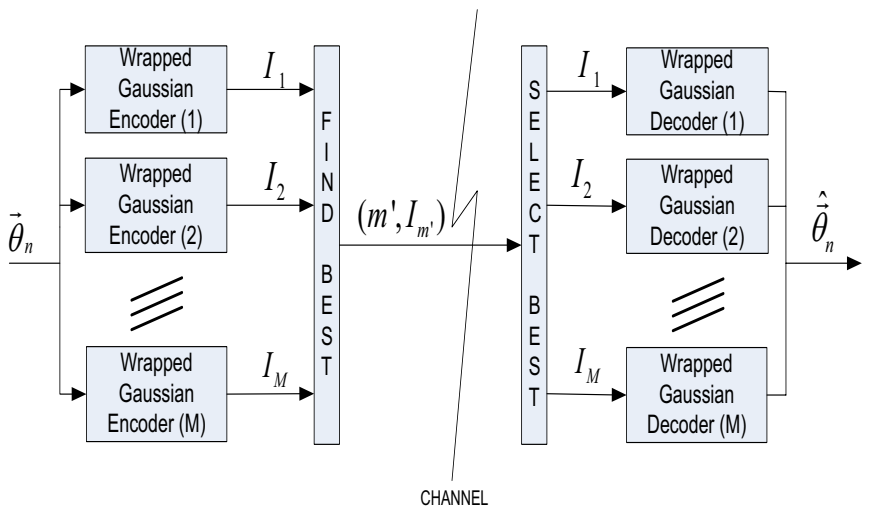


Figure 4.5 Basic scheme for WGMM-based vector quantization.

cording to:

$$d_w(\vec{\theta}, \hat{\vec{\theta}}) = \sum_{k=1}^{K} d_w(\vec{\theta}(k), \hat{\vec{\theta}}(k)). \tag{4.28}$$

The efficiency of GMM-based quantization is due to the decoupling of the estimation of the source pdf from the allocation of the codepoints. This idea can also be employed here using a WGMM estimator of the pdf. The corresponding scheme encodes the data vector $\vec{\theta}$ according to the pdf of each of the multivariate wrapped Gaussians. The procedure is depicted in Figure 4.5. Data $\vec{\theta}$ are quantized separately with M wrapped multivariate Gaussian *coders* and the best quantization is transmitted through the channel along with the corresponding indices. The design of a wrapped multivariate Gaussian quantizer is the subject of the following subsections. In subsection 4.5.1 we discuss the construction of a quantizer obtained by wrapping a Gaussian codebook to the circumference of the unit circle, and propose two bit allocation algorithms for WGMM-based quantization. This design is computationally appealing, but it provides sub-optimal codepoint allocation for variances $\sigma^2 > 0.5$. Section 4.5.2 provides a solution to this problem by introducing the concept of Polynomial CodeFunctions (PCF), which -in effect- is a set of functions that generate codepoints for given σ^2. A training algorithm for PCF is provided, followed by a greedy bit-allocation algorithm for WGMM.

4.5.1 Quantization using Wrapped Codebooks

The shape of the wrapped Gaussian pdf and the corresponding optimal codepoint density depends on the variance σ^2. Therefore, if the scalar quantizer for the wrapped Gaussian random variable is to be implemented with a set of precomputed codebooks,

one codebook needs to be stored for each required rate and variance. The storage complexity of this solution is prohibiting. For example, a WGMM with 32 components and 24 dimensions requires about $32 * 24 = 768$ codebooks. We propose to construct these wrapped codebooks by wrapping the codepoints of Gaussian $N(0, \sigma^2)$ codebooks around the circumference of the unit circle. In practice this is made by a simple modulo operation:

$$c_{wrapped} = mod(c_{linear}, 2\pi), \tag{4.29}$$

where c_{linear} is the codepoint of the linear Gaussian $N(0, \sigma^2)$ and $c_{wrapped}$ is the wrapped version of this codepoint. This solution works quite well for low variances $\sigma^2 \leq 1$ because the interval $(0, 2\pi]$ contains most of the pdf mass and the overlapping of the tiled Gaussian components is low, but it becomes less accurate for higher variances. However, we choose to accept this degradation for the benefit of low storage complexity, and we constrain the maximum overlapping by restricting the variances to $(0, 2\pi]$ during the training of the WGMM. Note that for variances above 2π the wrapped pdf is very close to the uniform distribution, as it is shown in Figure 4.2.

Bit Allocation

The allocation of quantization levels to the scalar variables of a Gaussian component requires a function that links the rate with the WSE. Two bit allocation algorithms will be presented. The first relies on a greedy bit allocation using precomputed tabulated distortions and the second is based on assumptions regarding the WGMM.

Algorithm A: Tabulated Distortions & Greedy Bit Allocation

Let R be the encoding rate and $N = 2^R$ be the number of quantization levels. A number of $N_m = \lfloor \alpha_m N \rfloor$ quantization levels is assigned to each of the M components of the WGMM. Within each Gaussian component, the N_m quantization levels were allocated with a greedy algorithm similar to [55] (pg. 234) that minimizes the expected component distortion D_m:

$$D_m = \sum_{k=1}^{K} D(N_{m,k}, \sigma_m^2(k)), \tag{4.30}$$

where $D(N_{m,k}, \sigma_m^2(k))$ is the expected WSE when the k-th variable of the m-th Gaussian component is encoded with $N_{m,k}$ quantization levels. The greedy bit allocation algorithm is the following:

Initialization: Set $N_{m,k} = 1$ for all $k = 1, \ldots, K$.

Step 1: Find the variable with the largest distortion $k' = \arg\max_k \{D(N_{m,k})\}$ subject to the rate constraint $\frac{N_{m,k'}+1}{N_{m,k'}} \prod_{k=1}^{K} N_{m,k} \leq N_m$.

Step 2: Increment the corresponding quantization level: $N_{m,k'} \leftarrow N_{m,k'} + 1$.

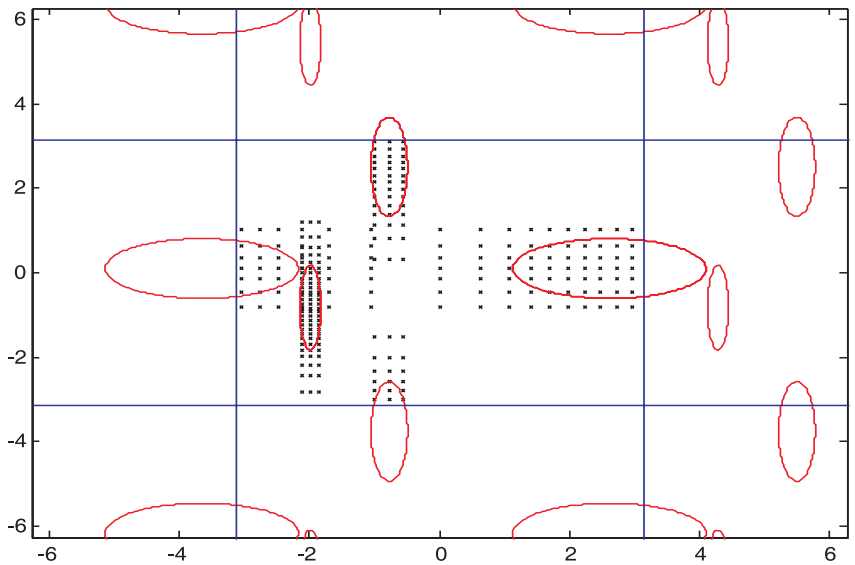


Figure 4.6 Two-dimensional WGMM and the corresponding codepoints, according to bit-allocation algorithm B.

Step 3: Repeat from Step 2 until no $N_{m,k}$ can be incremented without violating the rate constraint.

When the variances $\sigma_m^2(k) \leq 0.5$, the wrapped univariate Gaussian is well approximated by the linear Gaussian, and the high-rate formula (3.6) provides an approximation of the wrapped distortion function $D(.,.)$. For higher variances, $\sigma_m^2(k) > 0.5$, we use linear interpolation of tabulated distortions, sampled for a wide range of quantization levels and variances. The distortions were computed using 100.000 samples of a wrapped $N(0, \sigma^2)$ and evaluated with the WSE, for quantization levels $l = 1, 2, \ldots, 2^6$ and for densely sampled variances $\sigma^2 = \{0.5, 0.51, 0.52, \ldots, 2\pi\}$.

Algorithm B: Small Variance Assumptions & Optimal Bit Allocation

The second bit allocation algorithm is based on the assumption that the variances $\sigma_m^2(k)$ are small. In this case, the mean wrapped-squared-error D_{WSE} associated with the encoding of a wrapped Gaussian with a wrapped Gaussian codebook can be approximated by the mean-squared-error D_{MSE} associated with the encoding of a linear Gaussian with a linear Gaussian codebook.

This is illustrated in Figure 4.7 which shows the difference between the two errors $\Delta D = D_{MSE} - D_{WSE}$ (in decibel) as a function of the variance σ^2, for $1, \ldots, 6$ bits. For $\sigma^2 \leq 0.5$, the two distortions are identical. For $\sigma^2 \leq 1$, the linear distortion D_{MSE} is still a fairly accurate estimation of the wrapped distortion D_{WSE}, up to an accuracy of 0.5 dB ($\approx \frac{1}{6}$ bits). In practice, the most common rates are below 3 bits/dimension which are modeled with an accuracy of 0.25 dB when $\sigma^2 \leq 1$. The approximation becomes worst as the variance σ^2 increases. For example, for a variance equal to π the approximation error is approximately 2 dB which corresponds to an overestimation

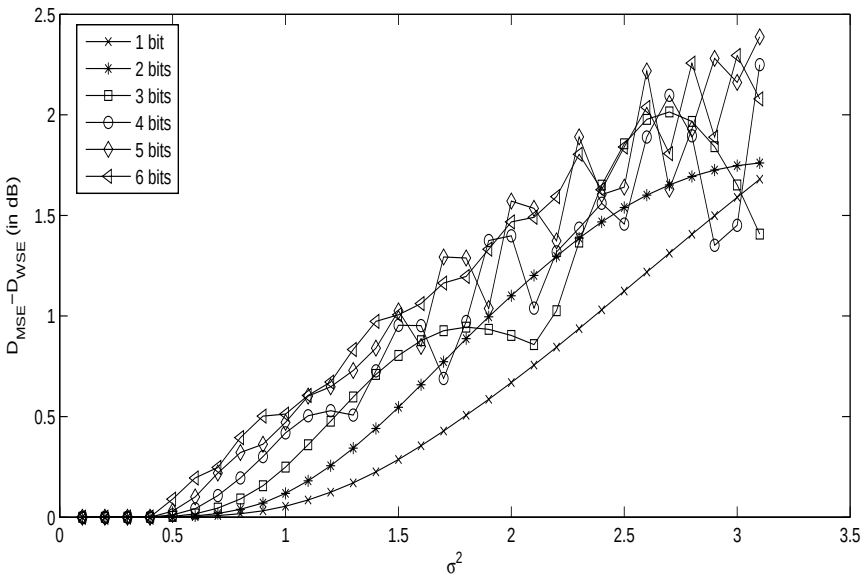


Figure 4.7 The difference (in decibel) between the mean-squared-error D_{MSE} and the mean wrapped-squared-error D_{WSE} for several variances σ^2.

of $\approx \frac{1}{2}$ bits/dimension. Note that the rapid variations above 3 bits/dimension are caused by the coincidence of the wrapped codepoints to the circumference of the unit circle.

If we assume that the variances $\sigma_m^2(k)$ are small, then the scalar WSE can be approximated by the high-rate formula (3.6). This results to a bit allocation algorithm that is the same with the bit allocation made for linear variables. Therefore, formula (3.10) provides the optimal bit-allocation for multivariate quantization and formula (3.21) the optimal allocation of bits to the WGMM components. In other words, the bit allocation is the same with the bit allocation made for GMM-based quantization, as it is presented in Chapter 3.

An evaluation between the two bit allocation algorithms will be made in Section 4.6 using measured phases of narrowband speech harmonics. An example of the reconstruction codepoints of a two-dimensional WGMM-based quantizer with bit allocation according to algorithm B is given in Figure 4.6.

4.5.2 Quantization using Polynomial CodeFunctions

Wrapping a linear Gaussian codebook on the unit circle in order to obtain a codebook for the wrapped Gaussian pdf is a practical but suboptimal choice dictated by the increased complexity of the optimal solution. This section proposes a novel method to construct codebooks optimized for symmetric circular random variables by introducing the concept of Polynomial CodeFunctions (PCF). The optimal codebook for a wrapped scalar Gaussian $N_w(\mu, \sigma^2)$ is a function of the variance σ^2 and the number

of codepoints M, since the translation term μ does not affect the shape of the pdf. Therefore, the idea is to construct a set of polynomial functions:

$$c_m(\sigma^2) = \sum_{p=0}^{P} c_{m,p}\sigma^{2p},\ m = 1, ..., M \tag{4.31}$$

that generate a codebook with M entries for $N_w(0, \sigma^2)$ for each σ^2 belonging to a continuous interval S. These functions could be referred to as "polynomial codepoint generator functions", or -in short- as Polynomial CodeFunctions. The PCF quantizer is based on the assumption that the optimal codepoints evolve smoothly over σ^2 in S. Let P be the order of the polynomial, $c_{m,p}$ its parameters and M be the size of the codebook. Assume that $c_m(.)$ and all circular variables take values in $(-\pi, \pi]$, and that $c_m(.)$ are sorted so that $c_1(\sigma^2) > c_2(\sigma^2) > ... > c_M(\sigma^2)$, for all $\sigma^2 \in S$. Since $N_w(0, \sigma^2)$ is symmetric around zero, only $M/2$ PCF are needed, and the following holds:

$$c_m(\sigma^2) = -c_{M-m+1}(\sigma^2),\ m = 1, ..., M. \tag{4.32}$$

If M is odd, then the central PCF is zero:

$$c_m(\sigma^2) = 0,\ m = \lfloor M/2 \rfloor + 1. \tag{4.33}$$

Enforcing symmetry using equation (4.32) makes sure that the partitioning of the unit circle made by the PCF has the point at π (or equivalently at $-\pi$) at the boundary between the quantization cells of $c_1(\sigma^2)$ and $c_M(\sigma^2) = -c_1(\sigma^2)$. In this case, no wrappings should be taken into account when quantizing the circular random variable $\theta \in (-\pi, \pi]$, and the linear squared error

$$d(\theta, \hat{\theta}) = (\theta - \hat{\theta})^2$$

can be used instead of the wrapped-squared-error of equation (4.27).

Training PCF

An iterative k-means-like algorithm has been developed for the training of the PCF. The PCF are trained for $\sigma^2 \in S$. Let $\sigma_l^2 = \{\sigma_1^2, \sigma_2^2, ..., \sigma_L^2\}$ be L samples of σ^2 in S. For example, $\sigma_l^2 = \{0.5, 0.52, 0.54, ..., 0.68, 0.7\}$. Let $\theta_{l,n}$, $l = 1, ..., L$, $n = 1, ..., N$ be N random samples from $N_w(0, \sigma_l^2)$, for each variance σ_l^2. Let $c_{m,p}^{(k)}$ be the PCF parameters resulting from the k-th iteration of the algorithm. The algorithm is initialized with constant PCF, uniformly distributed over $(-\pi, \pi]$:

$$c_{m,0}^{(0)} = 2\pi\left(0.5 + \frac{0.5 - m}{M}\right),\ m = 1, ..., \lfloor \frac{M}{2} \rfloor \tag{4.34}$$

$$c_{m,1}^{(0)} = c_{m,2}^{(0)} = \cdots = 0 \tag{4.35}$$

Each iteration consists of two steps: a *classification step* and an *optimization step*. The classification step labels each sample $\theta_{l,n}$ to a PCF function, and the optimization step uses these labels to estimate each PCF function. The PCF functions converge after 20 to 50 iterations.

Classification Step

At the k-th iteration, the classification step finds the indices $I_{l,n}$, $l = \{1, ..., L\}$, $n = \{1, ..., N\}$ that minimize the square error:

$$I_{l,n} = \arg\min_{m}\{(\theta_{l,n} - c_m^{(k-1)}(\sigma_l^2))^2\},\ m = 1, ..., M \tag{4.36}$$

Optimization Step

Let $\Theta_{l,m} = \{\theta_{l,n} : I_{l,n} = m\}$ be the set of samples that have been classified to the m-th PCF for each variance σ_l^2. The optimized m-th PCF is the polynomial that best fits the pairs of variables $\{(\sigma_l^2, \theta_l) : l = 1, ..., L\ \&\ \theta_l \in \Theta_{l,m}\}$. In other words, we obtain the optimal m-th PCF by minimizing the corresponding mean-square error:

$$c_{m,p}^{(k)} = \arg\min_{c_{m,p}}\{\sum_{l=1}^{L}\sum_{\theta\in\Theta_{l,m}}\left(\theta - \sum_{p=0}^{P} c_{m,p}\sigma_l^{2p}\right)^2\}. \tag{4.37}$$

The optimization can be made using typical polynomial least squares fitting methods [117].

Examples and Practical Considerations

The wrapped Gaussian is closely approximated by the linear Gaussian for small variances $\sigma^2 < 0.5$. Therefore, for $\sigma^2 < 0.5$ we can use the wrapped Gaussian quantizer presented in Section 4.5.1, while for higher variances we use PCF quantizers. For variances higher than 2π, the wrapped Gaussian is approximated by the uniform distribution, as shown in Section 4.3.2. Accordingly, we limit the variance σ^2 to a maximum of 2π. The construction of PCF quantizers depends on two inter-related design parameters: the size of the variance interval S and the order of the PCF polynomial. We found that high-order polynomials cannot provide high-quality PCF for the whole range of interest $S = [0.5, 2\pi]$. It is better to divide $[0.5, 2\pi]$ into smaller intervals of length 0.2 and to construct low-order (i.e., quadratic) polynomials for each of these intervals.

An example of the trajectories of wrapped linear Gaussian codepoints and PCF over σ^2 is shown in Figure 4.8. The wrapped linear Gaussian codepoints are not optimally distributed for $\sigma > 1$ and they may occasionally coincide, like at $\sigma^2 = 2\pi$, resulting to a practical loss of some quantization points and a distortion penalty. On the other hand, the PCF codefunctions converge to the codepoint allocation of a uniform quantizer, as σ^2 increases. This is better illustrated in Figure 4.9 where the distortion (WSE) of a PCF quantizer for $\theta \sim N_w(0, \sigma^2)$ is compared to the distortion of a uniform quantizer for θ uniformly distributed in $(-\pi, \pi]$. When $\sigma^2 \to 2\pi$, $N_w(0, \sigma^2)$ tends to the uniform distribution (horizontal line) and the two distortions converge.

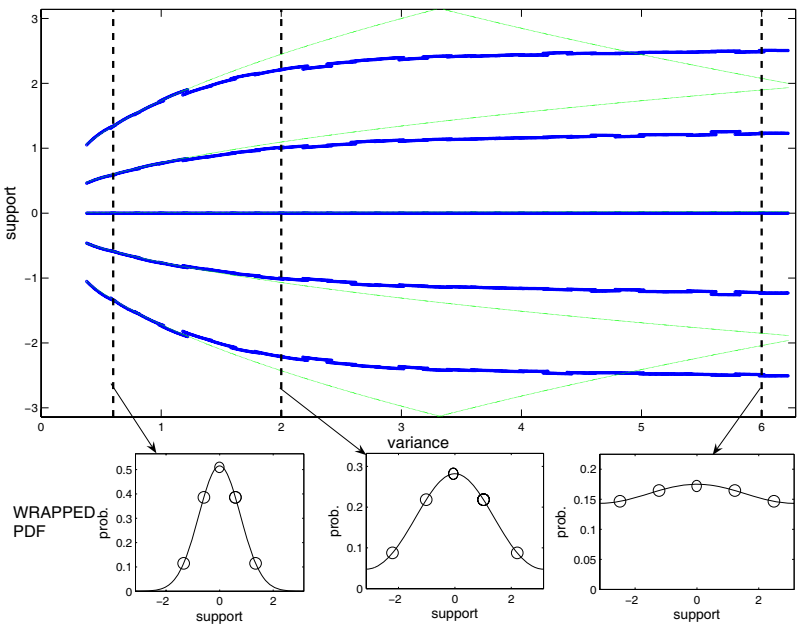


Figure 4.8 Codepoint trajectories over σ^2. The thin lines correspond to wrapped linear Gaussian codepoints and the thick lines to PCF generated codepoints. Three wrapped Gaussian pdf with variances $\sigma^2 = \{0.6, 2, 6\}$ are illustrated along with the corresponding PCF-generated codepoints.

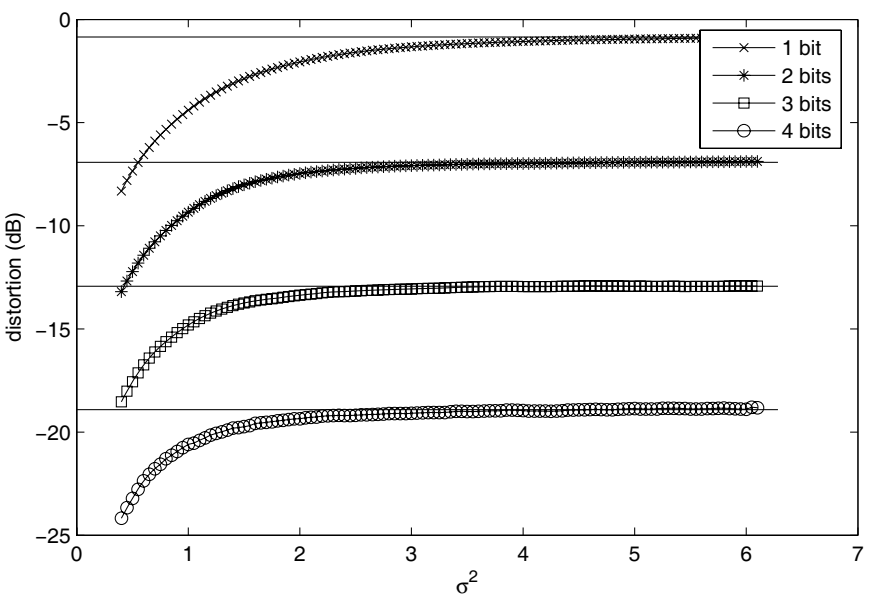


Figure 4.9 The distortion of PCF quantizers over σ^2 compared to the distortion of uniform quantizers, over σ^2.

Bit Allocation

A greedy bit-allocation algorithm similar to the one presented in Section 4.5.1 can be used to assign the quantization levels for the scalar PCF quantizers of a WGMM. The only difference now is that the tabulated distortions $D(N_{m,k}, \sigma_m^2(k))$ correspond to the expected wrapped square error distortion when the k-th variable of the m-th WGMM Gaussian component is encoded using a PCF quantizer with $N_{m,k}$ levels.

4.6 Phase Quantization for Narrowband Speech Coding

The presented WGMM-based quantization algorithm was used to quantize the dispersion phases ψ_k, $k = 1, ..., K$ of the narrowband speech harmonics below 3700 Hz. Only the phases of voiced frames were quantized, while the phases of unvoiced frames were randomly set.

An intrinsic difficulty in phase quantization is the variable dimensionality of the dispersion phase vectors $\vec{\psi}$. We address this problem by classifying pitch values in 7 classes (continuous intervals), referred to as Q1 to Q7 in Table 4.2, in order to reduce the variance of the dimensionality within each class. Note that this classification is just a plausible choice and that it is not critical for efficiency. The harmonics are separated in two bands; a low- and a high-frequency band, in order to provide more bits to the perceptually important low-frequency harmonics. A fixed number of low-frequency harmonics (depending on pitch class Q_i) are grouped together to form the lower-band dispersion phase vectors. For pitch classes Q1 and Q2, the lower-band consists of the first 24 harmonics. For pitch classes Q3 to Q6, the number of dimensions of the low-frequency harmonics is equal to the minimum size of the phase vectors of the corresponding class. For example, for class Q5, the number of harmonics for the low frequency band is given by: $\lfloor 3700/217 \rfloor = 17$ harmonics, where 217 is the lower pitch in Q5 and 3700 the bandwidth of the speech signal. The bandwidth of the lower-frequency band varies with the number of harmonics and the pitch. For the first 6 classes, Q1 to Q6, 6 fixed-dimension low-frequency dispersion phase datasets are obtained from TIMIT database. The number of dimensions of each dataset is shown in Table 4.2. Two more datasets are obtained for the high-frequency phases of pitch classes Q1 and Q2 with a size of 14 and 8 dimensions, respectively. These phases correspond to the first harmonics of the high-frequency band. An example is provided in Table 4.3: assume that the pitch is 100 Hz, giving a total of $\lfloor 3700/100 \rfloor = 37$ phases. The first 24 phases are used to train the low-frequency WGMM, while phases ψ_{25} to ψ_{32} are used to train the high frequency WGMM. Concluding, we derive 6 datasets from the low frequency band and 2 datasets from the high frequency band.

These datasets are used to train the corresponding WGMM according to Section 4.4. The circular mean (equation (4.3)) of each dataset is removed prior to training. This procedure moves the wrapped multivariate Gaussians closer to the

Pitch Class	Pitch Range	Low-Freq. WGMM dims.	High-Freq. WGMM dims.
Q1	<95 Hz	24	>14
Q2	95-115 Hz	24	>8
Q3	115-142 Hz	24	>0
Q4	142-176 Hz	21	>0
Q5	176-217 Hz	17	>0
Q6	217-250 Hz	14	>0
Q7	>250 Hz	<14	0

Table 4.2 Pitch Classes for WGMM-based Vector Quantization of phases.

Pitch Class	f_0	Low Frequency Harmonics	High Frequency Harmonics
Q2	100 Hz	$\underbrace{[\psi_1, ..., \psi_{21}]}_{\text{Quantized}}$	$\underbrace{[\psi_{25}, ..., \psi_{32}], \psi_{33}, ..., \psi_{37}}_{\text{Quantized}}$
Q4	150 Hz	$\underbrace{[\psi_1, ..., \psi_{21}]}_{\text{Quantized}}$	$\underbrace{\psi_{22}, \psi_{23}, \psi_{24}}_{\text{Quantized}}$
Q7	300 Hz	$[\underbrace{\psi_1, ..., \psi_{12}}_{\text{Quantized}}, \underbrace{\psi_{13}, \psi_{14}}_{\text{Ignored}}]$	$\oslash$

Table 4.3 Three examples of dispersion phase vectors with pitch values $f_0 = 100, 150$ and 300 Hz. Phases in brackets are modeled by a WGMM trained from data. Phases outside brackets are modeled by the "extended" WGMM.

center of the principal hypercube $(0, 2\pi]^K$ and increases the accuracy of the approximation that is made using only ± 2 tilings of each scalar Gaussian dimension. The number of dimensions of each low-frequency and high-frequency WGMM is shown in Table 4.2.

In most frames, the trained Wrapped Gaussian Mixture Models do not model all the harmonics and the statistics of a variable number of high frequency harmonics are not captured. In the examples provided in Table 4.3 these phases are shown to be outside the brackets. However, these harmonics are in high frequencies where the ear is less sensitive to individual phase distortions. Furthermore, we have observed that the bivariate marginal distributions of high-frequency harmonics have similar statistics. Therefore, for each frame we construct a high-band WGMM by replicating the means and the variances of the dispersion phase with the highest frequency that is modeled by a WGMM. Precisely, for pitch classes Q1 and Q2, where a high-frequency WGMM is already trained with 14 and 8 dimensions respectively, the trained WGMM is *extended* to the total number of harmonics using the means and the variances of the last dimension of the latter WGMM. In the first example of Table 4.3, this means that

the statistics of phases ψ_{33} to ψ_{37} are obtained from the statistics of ψ_{32}. For pitch classes Q3 to Q6, the high-frequency WGMM is constructed using the statistics of the last dimension of the corresponding low-frequency WGMM. In the second example of Table 4.3 the statistics of ψ_{22} to ψ_{24} are obtained from the statistics of ψ_{21}. A different procedure is used for the high-frequency class Q7 (above 250 Hz). Assuming that classes Q6 and Q7 have similar statistics, the phases of this class are modeled by *removing* the necessary number of higher frequency harmonics from the low-frequency WGMM of Q6. Therefore we have *less than* 14 dimensions as it is stated in Table 4.2 and illustrated in Table 4.3.

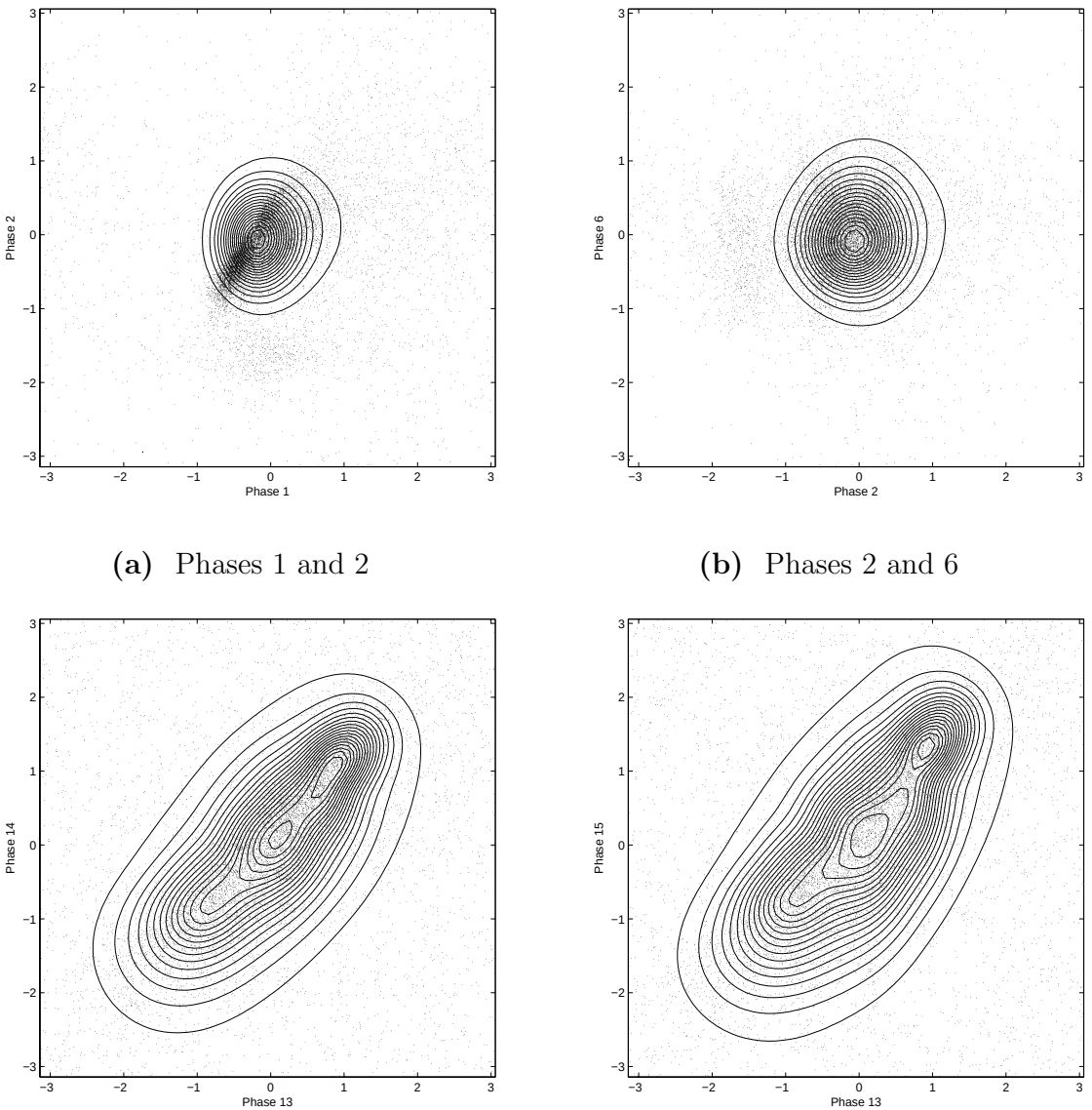
The training samples were obtained from training set of TIMIT database. The number of samples used for WGMM training for each class is shown in Table 4.2. The training samples belonged to frames with $SNR_{norm} \geq 3$ dB (see Section 2.5.2). Thus, the quantizers were trained mostly with voiced and transitional frames. Unvoiced frames can be transparently encoded with high distortion and a quantizer that is not specifically trained with unvoiced frames suffices. The unvoiced phases have approximately uniform distribution and the effect of including frames with lower SNR_{norm} is to raise the level of uniform noise at the phase statistics. Note that special classes of speech events like plosives are receiving high SNR_{norm} values in our analysis system and they are treated as voiced speech.

As an example, Figure 4.10 provides some insight regarding the statistics of the phases and the behavior of the WGMM for pitch class Q2. The plotted samples (dots) indicate the distribution of the phases, while the iso-contours show the pdf as it is modeled by the 32-component WGMM. We can observe that the first phases are not as noisy as the higher frequency phases, and that there are dependencies between the phases which benefit vector quantization. Furthermore, note that the dependencies are stronger between phases of neighboring harmonics. The statistics of the phases are well captured in higher harmonics but not so well in lower harmonics. This is due to the fact that the WGMM has diagonal covariance matrices which fail to accurately model the dependencies that exist, for example, between phases 1 and 2. The WGMM captures the dependencies using the location of the wrapped Gaussian means and in the latter case the means are located to account for the large diagonal structures arising between higher frequency harmonics. A better fit can be obtained at the cost of increased complexity if the number of WGMM components is increased.

An objective evaluation of the WGMM-based quantizers is made using a *Mean-Root-Wrapped-Square-Error* (MRWSE) criterion:

$$D = \left(\frac{180}{\pi}\right) \frac{1}{N} \sum_{n=1}^{N} \sqrt{\frac{1}{K} d_w(\vec{\theta}_n, \hat{\vec{\theta}}_n)}. \tag{4.38}$$

The MRWSE provides a measure of the average per-symbol distortion in degrees and it is useful for insight to the behavior of the quantizer. Figure 4.11 depicts the MRWSE for pitch classes Q1 to Q6 using the low-frequency WGMM-based quantizer at rates $30, 35, \ldots, 60$ bits. Three WGMM-based quantization methods are evaluated:



(a) Phases 1 and 2

(b) Phases 2 and 6

(c) Phases 13 and 14

(d) Phases 13 and 15

Figure 4.10 Scatter plots of harmonic phases from pitch class Q2 and iso-contours computed using the pdf of the corresponding low-frequency WGMM.

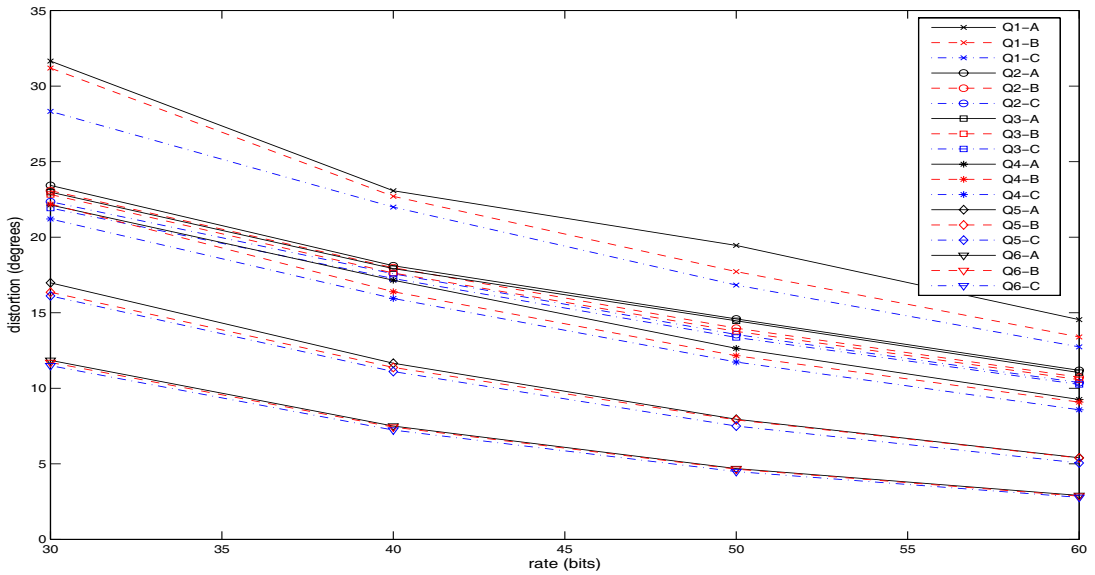


Figure 4.11 The Mean-Root WSE for pitch classes Q1 to Q6 and several rates. Three quantization methods are evaluated.

A: wrapped codebooks with bit allocation that uses small-variance assumptions

B: wrapped codebooks with greedy bit allocation

C: PCF Gaussian codebooks with greedy bit allocation

The first two methods use the bit allocation algorithms presented in Section 4.5.1 and they are evaluated using contours Q_i-A and Q_i-B, $i = \{1, .., 6\}$ respectively. The third method uses the PCF quantization algorithm described in Section 4.5.2 and it corresponds to contours Q_i-C, $i = \{1, .., 6\}$.

In pitch classes Q1 to Q4 method B is better than method A saving about 1-2 bits. The performance of both methods in all other pitch classes is similar. We speculate that this is due to the fact that the variances of the WGMM increase as the pitch decreases. The behavior of the variances can be partially attributed to the fact that transitional and unvoiced frames judged as voiced are usually classified to the low pitch classes. Note that the voiced/unvoiced decision is biased towards labeling unvoiced frames as voiced. Finally, the PCF quantizers (method C) outperform all other methods, saving about 1-5 bits over method A and about 0.5-3 bits over method B, depending on the class.

The rates of the phase quantizers were defined experimentally. Almost transparent quantization is achieved if the low-frequency WGMM-based quantizers use 60 bits/frame and the expanded high-frequency WGMM-based quantizers use 20 bits. Therefore, a total of 60+20=80 bits/frame are used to quantize the phases of nar-

rowband speech harmonics. Additionally, 7 bits are required for the quantization of the linear phase parameter τ (see Section 4.2).

The proposed phase quantization method provides the framework for high-rate quantization taking into account the circular (modulo-2π) nature of the phases both at modeling and quantization. However, a lot of improvements can also be made; perceptual weighting of the harmonics, voiced/unvoiced classification of the harmonics, a proper "decorrelation" of the circular random variables prior to modeling and quantization, etc. In the author's perspective, the decoupling of the pdf estimation from the allocation of codepoints breaks the phase quantization problem into a series of smaller manageable subproblems. Furthermore, the WGMM modeling of phases allows the statistical treatment of phases, which may benefit a number of applications like Speaker Recognition, where it is well known that the excitation bares speaker-specific information [118], generative models for Text-To-Speech (TTS) synthesis [119], detection of pathological speech [120]. Concluding, another application of the proposed phase quantization scheme is to reduce the footprint of large-corpus concatenative TTS systems [121].

Chapter 5

Packet Loss Concealment for Harmonic Models

5.1 Introduction

Packet loss concealment (PLC) is a vital part of a speech codec that attempts to hide the packet losses from the listener. This is a difficult task in predictive CELP-like codecs because a single packet loss can desynchronize the decoder from the encoder for a few subsequent frames. In that case, the major source of error propagation is the erroneous Adaptive CodeBook (ACB) excitation. Thus, most PLC schemes in CELP-like codecs are focused on reducing that desynchronization effect. A solution is to insert redundancy to the bit-stream using FEC (Forward Error Correction) techniques. In [35], a PLC scheme for ITU-T recommendation G.729 [31], redundant information from past frames is repeated regularly every 2 frames of coded speech. In [122], a FEC scheme repeats the excitation parameters only for frames that are judged to be perceptually important. In [123], a lower-quality low-bitrate Waveform Interpolation (WI) speech encoding is used as FEC data. The lower-quality representation is employed only when a packet is lost. In [38], the content of frames that have arrived after their playback time is used to reduce the error propagation in case of a frame erasure. In [124], time-scale modification is used to reduce the desynchronization effect. In [37] and [38], the contribution of the ACB excitation is constrained during the quantization of the innovative excitation of AMR-WB codec [125] so that the codec recovers faster after a packet loss. The latter method reduces the dependencies between frames to improve the robustness at the expense of lower speech quality when there are no packet losses.

The current solutions are effective for packet loss rates up to 3%-5% but face a rapid quality degradation upon higher loss rates and bursty losses. Most of the effort is given at recovering the state of the codec and not at concealing the packet loss. Clearly, there is a trade-off between coding efficiency and robustness. In that aspect, some researchers propose PLC schemes for codecs that encode each frame independently of the previous frames, like the ITU-T Recommendation G.711 [26],

the iLBC [40] codec and non-standardized, experimental sinusoidal codecs in [54], [51]. In practice, most of the research regarding PLC is made using the old G.711 codec that employs 8-bit companded Pulse-Code Modulation (PCM).

The ITU has proposed two PLC algorithms for G.711. The first one [126] detects and repeats the last received pitch period of the speech signal and has very low complexity while the second one [127] uses the short-term and the long-term excitation of the previously received speech signal to synthesize the lost speech samples. A simple scheme for waveform repetition was proposed in [128]. A periodic replication of the excitation signal is also used in [129], where an MOS of approximately 3.4 is reported for 10% packet losses. In [130], the lost samples are generated using a combination of linear prediction and waveform replication. Linear prediction has also been used in the backward direction when future speech samples are available in the jitter buffer, like in [131] where forward and backward linear prediction is used to compensate the speech gap. In [132], the LP residual signal is split into 8 sub-bands and each band is classified as voiced or unvoiced. In voiced bands, the excitation is replicated with respect to the estimated pitch while unvoiced bands are excited using white noise. An MOS of approximately 3 is reported for 10% packet losses. In [133], the consecutive speech samples are interleaved into L packets so that each packet contains one every L samples. Packet losses result to sparse sample losses that are recovered using a multi-rate state-space model of speech and interpolation based on a Kalman filter. The paper reports an MOS of approximately 3.4 for 15% packet losses. In [134], [49], a PLC algorithm based on a sinusoidal model shows significant improvement over the G.711, Appendix A algorithm [126]. All aforementioned MOS ratings are made using 10 ms packets of G.711. The MOS degradation is higher when the PCM samples are grouped in longer packets (i.e., 20 ms packets). It is evident that the PLC schemes for PCM encoded speech provide adequate subjective speech quality for packet loss rates up to 10%-15%.

The robustness of PCM speech codecs to packet losses is achieved at the expense of a very high bit-rate (64 kbits/sec). Sinusoidal codecs are well posed for high quality PLC because the harmonic representation allows efficient interpolation, extrapolation, and time scaling of the speech signal. Furthermore, these operations do not require extra analysis steps and can be made natively with appropriate modifications of the codec parameters. However, since there are no ITU standardized sinusoidal codecs, the corresponding sinusoidal PLC schemes are demonstrated on experimental codecs. In [51], time-scaling is used to stretch the received speech frames to fill the gap resulting from a packet loss, in the context of a 8 kbps sinusoidal speech codec. The authors report MOS ratings of 3.3 and 3.2 for loss rates of 10% and 20% respectively. In [54], a high-quality/high-rate sinusoidal speech codec classifies frames as unvoiced, voiced and transitional and the PLC algorithm treats differently each class. The author reports a slight quality degradation for packet losses up to 30%. Both aforementioned speech codecs encode each frame independently of the previous frames.

In this chapter we will propose a novel high-quality PLC algorithm suitable for

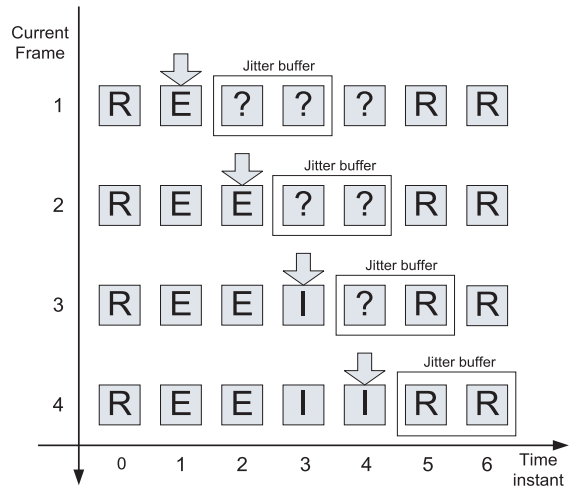


Figure 5.1 A combination of extrapolation and interpolation for PLC. Box labels "R","E","I" and "?" indicate a received, an extrapolated, an interpolated and a lost frame, respectively.

harmonic speech codecs. The proposed algorithm requires no extra analysis steps because it uses the quantized parameters of the harmonic model. The algorithm performs interpolation when a future speech frame is available in the jitter buffer or extrapolation when the jitter buffer is empty. Each sinusoid is characterized as voiced or unvoiced and treated accordingly. A sinusoid is considered voiced if it is below a voicing cutoff frequency threshold and unvoiced otherwise.

5.2 A novel high-quality PLC algorithm

The Harmonic Model (HM) allows efficient interpolation and extrapolation of the harmonically related sinusoids. Interpolation is used when a future speech frame is available in the jitter buffer, while extrapolation is used when the jitter buffer is empty. Each frame is 20 ms long (160 samples) at a sampling rate of 8000 Hz and the frame rate is 100 Hz (one frame every 10 ms).

Assume that the decoder has already played all the samples until the middle of the last received frame and that the current frame is not available due to a packet loss. The proposed PLC algorithm searches the jitter buffer for the nearest received future frame. If the jitter buffer is empty, then the decoder enters the *"extrapolation mode"* and uses extrapolation to fill the next 10 ms (80 samples) of speech. The extrapolation procedure is discussed in Section 5.4. If the jitter buffer is not empty, then the decoder enters the *"interpolation mode"* and uses interpolation to fill the samples from the last received frame to the nearest future received frame. Furthermore, the decoder may enter the interpolation mode after the extrapolation mode. The interpolation

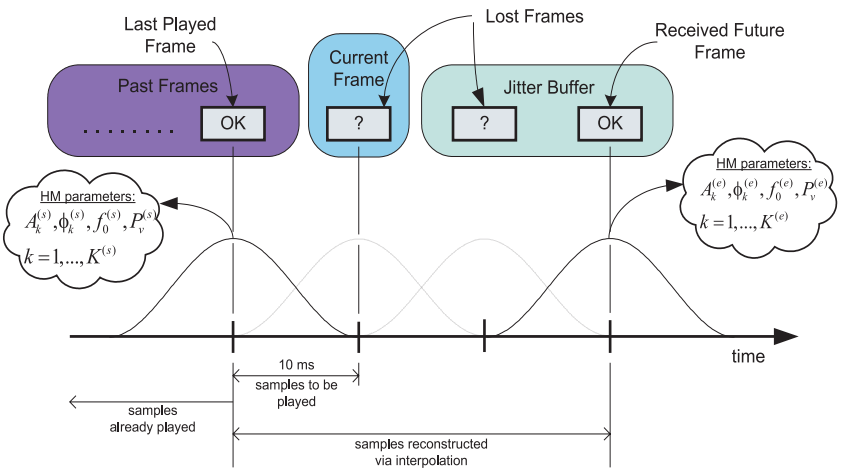


Figure 5.2 PLC with available future frames in the jitter buffer.

procedure is discussed in Section 5.3.

Figure 5.1 shows an example of interpolation following extrapolation. The decoder has to face a 4 packet long bursty loss. The boxes represent speech frames at successive time instants 0,1,...,6. The frames/boxes are labeled with a "?","R","E" or "I" to state that they are lost, received, extrapolated or interpolated, respectively. The first two lost frames (time instants 1,2) are extrapolated because the corresponding jitter buffer is empty, while the last two lost frames are interpolated since the jitter buffer contains a received frame (at time instant 5). The interpolation is made with the parameters of the last extrapolated frame and not the original received frame (at time instant 0). This ensures a smooth transition between extrapolation and interpolation in the proposed PLC scheme and it is suitable for the relatively infrequent long bursty losses.

The interpolation procedure is schematically shown in Figure 5.2. The jitter buffer holds two frames and there are two consecutive frame losses. The lost speech content is filled by interpolating the last played frame with the last frame of the jitter buffer. Let $A_k^{(s)}$, $\phi_k^{(s)}$, $f_0^{(s)} = kf_0^{(s)}$, $k = 1,\ldots,K^{(s)}$ be the amplitudes, the phases and the pitch of the last played frame (the "(s)tart-frame"), and $A_k^{(e)}$, $\phi_k^{(e)}$, $f_0^{(e)} = kf_0^{(e)}$, $k = 1,\ldots,K^{(e)}$ the corresponding parameters of the nearest future received frame (the "(e)nd-frame") in the jitter buffer. Let $P_v^{(s)}$ and $P_v^{(e)}$ be the *voicing probability* for the start-frame and the end-frame, respectively. The voicing probability is defined according to the normalized SNR of the speech frame (see equation 2.37):

$$P_v(SNR_{norm}) = \begin{cases} 0, & SNR_{norm} < 5 \\ \frac{SNR_{norm}-5}{5}, & 5 \leq SNR_{norm} \leq 10 \\ 1, & SNR_{norm} > 10 \end{cases} \tag{5.1}$$

A sinusoidal component is characterized as "voiced" if it is below the following cutoff

frequency:

$$f_c = 4000 P_v \tag{5.2}$$

and "unvoiced" otherwise. All sinusoidal components of unvoiced frames are considered to be unvoiced and are assigned to $P_v = 0$, because, occasionally, unvoiced frames may yield high SNR_{norm} values.

5.3 Interpolation Mode

An interpolation is made between the harmonic sinusoids of the start-frame and the harmonic sinusoids of the end-frame. The harmonic sinusoids are interpolated only when the corresponding fundamental frequencies $f_0^{(s)}$ and $f_0^{(e)}$ are close enough: $|f_0^{(e)} - f_0^{(s)}| \leq \Delta f_0$. The maximum difference for 1 frame loss is $\Delta f_0 = 20$ Hz, for 2 consecutive frame losses is $\Delta f_0 = 30$ Hz, and $\Delta f_0 = 40$ Hz for 3 or more consecutive losses. Since the number of sinusoids $K^{(s)}$ may differ from $K^{(e)}$, the remaining sinusoids are treated with a birth-death approach [45] (pg. 443). A birth-death approach is also used when the maximum difference is higher than Δf_0: the sinusoids of the start frame extrapolate with an amplitude that fades to zero while the sinusoids of the end frame extrapolate (backwards) with an amplitude that gradually rises from zero to $A_k^{(e)}$. The result of this matching/birth-death procedure is a new set of start-frame and end-frame sinusoids, with amplitudes $B_k^{(s)}$, $B_k^{(e)}$, phases $\varphi_k^{(s)}$,$\varphi_k^{(e)}$, and frequencies $f_k^{(s)}$,$f_k^{(e)}$, respectively.

In detail, when both start-frame and end-frame are voiced and $|f_0^{(e)} - f_0^{(s)}| \leq \Delta f_0$, the PLC method synthesizes the samples of the gap using *interpolation synthesis* with the following sinusoidal parameters:

Start-Frame parameters:

$$B_k^{(s)} = \begin{cases} A_k^{(s)}, & k = 1, ..., K^{(s)} \\ 0, & k = K^{(s)} + 1, ..., \max\{\mathrm{K^{(s)}}, \mathrm{K^{(e)}}\} \end{cases} \tag{5.3}$$

$$\varphi_k^{(s)} = \begin{cases} \phi_k^{(s)}, & k = 1, ..., K^{(s)} \\ \phi_k^{(e)} - 2\pi \frac{k f_0^{(e)}}{F_s} S', & k = K^{(s)} + 1, ..., \max\{\mathrm{K^{(s)}}, \mathrm{K^{(e)}}\} \end{cases} \tag{5.4}$$

$$f_k^{(s)} = \begin{cases} k f_0^{(s)}, & k = 1, ..., K^{(s)} \\ k f_0^{(e)}, & k = K^{(s)} + 1, ..., \max\{\mathrm{K^{(s)}}, \mathrm{K^{(e)}}\} \end{cases} \tag{5.5}$$

$$V_k^{(s)} = \begin{cases} \text{true}, & f_k^{(s)} \leq f_c^{(s)} \\ \text{false}, & f_k^{(s)} > f_c^{(s)} \end{cases} \tag{5.6}$$

End-Frame parameters:

$$B_k^{(e)} = \begin{cases} A_k^{(e)}, & k = 1, ..., K^{(e)} \\ 0, & k = K^{(e)} + 1, ..., \max\{\mathrm{K^{(s)}}, \mathrm{K^{(e)}}\} \end{cases} \tag{5.7}$$

$$\varphi_k^{(e)} = \begin{cases} \phi_k^{(e)}, & k = 1, ..., K^{(e)} \\ \phi_k^{(s)} + 2\pi \frac{k f_0^{(s)}}{F_s} S', & k = K^{(e)} + 1, ..., \max\{\mathrm{K^{(s)}}, \mathrm{K^{(e)}}\} \end{cases} \tag{5.8}$$

$$f_k^{(e)} = \begin{cases} k f_0^{(e)}, & k = 1, ..., K^{(e)} \\ k f_0^{(s)}, & k = K^{(e)} + 1, ..., \max\{\mathrm{K^{(s)}}, \mathrm{K^{(e)}}\} \end{cases} \tag{5.9}$$

$$V_k^{(e)} = \begin{cases} \text{true}, & f_k^{(e)} \leq f_c^{(e)} \\ \text{false}, & f_k^{(e)} > f_c^{(e)} \end{cases} \tag{5.10}$$

where $S' = N_{loss} S$ is the size of the gap in samples, N_{loss} is the number of lost frames, $S = 80$ samples is the hop size, $V_k^{(s)}$, $V_k^{(e)}$ are boolean variables stating the voicing of the k-th sinusoid, and $f_c^{(s)}$, $f_c^{(e)}$, are the voicing cutoff frequencies for the start-frame and the end-frame respectively.

When either the start-frame or the end-frame is unvoiced, or $|f_0^{(e)} - f_0^{(s)}| > \Delta f_0$, the PLC method synthesizes the samples of the gap using *birth-death synthesis* with the following sinusoidal parameters:

Start-Frame parameters:

$$B_k^{(s)} = \begin{cases} A_k^{(s)}, & k \in \text{“start} - \text{frame”} \\ 0, & k \in \text{“end} - \text{frame”} \end{cases} \tag{5.11}$$

$$\varphi_k^{(s)} = \begin{cases} \phi_k^{(s)}, & k \in \text{“start} - \text{frame”} \\ \phi_k^{(e)} - 2\pi \frac{k f_0^{(e)}}{F_s} S', & k \in \text{“end} - \text{frame”} \end{cases} \tag{5.12}$$

$$f_k^{(s)} = \begin{cases} f_k^{(s)}, & k \in \text{“start} - \text{frame”} \\ f_k^{(e)}, & k \in \text{“end} - \text{frame”} \end{cases} \tag{5.13}$$

$$V_k^{(s)} = \begin{cases} \text{true}, & f_k^{(s)} \leq f_c^{(s)} \\ \text{false}, & f_k^{(s)} > f_c^{(s)} \end{cases} \tag{5.14}$$

End-Frame parameters:

$$B_k^{(e)} = \begin{cases} A_k^{(e)}, & k \in \text{“start} - \text{frame”} \\ 0, & k \in \text{“end} - \text{frame} \end{cases} \tag{5.15}$$

$$\varphi_k^{(e)} = \begin{cases} \phi_k^{(e)}, & k \in \text{“start} - \text{frame”} \\ \phi_k^{(s)} + 2\pi \frac{k f_0^{(s)}}{F_s} S', & k \in \text{“end} - \text{frame”} \end{cases} \tag{5.16}$$

$$f_k^{(e)} = \begin{cases} f_k^{(e)}, & k \in \text{“start} - \text{frame”} \\ f_k^{(s)}, & k \in \text{“end} - \text{frame”} \end{cases} \tag{5.17}$$

$$V_k^{(e)} = \begin{cases} \text{true}, & f_k^{(e)} \leq f_c^{(e)} \\ \text{false}, & f_k^{(e)} > f_c^{(e)} \end{cases} \tag{5.18}$$

In both cases, the start-frame parameters $B_k^{(s)}$, $\varphi_k^{(s)}$, $f_k^{(s)}$ and the end-frame parameters $B_k^{(e)}$, $\varphi_k^{(s)}$, $f_k^{(s)}$ describe the evolution of a single sinusoid over time. The synthesis of the speech gap is made on a per-sinusoid basis, according to the voicing state of each sinusoid at the start-frame and the end-frame. Three cases are considered:

case 1: Voiced-Voiced (V*V): The sinusoid is voiced in both endpoints.

case 2: Unvoiced-Unvoiced (U*U): The sinusoid is unvoiced in both endpoints.

case 3: Voiced-Unvoiced (V*U) or (U*V): The sinusoid is unvoiced only in one endpoint.

The synthesis method differs for each case, as it will be presented in the following subsections. This type of synthesis provides a flexible way to interpolate between voiced speech frames, transitional speech frames and unvoiced speech frames.

5.3.1 Voiced-Voiced synthesis

The S' samples of the interpolated speech signal are provided by the following equation:

$$\tilde{x}(n) = \sum_{k=1}^{K} B_k(n) \cos\left(\theta_k(n)\right), \; n = 0.5, 1.5, \ldots, S' - 0.5 \tag{5.19}$$

where K is the total number of sinusoids, $B_k(n)$ are the instantaneous amplitudes and $\theta_k(n)$ are the instantaneous phases of the sinusoids. Note that the $+0.5$ sample displacement is due to the location of the center of the analysis window (time instant 0 is between samples 80 and 81, at position 80.5). Therefore, the first reconstructed sample is at position $n = 0.5$, and the last at position $n = S' - 0.5$. The analysis of the start-frame and the end-frame was made at time instants $n = 0$ and $n = S'$ respectively.

The instantaneous amplitudes are easily computed with linear interpolation:

$$B_k(n) = B_k^{(s)} + \left(B_k^{(e)} - B_k^{(s)}\right) \frac{n - 0.5}{S' - 1},\ n = 0.5, 1.5, \ldots, S' - 0.5 \tag{5.20}$$

The computation of the instantaneous phases is somewhat more complicated. When the k-th sinusoid is voiced at the start-frame and the end-frame ($V_k^{(s)} = V_k^{(e)} =$ true) then the instantaneous phases are computed using cubic phase interpolation [45] (pg. 446). The following description of the cubic phase interpolation is adapted from [45] (pg. 446). For convenience, let $\omega_k^{(s)} = \frac{2\pi f_k^{(s)}}{F_s}$, $\omega_k^{(e)} = \frac{2\pi f_k^{(e)}}{F_s}$. The cubic phase model assumes that the phase of the sinusoid is a third order polynomial over time:

$$\theta_k(n) = \varphi_k^{(s)} + \omega_k^{(s)} n + \alpha_k n^2 + \beta_k n^3,\ n \in [0, S'] \tag{5.21}$$

where the time variable n is now considered to vary continuously over $[0, S']$. The time positions $n = 0$ and $n = S'$ refer to the center of the start-frame and the end-frame, respectively. The polynomial satisfies the start-frame conditions:

$$\theta_k(0) = \varphi_k^{(s)} \tag{5.22}$$

$$\left.\frac{\partial \theta_k(n)}{\partial n}\right|_{n=0} = \omega_k^{(s)}, \tag{5.23}$$

that ensure continuity at $n = 0$. The parameters α_k and β_k are computed using the end-frame conditions at ensure continuity at $n = S'$:

$$\theta_k(S') = \varphi_k^{(e)} + 2\pi M \tag{5.24}$$

$$\left.\frac{\partial \theta_k(n)}{\partial n}\right|_{n=S'} = \omega_k^{(e)} \tag{5.25}$$

where M is an integer that *unwraps* the end-frame phase. The solution of this linear system for given M yields the following a_k, b_k:

$$\begin{bmatrix} \alpha_k \\ \beta_k \end{bmatrix} = \begin{bmatrix} \frac{3}{S'^2} & \frac{-1}{S'} \\ \frac{-2}{S'^3} & \frac{1}{S'^2} \end{bmatrix} \begin{bmatrix} \varphi_k^{(e)} - \varphi_k^{(s)} - \omega_k^{(s)} S' + 2\pi M \\ \omega_k^{(e)} - \omega_k^{(s)} \end{bmatrix}. \tag{5.26}$$

The unwrapping integer M can be obtained using a *smoothness criterion* on $\theta_k(n)$. A reasonable smoothness criterion is the energy of the second-order derivative of $\theta_k(n)$:

$$\epsilon = \int_0^{S'} \left[\frac{\partial^2 \theta_k(n)}{\partial n^2}\right]^2 dn \tag{5.27}$$

The M that minimizes ϵ is:

$$M = \text{round}\left(\frac{1}{2\pi}\left[\left(\varphi_k^{(s)} + \omega_k^{(s)} S' - \varphi_k^{(e)}\right) + \left(\omega_k^{(e)} - \omega_k^{(s)}\right)\frac{S'}{2}\right]\right) \tag{5.28}$$

The cubic phase model assures that the phase and the frequency of the evolving sinusoid at the start-frame and the end-frame are the measured corresponding phases and frequencies, while the evolution of the phase over time is smooth.

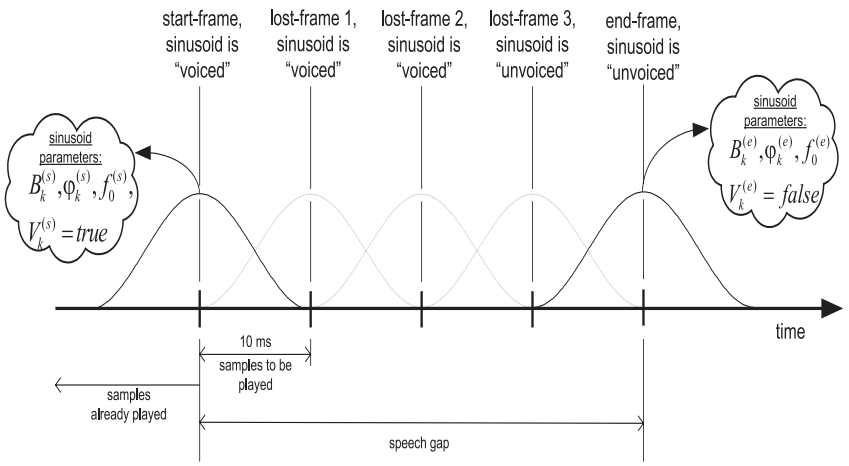


Figure 5.3 PLC synthesis of a single sinusoid that is voiced at the start-frame and unvoiced at the end-frame.

5.3.2 Unvoiced-Unvoiced synthesis

When the sinusoid is unvoiced at both endpoints ($V_k^{(s)} = V_k^{(e)} = \text{false}$), each lost frame is reconstructed using a random phase and the gap is synthesized with OLA of the reconstructed frame(s). This provides the desirable noisy character to the sinusoids.

5.3.3 Voiced-Unvoiced synthesis

Let N_{loss} be the number of the lost speech frames and l the index of the lost frame counting from past to future. When the k-th sinusoid is unvoiced only at one of the endpoints, ($V_k^{(s)} = \text{false}$ or $V_k^{(e)} = \text{false}$), the synthesis is guided by a voicing decision made for every lost frame according to a linear interpolation of the voicing cutoff frequency. The voicing decision states the type of synthesis that will be used at the corresponding speech frame. If the k-th sinusoid at the l-th lost frame is judged to be voiced, we perform cubic phase interpolation between the two endpoints to reconstruct the lost frame. If the k-th sinusoid at the l-th lost frame is judged to be unvoiced, we use a random phase that is uniformly distributed over $(-\pi, \pi]$ to synthesize the lost frame. The reconstructed frames are then properly overlap-added to the synthesis buffer.

The linear interpolation of the voicing cutoff frequency is given by equation:

$$f_c^{(int)}(l) = f_c^{(s)} + \left(f_c^{(e)} - f_c^{(s)}\right) \frac{l}{N_{loss}}, l = 1, \ldots, N_{loss}. \tag{5.29}$$

The corresponding voicing decision is then made according to the interpolated cutoff

frequency $f_c^{(int)}(l)$:

$$V_k^{(int)}(l) = \begin{cases} \text{true}, & f_k^{(s)} \leq f_c^{(int)}(l) \ \ \& \ \ f_k^{(e)} \leq f_c^{(int)}(l) \\ \text{false}, & \text{otherwise} \end{cases} \tag{5.30}$$

In other words, a voicing decision is made for every lost frame and every sinusoid that is unvoiced at the beginning or at the end of the gap. Since sinusoids from unvoiced frames are marked as unvoiced in both endpoints, a sinusoid that is marked as voiced at one of the endpoints has voiced frames at both endpoints.

An example of voiced-unvoiced synthesis is illustrated in Figure 5.3, where a single sinusoid is interpolated to fill a three-frame gap. The sinusoid is judged to be voiced at the start frame and unvoiced at the end-frame. The voicing cutoff frequency is interpolated for the three lost frames. The sinusoid is considered to be voiced at the first two lost frames because both frequencies $f_k^{(s)}, f_k^{(e)}$ are below $f_c^{(int)}(l), l = 1, 2$, and is considered to be unvoiced at the third frame because $f_c^{(int)}(3)$ is below $f_k^{(s)}$ or $f_k^{(e)}$. At the first two frames, the sinusoid is constructed using cubic phase interpolation between the start-frame and the end-frame parameters while at the third frame, the sinusoid is constructed using random phase. At all frames, the amplitude is linearly interpolated using equation 5.20. Finally, the speech gap is synthesized by overlap-adding the three reconstructed frames to the playout buffer.

5.4 Extrapolation Mode

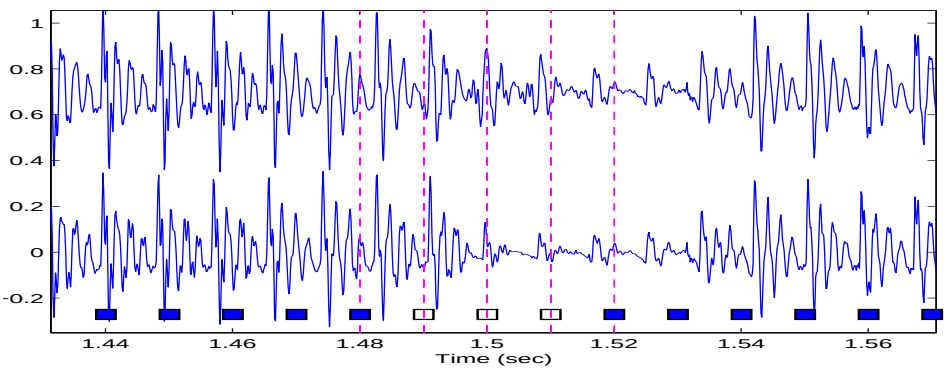
The harmonic sinusoids $A_k^{(s)}$, $\phi_k^{(s)}$, $f_0^{(s)}$, $k = 1, \ldots, K^{(s)}$ of the last played frame are extrapolated to fill the next 10 ms (80 samples) of the speech signal. The parameters of the extrapolated lost frame (next 160 samples) are estimated according to the following formulae:

Amplitudes: $A_k^{(ext)} = \gamma A_k^{(s)}$

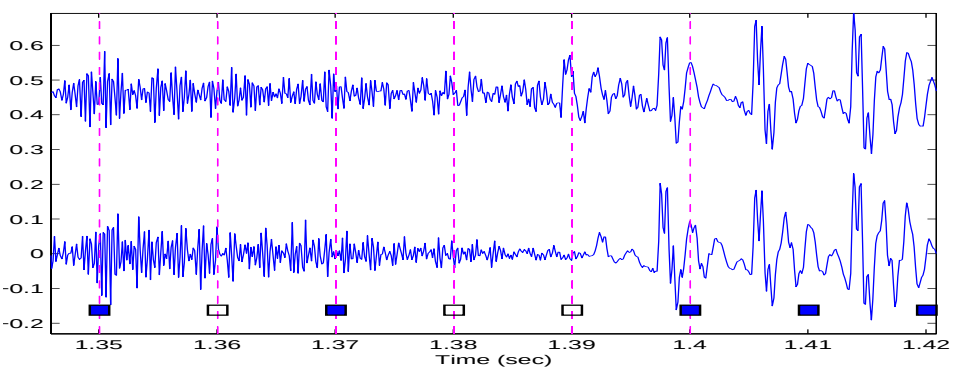
Phases: $\phi_k^{(ext)} = \phi_k^{(s)} + 2\pi \frac{k f_0^{(s)}}{F_s} S + \epsilon_k$

Frequencies: $f_k^{(ext)} = k f_0^{(s)}$

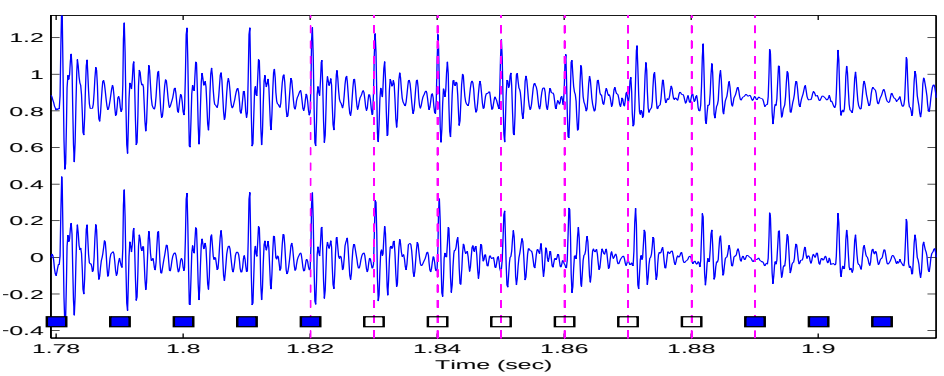
where $k = 1, \ldots, K^{(s)}$, $S = 80$ samples is the hop size, $F_s = 8000$ is the sampling rate, and $\epsilon_k^{(s)}$ is a noise parameter which is zero when the k-th harmonic is voiced and uniform noise in $(-\pi, \pi]$ when the k-th harmonic is unvoiced. The variable $\gamma \in [0, 1]$ is a scaling factor that reduces the energy of the frame in order to avoid extensive extrapolation over time that may lead to metallic sounds. For the first extrapolated frame, $\gamma = 1$, for the second and the third extrapolated frame, $\gamma = 0.9$, while for 4 or more extrapolated frames, $\gamma = 0.5$.



(a) Three consecutive losses at a non-stationary part of voiced speech.



(b) A single loss of unvoiced speech and two consecutive losses at a transition from unvoiced to voiced speech.



(c) Six consecutive losses at stationary voiced speech.

Figure 5.4 Three examples of PLC. The lower waveform presents the original signal while the upper waveform the concealed signal. The boxes are centered at the center of the analysis/synthesis frames. The empty boxes represent the lost frames. The dashed lines are located at the centers of the analysis/synthesis frames.

5.5 Results

A subjective evaluation of the proposed PLC algorithm is made in Chapter 9, Section 9.4 in the context of a sinusoidal speech codec. Some examples of concealment that provide further insight to the operation of the PLC algorithm are shown in Figure 5.4. The analysis/synthesis of the speech signal is made with the Harmonic Model discussed in Chapter 2. A 20 ms Hanning window was used for analysis/synthesis with a 10 ms step. The harmonic amplitudes were derived from a 20-th order cepstral envelope. At all cases, we consider a jitter buffer of 40 ms. Therefore, interpolation is used for up to 4 frame losses and extrapolation for more than 4 frame losses.

The first example (Figure 5.4a) is about the loss of three consecutive frames located at a non-stationary part of voiced speech. The PLC algorithm uses cubic phase interpolation to provide a smooth transition between the two endpoint speech frames. The concealment in this case provides speech of high quality and no degradation is perceived.

The second example (Figure 5.4b) shows the concealment of unvoiced/voiced transitions. Two frames are lost exactly in a unvoiced/voiced transition. The PLC algorithm uses birth/death synthesis to link the two endpoints. Furthermore, a single frame is lost at the unvoiced part of the speech. The frame is synthesized with birth/death synthesis using random phases. The concealment in this case provides a noisy character to the transition that is evident to the experienced listener. However, the perceived degradation is minimal to the average listener.

The third example (Figure 5.4c) is about a 6-frame long bursty loss that occurs in voiced speech. The first two frames are extrapolated while interpolation is used to fill the last 4 frames. The concealment in this case is of high quality, almost indistinguishable from the original signal.

Chapter 6

Multiple Description Coding

The content of a lost packet cannot be recovered with a PLC algorithm and redundancy is needed to compensate the packet loss. Multiple Description Coding (MDC) is a plausible way to introduce controlled redundancy into the bitstream. In MDC, each frame is encoded in two correlated descriptions that are independently transmitted through the network. If both descriptions arrive, then the central decoder is used to provide a high quality reconstruction of the encoded source. If only one description arrives then one of the two side decoders is used to provide a lower quality reconstruction of the source. If no description arrives then PLC is used to fill the gap. The generic MDC procedure is illustrated in Figure 6.1.

The goal in MDC is to construct the side descriptions and the central description in an optimal manner for the given channel conditions. Let D_0, D_1 and D_2 be the average distortions associated with the central decoder and the two side decoders. Assuming that each description is routed through an independent symmetric channel with packet loss probability ρ, the total distortion of an MDC system is provided by the following equation:

$$D_{MDC} = (1-\rho)^2 D_0 + \rho(1-\rho)(D_1 + D_2) + \rho^2 D_3 \ , \tag{6.1}$$

where D_3 is the distortion when both descriptions are lost (for example, the variance of the source). The two descriptions are called "*balanced*" when they have the same rate and $D_1 \approx D_2$. Most of the attention in MDC is given to *balanced* MDC systems. We can classify MDC techniques in three categories, discriminated by the way the correlations between the descriptions are captured: MDC based on an *index-assignment* matrix, MDC based on *lattices* and *transform*-based MDC.

The techniques based on "index-assignment" require three codebooks; two *side codebooks* that decode the indices of each of the descriptions, and one *central codebook* which decodes the indices of both descriptions. A *lookup table* called *"index assignment"* function maps the received indices to the central codebook and the side codebooks. Intuitively, the index assignment function captures the correlations between the descriptions. A scalar version of these techniques was initially introduced by Vaishampayan in [74] with the *Multiple Description Scalar Quantization* (MDSQ)

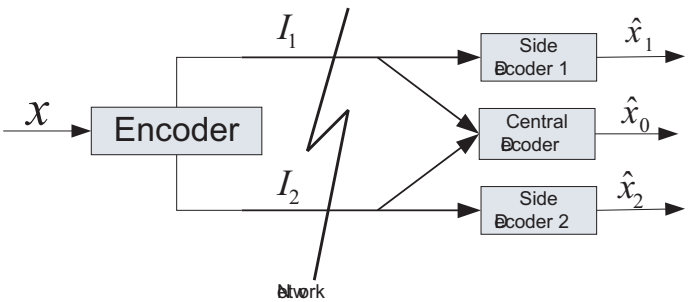


Figure 6.1 Multiple Description Coding

framework. A theoretical analysis can be found in [135], [136], [137], [138]. An extension to vector data and more than two descriptions was made in [139]. The performance of these methods is sensitive to the initialization of the index assignment table prior to training. A way to by-pass this problem is to assume a stochastic index assignment mapping that converges to the index assignment function in a deterministic annealing framework [140]. The major drawback of these techniques is that the computational complexity and the memory requirements for training, encoding and decoding increase rapidly with the encoding rate, the number of descriptions, the dimensionality of the data and the complexity of the network model. Multiple description transform quantizers based on MDSQ can provide a solution with much lower complexity [141].

Lattice-based MDC uses central codebook and side codebooks generated from lattices. The correlations between the descriptions are captured through the relative position/size of these lattices. MDC based on lattices can be found in [142], [143], [144].

In Multiple Description Transform Coding ($MDTC$), a *correlating transform* is used to introduce correlations between uncorrelated data [145]. In [146], [147], the correlations are introduced in a pairwise manner, via a linear volume preserving transform. The correlations can also be introduced via *frame expansions* [148]. The quantization of frame expansions and *reconstruction* issues are discussed in [149], [150].

In this chapter we study MDC systems suitable for speech coding. Focus is given to resolution-constrained quantization which is typically used in speech coding. Section 6.1 presents the principles of MDSQ (Multiple Description Scalar Quantization), the MDC method for scalar quantization originally proposed by Vaishampayan in [74]. The complexity of a vector quantizer that uses the index assignment matrix as in MDSQ is overwhelming for the rates and dimensionalities encountered in sinusoidal speech coding. Therefore, we resort to transform coding solutions. Section 6.2 discusses the construction of transform coders based on MDSQ [141]. Such transform coders will be referred to as MDSQ$_{TC}$ (MDSQ Transform Coding). A special note is made regarding the behavior of the bit-allocation formula for MDSQ$_{TC}$ which lower bounds the distortion of the central description according to the statistics of the

source. Section 6.3 presents GMM-MDSQ$_{TC}$, the extension of MDSQ$_{TC}$ to sources that can be accurately modeled by GMM, like spectral envelopes [151]. Section 6.4 extends GMM-MDSQ$_{TC}$to sources with a modulo-2π behavior (phase data) using Wrapped GMM. Finally, section 6.5 proposes a novel method that uses the redundancy between the descriptions in GMM-MDSQ$_{TC}$ to alleviate the effect of bit-errors. Experiments using RCC cepstral coefficients show that single and double bit-errors that occur solely in one description can be corrected with minor impact to the average central distortion.

6.1 Multiple Description Scalar Quantization

In MDSQ, the correlations between the two descriptions are introduced via an *index-assignment* matrix that links the two side description codebooks with the central description codebook [74]. Let $\lambda \sim p(\lambda)$ be the scalar random variable which describes the data that we encode. Assuming balanced descriptions with equal rates, let each description have a rate of r bits. Thus, $2r$ is the total rate of the MDSQ system. Let $C_0 \equiv \{\hat{\lambda}_{i,j}^{(0)} : i = 1, ..., 2^r, j = 1, ..., 2^r\}$ be the codebook of the central description, $C_1 \equiv \{\hat{\lambda}_i^{(1)} : i = 1, ..., 2^r\}$ be the codebook of description 1 and $C_2 \equiv \{\hat{\lambda}_j^{(2)} : j = 1, ..., 2^r\}$ be the codebook of description 2.

The two description indices i and j are routed independently to the receiver that has three decoders, a central decoder and two side decoders, one for each description. When both descriptions are received, the central decoder returns $\hat{\lambda}_{i,j}^{(0)} \in C_0$. When only the first description is received, side decoder 1 returns $\hat{\lambda}_i^{(1)} \in C_1$. Accordingly, when only the second description is received, side decoder 2 returns $\hat{\lambda}_j^{(2)} \in C_2$. The encoder finds the pair of indices (i, j) that minimizes the *total distortion*:

$$d_{tot} = d_0 + \frac{\rho}{1-\rho}(d_1 + d_2), \tag{6.2}$$

where ρ is the packet loss probability, $d_0 = (\lambda - \hat{\lambda}_{i,j}^{(0)})^2$ is the central distortion and $d_1 = (\lambda - \hat{\lambda}_i^{(1)})^2$, $d_2 = (\lambda - \hat{\lambda}_j^{(2)})^2$ the distortions of the two side descriptions, accordingly. This total distortion measure is based on the assumption that each description is routed through a symmetric channel and that both channels have the same packet loss probability ρ. Note that the distortion when both descriptions are lost is constant and depends on the variance of the source; thus it can be ignored during the minimization process. In practice, varying ρ provides a mechanism to exchange central distortion for side distortions. The index assignment matrix links each possible pair of side codebooks entries (i, j) to one central codebook entry. Note that at non-zero loss probabilities ρ, C_0 has less than 2^{2r} codepoints. Therefore, not all possible pairs of indices (i, j) correspond to a central description codepoint. The pairs (i, j) that correspond to a central description codeword will be referred to as "valid".

Let $Q_{i,j}^{(0)}$ be the quantization cell associated with central description codepoint $\hat{\lambda}_{i,j}^{(0)}$, $Q_i^{(1)}$ and $Q_j^{(2)}$ be the quantization cells associated with side description codepoints $\hat{\lambda}_i^{(1)}$ and $\hat{\lambda}_j^{(2)}$ respectively. Let $I_j^{(2)}(i) \equiv \{j : \hat{\lambda}_{i,j}^{(0)} \in C_0\}$ be the set of valid indices j when i is known, and $I_i^{(1)}(j) \equiv \{i : \hat{\lambda}_{i,j}^{(0)} \in C_0\}$ be the set of valid indices i when j is known. Then, the side description quantization cells $Q_i^{(1)}$, $Q_j^{(2)}$ can be expressed as a union of several $Q_{i,j}^{(0)}$ cells:

$$Q_i^{(1)} = \bigcup_{j \in I_j^{(2)}(i)} Q_{i,j}^{(0)}, \quad Q_j^{(2)} = \bigcup_{i \in I_i^{(1)}(j)} Q_{i,j}^{(0)}, \tag{6.3}$$

When the quantization cells $Q_{i,j}$ and the mappings $I_i(j)$, $I_j(i)$ are known, the optimal (in the MSE sense) codepoints can be computed by taking expectations over the quantization cells using $p(\lambda)$:

$$\hat{\lambda}_{i,j}^{(0)} = \int_{Q_{i,j}^{(0)}} \lambda p(\lambda|i,j) d\lambda, \tag{6.4}$$

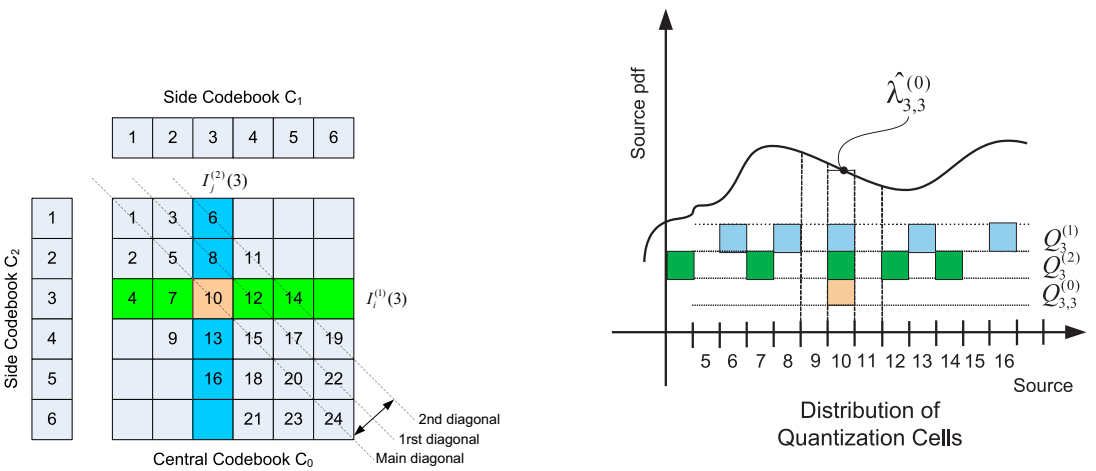
$$\hat{\lambda}_i^{(1)} = \sum_{j \in I_j^{(2)}(i)} \int_{Q_{i,j}^{(0)}} \lambda p(\lambda|i) d\lambda, \tag{6.5}$$

$$\hat{\lambda}_j^{(2)} = \sum_{i \in I_i^{(1)}(j)} \int_{Q_{i,j}^{(0)}} \lambda p(\lambda|j) d\lambda, \tag{6.6}$$

where $p(\lambda|i,j)$, $p(\lambda|i)$ and $p(\lambda|j)$ is the pdf of λ inside the quantization cells $Q_{i,j}^{(0)}$, $Q_i^{(1)}$ and $Q_j^{(2)}$, respectively.

Figure 6.2 provides an example of a source pdf, the corresponding quantization cells and their relationship as it is captured by the index assignment table. The two side description codebooks C_1 and C_2 index the column and the row of the sparse index assignment matrix which links to the central description codebook C_0. The index assignment matrix has a maximum of $6 * 6 = 36$ entries but only 24 of these 36 entries are linked to central description codewords, thus, the central description has a loss of 12 codewords. Assume that $C_0(10)$ and $C_1(3)$,$C_2(3)$ are the central and side description codewords that minimize the MDC distortion (6.2). The source pdf and the corresponding quantization cells $Q_3^{(1)}$, $Q_3^{(2)}$ and $Q_{3,3}^{(0)}$ are shown in Figure 6.2b. Note that the side description quantization cells are disjoint and consist of several central description quantization cells.

Construction techniques for the design of $N(0,1)$ MDSQ codebooks and the corresponding index assignment tables can be found in [74] and [140]. Both methods use a k-means approach with two steps, where the first step labels the samples of the source according to the central description quantization cells and the second step computes the optimal codepoints using equations (6.4),(6.5),(6.6). The method in [140] uses *deterministic annealing* to avoid local minima of the average total distortion but



(a) index assignment table

(b) central and side description quantization cells

Figure 6.2 An illustration of MDSQ. Sub-figure (a) shows the central description codebook C_0, the two side description codebooks C_1, C_2 and the index assignment matrix. Sub-figure (b) shows the source pdf and the corresponding quantization cells.

does not explicitly provide balanced descriptions. However, we found that balanced descriptions can be obtained by penalizing the average total distortion $E\{d_{tot}\}$ using the difference $|E\{d_1\} - E\{d_2\}|$ during the first step (labeling) of each k-means iteration ($E\{\cdot\}$ denotes expectation). The deterministic annealing method provides high-quality tradeoffs between central distortion and side distortions.

The behavior of an MDSQ system is studied using asymptotic high-rate analysis. Under the assumption that the source pdf is approximately constant over the side description quantization cells (the diameter of the disjoint cells $Q_i^{(1)}$, $Q_j^{(2)}$ goes to zero as $r \to \infty$) [135] (pg. 281), the average central and side distortions are given by the following equations:

$$\bar{d}_0 = C\sigma^2 2^{-2(1+\alpha)r} \tag{6.7}$$

$$\bar{d}_1 = \bar{d}_2 = S\sigma^2 2^{-2(1-\alpha)r} \tag{6.8}$$

where $\alpha \in (0,1)$ states the tradeoff between the central and the side distortions, σ^2 is the variance of the source, C and S are constants determined by the source pdf [135]. Note that the product $\bar{d}_0\bar{d}_1$ is constant, indicating the aforementioned *tradeoff* between the central and the side distortions. The parameter $\alpha = \frac{\log_2(k)}{r}$ where k is the number of diagonals of the index assignment matrix [135], [141]. For example, in Figure 6.2, the index assignment matrix has $k = 2$ diagonals.

An example of the behavior of an MDSQ system for resolution-constrained quan-

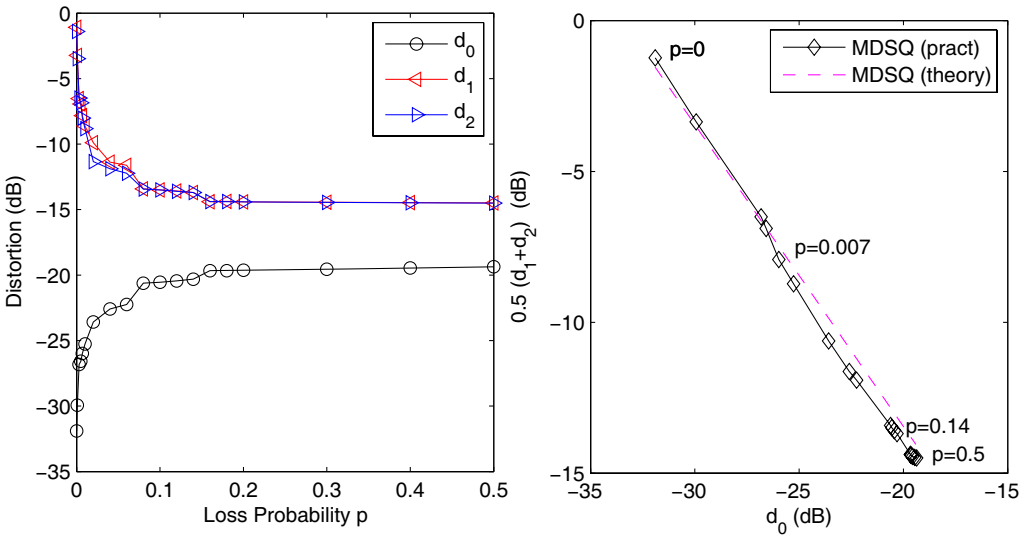


Figure 6.3 An example of resolution-constrained MDSQ of a $N(0,1)$ Gaussian with $r = 3$ bits. The left plot shows the mean central distortion d_0 and the mean side distortions d_1,d_2 for several loss probabilities. The right plot shows the central/side distortion tradeoffs.

tization of the Gaussian $N(0,1)$ case is provided in Figure 6.3. MDSQ codebooks were trained for several loss probabilities ranging from 0 to 0.5. The left plot shows the central and side distortions for each loss probability ρ. As ρ increases, the central distortion $\bar{d}_0$ becomes worst at the benefit of better side distortions d_1 and d_2. For each loss probability ρ, there is a tradeoff between d_0 and d_1,d_2. These tradeoffs are depicted in the right plot of Figure 6.3, along with the theoretical tradeoffs provided by the constant product $\bar{d}_0\bar{d}_1$. Note that d_0,d_1 and d_2 indicate data-driven averages (using 50.000 samples) and that the constants C and S for the resolution-constrained quantization of an $N(0,1)$ Gaussian are $C = \frac{\pi 3^{3/2}}{24}$, $S = \frac{\pi 3^{3/2}}{6}$ [141] (pg. 706).

6.2 Transform Coding using Multiple Description Scalar Quantization

A practical vector quantizer can be designed using transform coding and precomputed MDSQ codebooks. Let $x_p \sim N(0, \sigma_p^2)$, $p = 1, \cdots, P$ be a multivariate source of uncorrelated Gaussian random variables. Each variable x_p is encoded with MDSQ codebooks at a rate of $\acute{r}_p$ bits. Let $R_s = \sum_{p=1}^{P} \acute{r}_p$ be the rate of each side description and $R = 2R_s$ be the total rate for both descriptions. Theoretical approximations of the average central distortion $\bar{D}_0$ and average side distortions $\bar{D}_1$,$\bar{D}_2$ of the transform

coding system can then be computed according to the formulae:

$$\bar{D}_0 = \frac{1}{P}\sum_{p=1}^{P} \bar{d}_0(r_p) \tag{6.9}$$

$$\bar{D}_2 = \bar{D}_1 = \frac{1}{P}\sum_{p=1}^{P} \bar{d}_1(r_p). \tag{6.10}$$

The behavior the MDSQ-based transform coder (MDSQ_{TC}) is studied by Batllo and Vaishampayan in [141] using asymptotic high-rate arguments. The product $\bar{D}_0\bar{D}_1$ is constant as in the scalar case, indicating the existence of a distortion tradeoff that is linear on the logarithmic domain. Furthermore, the optimal bit allocation for MDSQ_{TC} is:

$$\acute{r}_p = \frac{R_s}{P} + \frac{1}{2}log_2\left(\frac{\sigma_p^2}{\left(\prod_{i=1}^{P}\sigma_i^2\right)^{\frac{1}{P}}}\right). \tag{6.11}$$

This result is the same with the bit allocation formula (3.10) for the single channel case at the half rate $R_s = \frac{1}{2}R$. Furthermore, the KLT transform is (again) the optimal transform for the MDC case [141].

An interesting note can now be made regarding the behavior of an MDSQ_{TC} quantizer at low loss rates. Assume that there are no packet losses ($\rho = 0$). Then, the scalar MDSQ codebooks are trained to minimize only the average central distortion: $E\{d_{tot}\} = E\{d_0\}$ (see equation (6.2)) resulting to a central codebook that is the same with the optimal single channel codebook with $2^{2\acute{r}_p}$ entries. Therefore, the p-th dimension of the central description will have a rate of $2\acute{r}_p$ bits. However, if the bit allocation is optimized for a single description, then r_p bits will be available for the p-th dimension, allocated according to equation (3.10). Let $\acute{D}_0$ be the average central distortion of $MDSQ_{TC}$ at $\rho = 0$ and D_0 the average distortion of typical transform coding with R bits. The ratio of the two distortions can be obtained using equations (3.9), (3.10), (3.11). and (6.11):

$$D_{penalty} = \frac{\acute{D}_0}{D_0} = \frac{\frac{1}{P}\sum_{i=1}^{P}\sigma_i^{-2}}{(\prod_{i=1}^{P}\sigma_i^{-2})^{\frac{1}{P}}}, \tag{6.12}$$

which is the arithmetic-to-geometric mean ratio of σ_i^{-2}. Since the arithmetic mean is always greater or equal to the geometric mean, $\acute{D}_0 \geq D_0$, the $MDSQ_{TC}$ scheme is suboptimal at $\rho = 0$. Furthermore, the performance loss depends on the distribution of the σ_i^{-2}. When $\sigma_i = \sigma_j, \forall i, j$, there is no penalty, but this case is rare in practice. The distortion penalty sets a bound to the central distortion $\acute{D}_0$ that can be achieved by the $MDSQ_{TC}$ scheme. Since $\acute{D}_0$ can only increase with increasing ρ, this bound is limiting the performance of the $MDSQ_{TC}$ quantizer at a whole range of low loss probabilities ρ.

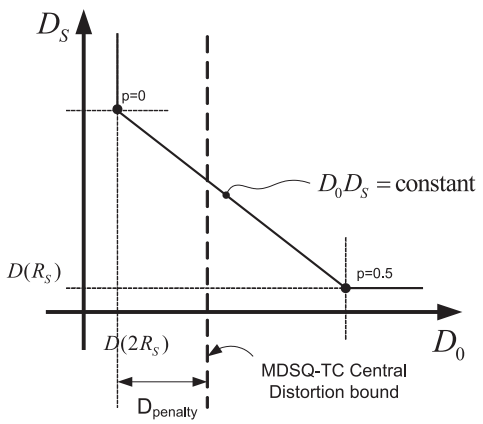


Figure 6.4 Central/Side distortion penalty (in log-scale) for $MDSQ_{TC}$ quantization. $D(R)$ is the average single description distortion at rate R, while R_s is the side description rate.

The observation made in this section states that the bit allocation in [141] is optimal only within a range of the available central-side distortion tradeoffs. It may be tempting to use a different (ad-hoc) bit-allocation for $MDSQ_{TC}$. For example, one could minimize only the central distortion at $\rho = 0$. Such a scheme may produce good results at $\rho = 0$ but it will be suboptimal at all other loss probabilities $\rho > 0$. Another option would be to use a greedy bit-allocation algorithm with tabulated central and side description distortions. Such an algorithm would perform better at lower loss probabilities, but it would also require higher side description rates, resulting to increased storage requirements.

The suboptimal behavior results from the limited range of validity of the equations (6.7) and (6.8) that describe the central distortion and the side distortions respectively, in [141], [135]. Similar formulae are also derived with less restrictive assumptions regarding the central description cells in [138]. Both high-rate analysis in [135] and [138] are based on the assumption that the source pdf is approximately constant inside the side description cells, as it is shown in [138], pg. 2096 and [135], pg. 281. This "smoothness" assumption restricts the index assignment matrix to be a "thin"-banded matrix, as stated in [138], pg. 2096. The central/side distortion tradeoffs and the distortion penalty for MDSQ_{TC} are illustrated in Figure 6.4. The optimal tradeoffs lie in the line D_0D_s = constant, above the full-rate bound $D_0 \geq D(2R_s) + D_{penalty}$ and above the half-rate distortion $D_s \geq D(R_s)$. An ideal MDC quantizer though, should operate like a single channel quantizer at $\rho = 0$ (that is $D_0 = D(2R_s)$).

6.3 GMM-based Multiple Description Coding

The extension of the MDSQ$_{TC}$ to GMM-based quantizers (GMM-MDSQ$_{TC}$) is relatively straightforward [151]. The source vector $\mathbf{x} \in \mathbb{R}^P$ is MDSQ$_{TC}$ quantized with each of the M Gaussian components of the GMM. Let $\hat{\mathbf{x}}_{0,m}$ and $\hat{\mathbf{x}}_{1,m}$, $\hat{\mathbf{x}}_{2,m}$ be the output of the central decoder and the two side decoders of the m-th Gaussian component, respectively. Let m' be the component that yields the least MDC distortion:

$$D_{tot,m} = \|\mathbf{x} - \hat{\mathbf{x}}_{0,m}\|_2^2 + \frac{\rho}{1-\rho}\left[\|\mathbf{x} - \hat{\mathbf{x}}_{1,m}\|_2^2 + \|\mathbf{x} - \hat{\mathbf{x}}_{2,m}\|_2^2\right]. \tag{6.13}$$

Each GMM-MDSQ$_{TC}$ description is composed of the index m' of the "best" encoding and the corresponding MDSQ$_{TC}$ indices. Therefore, there is a slight loss of efficiency (since the index m' is transmitted twice) at the benefit of simple implementation.

Assuming balanced descriptions, let R_m be the number of bits allocated for each side description of each Gaussian component and $R_s = \sum_{m=1}^{M} 2^{R_m}$ be the total side description rate of GMM-MDSQ$_{TC}$. A surprising result is that the optimal component bit allocation for the GMM-MDSQ$_{TC}$ quantizer is the same with that of a GMM-based quantizer at a rate of R_s bits (provided by equation 3.21), as stated in [151]. Therefore, similarly to the MDSQ$_{TC}$ quantizer, the bit allocation of the GMM-MDSQ$_{TC}$ quantizer is optimized to minimize solely the side distortions inheriting the suboptimal behavior of MDSQ$_{TC}$ at lower loss probabilities.

The MDSQ quantizer at bit rates lower than 2 bits/dimension fails to provide balanced descriptions. This is a significant problem because for spectral envelopes and harmonic phases most of the scalar MDSQ quantizers in a GMM-MDSQ$_{TC}$ system operate at low rates of 1-2 bits. A practical and efficient solution to this problem is to flip the MDSQ indices (i, j) of the odd-indexed dimensions of $\mathbf{x}$.

6.4 WGMM-based Multiple Description Coding of Phase data

The construction of a scalar multiple description quantizer for circular $N_w(0, \sigma^2)$ wrapped Gaussian random variables is not trivial due to the fact that the shape of the wrapped pdf $N_w(0, \sigma^2)$ depends of the variance σ^2. In practice, this means that we need a different set of $N_w(0, \sigma^2)$ quantizers for each possible rate, variance and loss probability. The number of codebooks needed for this solution is overwhelming.

In Section 4.5 we saw that it is possible to construct a quantizer for the phase data using WGMM and circular scalar quantizers of $N_w(0, \sigma^2)$ random variables created by wrapping a linear Gaussian codebook on the unit circle. This construction of circular codebooks is not optimal and it does not minimize the WSE as the PCF-based quantizers do (in Section 4.5.2) but it is fairly accurate for lower variances σ^2 and in practice it is only 2-5 bits worst than the PCF-based method, as it is demonstrated in the experiments of Section 4.6. Therefore, it is possible to construct a scalar multiple

description phase quantizer by wrapping the central and side description codebooks of an MDSQ quantizer to the circumference of the unit circle.

The only difference of this scalar quantizer from the MDSQ quantizer is that the encoder finds the pair of indices (i, j) of the first and the second description by minimizing an appropriate version of the total distortion in (6.2). The *wrapped total distortion* is of the form:

$$d_{w,tot}(\lambda, \hat{\lambda}_{i,j}^{(0)}, \hat{\lambda}_{i,j}^{(1)}, \hat{\lambda}_{i,j}^{(2)}) = d_w(\lambda, \hat{\lambda}_{i,j}^{(0)}) + \frac{\rho}{1-\rho}\left(d_w(\lambda, \hat{\lambda}_{i,j}^{(1)}) + d_w(\lambda, \hat{\lambda}_{i,j}^{(2)})\right), \quad (6.14)$$

where ρ is the packet loss probability, $\hat{\lambda}_{i,j}^{(0)}$, $\hat{\lambda}_{i,j}^{(1)}$ and $\hat{\lambda}_{i,j}^{(2)}$ are the central/side description MDSQ codepoints that correspond to the index pair (i, j), and $d_w(\cdot, \cdot)$ is the wrapped square error (WSE) function (equation (4.27)).

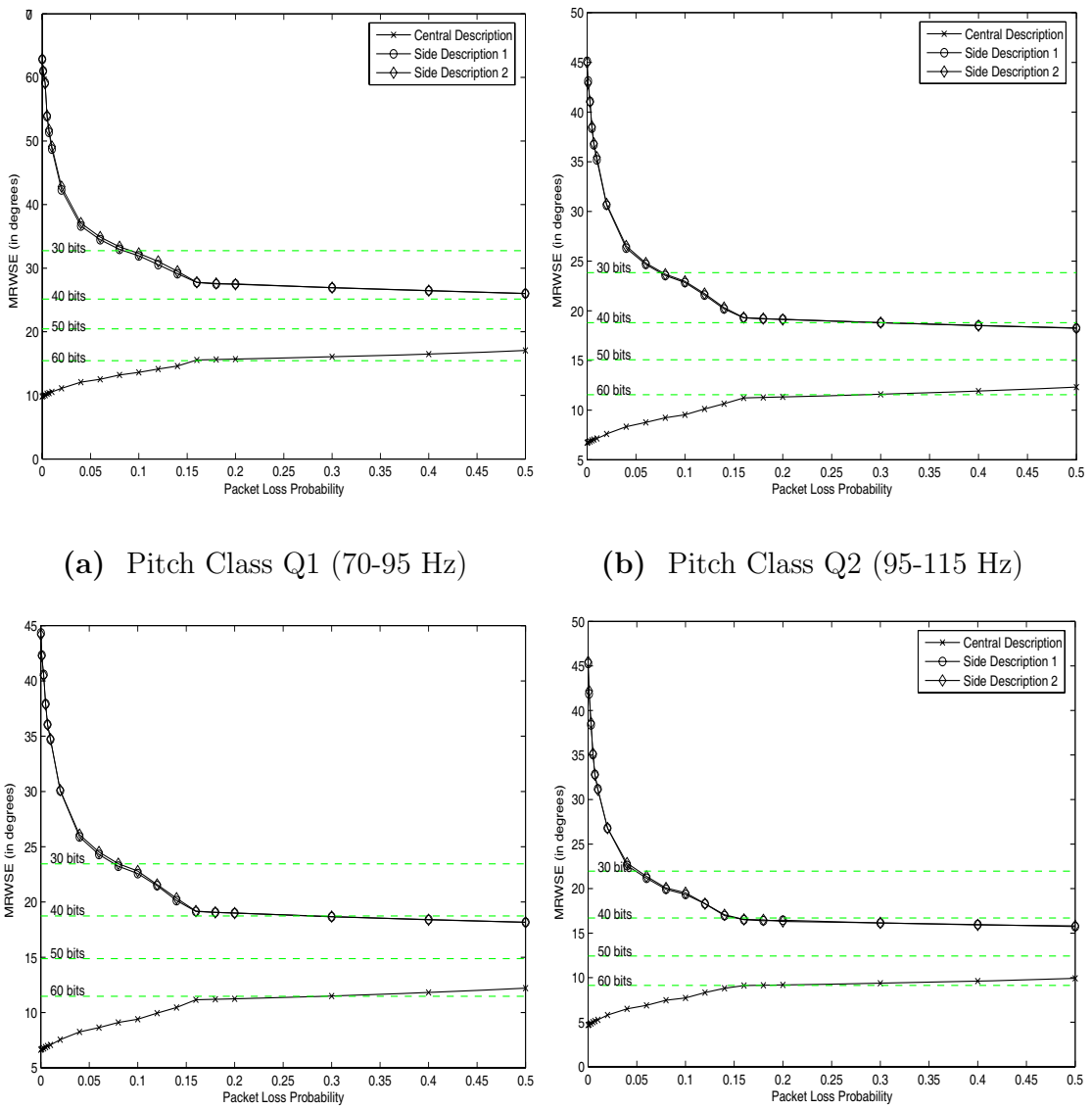
A phase vector $\vec{\theta}$ is quantized according to each of the M wrapped multivariate Gaussian components of the WGMM. Let $\hat{\vec{\theta}}_m^{(0)}$, $\hat{\vec{\theta}}_m^{(1)}$ and $\hat{\vec{\theta}}_m^{(2)}$ the central description and the two side description reconstructions associated with the m-th wrapped Gaussian component of the WGMM. The WGMM-based phase quantizer selects the m'-th wrapped Gaussian component that minimizes the vectorized form of the wrapped total distortion:

$$d(\vec{\theta}, \hat{\vec{\theta}}_m^{(0)}, \hat{\vec{\theta}}_m^{(1)}, \hat{\vec{\theta}}_m^{(2)}) = \sum_{k=1}^{K} d_{w,tot}(\vec{\theta}(k), \hat{\vec{\theta}}_m^{(0)}(k), \hat{\vec{\theta}}_m^{(1)}(k), \hat{\vec{\theta}}_m^{(2)}(k)), \quad (6.15)$$

where K is the number of dimensions and $\vec{\theta}(k), \hat{\vec{\theta}}_m^{(0)}(k), \hat{\vec{\theta}}_m^{(1)}(k), \hat{\vec{\theta}}_m^{(2)}(k)$ are the k-th dimensions of the corresponding vectors. The index m' of the selected wrapped Gaussian component is transmitted together with the indices of the associated scalar multiple description quantizers, as it is made in GMM-based MDC (Section 6.3).

The allocation of bits to the wrapped Gaussian components of the WGMM and the scalar quantizers within each component is made using the bit-allocation method for multiple description coding of GMM. This strategy is justified when the variances of the WGMM are small (below 1.0) because in that case the interval $(0, 2\pi]$ contains most of the mass of the Gaussian pdf and the overlapping of the tiled Gaussian components is low. A similar strategy was used in single description WGMM-based quantization by the bit-allocation algorithm B at Section 4.5.

Figure 6.5 shows an evaluation of the proposed WGMM-based MDC algorithm using the mean-root-wrapped-square-error criterion (MRWSE) that was introduced in equation (4.38). The algorithm is evaluated for the quantization of the first 21-24 dispersion phases of pitch classes Q1 to Q4, as they were introduced in Table 4.2, Section 4.6. The dispersion phases were modeled with the corresponding low-frequency WGMM that was trained according to Section 4.6 and the same test-set was used. The central and side distortions of the WGMM-based MDC quantizers are plotted against the packet loss probability ρ, for a range of values ranging from 0 to 0.5. The distortion is associated with the rate using four horizontal lines which correspond



(a) Pitch Class Q1 (70-95 Hz)

(b) Pitch Class Q2 (95-115 Hz)

(c) Pitch Class Q3 (115-142 Hz)

(d) Pitch Class Q4 (142-176 Hz)

Figure 6.5 WGMM-based MDC examples. Central and Side Distortions (MRWSE) for several loss probabilities.

to the distortion levels achieved by a single description WGMM-based quantizer at rates of 30, 40, 50 and 60 bits. The single description WGMM-based quantizer uses wrapped codebooks and the bit allocation algorithm B (Section 4.5). The dispersion phases were encoded with 45 bits/description. We can observe that at the highest correlation point ($\rho = 0.5$), each side description provides distortion below the 40-bit level, while the central description operates at a little higher than the 60-bit level. Therefore, we can expect a high-quality reconstruction when both descriptions are available and a lower quality reconstruction when only one description is available.

6.5 Erasure Channel Decoding for GMM-MDSQ$_{TC}$ quantizers

The redundancy that is introduced between the two descriptions in GMM-MDSQ$_{TC}$ can be used to combat bit-errors that may have occurred during the transmission. Let the data vector $\mathbf{x} \in \mathbb{R}^P$ be GMM-MDSQ$_{TC}$-encoded with R_s bits per description and assume that each description is error protected with a channel code that can correct e_{cor} bits and detect e_{det} bit-errors. Let e_1, e_2 be the number of bit-errors that are detected but not corrected by the channel coder for description 1 and 2, respectively. The channel decoder outputs a set of candidate indices U_1, U_2 with 2^{e_1}, 2^{e_2} elements for the corresponding description.

The case where both descriptions contain bit-errors is similar to the case where we have a single description with $2R_s$ bits and $e_1 + e_2$ bit-errors. The set of candidate indices then is the product set $U = (U_1 \times U_2) \backslash \mathcal{X}$, where $\mathcal{X}$ is the set of "non-valid" indices: the pairs (I_1, I_2) that do not correspond to a codevector due to the fact that the index assignment matrix is sparse or that they do not refer to the same Gaussian component of the GMM. A solution to this problem is provided in [88] (chapter 6) where correlated information from past speech frames is used to average the candidate codevectors.

This section presents a novel decoding method for the case where only one description contains bit-errors. In this case, the correlations between the received descriptions can be used to significantly reduce the average distortion of the reconstruction. Assume that the second description contains e_1 uncorrected bit-errors while the first is received correctly. Initially, the problem will be addressed for MDSQ and then the solution will be extended to GMM-MDSQ$_{TC}$. Finally, the proposed method is experimentally evaluated using RCC cepstral coefficients.

6.5.1 MDSQ Case

The optimal decoding of an MDSQ system for a squared error distortion measure when one description is lost is given by equations (6.5),(6.6). These reconstructions are based on the pdf $p(\lambda)$ of the data λ. The conditional probability of the data when

the first description is received, is:

$$p(\lambda|i) = \frac{p(\lambda)}{\sum_{j\in I_j^{(2)}(i)} \int_{Q_{i,j}^{(0)}} p(\lambda)d\lambda} 1(\lambda \in Q_i^{(1)}), \tag{6.16}$$

where $1(\cdot)$ is the characteristic function. The corresponding reconstruction is:

$$\hat{\lambda}_i^{(1)} = \sum_{j\in I_j^{(2)}(i)} \int_{Q_{i,j}^{(0)}} \lambda p(\lambda|i)d\lambda. \tag{6.17}$$

Note that the integration in this case is made over all central quantization cells with indices $(i,j) = (i, I_j^{(2)}(i))$, where i is the index from description 1 which is received and $I_j(i)$ is the set of valid indices from description 2, according to the index assignment matrix (see Section 6.2). However, the set $I_j^{(2)}(i)$ of description 2 indices corresponds to the case where description 1 is totally lost. In the case of a few bit-errors, a much smaller set of candidate indices U_2 can be obtained from the channel decoder. Thus, the expectation can be made over a disjoint quantization cell with much smaller diameter, resulting to a better reconstruction. The optimal MSE reconstruction is:

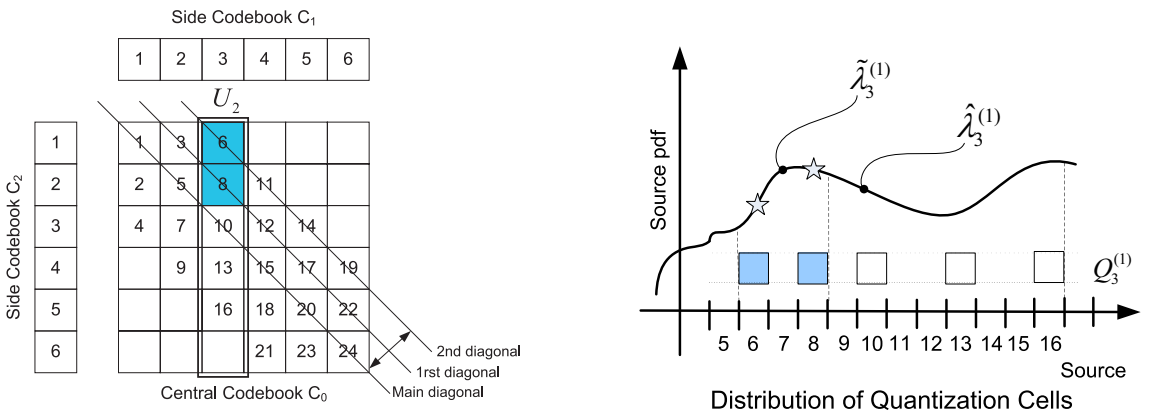
$$\tilde{\lambda}_i^{(1)} = \sum_{j\in U_2} \left(\frac{\int_{Q_{i,j}^{(0)}} \lambda p(\lambda)d\lambda}{\sum_{j'\in U_2} \int_{Q_{i,j'}^{(0)}} p(\lambda)d\lambda} \right). \tag{6.18}$$

This formula is amenable to analytic computation for the Gaussian case but it is computationally expensive. A fast approximation can be obtained under the "high-rate" assumption that $p(\lambda)$ is constant within each central quantization cell $Q_{i,j}^{(0)}$. The resulting fast reconstruction formula is:

$$\tilde{\lambda}_i^{(1)} \approx \sum_{j\in U_2} \left(\frac{p(\hat{\lambda}_{i,j}^{(0)})S_{i,j}}{\sum_{j'\in U_2} p(\hat{\lambda}_{i,j}^{(0)})S_{i,j}} \right) \hat{\lambda}_{i,j}^{(0)}, \tag{6.19}$$

where $S_{i,j}$ is the length of the quantization cell $Q_{i,j}^{(0)}$ which can be precomputed and stored off-line.

An example is provided in Figure 6.6. The second description is received with bit-errors. The first description i states that the integration should be made over the quantization cells $Q_{i,j}^{(0)}$ with indices $j = \{6, 8, 10, 13, 16\}$ as shown at the 3-rd column of the index assignment matrix (Figure 6.6a). The second description states that the candidate indices are only $j = \{6, 8\}$. Therefore the integration is made using the quantization cells $Q_{3,6}^{(0)}$ and $Q_{3,8}^{(0)}$. The side decoder reconstruction $\hat{\lambda}_i^{(1)}$ and the reconstruction $\tilde{\lambda}_i^{(1)}$ made by the proposed method are depicted in Figure 6.6b. Note that $\tilde{\lambda}_i^{(1)}$ is closer to both candidate central description reconstructions (stars).



(a) index assignment table

(b) central and side description quantization cells

Figure 6.6 An example of MDSQ decoding when description 2 contains bit-errors. Sub-figure (a) shows the central description codebook C_0, the two side description codebooks C_1, C_2 and the index assignment matrix. The two candidate quantization cells are highlighted. Sub-figure (b) shows the source pdf, the side decoder reconstruction $\hat{\lambda}_i^{(1)}$ and the proposed reconstruction $\tilde{\lambda}_i^{(1)}$. The central description reconstructions are indicated with the stars.

6.5.2 GMM-MDSQ$_{TC}$ Case

The extension to the GMM-MDSQ$_{TC}$ quantizer is relatively straightforward. Assume that description 1 (index I_1) is received without bit-errors, while description 2 contains bit-errors. The channel decoder provides a set of L candidate indices $U_2 = \{I_{2,l} : l \in 1, \cdots, L\}$ for the second description. The indices I_1, $I_{2,l}$ are decomposed to the following set of component and scalar quantization indices:

$$I_1 = \{m', i_1, i_2, \cdots, i_P\}, \tag{6.20}$$

$$I_{2,l} = \{m_l, j_{1,l}, j_{2,l}, \cdots, j_{P,l}\},\ l = 1, \cdots, L \tag{6.21}$$

where P is the number of dimensions. Only a subset of the candidate indices U_2 are valid, in the sense that the pair $(I_1, I_{2,l})$ can be produced from the encoder. Let U_2' be the set of *valid candidate indices*:

$$U_2' = \{I_{2,l} : m_l = m' \ \&\ (i_p, j_{p,l}) \text{ is valid } \forall p \in \{1, \cdots, P\}\}, \tag{6.22}$$

where a pair of indices $(i_p, j_{p,l})$ is considered to be valid if the index assignment matrix maps it to a central codebook entry. Note that the component index m' is transmitted twice, one in each description, therefore every valid candidate index must have the same component index. For convenience, let us renumber the elements in U_2' so that $U_2' = \{I_{2,l}, l = 1, \cdots, L'\}$, where L' is the size of the set U_2'.

Following the nomenclature of Chapter 3, let $\mathbf{x}'_{m'} = [x'_{m',1}, x'_{m',2}, \cdots, x'_{m',P}]^T$ be the P uncorrelated variables that are encoded with the m'-th MDSQ$_{TC}$ quantizer, and $\sigma^2_{m',1},\ \sigma^2_{m',2},\ \ldots,\ \sigma^2_{m',P}$ be the corresponding variances. Let $r_{m',p}$ be the number of bits allocated to each of the $x'_{m',p}$, $p = 1, ..., P$. A set of pre-trained MDSQ codebooks for the $N(0,1)$ Gaussian is obtained using the methods discussed in Section 6.1: one pack of MDSQ codebooks (central and side codebooks) for each loss probability ρ and each rate r_p. The GMM-MDSQ$_{TC}$ encoder/decoder operates with the subset of MDSQ codebooks that corresponds to the loss probability of the network. Let $\hat{\lambda}^{(0)}_{i,j}(r)$ be the (i,j)-th entry of the central description codebook trained for $N(0,1)$ variables at a side description rate of r bits, and $S_{i,j}(r)$ the length of the corresponding quantization cell. Let $\mathbf{y}^{(l)}$, $l = 1, ..., L'$ be the *valid candidate codevectors*:

$$\mathbf{y}^{(l)} = \begin{bmatrix} \sigma_{m',1}\hat{\lambda}^{(0)}_{i_1,j_{1,l'}}(r_{m',1}) \\ \sigma_{m',2}\hat{\lambda}^{(0)}_{i_2,j_{2,l'}}(r_{m',2}) \\ \vdots \\ \sigma_{m',P}\hat{\lambda}^{(0)}_{i_P,j_{P,l'}}(r_{m',P}) \end{bmatrix}. \tag{6.23}$$

Let $Q^{(l)}$ be the P-dimensional quantization cell associated with codevector $\mathbf{y}^{(l)}$:

$$Q^{(l)} = Q_{i_1,j_{1,l}} \times Q_{i_2,j_{2,l}} \times \cdots \times Q_{i_P,j_{P,l}}, \tag{6.24}$$

where $Q_{i_p,j_{p,l}}$ is the (scalar) central description quantization cell associated with the codepoint $\sigma_p\hat{\lambda}^{(0)}_{i_p,j_{p,l}}(r_{m',1})$. The cell $Q_{i_p,j_{p,l}}$ is an interval of length $S_{i_p,j_{p,l}}(r_{m',p})\sigma_p$. The optimal MSE (Mean Square Error) reconstruction is given by the following formula:

$$\tilde{\mathbf{x}}'_{1,m'} = \sum_{l=1}^{L'} \left(\frac{\int\limits_{\mathbf{x}\in Q^{(l)}} \mathbf{x}p(\mathbf{x})d\mathbf{x}}{\sum\limits_{l'=1}^{L'} \int\limits_{\mathbf{x}\in Q^{(l')}} \mathbf{x}p(\mathbf{x})d\mathbf{x}} \right), \tag{6.25}$$

where $p(\mathbf{x})$ is the pdf of $\mathbf{x}$. A computationally attractive high-rate approximation can be made if we assume that $p(x)$ is approximately constant inside each quantization cell $Q^{(l)}$. The resulting reconstruction formula is:

$$\tilde{\mathbf{x}}'_{1,m'} = \sum_{l=1}^{L'} \left(\frac{\prod\limits_{p=1}^{P} p(\hat{\lambda}^{(0)}_{i_p,j_{p,l}}(r_p))S_{i_p,j_{p,l}}\sigma_{m',p}}{\sum\limits_{l'=1}^{L'}\prod\limits_{p=1}^{P} p(\hat{\lambda}^{(0)}_{i_p,j_{p,l'}}(r_p))S_{i_p,j_{p,l'}}\sigma_{m',p}} \right)\mathbf{y}^{(l)}, \tag{6.26}$$

where $p(\lambda)$ is the $N(0,1)$ pdf.

The decoded GMM-MDSQ$_{TC}$ reconstruction can be obtained by rotating and translating $\tilde{\mathbf{x}}'_{1,m'}$ according to the statistics of the m'-th Gaussian component of the GMM (see equation 3.17):

$$\tilde{\mathbf{x}}_1 = \mu_{m'} + V_{x,m'}\tilde{\mathbf{x}}'_{1,m'}. \tag{6.27}$$

Complexity Issues

The reconstruction formula (6.26) has a complexity that increases linearly with L', but the number L' of valid candidate codevectors increases rapidly with the number of detected but uncorrected bit-errors. Assume that each description is quantized with 30 bits, and that the channel code only detects bit-errors. The number of candidate codevectors when there are K bit-errors is $L = \begin{pmatrix} 30 \\ K \end{pmatrix}$. For $K = 1 \Rightarrow L = 30$, for $K = 2 \Rightarrow L = 30 * 29 = 870$, while for $K = 3 \Rightarrow L = 30 * 29 * 28 = 24360$. The number of valid candidate codevectors is smaller that L: $L' \leq L$, but the complexity remains overwhelming for more than 2 bit-errors because all candidate codevectors should be checked for validity.

6.5.3 Results

An evaluation is made using 20-dimensional RCC cepstral coefficients derived from 20 ms speech frames. The training-set was the same one used in Section 3.3, while the test-set consisted of 10.000 samples. The RCC source was encoded with 60 bits using two balanced 30 bit descriptions. The source was encoded and decoded using the central decoder (distortion D_0) and the two side decoders 1 and 2 (distortions D_1 and D_2, respectively). A number of 1, 2 bit-errors was introduced to description 2 and the proposed decoder was evaluated using the measured distortions $D_{ber,1}$, $D_{ber,2}$, respectively. The candidate indices for each vector were computed by introducing 1-2 random bit-errors to the 30 bit description and changing every possible set of two bits; a total of $\begin{pmatrix} 30 \\ 1 \end{pmatrix} = 30$ candidates for the 1 bit error case and a total of $\begin{pmatrix} 30 \\ 2 \end{pmatrix} = 30 * 29 = 870$ candidates for the 2 bit error case. The evaluation is made with the MSE (Mean Square Error) criterion using the raw RCC parameters.

The results are depicted in Figure 6.7. The distortions were evaluated for several loss probabilities ρ ranging from 0 to 0.5. At $\rho = 0$, the central distortion D_0 is very low but the side distortions D_1, D_2 are very high. As the loss probability increases, the side distortions become lower at the cost of a higher central distortion. From $\rho = 0.2$ and above, the central/side descriptions almost converge to the state of highest correlation. At $\rho = 0.5$, the side distortions D_1, D_2 are equal to the distortion provided by a GMM-based quantizer at a rate of 30 bits. Having both descriptions provides a much better reconstruction. When the second description has bit-errors, the distortion $D_{ber,1}$ (or $D_{ber,2}$) provided by the proposed method is much lower than D_1 which corresponds to a complete loss of description 2. At the higher correlation point ($\rho = 0.5$) the proposed method almost corrects the single/double bit errors, providing a reconstruction that is close to the one provided by the central description. Furthermore, the benefits of the proposed method appear even from low correlations ($\rho \geq 0.03$).

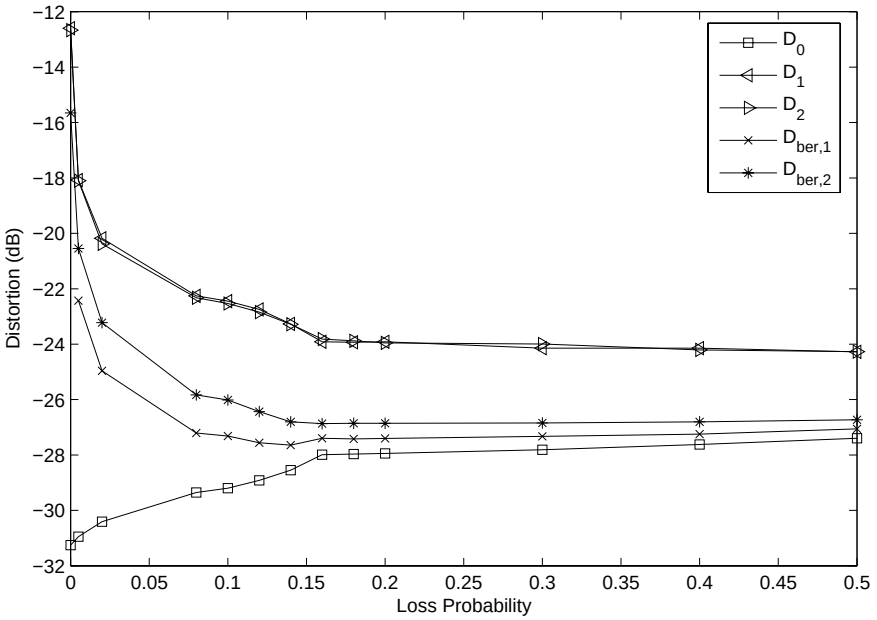


Figure 6.7 GMM-MDSQ$_{TC}$ for an 20-dimensional RCC source of speech spectral envelopes. Each description is encoded using 30 bits. D_0,D_1 and D_2 correspond to the average distortions of the central and the two side descriptions, respectively. $D_{ber,1}$ and $D_{ber,2}$ corresponds to the average distortion when description 2 contains 1 and 2 bit-errors. The distortions are depicted in dB scale.

A practical aspect of this work is that it makes possible to reallocate a portion of the redundancy given for channel coding to the GMM-MDSQ$_{TC}$ quantizer. For example, the error correction capability of the channel coder could be reduced by 1, allowing to reallocate the saved bits to the GMM-MDSQ$_{TC}$ quantizer in order to provide a better reconstruction when there are no bit-errors. Furthermore, in a wireless transmission setting, the two descriptions could be routed through different (independent) channels in order to minimize the possibility of damaging both descriptions of a single speech frame.

Chapter 7

Multiple Description Transform Coding

The complexity of the GMM-MDSQ$_{TC}$ quantizer increases rapidly with the bit-rate and the complexity of the channel. A set of precomputed MDSQ codebooks (one central, two side codebooks, and an index assignment mapping) has to be stored for each possible rate and each possible channel condition. In practice, GMM-MDSQ$_{TC}$ is effective only under several constraints: a rather simple channel model (for example, two independent symmetric channels with equal loss probabilities), balanced descriptions, and a source that does not require scalar MDSQ quantization at increased rates. The latter constrain is more-or-less satisfied when the 20 RCC coefficients (or the 12-24 coefficients of the harmonic phases) of a single frame are quantized with GMM-MDSQ$_{TC}$ but it is violated when the coefficients of more than one frame are block quantized together. A solution to this problem is to further constrain the side description MDSQ codebooks to be integer-sized [151], at the cost of performance loss.

Another drawback of GMM-MDSQ$_{TC}$ is related to the bit-allocation procedure of MDSQ$_{TC}$. Both GMM-MDSQ$_{TC}$ and MDSQ$_{TC}$ are optimized solely for the side distortions (equiv. high loss probabilities). This poses a bound to the central distortion of the MDSQ$_{TC}$ quantizer as it is addressed in Section 6.2. The bound consequently effects the behavior of GMM-MDSQ$_{TC}$ at lower loss probabilities. Another approach to the bit allocation problem (i.e., a greedy optimization) would require higher side description rates for the MDSQ quantizers. Depending on the source, the latter may considerable increase the complexity and the storage requirements of GMM-MDSQ$_{TC}$.

This chapter contributes to another type of multiple description quantizers, which are inherently scalable to high bit-rates. In Multiple Description Transform Coding (MDTC), the redundancy between the descriptions is introduced via a multiplication with a *correlating matrix*. Currently, there are two choices for the construction of the correlating matrix: a transform and a frame expansion. In [152], [147], [146], the correlations are introduced in a pairwise manner via a transform matrix. The optimal pairing of the encoded coefficients is discussed in [145]. However, as noted

in [145] (pg. 2211), transform coding is better than MDSQ in lower redundancies (equiv. loss probabilities) and MDSQ is better than transform coding in higher redundancies. The suboptimal behavior of transform coding in higher redundancies is partially corrected in [153] via the introduction of a layered coding approach which actually corresponds to a frame expansion.

A different approach to transform coding is made when the correlating matrix is a frame expansion [154], [148]. Quantization issues regarding frame expansions are discussed in [149], [150]. Frame expansions can also be used as source-channel codes that feature a soft degradation compared to conventional block-channel codes [155]. The construction and optimization (for given channel conditions) of the frame expansion matrix F for MDTC is, in general, a complicated problem. In [154], frame expansions are optimized by varying one coefficient at a time using numerical gradient descent techniques. However, for increased dimensions this approach is prohibitively expensive in terms of storage requirements and optimization complexity, while its effectiveness is demonstrated for a limited number of dimensions. Furthermore, all reported MDTC methods focus to entropy-constrained MDTC [146], [147], [148], [154], [145], aiming mostly to audio, image and video coding applications which benefit from large encoding buffers. Therefore, the analysis and the design of the presented MDTC methods cannot be applied to resolution-constrained quantization. However, many ideas from entropy-constrained MDTC can also be used in the case of our interest: resolution-constrained MDTC.

This chapter proposes a novel resolution-constrained MDC scheme based on *transform coding*. Section 7.1 presents a novel MDTC quantizer for multivariate Gaussian sources. The quantizer is based on a special class of *tight frame expansions* called *Parseval frames*. The Parseval frames are constructed in a systematic manner with a predefined number of degrees of freedom. Two special cases are then studied, MDTC with 1 and MDTC with P degrees of freedom (where P is the number of dimensions). For these cases, we propose a novel *optimal consistent reconstruction* algorithm for the central quantizer. Section 7.2 extents the proposed MDTC quantizer to sources that can be modeled with a GMM. The quantizers are evaluated and compared to MDSQ$_{TC}$ and GMM-MDSQ$_{TC}$. The proposed quantizers feature very low complexity and storage requirements, rate scalability, excellent performance at lower redundancies and competitive performance at higher redundancies. In combination with GMM-MDSQ$_{TC}$ they provide a practical resolution-constrained MDC system with very good performance and low computational overhead at the whole range of central/side distortion tradeoffs. Finally, Section 7.3 identifies a sub-optimality in MDTC and proposes a modification to the MDTC encoding procedure that improves the central distortion in higher redundancies. In the new algorithm, the two Gaussian encoders cooperate in order to minimize a distortion criterion that takes into account the central decoder. As illustrated by example, the proposed quantizer provides much better central/side description tradeoff points. This idea can also be extended to entropy-constrained multiple description transform coders.

7.1 Multiple Description Transform Coding of Multivariate Gaussian Sources

In this section, we propose a framework for resolution-constrained MDTC that uses Parseval frame expansions with a predefined number of degrees of freedom. The proposed algorithm optimizes MDTC for given network conditions and requires minimal memory and computational resources for optimization, encoding and decoding. Section 7.1.1 presents the necessary background knowledge and defines the notion of optimal consistent reconstruction. Section 7.1.2 provides an overview of the proposed MDTC system and the methodology of Parseval frame construction. In Section 7.1.3 we focus on two computationally tractable cases; expansions with one and expansions with P degrees of freedom where P is the number of the Gaussians. An optimal consistent reconstruction algorithm for these cases is given in Section 7.1.4. The proposed MDTC schemes are tested in two experiments shown in Section 7.1.5: MDTC for a *simple symmetric channel* with equal loss probabilities, and *hierarchical coding*. Furthermore, a comparison is made between the proposed methods and MDSQ$_{TC}$. We experimentally show that a combined MDTC and MDSQ$_{TC}$ system achieves a wide range of admissible central/side distortion tradeoffs while having low complexity and storage requirements. Section 7.1.6 discusses the significance of this work in comparison to related work and proposes possible improvements.

7.1.1 Frames and Frame Expansions

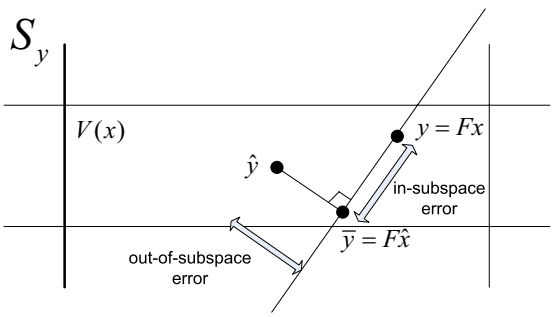
Frames can be seen as a generalization of linear transforms. Let x be a random vector in $\Re^P$ and $F \in \Re^{D\times P}$. The rows of F form a *frame* iff the frame condition holds:

$$Ax^Tx \leq x^TF^TFx \leq Bx^Tx,\ \forall x \in \Re^P. \tag{7.1}$$

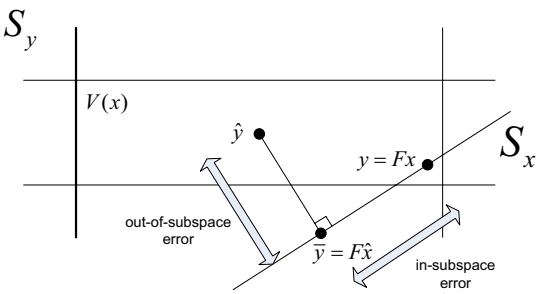
The lower bound in (7.1) states that F spans $\Re^P$, therefore, if (7.1) holds then $D \geq P$. The constants A and B are equal to the minimum and maximum eigenvalues of F^TF, respectively. A frame F is called *tight frame* when $A = B$. When $A = B = 1$, the frame is called *1-tight* or *Parseval*.

Now assume that we use a quantizer $Q_y(\cdot)$ to encode $y = Fx$: $\hat{y} = Q_y(y)$, and that we recover a decoded value $\hat{x}$ from $\hat{y}$ using a matrix operation $\hat{x} = F^{\#}\hat{y}$. The matrix that minimizes the mean square error $E\{\|x-\hat{x}\|_2^2\}$ is given by the *pseudoinverse* of F: $F^{\#} = (F^TF)^{-1}F^T$. This MMSE solution is also referred to as *MMSE Reconstruction* and corresponds to the projection of $\hat{y}$ onto the column space of F. A very useful property is that for a Parseval frame, the pseudoinverse of F is its transpose $F^{\#} = F^T$.

A schematic display of the quantization process in frame expansions and MMSE reconstruction can be found in Figure 7.1. Let $S_x = F(\Re^P)$ be the *image* of F (*column space* of F) in $S_y = \Re^D$. The initial vector x is mapped to $y = Fx$, $y \in S_x$. Then y is encoded to $\hat{y}$ which is the representative of the cell $V(y) = V(Fx) \in \Re^D$ of x. The MMSE reconstruction is $\hat{x} = F^{\#}\hat{y}$ and it corresponds to the point $\overline{y} = F\hat{x} \in S_x$.



(a) Consistent MMSE Reconstruction



(b) Inconsistent MMSE Reconstruction

Figure 7.1 Examples of consistent and inconsistent MMSE reconstructions in frame expansions.

Depending on the geometry of the quantization cell, the reconstruction point $\bar{y}$ may not be in the same cell as y. A mapping that assures that y and $\bar{y}$ belong to the same cell is called *Consistent Reconstruction* [149]:

Definition 1 *Let $\hat{x}$ be the reconstruction of x. The reconstruction $\hat{x}$ is called* consistent reconstruction *iff $Q_y(Fx) = Q_y(F\hat{x})$.*

The property of consistency does not take into account the statistics of $y = Fx$ inside the quantization cell $V(Fx)$. Let $G(x) = \{a \in \Re^P : Q_y(Fa) = Q_y(Fx)\}$ be the set of points mapped onto the image of F inside the quantization cell $V(Fx)$. Let $p(x)$ be the pdf of x in $\Re^P$. The pdf of $x' \in G(x)$ is:

$$g(x') = \frac{p(x')}{\int\limits_{G(x)} p(a)da},\ x' \in G(x), \tag{7.2}$$

Now we can provide a formal definition of the Optimal Consistent Reconstruction:

Definition 2 Optimal Consistent Reconstruction *(OCR) under a distortion measure $d(\cdot,\cdot)$ is the consistent reconstruction $\hat{x}$ that minimizes the mean reconstruction error $E_{g(x')}\{d(x',\hat{x})\}$, $x' \in G(x)$, $\forall x \in \Re^P$.*

According to this definition, OCR is the optimal mean square error reconstruction according to the statistics of each cell $V(Fx)$, for all cells $\{V(Fx) : \forall x \in \Re^P\}$. Therefore, OCR is optimal with respect to the specific quantizer, while the MMSE reconstruction provided by the pseudoinverse is optimal according to a more generic but not so accurate quantizer model. The fine grained optimality of OCR is not without cost; for each cell, the expectation is taken over $G(x)$ (the image of F that lies inside the cell $V(Fx)$). The complexity required for this expectation increases rapidly

with the complexity of the shape of $V(Fx)$ and the number of dimensions P. However, fast solutions can be found for some cases, as will be reported in Section 7.1.4. OCR is mentioned in [150] (as "optimal reconstruction") but no formal definition is given and no solution is provided due to the complexity of the generic case.

It can be shown that for a squared error distortion measure the OCR is given by the equation:

$$\hat{x} = \int_{G(x)} ag(a)da. \tag{7.3}$$

7.1.2 MDTC using Parseval Frame Expansions

In this section we will present an *overview* of the proposed MDTC framework, the operation of the *encoder*, the *side decoders* and the *central decoder*. The construction of *Parseval frames* is presented, and a discussion regarding time and memory complexity is following.

Overview

At the sender, the zero-mean Gaussian random vector $x \in \Re^P$ is transformed into two correlated vectors $y_1 = F_1 x$, $y_2 = F_2 x$, such that F_1, F_2 form a Parseval frame $F = [F_1^T F_2^T]^T$. Since the rows of F are not restricted to be of unit norm, the frame bound does not indicate the redundancy. This unconventional choice of F is motivated by the resulting simplicity of the construction procedure. The vectors y_1, y_2 follow a multivariate Gaussian distribution and the encoding is made with a series of scalar quantizers using typical transform coding techniques (see Section 3.1.2) at rates R_1, R_2, respectively. Consequently, the corresponding indices I_1, I_2 are transmitted through the network. Let $R = R_1 + R_2$ be the total rate of the MDTC system.

At the receiver, the indices I_1, I_2 are decoded back to the correlated descriptions $\hat{y}_1$, $\hat{y}_2$. Then, according to the losses that occurred in the network the descriptions are fed to the appropriate decoder; the central decoder when both descriptions are received, side decoder 1 when only description 1 is received, etc. The output of the central decoder and the side decoders 1 and 2 is denoted by $\hat{x}_0$, $\hat{x}_1$, $\hat{x}_2$, respectively. The MDTC process is schematically presented in Figure 7.2.

Let $D_0 = E\{\| x - \hat{x}_0 \|_2^2\}$ be the distortion from the central decoder, $D_1 = E\{\| x - \hat{x}_1 \|_2^2\}$ and $D_2 = E\{\| x - \hat{x}_2 \|_2^2\}$ be the distortions from the side decoders 1 and 2 respectively. If we assume that each description is routed through an independent channel and that channels 1 and 2 have loss probabilities ρ_1 and ρ_2, respectively, then the total distortion is:

$$\begin{aligned} D_{tot} &= (1-\rho_1)(1-\rho_2)D_0 + \rho_2(1-\rho_1)D_1 + \\ &+ \rho_1(1-\rho_2)D_2 + \rho_1\rho_2 D_3, \end{aligned} \tag{7.4}$$

where $D_3 = \sum_{i=1}^{P} \sigma_i^2$ is the distortion when both descriptions are lost.

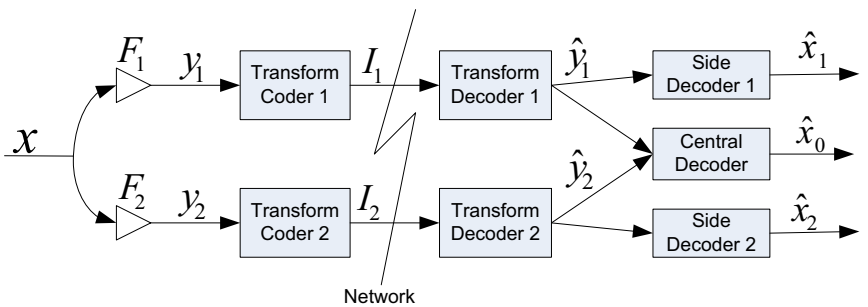


Figure 7.2 A schematic display of the proposed MDTC scheme.

Central Decoder

The central decoder receives both descriptions $\hat{y}_1$, $\hat{y}_2$, therefore it has the full quantized expansion $\hat{y} = [\hat{y}_1^T \hat{y}_2^T]^T$. The MMSE reconstruction can be computed using the pseudoinverse $F^{\#} = F^T$:

$$\hat{x}_0 = \begin{bmatrix} F_1^T & F_2^T \end{bmatrix} \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \end{bmatrix} = F_1^T \hat{y}_1 + F_2^T \hat{y}_2. \tag{7.5}$$

Better reconstructions can also be achieved, as it is shown in Section 7.1.4.

Side Decoders

When only one description is received, the lost description is *estimated* from the received description using the correlations introduced via the frame expansion. Note that depending on F, y_1 and y_2 have at most P non-zero dimensions, therefore they are potentially *incomplete* descriptions. Assume that description $\hat{y}_2$ is lost. The quantization cell defined by received description $\hat{y}_1$ may be unbounded in some directions and then no inverse transform can provide a reconstruction. Therefore, in order to have a reconstruction we must make an estimation. In Appendix B.1 we show that the optimal MSE reconstruction $\hat{x}_1$ is provided by:

$$\hat{x}_1 = (F_1^T + F_2^T \Sigma_{y_2 y_1} \Sigma_{y_1 y_1}^{-1}) \hat{y}_1, \tag{7.6}$$

where $\Sigma_{y_2 y_1}$ is the cross-covariance matrix between y_2 and y_1, $\Sigma_{y_1 y_1}$ is the covariance matrix of y_1, while both y_1, y_2 have zero means. The formula for the side decoder 2 is similar:

$$\hat{x}_2 = (F_2^T + F_1^T \Sigma_{y_1 y_2} \Sigma_{y_2 y_2}^{-1}) \hat{y}_2, \tag{7.7}$$

where $\Sigma_{y_1 y_2} = \Sigma_{y_2 y_1}^T$, and $\Sigma_{y_2 y_2}$ is the covariance matrix of y_2.

The covariance matrices can be computed using the frame expansion F to derive

the covariance matrix Σ_{yy} of y:

$$\begin{aligned}\Sigma_{yy} = F\Sigma_{xx}F^T = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} \Sigma_{xx} \begin{bmatrix} F_1^T & F_2^T \end{bmatrix}^T = \\ = \begin{bmatrix} F_1\Sigma_{xx}F_1^T & F_1\Sigma_{xx}F_2^T \\ F_2\Sigma_{xx}F_1^T & F_2\Sigma_{xx}F_2^T \end{bmatrix} = \begin{bmatrix} \Sigma_{y_1y_1} & \Sigma_{y_1y_2} \\ \Sigma_{y_1y_2} & \Sigma_{y_2y_2} \end{bmatrix}.\end{aligned} \tag{7.8}$$

Note that the inversion of $\Sigma_{y_1y_1}$ (resp. $\Sigma_{y_2y_2}$) is not possible when F_1 (resp. F_2) has rows/columns which are zero or are linearly dependent. The constraints can be satisfied by an appropriate construction of frame F. Such a construction is proposed in Section 7.1.3, where F_1 and F_2 are diagonal matrices.

7.1.3 MDTC with P degrees of freedom

This section describes a MDTC system with P degrees of freedom (DOF) $\vec{\phi} = \{\phi_i : i = 1, ..., P\}$. The main idea is to introduce the correlations in a scalar manner; i.e., component $x(1)$ is correlated with only one component $y_1(1)$ in the first description and one component $y_2(1)$ in the second description. The advantages of this system is that it has low complexity, limited storage requirements (only P degrees of freedom per channel condition), and that it simplifies the matrix inversions of $\Sigma_{y_1y_1}$ and $\Sigma_{y_2y_2}$.

For fixed source statistics Σ_{xx}, the total distortion D_{tot} in equation (7.4) is a function of the channel loss probabilities ρ_1 and ρ_2, the frame expansion parameters $\vec{\phi}$ and the description rates R_1 and R_2. It is convenient for practical purposes to have fixed rate descriptions R_1 and R_2. However, the determination of R_1 and R_2 is not a trivial task. An optimal set of rates R_1 and R_2 can found by minimizing D_{tot} for specific values of ρ_1, ρ_2 and $\vec{\phi}$. An MDC system should behave like a conventional coding system when there are no packet losses ($\rho_1 = \rho_2 = 0$). In that case, no correlations are needed between the two descriptions. The proposed MDTC system can be calibrated to have $\vec{\phi} = 0$ when there are no correlations between y_1 and y_2. The description rates R_1 and R_2 can then be allocated according to this operating point.

The section presents the construction of frame expansion F with P degrees of freedom, discusses the determination of the description rates R_1 and R_2 for the no-losses operating point and describes an algorithm that provides descriptions with predefined rates $\tilde{R}_1 \approx R_1$ and $\tilde{R}_2 \approx R_2$. The construction of frame expansion F with a single degree of freedom is then introduced and its training is presented.

Frame Expansion with P degrees of freedom

Let $\breve{x}_1 \in \Re^{P_1}$ and $\breve{x}_2 \in \Re^{P_2}$ be sub-vectors of $x = [\breve{x}_1^T \breve{x}_2^T]^T$ with $P_1 + P_2 = P$. We construct $F(\vec{\phi}) \in \Re^{2P \times P}$ as the concatenation of two diagonal matrices $F_1(\vec{\phi}) \in \Re^{P \times P}$, $F_2(\vec{\phi}) \in \Re^{P \times P}$ so that $F(\vec{\phi}) = [\ F_1^T(\vec{\phi})\ F_2^T(\vec{\phi})\ ]^T$. $F_1(\vec{\phi})$ and $F_2(\vec{\phi})$ are generated

according to the formulas:

$$F_1(\vec{\phi}) = \text{diag}\left(\underbrace{\cos(\phi_i)}_{i=\{1,...,P_1\}}, \quad \underbrace{\sin(\phi_i)}_{i=\{P_1+1,...,P\}} \right), \tag{7.9}$$

and

$$F_2(\vec{\phi}) = \text{diag}\left(\underbrace{-\sin(\phi_i)}_{i=\{1,...,P_1\}}, \quad \underbrace{\cos(\phi_i)}_{i=\{P_1+1,...,P\}} \right). \tag{7.10}$$

It can easily be verified that F is a Parseval frame: $F^T F = I$. Due to the symmetries of $\cos(\cdot)$ and $\sin(\cdot)$ it makes sense to restrict ϕ_i in $[0, \pi/4]$. As $\phi_i \in [0, \pi/4]$ increases, so do the correlations between the two components $y_1(i)$ and $y_2(i)$.

For this frame, the side decoder estimation matrices are greatly simplified:

$$F_1^T + F_2^T \Sigma_{y_2 y_1} \Sigma_{y_1 y_1}^{-1} = F_1 + F_2^2 F_1^{-1} = (F_1^2 + F_2^2) F_1^{-1} = F_1^{-1} \tag{7.11}$$

$$F_2^T + F_1^T \Sigma_{y_1 y_2} \Sigma_{y_2 y_2}^{-1} = F_2 + F_1^2 F_2^{-1} = (F_1^2 + F_2^2) F_2^{-1} = F_2^{-1}. \tag{7.12}$$

Which proves that the estimation matrices reverse the transform operations $y_1 = F_1 x$ and $y_2 = F_2 x$, respectively. The case where F_1 or F_2 has a zero on its diagonal is treated by inserting a zero to the corresponding dimension of x.

Intuitively, what F_1 does is to primarily preserve the information regarding $\breve{x}_1$ and secondarily insert a *backup copy* of the information regarding $\breve{x}_2$ at the expense of a higher distortion for $\breve{x}_1$. For example, when $\vec{\phi} = 0$, then $F_1 = \text{diag}(1, ..., 1, 0, ..., 0)$ and $F_2 = \text{diag}(0, ..., 0, 1, ..., 1)$. Therefore, y_1 holds $\breve{x}_1$ and y_2 holds $\breve{x}_2$: $y_1 = F_1 x = [\breve{x}_1^T \ 0, ..., 0]^T$ and $y_2 = F_2 x = [0, ..., 0 \ \breve{x}_2^T]^T$. Encoding y_1 in this case is the same as encoding $\breve{x}_1$. When $\vec{\phi} = \pi/4$, $F_1 = \text{diag}(\sqrt{2}/2, ..., \sqrt{2}/2, \sqrt{2}/2, ..., \sqrt{2}/2)$ and $F_2 = \text{diag}(-\sqrt{2}/2, ..., -\sqrt{2}/2, \sqrt{2}/2, ..., \sqrt{2}/2)$. In this case both descriptions are equivalent since both y_1 and y_2 hold the same information regarding x. Intermediate values of ϕ_i introduce intermediate correlations.

Description Rate Allocation

When there are no correlations, the optimal MDTC system should operate like a conventional transform coding system. The proposed MDTC system can be designed to satisfy this requirement if we set $R_1 = \sum_{i=1,...,P_1} r_i$ and $R_2 = \sum_{i=P_1+1,...,P} r_i$, where r_i are the rates of $x(i)$ for $i = 1, \ldots, P$. The rates r_i can be computed using the standard bit allocation algorithm for transform coding (see Section 3.1.2). It is a simple exercise to show that the two independent Gaussian encoders in Figure 7.2 will allocate the same rate r_i to the corresponding variable $x(i)$ they receive when $\vec{\phi} = 0$. Therefore, we expect excellent performance at zero loss rate.

Splitting x into two sub-vectors

It is sometimes convenient for practical purposes to predefine the desirable rates of the descriptions 1 and 2 to be $\tilde{R}_1$ and $\tilde{R}_2$, respectively. In that case, a "splitting"

algorithm is needed to distribute the variables $x(i)$ in two sub-vectors $\breve{x}_1$ and $\breve{x}_2$ so that $\tilde{R}_1 \approx R_1$ and $\tilde{R}_2 \approx R_2$.

Let r_i be the number of bits allocated to the scalar component $x(i)$ by the bit allocation algorithm, and $\tilde{R}_1, \tilde{R}_2$ the desirable rates allocated to the first description and the second description respectively. The following algorithm splits x into two sets $\breve{x}_1 \in \Re^{P_1}$ and $\breve{x}_2 \in \Re^{P_2}$, $P_1 + P_2 = P$:

a. Initialize $I_a = \{1, ..., P_1\}$, $I_b = \{P_1 + 1, ..., P\}$ as the sets of indices of the variables $x(i)$ that compose $\breve{x}_1 = \{x(i) : i \in I_a\}$ and $\breve{x}_2 = \{x(i) : i \in I_b\}$.

b. Let r_i be the rates of $x(i)$.

c. Set $R_1 = \sum_{i \in I_1} r_i$, $R_2 = \sum_{i \in I_2} r_i$, $dR = |\tilde{R}_1 - R_1| + |\tilde{R}_2 - R_2|$

d. While $dR > 0.25$ bits, swap every possible combination of two variables between the sets I_a and I_b and update I_a, I_b for the swaps that reduce dR. Break, if no swap reduces dR.

e. If $dR > 0.25$ bits, transfer one variable from the description with the higher rate to the description with the lower rate. Choose the variable with the lowest r_i inside the description with the higher rate.

f. Repeat steps (d), (e) until $dR \leq 0.25$ or a predefined number of iterations is reached.

Note that P_1 and P_2 are the number of variables in I_a and I_b respectively.

Frame Expansion with a single degree of freedom

The degrees of freedom for the construction of F can be further reduced to one, if we constrain all ϕ_i to take the same value $\phi_i = \phi$, $i = \{1, ..., P\}$. This is very tempting in terms of complexity and memory requirements. Also, we found that having a single DOF is as effective as having P DOF, for the typical MDC case of having equal rate descriptions and a simple network model of two symmetric independent channels with independent losses according to equal loss probabilities $\rho_1 = \rho_2 = \rho_{loss}$. This result reflects the symmetry of the network and the fact that the descriptions are balanced.

Training

The total distortion D_{tot} is optimized by performing a series of scalar optimization steps, varying one ϕ_i at a time. The fact that this is a scalar minimization over a fixed interval makes feasible a *data-driven* minimization of the distortion: Given the packet loss probabilities ρ_1, ρ_2, random samples of x are encoded and decoded using the proposed MDTC for N=4096 uniformly distributed values of $\phi_i \in [0, \pi/4]$, and the ϕ_i that minimizes the total distortion (7.4) is found. The encoder and the

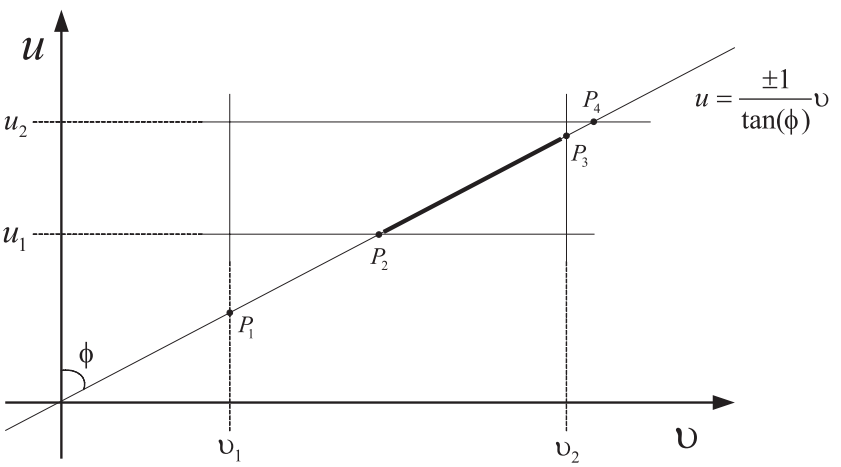


Figure 7.3 An example of Optimal Consistent Reconstruction. The normal samples are distributed on a line with angle ϕ to the u-axis. The integration is performed over the line segment (P_2, P_3) that lies inside the rectangular cell.

decoder share a small *lookup table* that stores one vector of parameter(s) $\vec{\phi}$ per network condition, for a predefined set of network conditions.

7.1.4 Optimal Consistent Reconstruction

In this section we will describe a method to have *Optimal Consistent Reconstruction* (OCR) for the MDTC scheme presented in Section 7.1.3. The method is based on the fact that the latter introduces the correlations in a scalar manner. Therefore the reconstruction of $\hat{x}_o$ when both descriptions are received can be expressed as a series of independent reconstructions; one for each set of correlated components. For example, $\hat{y}_1(i)$ and $\hat{y}_2(i)$ will be combined to reconstruct $\hat{x}_0(i)$. This greatly simplifies the geometry of the quantization cells and allows an efficient solution of the *optimal consistent reconstruction* problem.

Let u be a scalar component of one description and υ be the corresponding component of the other description. The image of F in the two dimensional plane is then a line of the form $u = \alpha\upsilon$, $\alpha = \pm 1/\tan(\phi)$, where ϕ is the rotation introduced by F.

The intersection of the line $u = \alpha\upsilon$ with the rectangular quantization cell is depicted in Figure 7.3. The line intersects the cell boundary. At least two of these intersections are inside the quantization cell. Let P_2 and P_3 be these intersections, as shown in Figure 7.3. A consistent reconstruction would then lie on the line segment (P_2, P_3). Let the coordinates of P_2 and P_3 with respect to the line be η_2 and η_3 respectively. Without loss of generality we can assume that $\eta_2 < \eta_3$.

The optimal consistent reconstruction is a point $\hat{\eta} \in (\eta_2, \eta_3)$. This point can eas-

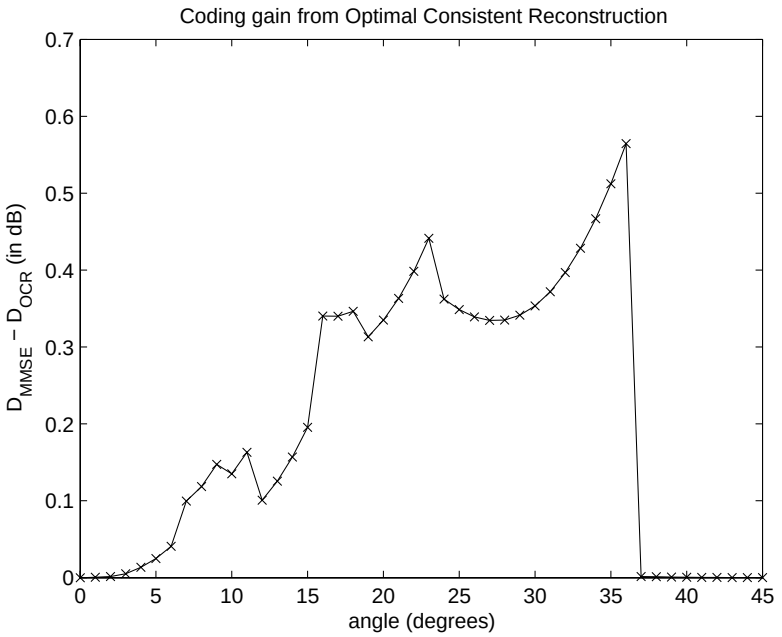


Figure 7.4 The benefit from using OCR reconstruction over MMSE reconstruction, in dB.

ily be found if we assume that the data in $u = \alpha v$ follow the Gaussian distribution $N(0, 1)$. In that case, according to equation (7.3) the optimal consistent reconstruction $\hat{\eta}$ for a squared error distortion measure is given by the formula:

$$\hat{\eta} = \frac{(2\pi)^{-0.5}}{F_g(\eta_3) - F_g(\eta_2)} \left(e^{-0.5\eta_2^2} - e^{-0.5\eta_3^2} \right), \tag{7.13}$$

where $F_g(\cdot)$ is the Gaussian cumulative density function.

An experiment was conducted to measure the benefit from using optimal consistent reconstruction when both descriptions are available. A large number (50.000) of eight dimensional Gaussian vectors $x \sim N(0, I_8)$ were generated, encoded and decoded at R=4 bits/component with MMSE reconstruction and OCR reconstruction, using the MDTC scheme with a single DOF. The distortion D_{MMSE} of the MMSE reconstruction and the distortion D_{OCR} of the OCR reconstruction were measured in dB, and their difference is plotted in Figure 7.4 for several angles $\phi \in [0, \pi/4]$. As expected, the gain $D_{MMSE} - D_{OCR}$ depends on the angle ϕ. Note that ϕ controls the bit assignment that affects the shape of the quantization cells. This explains the non-smooth evolution of the gain $D_{MMSE} - D_{OCR}$. The gain is significant in intermediate angles while the computational overhead is relatively low.

7.1.5 Results

Experiments were performed to measure the effectiveness of the proposed MDTC scheme. Two different network scenarios were tested: The first is the *Simple Symmetric Channel* scenario which refers to the case where the two descriptions are rooted through independent channels with independent losses according to the same loss probability $\rho_{loss} = \rho_1 = \rho_2$. The second is the *Hierarchical Coding* scenario, where description 1 always arrives ($\rho_1 = 0$) and description 2 may be lost with probability $\rho_{loss} = \rho_2$. We refer to this scenario as "hierarchical coding" because it can be seen as a formulation of the hierarchical coding problem within the MDC context.

The experiments were conducted using 50.000 random samples. For the *data-driven* minimization we used 10.000 samples. Optimal consistent reconstruction was used in the first scenario, while the second scenario was examined using MMSE reconstruction. All distortions in this section are presented in decibel for visualization purposes.

Simple Symmetric Channel

In this experiment we used the proposed MDTC with a single degree of freedom as stated in section 7.1.3. The optimized MDTC is compared to three different coding schemes:

- The *single packet* scheme that refers to typical single description scalar quantization of the Gaussian components with R bits.
- The *double packet* scheme, which is actually a repetition code of a half rate $R/2$ scalar encoding of the Gaussian components.
- The MDSQ$_{TC}$ scheme presented in Section 6.2.

When there are no losses ($\rho_{loss} = 0$), an optimal MDTC (or MDC) scheme is expected to behave like single description coding. Under severe losses ($\rho_{loss} = 0.5$) the optimal MDTC (or MDC) should operate better than a repetition scheme. Therefore the first two schemes indicate the performance boundaries for both MDTC and MDSQ$_{TC}$.

The third scheme uses the method provided by Vaishampayan [74] to train a scalar MDSQ encoder for the $N(0, 1)$ case. The MDSQ codebooks were trained using methods from [74] and [140] and validated according to the theoretical central/side distortion tradeoffs provided in [135].

The experiment is conducted using a multivariate Gaussian taken from a GMM (Gaussian Mixture Model) trained with 10-dimensional Line Spectrum Frequencies (LSF) spectral envelopes of narrowband 0-4 kHz speech. The variances of the encoded components correspond to the eigenvalues of the covariance matrix of the multivariate Gaussian. The results are depicted in Figures 7.5, 7.6 and 7.7. Figure 7.5 shows the total distortion for all the examined schemes. As expected, MDTC outperforms the single packet and the double packet schemes. In contrast, MDSQ$_{TC}$ is worse than

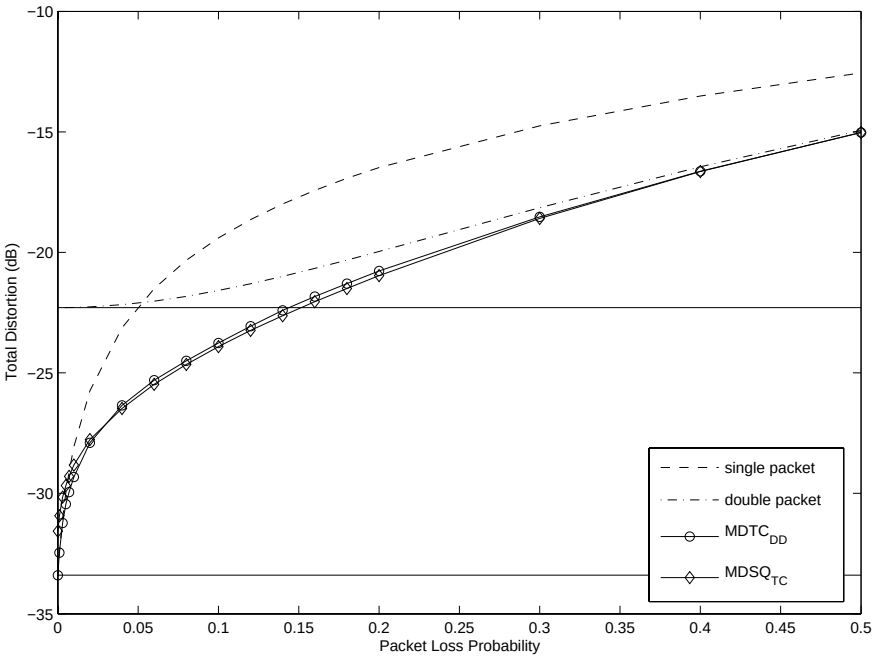


Figure 7.5 A comparison between the total distortions provided by the single packet scheme, the double packet scheme, MDTC with a single DOF (MDTC) and MDSQ$_{TC}$, for the LSF source.

the single packet scheme at packet loss probabilities near $\rho = 0$. According to the discussion in Section 6.2, this can be attributed to the fact that the variances of the components differ enough to provide a penalty of 2 dB to the central distortion at $\rho = 0$. However, at higher loss probabilities, the MDSQ$_{TC}$ scheme outperforms the MDTC scheme. Further insight about the behavior of the central and side descriptions is provided in Figure 7.6 where the central distortion and the side distortions for MDTC and MDSQ$_{TC}$ are depicted for a range of packet loss probabilities.

Figure 7.7 provides a better understanding of the behavior of the two schemes by showing the central distortion and side distortion tradeoffs. The dashed line corresponds to the optimal distortion tradeoff of a MDSQ$_{TC}$ scheme [141], the dashed-dotted line corresponds to the best central distortion achievable by MDSQ$_{TC}$ and the dotted line to the best central distortion achievable by MDTC. The latter two bounds show the distortion penalty of 2 dB for MDSQ$_{TC}$. Furthermore, we can clearly see that the bound restricts the distortion tradeoffs for MDSQ$_{TC}$ to be away from the optimal theoretical behavior of the MDSQ$_{TC}$ (achieved only when the variances of the components are equal). However, at intermediate and higher loss probabilities, MDSQ$_{TC}$ benefits from the increased number of the degrees of freedom and outperforms MDTC, progressively, up to 1 dB.

From a practical point of view, the two schemes are actually not competitive but complementary; a combination of MDTC (at lower loss probabilities) and MDSQ$_{TC}$

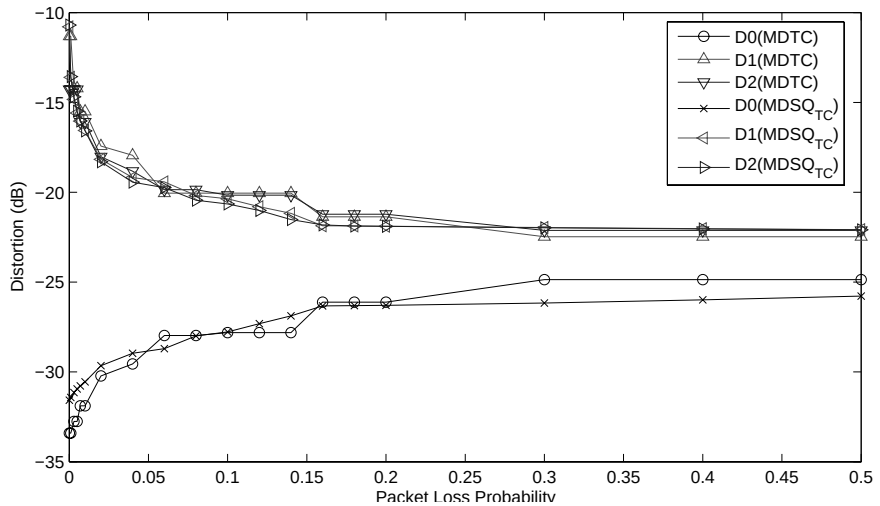


Figure 7.6 A comparison between the central distortion D_0 and the side distortions D_1, D_2 for the MDTC and MDSQ$_{TC}$ cases presented in Figure 7.5, for the LSF source.

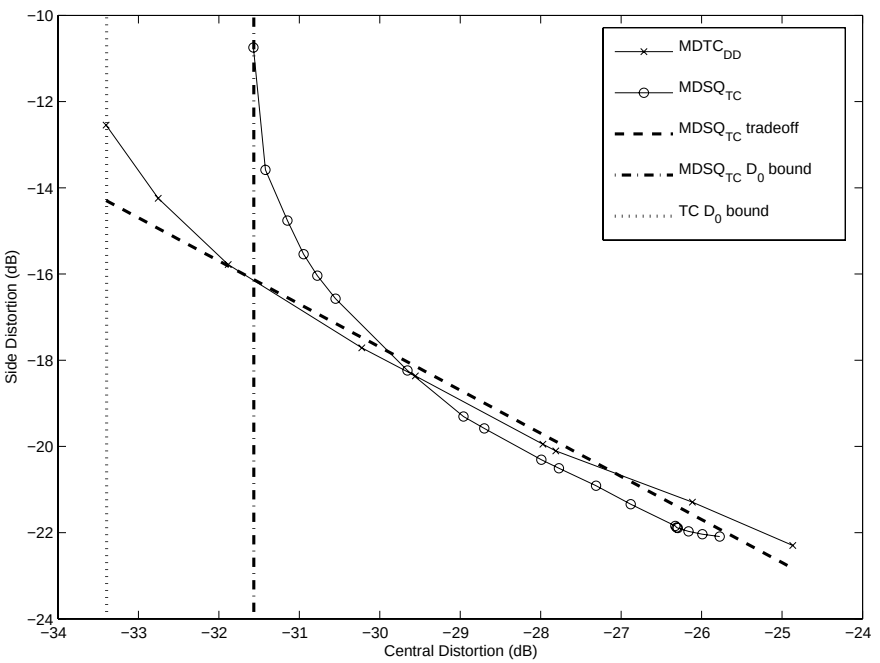


Figure 7.7 Central Distortion/Side Distortion tradeoffs for MDSQ$_{TC}$ and MDTC for the LSF source.

(at higher loss probabilities) provides a Multiple Description Coding system that performs well at all cases. This observation is experimentally validated for many real world sources typically used in speech coding, like LSF and Cepstrum Coefficients computed for narrowband and wideband spectra. Furthermore, such a combined system will have much lower memory requirements than the MDSQ_{TC}, because the central description codebooks of MDSQ are sparsely populated at higher loss probabilities.

The comparison made in this section does not take into account the fact that due to storage limitations in many practical applications, like GMM-based MDSQ_{TC} [151], the storage of all MDSQ codebooks required for good performance may become too expensive. Therefore, a limited number of codebooks must be used, i.e. using integer bit allocation schemes [151]. This introduces a loss in terms of distortion, and this loss partially compensates the distortion gap between MDSQ and MDTC at higher loss probabilities. Furthermore, the MDSQ_{TC} scheme is not practically scalable to complicated channel models, where the channel loss probabilities ρ_1,ρ_2 are different. In contrast, the MDTC scheme has a number of significant computational advantages like low complexity, low storage requirements and rate scalability (via companding [89]).

The two multiple description schemes that are compared in this section are quite different in nature. However, a proper theoretical comparison between the two methods is beyond the scope of this thesis. The proposed MDTC scheme constrains the partitions of the descriptions and ties the partitioning to a single degree of freedom. This imposes structure onto the quantizers and enables the scalability of the proposed MDTC scheme. The MDSQ_{TC} scheme, on the other hand, does not constrain the partitions, at the cost of decreased scalability and the requirement of a large number of parameters to model the relationship between the descriptions.

Finally, we argue that the worse performance of the proposed MDTC scheme at higher loss probabilities is associated with the nature of transform coding. A similar observation is made in [145] (pg. 2211) when comparing their version of entropy-constrained multiple description transform coding with MDSQ. In [155] it is shown that the MSE is increased by a factor of 2 (that is ~ 3 dB) when half of the components are lost, and the lost components are pairwise orthogonal. A solution to this problem is provided in [153] for the case of entropy-constrained transform-based coding of Gaussians. Therefore, the behavior of the proposed scheme in high redundancies is in accordance to the observations made for other transform coding schemes for the entropy-constrained case.

Hierarchical Coding

In this experiment, the first description is always received, while the second description may be lost with probability ρ_{loss}. This is equivalent to the "hierarchical coding" problem [55]. Following the notation of the previous subsection the MDTC is the measured distortion from the data-driven minimization. This example, however,

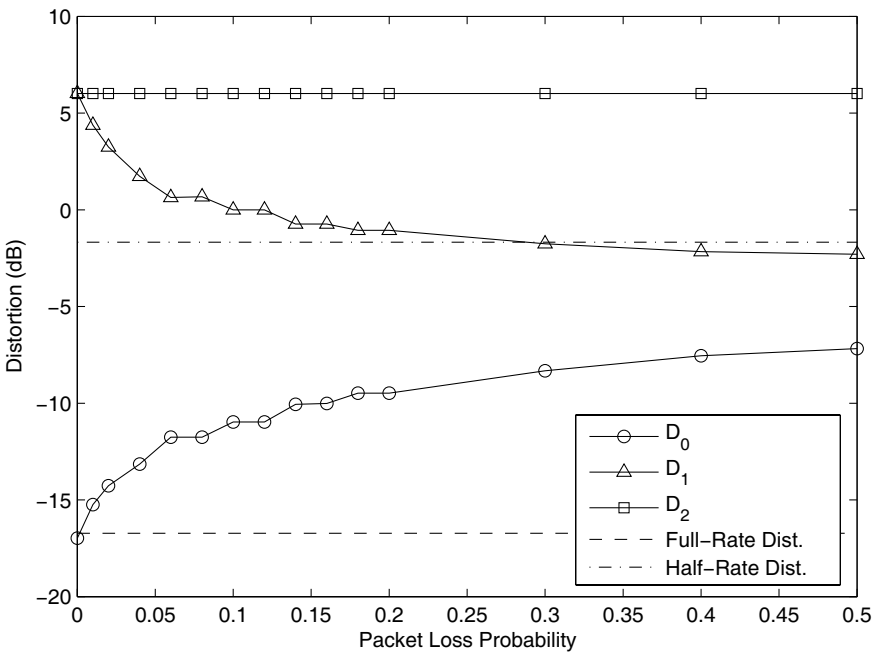


Figure 7.8 The central distortion D_0 and side distortions D_1, D_2 for the Hierarchical Coding experiment. The upper and lower horizontal lines correspond to the half rate and full rate distortion levels $D(R/2)$ and $D(R)$, respectively.

.

uses $|I| = 8$ degrees of freedom to compensate the need to have different rotations in different Gaussian components. Figure 7.8 shows the central distortion D_0 and the side distortions D_1, D_2 for the MDTC case. As expected, the distortion D_2 of description 2 is constantly high, at the benefit of a much better central reconstruction D_0, while D_1 converges to the half rate bound. This readily reflects the fact that description 2 will never be received alone.

7.1.6 Discussion

A novel scheme for MDTC of Gaussian variables was proposed. The scheme is designed for resolution-constrained MDTC, in contrast to other MDTC schemes presented in the literature that focus on entropy-constrained MDTC. The proposed MDTC scheme is based on structured Parseval frame expansions. Two special cases are presented and evaluated. For these cases, an optimal consistent reconstruction algorithm is provided. The optimization procedure directly minimizes the expected distortion for a given data-set. The result is a practical MDTC scheme that is computationally efficient and scalable with minimal memory requirements, both for training and encoding/decoding.

Compared to "conventional" multiple description transform coding schemes that are based on scalar MDSQ codebooks (MDSQ_{TC}), our method is fast, scalable and practically efficient (via companding [89]) for coding at high rates and complicated network models. However, this scalability comes at the cost of reduced performance at higher loss probabilities. In practice, this performance loss is partially compensated by the fact that in a practical $MDSQ_{TM}$ implementation the number of MDSQ codebooks and index-assignment tables is restricted by cost. On the other hand, the proposed MDTC scheme outperforms MDSQ_{TC} at low loss probabilities. This can be attributed to the bit-allocation method used in MDSQ_{TC} that bounds the central distortion, as it is shown in Section 6.2. The resulting distortion penalty for MDSQ_{TC} is considerable for real world sources (like LSF spectral envelopes of speech). Our observations motivate a combined MDC scheme where MDTC is used at lower loss probabilities and MDSQ_{TC} at higher loss probabilities.

The reduced performance of the proposed method at higher redundancies can be attributed to the nature of transform coding with multiple descriptions, and it is also observed by many other researchers [145], [153], [145]. The simplicity, scalability and computational advantages of MDTC come not without a cost.

This work can be extended in many ways. For example, an extension can be made to Generalized MDTC, where the data is encoded in more than two correlated descriptions. Layered coding approaches can be used to improve the performance in higher redundancies, as in [153]. Finally, the quantization of y_1 and y_2 can be made jointly with respect to the MDC distortion measure.

7.2 GMM-based MDTC

This section extends the basic MDTC scheme, introduced in Section 7.1, to a novel MDC quantizer suitable for GMM sources. Section 7.2.1 presents an overview of the proposed scheme (GMM-MDTC). Section 7.2.2 describes the bit-allocation procedure for the proposed quantizer. The training and the complexity of GMM-MDTC is discussed in Section 7.2.3. The proposed scheme is experimentally evaluated in Section 7.2.4.

7.2.1 Overview

Let $x \in \mathbb{R}^P$ follow a GMM distribution with M Gaussians. The basic scheme is depicted in Figure 7.9. Data x is encoded with each of the M MDTC Gaussian encoders giving M candidate sets of codevectors $\{\hat{x}_{0,m}, \hat{x}_{1,m}, \hat{x}_{2,m} : m = 1, ..., M\}$. Let m' be the index of the "best" candidate set, corresponding to the m'-th Gaussian of the GMM. The m'-th candidate set that minimizes the MDC distance:

$$\begin{aligned} m' = \arg\min_{m} \{&(1-p_1)(1-p_2) \parallel x - \hat{x}_{0,m} \parallel_2^2 + \rho_2(1-\rho_1) \parallel x - \hat{x}_{1,m} \parallel_2^2 + \\ &+\rho_1(1-\rho_2) \parallel x - \hat{x}_{2,m} \parallel_2^2\} \end{aligned} \tag{7.14}$$

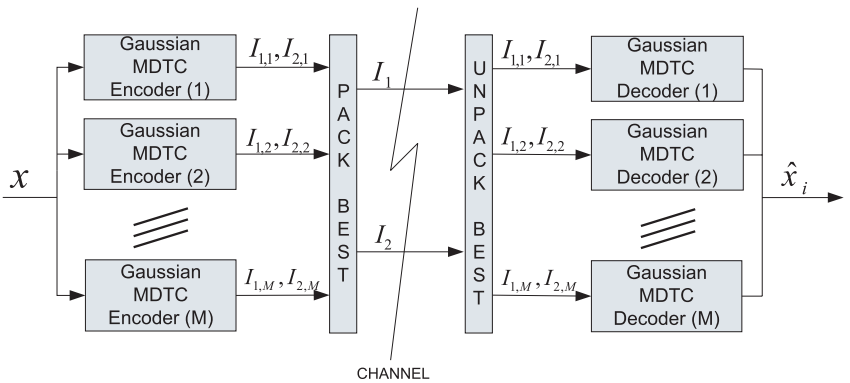


Figure 7.9 Proposed system for GMM-based MDTC.

is selected for transmission. The index m' together with the indices $I_{1,m'}, I_{2,m'}$ provided by the m'-th Gaussian MDTC encoder are packed and transmitted through the network in two descriptions $I_1 = \{m', I_{1,m'}\}$ and $I_2 = \{m', I_{2,m'}\}$. Let $R_{1,m}$, $R_{2,m}$ be the rates of $I_{1,m}$, $I_{2,m}$, respectively. Then, the rates of I_1, I_2 are:

$$R_1 = \log_2\left(\sum_{m=1}^{M} 2^{R_{1,m}}\right), \quad R_2 = \log_2\left(\sum_{m=1}^{M} 2^{R_{2,m}}\right) \tag{7.15}$$

and the index m' of the "best" Gaussian encoder can be represented by the location of the values of I_1, I_2 inside the (integer) intervals $[1, 2^{R_1}]$ and $[1, 2^{R_2}]$, respectively.

At the GMM-MDTC decoder, the m'-th Gaussian MDTC decoder is used to get one of the available reconstructions $\hat{x}_0$, $\hat{x}_1$, $\hat{x}_2$, depending on whether one or both descriptions are received. Therefore, the GMM-based MDTC scheme is a multi-coder scheme where the data is encoded with many Gaussian MDTC coders and the "best" encoding is selected for the transmission together with the corresponding information regarding the index of the coder. This treatment is typical in GMM-based coding, which is actually a generalization of KLT (Karhunen-Loeve Transform) coding for GMM, and details can be found in Section 3.2 and in [88], [151].

The m-th Gaussian MDTC encoder assumes that the statistics of x follow the statistics of the m-th Gaussian component of the GMM, namely $N(\mu_m, \Sigma_{xx,m})$. Vector x is translated and rotated in order to get a zero mean vector x'_m with diagonal covariance matrix:

$$x'_m = V_m^T(x - \mu_m) \tag{7.16}$$

where V_m is the eigenvector matrix taken from the eigenvalue decomposition of the covariance matrix $\Sigma_{xx,m} = V_m \Lambda_m V_m^T$. The variances of $x'_m(i)$ are the corresponding eigenvalues $\Lambda_m(i,i)$, $i = 1, ..., P$. From these variances we compute the rate required for each of the $x'_m(i)$ according to the well-known bit-allocation algorithm [55] (see Section 3.1.2). The eigenvectors (columns of V_m) are ordered to ensure that x'_m is

composed of two subvectors $x'_{1,m}$ and $x'_{2,m}$ in such a way that $x'_{1,m} \in \Re^{P_{1,m}}$ requires $R_{1,m}$ bits and $x'_{2,m} \in \Re^{P_{2,m}}$ requires $R_{2,m}$ bits. Note that the dimensions $P_{1,m}$ and $P_{2,m}$ may be different for each of the M Gaussian components. The rates $R_{1,m}$ and $R_{2,m}$ depend only on the variances $\Lambda_m(i,i)$ of $x'_m(i)$ and the total rates R_1, R_2 and they are computed by the bit allocation procedure described in Section 7.2.2. The frame expansion for the m-th Gaussian component is $F_m = [F^T_{1,m} F^T_{2,m}]^T$ and the corresponding descriptions are $y_{1,m} = F_{1,m} x'_m$ and $y_{2,m} = F_{2,m} x'_m$. The matrices $F_{1,m}$ and $F_{2,m}$ are provided by the following equations:

$$F_{1,m}(\vec{\phi}) = \mathrm{diag}\left(\underbrace{\cos(\phi_{i,m})}_{i=\{1,\ldots,P_{1,m}\}}, \underbrace{\sin(\phi_{i,m})}_{i=\{P_{1,m}+1,\ldots,P\}} \right), \tag{7.17}$$

and

$$F_{2,m}(\vec{\phi}) = \mathrm{diag}\left(\underbrace{-\sin(\phi_{i.m})}_{i=\{1,\ldots,P_{1,m}\}}, \underbrace{\cos(\phi_{i,m})}_{i=\{P_{1,m}+1,\ldots,P\}} \right) \tag{7.18}$$

where $\phi_{i,m}$ is the angle that corresponds to the i-th dimension of $y_{1,m}$ and $y_{2.m}$. The vectors $y_{1,m}$ and $y_{2,m}$ are then encoded into two indices (*descriptions*) $I_{1,m}$ and $I_{2,m}$ at rates $R_{1,m}$ and $R_{2,m}$ respectively according to the MDTC scheme for Gaussians presented in Section 7.1.

7.2.2 Bit Allocation

This section describes the bit allocation procedure for the proposed GMM-based MDTC scheme. We assume that R_1 and R_2 are predefined design parameters. The bit allocation procedure sets the rates $R_{1,m}$, $R_{2,m}$ for each of the Gaussian MDTC encoders $m = 1, ..., M$. Assuming that the Gaussians of the GMM are well separated, the total distortion of the GMM-based MDTC encoding is:

$$D_{tot} = \sum_{m=1}^{M} \alpha_m D_{tot,m} \tag{7.19}$$

where α_m are the weights of each Gaussian and $D_{tot,m}$ is the average total distortion associated with the m-th Gaussian component (see equation 7.4). For fixed source statistics, the total distortion D_{tot} is a function of $\vec{\phi}_m$, and ρ_1, ρ_2. Since we do not have a closed-form formula for the total distortion, an analytical minimization of D_{tot} for the optimal $R_{1,m}$ and $R_{2,m}$ is not feasible. Therefore, we will have to resort to a *strategy* that will provide us with a solution that is optimal for a single predefined operation point; the case when no packets are lost: $\rho_1 = \rho_2 = 0$. In that case, the total distortion D'_{tot} is equal to the distortion of the central description:

$$D'_{tot} = D_{tot}|_{p_1=p_2=0} = \sum_{m=1}^{M} \alpha_m \left(\acute{D}^{(1)}_{0,m} + \acute{D}^{(2)}_{0,m} \right). \tag{7.20}$$

When $\rho_1 = \rho_2 = 0$, the best performance is obtained if the descriptions are uncorrelated, thus when $\vec{\phi}_m = 0$, $m = \{1, ..., M\}$. In the latter case, it is easy to observe that the central distortions $\acute{D}_{0,m}^{(1)}$ and $\acute{D}_{0,m}^{(2)}$ for the first and the second description, respectively, are provided by:

$$\begin{aligned} \acute{D}_{0,m}^{(1)} &= Q_c P_{1,m} c_{1,m} 2^{\frac{-2R_{1,m}}{P_{1,m}}}, \\ \acute{D}_{0,m}^{(2)} &= Q_c P_{2,m} c_{2,m} 2^{\frac{-2R_{2,m}}{P_{2,m}}}, \end{aligned} \tag{7.21}$$

where $Q_c = \frac{\sqrt{3}\pi}{2}$ is the quantization constant for scalar resolution-constrained coding of Gaussians [55], and $c_{1,m}$, $c_{2,m}$ are the geometric means of the corresponding variances:

$$c_{1,m} = \left(\prod_{i=1}^{P_{1,m}} \Lambda_m(i,i) \right)^{\frac{1}{P_{1,m}}}, \quad c_{2,m} = \left(\prod_{i=P_{1,m}+1}^{P} \Lambda_m(i,i) \right)^{\frac{1}{P_{2,m}}} \tag{7.22}$$

The minimization of D'_{tot} with respect to the rate constrains stated in equations (7.15) can be made using typical Lagrangian methods, and results to the following bit-allocation formula:

$$\begin{aligned} R_{1,m} &= R_1 + \log_2 \left(\frac{(\alpha_m c_{1,m})^{\frac{P_{1,m}}{P_{1,m}+2}}}{\sum_{m=1}^{M} (\alpha_m c_{1,m})^{\frac{P_{1,m}}{P_{1,m}+2}}} \right), \\ R_{2,m} &= R_2 + \log_2 \left(\frac{(\alpha_m c_{2,m})^{\frac{P_{2,m}}{P_{2,m}+2}}}{\sum_{m=1}^{M} (\alpha_m c_{2,m})^{\frac{P_{2,m}}{P_{2,m}+2}}} \right), \end{aligned} \tag{7.23}$$

which is equal to the GMM "cluster" allocation formula for resolution-constrained coding, in [88]. Note, however, that these rates depend on the splitting of $x'_m = [\ \acute{x}_{1,m}^T\ \acute{x}_{2,m}^T\ ]^T$ in two subvectors $x'_{1,m}$ and $x'_{2,m}$. This ordering defines the dimensions $P_{1,m}$ and $P_{2,m}$ and it is implemented with an appropriate ordering of the eigenvectors (columns of V_m) of $\Sigma_{xx,m}$.

The splitting of x'_m in $\acute{x}_{1,m}^T$ and $\acute{x}_{2,m}^T$ is performed according to the splitting algorithm in Section 7.1.3 using the following rates:

$$R'_{1,m} = \frac{R_1}{R_1 + R_2} R'_m, \quad R'_{2,m} = \frac{R_2}{R_1 + R_2} R'_m, \tag{7.24}$$

where R'_m are the GMM "cluster" bit allocation rates (according to equation (3.21)) when the GMM is encoded with a total rate $R = R_1 + R_2$. Concluding, the bit-allocation algorithm is the following:

a. Compute R'_m according the "cluster" bit allocation formula in (3.21), using a total rate $R = R_1 + R_2$. Then use equation (7.24) to compute $R'_{1,m}$ and $R'_{2,m}$.

b. Order the eigenvectors in V_m (and the corresponding eigenvalues in Λ_m) so that the first $P_{1,m}$ parameters of x'_m are $\acute{x}^T_{1,m}$ and the last $P_{2,m}$ parameters of x'_m are $\acute{x}^T_{2,m}$, requiring rates $R'_{1,m}$ and $R'_{2,m}$, respectively.

c. Compute the rates $R_{1,m}$ and $R_{2,m}$ using equation (7.23).

A nice property of the presented bit allocation algorithm is that it makes the proposed GMM-based MDTC scheme to operate like a conventional GMM-based encoder when $\vec{\phi}_m = 0$ or equivalently, when there are no packet losses.

7.2.3 Training and Complexity

The proposed GMM-based MDTC scheme requires the determination of M angle vectors $\vec{\phi}_m$ for a specific loss probability set (ρ_1, ρ_2). In practice, we can quantize the set (ρ_1, ρ_2) to a predefined set of K *channel configurations* and compute offline a set of optimal angle vectors $\vec{\phi}_m$, $m = 1, ..., M$ for each of these configurations. This strategy will make GMM-MDTC a practical and viable solution with reasonable storage requirements ($K \times M \times P$ angles). For specific loss probabilities ρ_1 and ρ_2 the training of the proposed scheme is made by a series of M separate trainings, one for each of the M MDTC Gaussian encoders. This treatment is justified only when the Gaussian components of the GMM are well separated and it is a common practice in GMM-based coding [88].

The typical network model used for MDC consists of two simple symmetric channels with equal loss probabilities $p = p_1 = p_2$. For this model and for balanced equal-rate descriptions $R_1 = R_2$ an interesting observation can be made: the descriptions must be equally correlated. Note that description 1 holds a quantized version of $x'_{1,m}$ and a lower fidelity version of $x'_{2,m}$ and vice versa. The two descriptions are correlated via the lower fidelity subvectors which hold information regarding the other description. The symmetric network model states that no description is more important than the other. Since the descriptions are balanced, an equal amount of correlation should be introduced via frame F_m. Therefore, it is reasonable to constrain $\phi_{i,m} = \varphi_m$, for all $i = 1, ..., P$ and parameterize each angle vector $\vec{\phi}_m$ with a single parameter φ_m. In this case, the training procedure for each MDTC Gaussian encoder is a scalar minimization in interval $[0, \pi/4]$. Furthermore, the storage requirements for K channel configurations are considerably reduced to $K \times M$ angle parameters.

The complexity of using the proposed GMM-based MDTC scheme is very low and comparable to the complexity of the typical GMM-based quantization [88]. The storage requirements for the precomputed angles are also very low allowing the usage of the proposed scheme under more complicated network models. Furthermore, high rates are achievable if companding is used for the quantization of the scalar variables $y_1(i)$, $y_2(i)$ in Section 7.1. This is a clear computational advantage over GMM-MDSQ$_{TC}$ because the storage requirements for the latter increase rapidly with the complexity of the network model and the encoding rates. Furthermore, with an appropriate construction of F, the proposed method can directly be generalized to

MDTC with more than two descriptions using relatively minor additional complexity and storage requirements.

7.2.4 Experiments and Results

The proposed method is evaluated on the context of MDC of RCC cepstral envelopes derived from narrowband speech. The cepstral envelopes are encoded at a total rate of $R = 60$ bits. The basic hypothesis for the experiments is that the network consists of two symmetric channels with equal loss probabilities $\rho = \rho_1 = \rho_2$ and that the descriptions have equal rates $R_1 = R_2 = 30$ bits. The proposed method (**GMM-MDTC**) is compared to four different schemes:

a. **GMM-SD**: A *single description scheme* with one full rate (60 bit) description, using GMM-based quantization [88].

b. **GMM-SD**: A *double description (repetition) scheme* with two equal half rate (30 bit) descriptions, using GMM-based quantization [88].

c. **GMM-MDSQ$_{\mathbf{TC}}$**: The GMM-based MDC scheme presented in [151], with integer-level side description codebooks.

d. **GMM-MDSQ$_{\mathbf{TC}}$ (ibit)**: The GMM-based MDC scheme presented in [151], with integer-bit side description codebooks.

The GMM-SD scheme and the GMM-DD scheme are used to indicate the performance of single description codes and repetition codes in the context of an erasure channel. The latter two schemes are chosen as "transform coding" alternatives to the proposed GMM-MDTC scheme; competitive, in terms of complexity and performance. The GMM-MDSQ$_{TC}$ (ibit) scheme uses integer-bit side codebooks, following the implementation in [151]. However, this introduces a loss of performance because it constrains the rate of each encoded scalar gaussian of the central description to be allocated in steps of 2 bits. Therefore, the latter method is also evaluated in GMM-MDSQ$_{TC}$ using side codebooks with integer-level sizes. The GMM-MDSQ$_{TC}$ scheme requires a large number of codebooks for each description loss probability ρ. For example, the cepstral envelope source needs 16 MDSQ sets of codebooks when encoded with the GMM-MDSQ$_{TC}$ scheme, compared to 4 MDSQ sets of codebooks when encoded with the GMM-MDSQ$_{TC}$ (ibit) scheme. Note that each set of codebooks consists of one central description codebook and two side description codebooks for each channel loss probability ρ.

The experiments were conducted using 400.000 samples from the training set of TIMIT database to estimate the GMM of the cepstral envelope (with $M = 16$ Gaussians) and 100.000 samples from the testing set of TIMIT for testing. The test-set is the same with the one used in Section 3.3. The 20-dimensional cepstral envelopes were derived with a least squares fit of the harmonic amplitudes at the log-domain, according to Section 2.4.1. For the GMM-MDTC scheme, we used MDTC with a

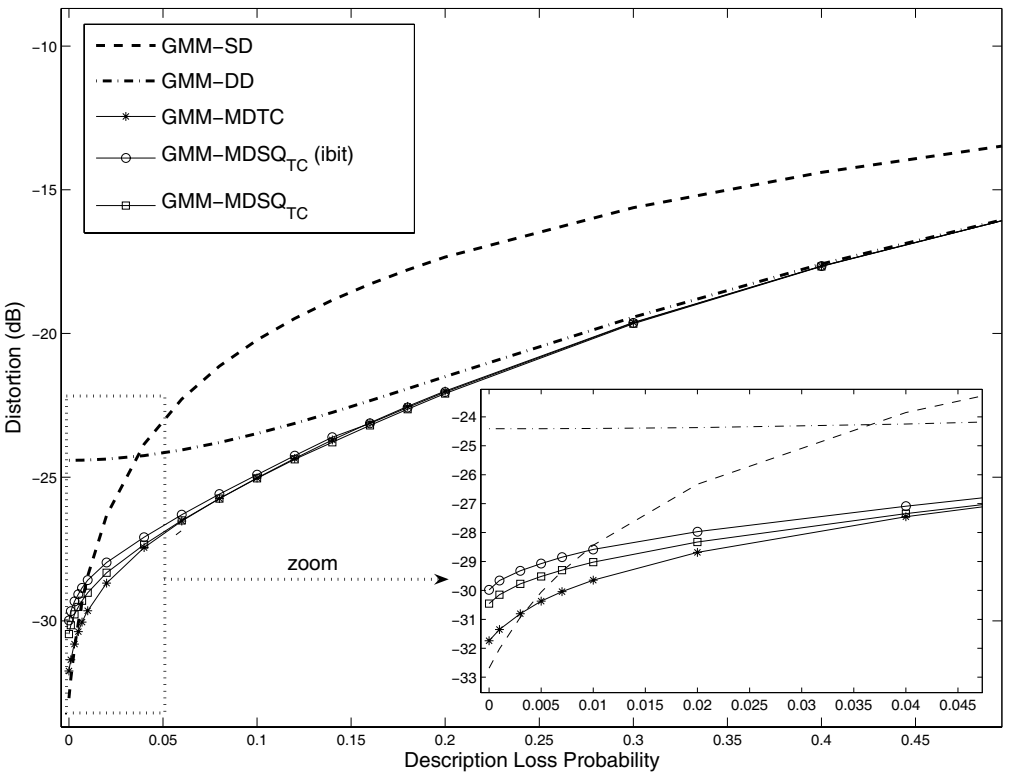


Figure 7.10 Total distortion for several loss probabilities.

single degree of freedom per Gaussian component. Thus, the memory requirements of the proposed method are M parameters per channel condition. The performance of the four schemes is measured in terms of mean square error and expressed in decibel for visualization purposes.

Figure 7.10 shows the total distortion for the four tested encodings. The total distortion for GMM-SD is provided by equation:

$$D_{tot,SD} = (1-\rho)D_{0,SD} + \rho D_3, \tag{7.25}$$

while the total distortion for GMM-DD is provided by equation:

$$D_{tot,DD} = (1-\rho^2)D_{0,DD} + \rho^2 D_3, \tag{7.26}$$

where ρ is the description loss probability and D_3 is the distortion when both descriptions are lost. The total distortion of the three MDC methods is computed by equation:

$$D_{tot,MDC} = (1-\rho)^2 D_0 + \rho(1-\rho)(D_1 + D_2) + \rho^2 D_3. \tag{7.27}$$

Note that the above formulas do not take into account the beneficial effect of the packet loss concealment algorithm to the distortion. However, this does not effect

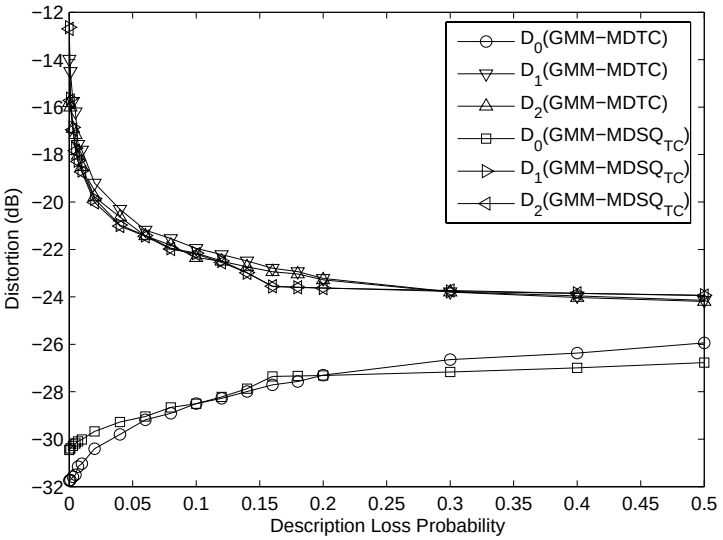


Figure 7.11 Central distortion D_0 and side distortions D_1, D_2 for several loss probabilities. Note that the side distortions coincide, reflecting the symmetry of the descriptions.

the insight that can be gained regarding the performance of the four schemes when packets are lost.

As it is shown in Figure 7.10, all MDC schemes outperform the double description scheme in all loss probabilities ρ, and the single description scheme at $\rho > 0.01$. Note that the single description scheme outperforms the three MDC schemes at $\rho = 0$, due to the fact that the MDC schemes encode the index of the Gaussian component twice (once to each description) instead of once. The proposed GMM-MDTC scheme outperforms the other MDC schemes in lower loss probabilities, while it converges to more or less the same performance in higher loss probabilities. Furthermore, Figure 7.10 indicates the penalty from using integer-bit side codebooks instead of integer-level side codebooks. The penalty is more evident at lower loss probabilities.

A further insight into the results of the experiment is provided by Figure 7.11, where the central distortion D_0 and the side distortions D_1, D_2 for GMM-MDTC and GMM-MDSQ$_{TC}$ are plotted. As the loss probability increases, the central reconstruction becomes worst and the side reconstructions become better to compensate the probable loss. It can be clearly seen that the proposed method provides much lower central distortion at lower loss probabilities. This is can be attributed to the bit-allocation procedure of MDSQ$_{TC}$ [141], [74], as it is shown in Section 6.2. In contrast, the GMM-MDSQ$_{TC}$ scheme outperforms the GMM-MDTC scheme at higher loss probabilities ρ. This is consistent with the findings in Section 6.2 where it is shown that MDSQ$_{TC}$ outperforms MDTC at higher loss probabilities, primarily by taking advantage of the high degree of freedom non-parametric nature of the MDSQ codebook structure. The MDTC scheme introduces the dependencies between the

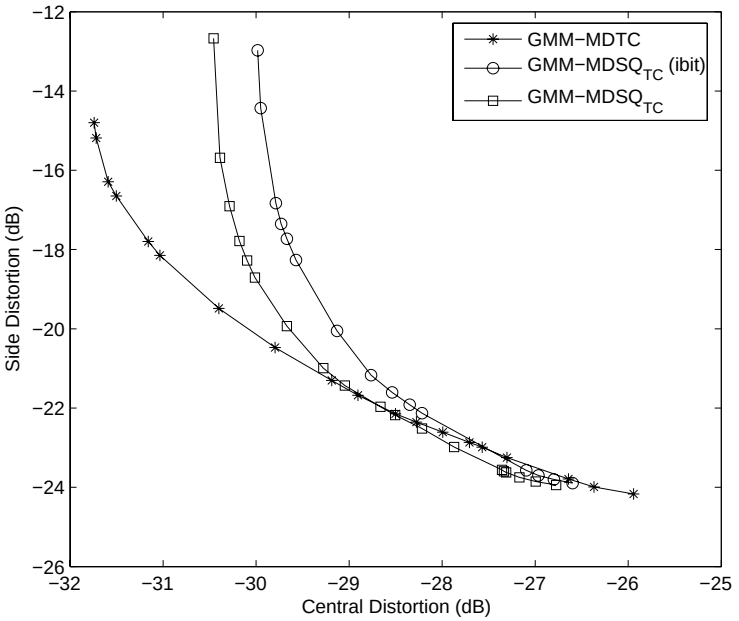
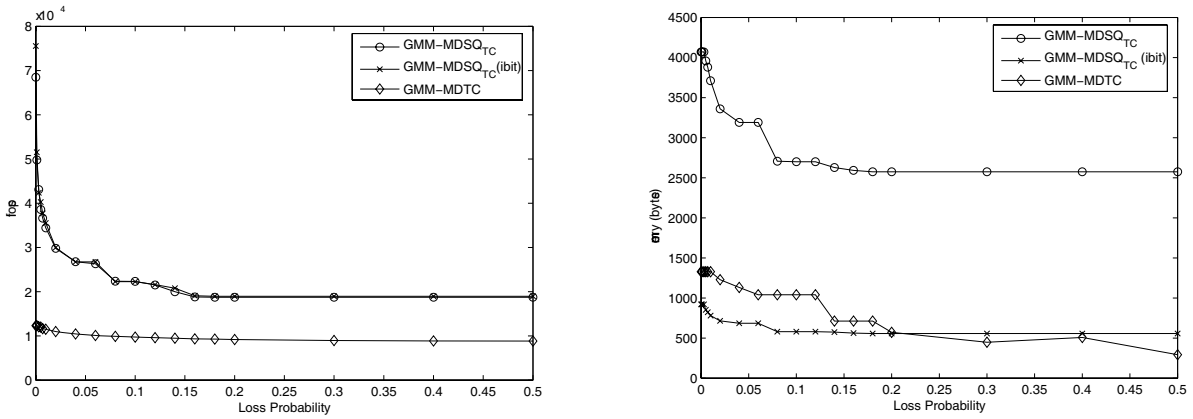


Figure 7.12 Tradeoff between the central distortion D_0 and side distortions D_1, D_2.

descriptions in a structured manner (via the Parseval frame expansion F), for the benefit of scalability and reduced complexity, but also at the cost of reduced performance when increased dependencies are needed.

A closer look, however, reveals that the performance reduction at higher redundancies is not as high as Figure 7.11 implies. Figure 7.12 shows the tradeoff between the central distortion and the side distortions for the three evaluated MDC schemes. At higher redundancies, GMM-MDSQ$_{TC}$ is only slightly better than GMM-MDTC because the central/side distortion tradeoffs offered by GMM-MDSQ$_{TC}$ are closely followed by the GMM-MDTC tradeoffs. For example, the highest redundancy (lowest, right-most) tradeoff point of GMM-MDSQ$_{TC}$ is closely followed by the 3-rd to the right tradeoff point of GMM-MDTC. Furthermore, the superior performance of the proposed GMM-MDTC scheme is clearly shown in lower loss probabilities. We can observe that GMM-MDTC is better than GMM-MDSQ$_{TC}$ in almost half of the available central/side distortion tradeoffs, and much better than GMM-MDSQ$_{TC}$ (ibit) in most of the available central/side distortion tradeoffs.

A comparison between GMM-MDSQ$_{TC}$ and GMM-MDTC is performed in terms of computational complexity and memory requirements. The computational complexity is measured in flops (floating point operations) while the memory requirements are measured in bytes. The evaluation is made using pseudo-code implementations of GMM-MDTC and GMM-MDSQ$_{TC}$. The parts of code that are common in both GMM-based quantizers (like the translation and decorrelation operations) are not taken into account. Details are omitted to Appendix B.2. The results are depicted in



(a) floating point operations **(b)** storage requirements (bytes)

Figure 7.13 Complexity and storage requirements for GMM-MDSQ$_{TC}$ and GMM-MDTC.

Figure 7.13. The two algorithms where evaluated for several loss probabilities (central/side distortion tradeoff points). Note that the storage requirements show the amount of static memory that holds precomputed information. The proposed method is twice as fast as GMM-MDSQ$_{TC}$at loss probabilities above $\rho \geq 0.2$, and more than two times faster for lower loss probabilities. Furthermore, the storage requirements are as lower than those of GMM-MDSQ$_{TC}$ (ibit) at $\rho = 0.5$, approximately the same at $0.2 < \rho < 0.4$ and higher at $\rho = 0.5$.. Compared to GMM-MDSQ$_{TC}$, though, the storage requirements are approx. 2.5 to 7 times lower at higher correlations. It is evident that the proposed method has considerable computational advantages. It achieves competitive multiple description coding performance with minimal storage and complexity requirements.

The analysis so far has examined the performance of the MDC methods according to the description loss probability ρ. However, the subjective quality of the resulting encoding cannot be captured by equation (7.27). Furthermore, the effect of interframe dependencies and packet loss concealment is not captured in the results of Figure 7.11. Therefore, the loss probability ρ should not be handled as a parameter directly associated with the conditions of the network, but as a parameter that allows the MDC encoder to switch to different central/side distortion tradeoff points. MDC is based on the minimization of a total distortion measure, i.e. equation (7.27). It is very difficult to quantify all the necessary parameters that effect the overall subjective quality in a distortion metric. It is up to the speech codec designer to establish a link between the central/side distortion tradeoffs and the network conditions, according to the overall subjective quality of the speech coder.

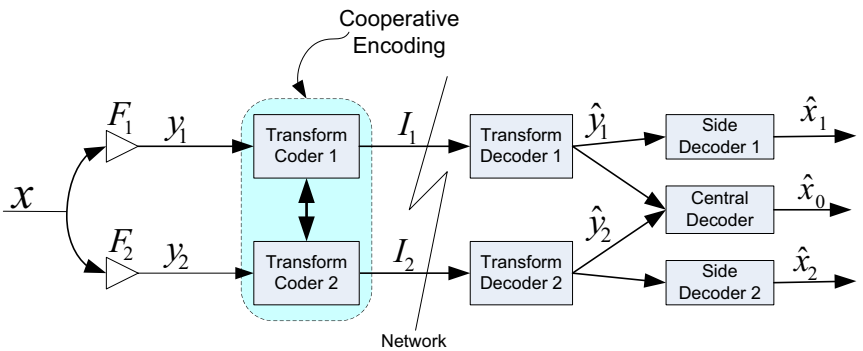


Figure 7.14 A schematic display of MDTC with Cooperative Encoding.

7.3 Improving MDTC: Cooperative Encoding

The performance of MDTC in higher packet loss probabilities is worse than the performance of MDSQ$_{TC}$. Similar observation have been made in the literature for other transform coding systems, as discussed in Section 7.1. This section identifies a source of performance degradation in MDTC and proposes an improvement. In MDTC, the source vector $\mathbf{x}$ is transformed in two vector $\mathbf{y}_1$ and $\mathbf{y}_2$ which are encoded independently by Gaussian quantizers. This means that for a given frame F the MDTC encoder operates in a manner that is optimal only for the side decoders and does not take into account the central decoder, resulting to higher central distortion as it is evident in Figure 7.6. In this Section, without loss of generality, the focus will be given to the case where the descriptions are of equal rate, balanced and transmitted through two independent channels with equal loss probability ρ. In that case, the central distortion is taken into account if the two encoders operate with respect to the total distortion:

$$d_{tot} = \|\mathbf{x} - \hat{\mathbf{x}}_0\|_2^2 + \frac{\rho}{1-\rho}\left(\|\mathbf{x} - \hat{\mathbf{x}}_1\|_2^2 + \|\mathbf{x} - \hat{\mathbf{x}}_2\|_2^2\right), \tag{7.28}$$

where $\mathbf{x}_0$, $\mathbf{x}_1$ and $\mathbf{x}_2$ are the output of the central decoder and the two side decoders, respectively. The total distortion can be broken to the sum of the individual distortions per dimension:

$$d_{tot} = \sum_{i=1}^{P}\left[(\mathbf{x}(i) - \hat{\mathbf{x}}_0(i))^2 + \frac{\rho}{1-\rho}\left[(\mathbf{x}(i) - \hat{\mathbf{x}}_1(i))^2 + (\mathbf{x}(i) - \hat{\mathbf{x}}_2(i))^2\right]\right], \tag{7.29}$$

where $\mathbf{a}(i)$ is the i-th dimension of vector $\mathbf{a}$. The minimization of d_{tot} is made by individually minimizing the scalar expressions inside the brackets. When the encoders do not cooperate, the terms $(\mathbf{x}(i) - \hat{\mathbf{x}}_0(i))^2$ are ignored. An MDTC scheme where the encoders cooperate will be referred to as *"Cooperative Encoding MDTC"* (MDTC$_{CE}$). A schematic representation of an MDTC$_{CE}$ scheme is shown in Figure 7.14.

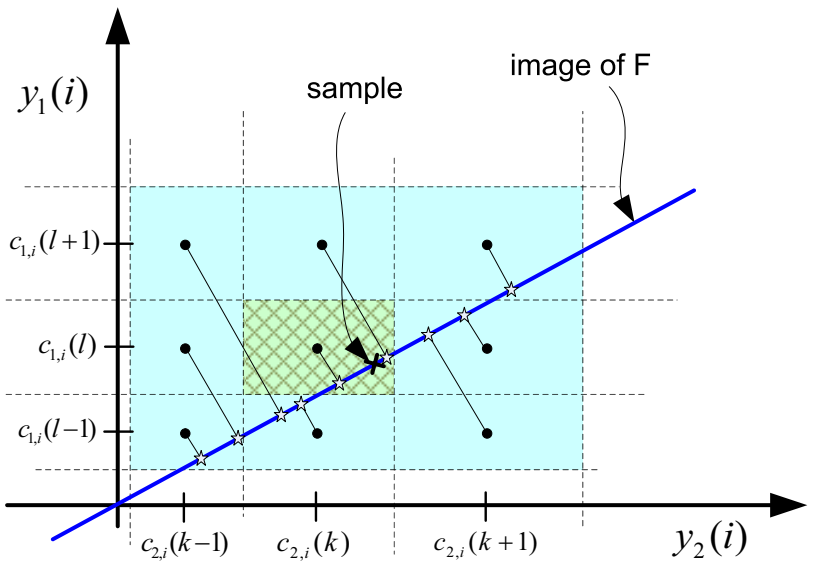


Figure 7.15 Cooperative Encoding in the side description $\mathbf{y}_1(i)$-$\mathbf{y}_2(i)$ plane.

The construction of an MDTC$_{CE}$ encoder is simplified by the frame F that was presented in Section 7.1.3. Each dimension of the source is encoded using one component in each side description. For example $\mathbf{x}(i)$ is encoded in $\mathbf{y}_1(i)$ and $\mathbf{y}_2(i)$. Figure 7.15 shows the $\mathbf{y}_1(i)$-$\mathbf{y}_2(i)$ plane and the corresponding quantization cells. The source point $\mathbf{x}(i)$ is projected to the image of F which in our case is a straight line passing through the origin of the axes. If the central distortion is ignored, the source point will be quantized to the patterned quantization cell that corresponds to the side description codepoints $c_1(l)$ and $c_2(k)$. The central description will be reconstructed with the projection of the codepoint $(c_1(l), c_2(k))$ to the image of F. Clearly, the central description codepoints are limited by the number of quantization cells that intersect the image of F, reducing the performance of the central decoder. The central distortion can be reduced if the neighboring quantization cells are used to provide more central description codepoints. Figure 7.15 depicts the candidate central description codepoints as stars that lay on the image of F. It is possible that a neighboring quantization cell provides less total distortion $\left[(\mathbf{x}(i)-\hat{\mathbf{x}}_0(i))^2+\frac{\rho}{1-\rho}\left[(\mathbf{x}(i)-\hat{\mathbf{x}}_1(i))^2+(\mathbf{x}(i)-\hat{\mathbf{x}}_2(i))^2\right]\right]$ than the quantization cell that minimizes the side distortions (patterned one). However, the possibility that a quantization cell minimizes the total distortion is greatly reduced with the distance from that cell. Therefore, in practice it suffices to search only it's direct neighbors.

Let $\mathbf{c}_{1,i}(\cdot)$ and $\mathbf{c}_{2,i}(\cdot)$ be the codepoints used to quantize $\mathbf{y}_1(i)$ and $\mathbf{y}_2(i)$, respectively. The algorithm of the cooperative encoder could be summarized as follows:

a. For each dimension $i = 1, ..., P$

b. Quantize $\mathbf{y}_1(i)$ and $\mathbf{y}_2(i)$ with the side encoders:
$l = \arg\min_j \left\{ \left(\mathbf{y}_1(i) - \mathbf{c}_{1,i}(j)\right)^2 \right\}$,
$k = \arg\min_j \left\{ \left(\mathbf{y}_2(i) - \mathbf{c}_{2,i}(j)\right)^2 \right\}$

c. Search the neighboring quantization cells for the one that minimizes the total distortion:
$(I_1(i), I_2(i)) = \arg\min_{(j_1,j_2)\in\mathbb{S}} \left\{ \left(\mathbf{x}(i) - \hat{\mathbf{x}}_0(i; j_1, j_2)\right)^2 + \frac{\rho}{1-\rho} \left[\left(\mathbf{x}(i) - \hat{\mathbf{x}}_1(i; j_1)\right)^2 + \left(\mathbf{x}(i) - \hat{\mathbf{x}}_2(i; j_2)\right)^2 \right] \right\}$
where $\mathbb{S} = \{l-1, l, l+1\} \times \{k-1, k, k+1\}$ is the set of neighboring cells and $\hat{\mathbf{x}}_0(i; j_1, j_2)$, $\hat{\mathbf{x}}_1(i; j_1)$ and $\hat{\mathbf{x}}_2(i; j_2)$ are reconstructions from the central decoder and the two side decoders that were made using the codepoints $\mathbf{c}_1(j_1)$ and $\mathbf{c}_2(j_2)$, respectively.

Cooperative encoding was evaluated for the LSF source presented in Section 7.1.5, Figure 7.7. The training of the system was made using the cooperative encoders instead of the independent encoders and the new method is evaluated in comparison to MDTC and MDSQ$_{TC}$. Figure 7.16 depicts the central/side distortion tradeoffs provided by MDSQ$_{TC}$, MDTC and MDTC$_{CE}$. MDTC$_{CE}$ outperforms MDTC in higher loss probabilities while it has the same performance in lower loss probabilities. Furthermore, at $\rho = 0.5$, the proposed method provides a central distortion that is only 0.35 dB away from the central distortion of MDSQ$_{TC}$, for a slightly (0.04 dB) better side distortion.

Another evaluation was made in the context of GMM-MDTC. The corresponding quantizer will be referred to as GMM-MDTC$_{CE}$. The evaluation was made with the RCC source used in Section 7.2.4, Figure 7.12. GMM-MDTC$_{CE}$ was trained using cooperative encoders. The results are depicted in Figure 7.17. It is clearly shown that the proposed quantizer offers tradeoff points which are similar to the tradeoff points of GMM-MDSQ$_{TC}$ and better that the tradeoff points of GMM-MDTC.

Concluding, a modification was made to the MDTC quantizer. This modification improved the central distortion and allowed multiple description transform coding with much better central/side distortion tradeoff points. The improvement though doesn't come without a computational cost. However, MDTC$_{CE}$ retains the advantages of MDTC: it is scalable, it has low storage requirements and it can be used with complicated channel models. Motivated by the findings of this section, we strongly suggest cooperative encoding also for the case of entropy-constrained multiple description transform coding.

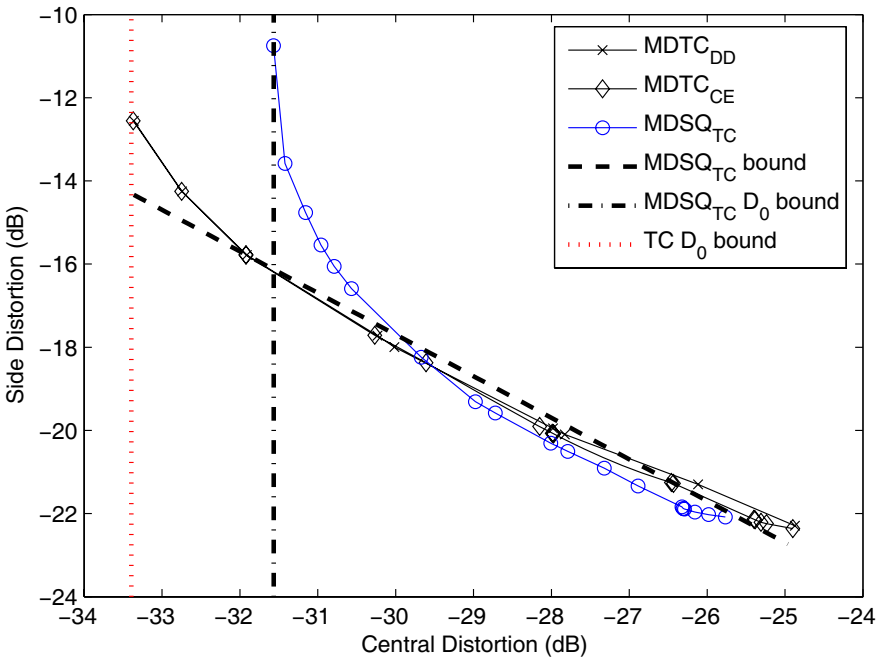


Figure 7.16 MDTC$_{CE}$ evaluation for the multivariate Gaussian LSF source.

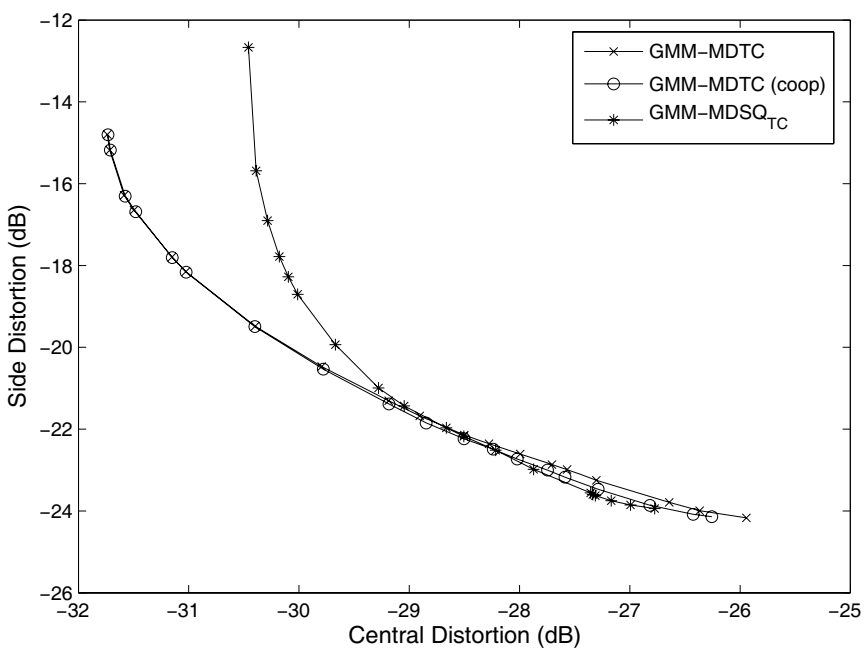


Figure 7.17 GMM-MDTC$_{CE}$ evaluation for the RCC source.

Chapter 8

Coding with Side Information

Several speech coding problems, like Speech Spectrum Expansion (the reconstruction of 4-8 kHz speech spectrum) and the recovery from packet losses in VoIP (Voice-over-IP), face the following situation: there is *available* information and *lost* information, and the lost information has to be -somehow- *recovered* from the available information. This is an *estimation* problem when there is no possibility to transmit additional data, and a *coding* problem when data transmission is permitted. In a simple coding scenario where the available information is coded independently of the lost information (however, useful to the decoder) there is no benefit from the mutual information between the two sources: the *lost* information and the *available* information. Therefore, it is desirable to encode the former having the latter as *side* information.

In terms of (Conditional) Rate-Distortion theory, this is referred to as Coding with Side Information (CSI) problem [156], [157] schematically shown in Figure 8.1, where Y is the information that will be coded, and $\hat{X}$ the side information (with distortion) available at the encoder and the decoder. Estimation can be seen as a particular case of CSI where the transmitted bit stream is empty. This chapter investigates the potential of CSI for applications like wideband speech coding, bandwidth expansion and packet loss concealement.

There has been much effort in the enhancement of the narrow-band (0.3-3.4 kHz) PSTN (Public Switch Telephone Network) speech signal by bandwidth expansion; the high-band is estimated from the narrow-band using several methods like VQ mapping

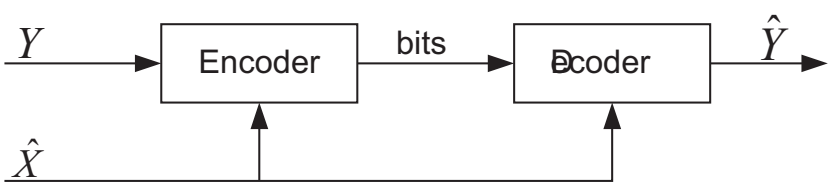


Figure 8.1 Coding with Side Information.

[158], GMM (Gaussian Mixture Models) based estimators [159], [160] and HMMs (Hidden Markov Models) [161]. These attempts report an improvement over narrow-band speech, although the resulting speech signal suffers from artifacts. The quality of the reconstructed speech is bounded by the relatively low mutual information between the two bands [162], [163] and the poor performance of estimation [164]. On the other hand, the acceptable performance of these methods indirectly states that the required bit-rate for high quality highband reconstruction should be low. Coding the highband without taking advantage of the highband knowledge carried at the narrow-band, results in a higher bit-rate. Therefore, it is beneficial to encode the highband having the narrow-band as side information available to the encoder and the decoder.

It is widely accepted that for many speech sounds the lower frequencies are perceptually more important than the higher frequencies. Therefore, in wideband speech coding it may be desirable to separately encode the spectral envelope of the higher frequencies from the spectral envelope of the lower frequencies. Moreover, different fidelity requirements may be used in each band. For example, memoryless coding of the wideband spectral envelopes (0-8 kHz) using 14 Line Spectrum Frequencies (LSFs) requires ≈ 41 bits/frame, while coding narrow-band spectral envelopes (0-3.4 kHz) using 10 LSFs requires ≈ 24 bits/frame [165]. Because a high distortion is in general acceptable at the higher frequencies the use of a non-weighted single fidelity criterion to the whole wideband spectral envelope is perceptually not optimal. Furthermore, different bands may need to be encoded using different analysis/synthesis rates. Splitting the wideband spectral envelope in two bands and coding them with different fidelity criteria can be quite advantageous, but it results to an information loss equal to the mutual information between the two spectra. Coding with Side Information may use most of the mutual information, by reestablishing the broken dependencies between the two information sources [156].

New packet-based applications like Voice-over-IP (VoIP) generate new demands for codecs. Packets, typically containing 10-30 ms of encoded speech, may be lost or unacceptably delayed. A lookahead buffer called "jitter buffer" containing a few packets of speech is used to counteract small delays of packet arrivals. One lost packet results to the loss of 1-2 speech frames and depending on the speech codec used, the reconstruction error can be propagated to several following frames [35]. An obvious way to cope with this is to use Forward Error Correction (FEC) [35]; the information of the current frame is repeated in the next frame, but the added redundancy does not take into account the information carried at the neighboring frames. Some researchers try to estimate the lost spectral envelope from the previous frame(s) [166], [167]. Coding with Side Information can be used to introduce a small size corrective bit-stream that provides an enhanced estimation/coding of the lost spectral envelope(s), up to a pre-defined fidelity requirement. In other words, the idea is to repair the loss, not to repeat the loss.

Coding with Side Information is not something completely new in speech coding. In fact, various forms of Predictive Coding can be seen as CSI; the current frame is coded having the previous frame as side information under certain distortion re-

quirements. In this perspective, CSI can be seen as a generalization of Predictive Coding, with complex non-linear input-output space relationships, where adverse but relevant information sources (like LSFs, energy, voicing, pitch) can be used as side information.

This chapter deals with two distinct speech coding problems which are formulated in the CSI context; the loss of packets in VoIP and speech spectrum expansion. Focus is given to the recovery of the lost spectral envelope information. A VQ-based solution to the CSI problem is proposed. In Section 8.1 the CSI problem is discussed using Conditional Rate-Distortion theory arguments, in comparison with estimation and simple VQ. The role of mutual information is discussed and a distortion-rate bound for CSI is given. The discussion is supported by a toy example. In Section 8.2 we formulate/simplify the CSI problem as a generalization of VQ, which will be referred to as the Conditional Vector Quantization (CVQ) problem, and suggest a fast divide-and-conquer two-step solution. CVQ assumes a piecewise one-to-many mapping between input space X (the side information) and output space Y (the coded information). Section 8.3 describes three estimation methods. The following sections discuss two applications of CSI. In Section 8.4 CVQ is used to generate a repairing bit-stream for the VoIP problem and encode the current spectral envelope, using the previous and the next spectral envelopes as side information. Using LSFs for the parameterization of the spectral envelopes, we show that a very low bit-stream of 400 bits/sec can significantly reduce the reconstruction distortion for single and double packet losses. In Section 8.5 we use CVQ to encode the highband 4-8 kHz LSFs using the narrow-band 0-4 kHz LSFs as side information. It is shown that, provided an appropriate excitation, only 134 bits/sec are enough for a high quality reconstruction of the highband spectral envelopes.

8.1 Background

Lets us consider two correlated sources X,Y, and their joined source $Z = [\ X\ Y\ ]^T$. Source X is already transmitted from the encoder to the decoder, while source Y must be, somehow, reconstructed at the decoder. Three options are available then:

- estimate Y given X. In most cases mutual information $\mathcal{I}(x;y)$ between the two sources cannot be fully utilized.
- encode Y with a CSI system having X as side information. Mutual information $\mathcal{I}(x;y)$ can be effectively utilized.
- encode Y. In this case, mutual information is lost.

The best option for reconstructing Y will depend on the amount of mutual information, the available bit-rate and the fidelity requirement. In this section we discuss about the benefits and the limits of Coding with Side Information (as shown in Figure 8.1), using rate-distortion theory arguments. The distortion-rate Shannon Lower

Bound for CSI will be provided, and a non-tight distortion bound for estimation will be given as a special case.

8.1.1 Conditional Rate-Distortion Theory

Let $\mathcal{R}_x(\Delta_x)$, $\mathcal{R}_y(\Delta_y)$ and $\mathcal{R}_{xy}(\Delta_x, \Delta_y)$ be the rate-distortion functions for X,Y and Z, respectively, where Δ_x, Δ_y is the fidelity constraint for each of the corresponding variables. Let $D_x(x, \hat{x})$, $D_y(y, \hat{y})$ be some distortion measures over X-space and Y-space, respectively. Rate-Distortion theory [59] states that :

$$\mathcal{R}_y(\Delta_y) = \inf_{p(\hat{y}|y):E_{y,\hat{y}}\{D_y(y,\hat{y})\}\leq\Delta_y} \mathcal{I}(y;\hat{y}) \tag{8.1}$$

where $\mathcal{I}(y;\hat{y})$ is the mutual information between the source and the encoded source. For the Coding with Side Information problem we are mainly interested in rate $\mathcal{R}_{y|x}(\Delta_y)$ which is the rate of the system depicted in Figure 8.1. The formula for the conditional rate-distortion function [156] is analogous to (8.1):

$$\mathcal{R}_{y|x}(\Delta_y) = \inf_{p(\hat{y}|y,x):E_{x,y,\hat{y}}\{D_y(y,\hat{y})\}\leq\Delta_y} \mathcal{I}(y;\hat{y}|x) \tag{8.2}$$

Note that $\mathcal{R}_{y|x}(\Delta_y)$ is the rate of the CSI system when side information X is provided with zero distortion. The conditional rate-distortion function satisfies the following inequalities [156]:

$$\mathcal{R}_{xy}(\Delta_x, \Delta_y) \geq \mathcal{R}_{y|x}(\Delta_y) + \mathcal{R}_x(\Delta_x) \tag{8.3}$$

$$\mathcal{R}_{y|x}(\Delta_y) \geq \mathcal{R}_y(\Delta_y) - \mathcal{I}(x;y) \tag{8.4}$$

$$\mathcal{R}_{xy}(\Delta_x, \Delta_y) \geq \mathcal{R}_y(\Delta_y) + \mathcal{R}_x(\Delta_x) - \mathcal{I}(x;y) \tag{8.5}$$

where $\mathcal{I}(x;y)$ is the mutual information between the two sources. Under moderate assumptions, inequalities (8.3) to (8.5) become equalities [156]. The assumptions are that there are no restricted transitions between X and Y (for any x and y, $P(y|x)$ is non-zero), and that distortions Δ_x and Δ_y are sufficiently small. When these assumptions do not hold, the above inequalities provide the performance bounds. On the other hand, when the assumptions hold there is no rate penalty for encoding source Y with a CSI system instead of jointly encoding X and Y. Therefore coding X with fidelity Δ_x, and Y with fidelity Δ_y at a specific rate can be made either way: with typical source coding of the joined source Z or with CSI. Additionally, CSI has the advantage of being applicable in cases where the two sources X and Y are defacto separated. Furthermore, (8.4) states the role of mutual information: $\mathcal{I}(x;y)$ is the rate loss for encoding Y without knowing X.

Note that in [156] inequalities (8.3) to (8.5) are proven for X and Y taking values from finite alphabets. However, it is quite straightforward to extend the proof of the corresponding theorem to continuous sources.

8.1.2 Mutual Information

Mutual information provides the rate gain when a CSI system is used for coding Y, instead of a typical source coding system. Furthermore, mutual information is provided in closed form [59]:

$$\mathcal{I}(x;y) = E_{x,y}\{\log \frac{p(x,y)}{p(x)p(y)}\} \tag{8.6}$$

When densities $p(x,y)$,$p(x)$,$p(y)$ are available through a continuous parametric model like a GMM, the integral in (8.6) can be approximated by stochastic integration [162], [163], according to the law of big numbers:

$$\mathcal{I}(x;y) \approx \frac{1}{N}\sum_{n=1}^{N} \log \frac{p(x_n,y_n)}{p(x_n)p(y_n)} \tag{8.7}$$

where x_n, y_n are drawn from the join pdf $p(x,y)$.

Several properties of mutual information provide further insight to the CSI problem. For example, theoretically we cannot increase the rate gain of a CSI system by using other transformations (1-1 mapping functions $g(\cdot)$, $f(\cdot)$) of either X or Y, because a transformation can only decrease mutual information, as stated by the data processing inequality [59]:

$$\mathcal{I}(X,Y) \geq \mathcal{I}(g(X),f(Y)) \tag{8.8}$$

8.1.3 Distortion-Rate for CSI

A distortion-rate bound for CSI and squared error distortion measure can easily be derived via Shannon's Lower Bound (SLB) for vector processes:

$$\mathcal{D}_y(R_y) \geq \frac{d}{2\pi e}\exp\left(\frac{2}{d}(h(y) - R_y)\right) \tag{8.9}$$

where $h(y)$ is the differential entropy of source Y, and d the dimensionality of Y-space. Using inequalities (8.4) and (8.9) we can derive a Shannon Lower Bound for the distortion rate function of vector processes for CSI:

$$\mathcal{D}_y(R_{y|x}) \geq \frac{d}{2\pi e}\exp\left(\frac{2}{d}(h(y) - R_{y|x} - \mathcal{I}(x;y))\right) \tag{8.10}$$

Note that inequality (8.4) is also valid for vector processes (exercise 4.4 in [168]) and continuous sources.

In the CSI framework, estimation can be seen as the attempt to recover Y at the decoder without transferring any bits ($\mathcal{R}_{y|x} = 0$). By setting $R_{y|x} = 0$ we obtain a boundary to the performance of an estimator of Y given X:

$$\mathcal{D}_y \geq \frac{d}{2\pi e}\exp\left(\frac{2}{d}(h(y) - \mathcal{I}(x;y))\right) \tag{8.11}$$

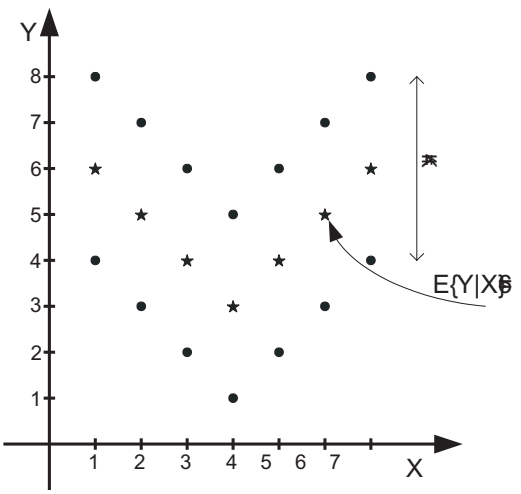


Figure 8.2 A Toy Example.

This is the same estimation bound with the one provided in [162]. However, note that the bound is not tight [162]. Based on the discussion developed in section 8.1.1 this is expected since the estimation distortion is rather high and mutual information is gained only when distortions Δ_x and Δ_y are sufficiently small.

The evaluation of CSI via the SLB is not practical for many sources (including the speech spectral envelopes) for two reasons: it is not always feasible to determine the tightness of the SLB and it is not always possible to make an accurate estimation of the differential entropy $h(y)$. Note that the estimation of differential entropy is not a trivial task when data lay on a manifold, since then $h(y)$ must be computed over the manifold. Furthermore, there is evidence that the spectral envelopes of speech lay on manifolds [169]. In such cases, the evaluation of CSI can be made via an estimation of the mutual information, e.g. as presented in Section 8.1.2.

8.1.4 A Toy Example

A toy example, similar to the one provided in [162], will be given to illustrate the notions described in previous subsections. Let $X \in \{1...7\}$ and $Y \in \{1...8\}$ be random variables taking values from finite alphabets. Let X,Y follow the joined distribution depicted in Figure 8.2. The joint distribution codepoints (dots) have equal probability $p = \frac{1}{14}$. Three bits are needed to describe Y. If we perform an estimation $\hat{y} = E_y\{Y|X\}$ of Y from X, we get the stars between the codepoints. Estimation $\hat{y}$ depends on the distance k between the two codepoints corresponding to the value of X. Note that for any $k \geq 3$, the mutual information is constant ($\mathcal{I}(x;y) = 1.95$) bits and the entropy is fixed to $\mathcal{H}(y) = 2.95$ bits. Therefore the distortion-rate function $\mathcal{D}_y(R_{y|x})$ is independent of k. Obviously, estimation distortion

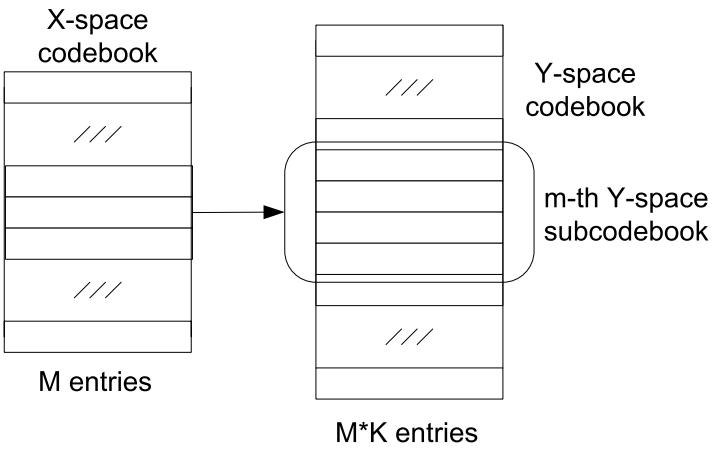


Figure 8.3 Conditional Vector Quantization.

can be arbitrary large for the given statistics. An important remark can be made: if 1 bit is provided, the reconstruction distortion falls to zero. For a given X, 2 codepoints may be chosen. The extra bit helps choosing among these codepoints. In terms of our previous discussion, distortion Δ_y in the case of estimation (rate $R_{y|x} = 0$) is too large to take advantage of the mutual information. If 1 bit is provided, Δ_y becomes small enough ($= 0$) to gain $\mathcal{I}(x; y)$.

8.2 Conditional Vector Quantization

Intuitively, each value of X-space generates a different conditional pdf $p(y|x)$ for Y-space. We will try to capture the coarse structure of this mapping, using a Vector Quantization framework, which is referred to as Conditional Vector Quantization. The main idea is that each region in X-space is mapped to a different codebook of Y-space.

The problem of Conditional Vector Quantization (CVQ) will be approached through a probabilistic point of view. Let $\vec{x} \in \mathcal{R}^P$, $\vec{y} \in \mathcal{R}^D$ be random vectors of X-space and Y-space respectively. The CVQ problem consists of constructing two linked codebooks $C_x \equiv \{\hat{\vec{x}}_m : m = 1...M\}$ and $C_y \equiv \{\hat{\vec{y}}_{m,k} : m = 1...M, k = 1...K\}$, for X-space and Y-space respectively. Each codevector in C_x is linked to K codevectors in C_y, which form the m-th subcodebook of C_y. The encoder finds the nearest C_x codevector and transmits the index of the nearest C_y codevector of the linked C_y subcodebook. The decoder locates the nearest C_x codevector and takes the estimation from the linked C_y subcodebook according to the transmitted index. Figure 8.3 illustrates the two codebooks C_x and C_y, for $K = 4$. CVQ can be seen as a form of classified vector quantization [55], where the classification rule is taken from a VQ of X-space.

The CVQ reconstruction of $\vec{y}$ is a function of $\vec{y}$, $\vec{x}$, C_x, C_y:

$$\hat{\vec{y}}_{m,k} = Q_{y|x}(\vec{y}, Q_x(\vec{x}, C_x), C_y) \tag{8.12}$$

where $Q_x(.)$ is the quantization rule for X-space and $Q_{y|x}(.)$ the quantization rule for Y-space depending on X-space. The encoding rule can be expressed as:

$$k = \arg\min_{k'}\{d(\vec{y}, \hat{\vec{y}}_{m,k'})\}, \text{where } m = \arg\min_{m'}\{d(\vec{x}, \hat{\vec{x}}_{m'})\} \tag{8.13}$$

where $d(.,.)$ is some distortion measure. If we assume that $\vec{x}_m$, $\vec{y}_{m,k}$ are random vectors spanning the discrete spaces C_x, C_y, respectively, the average distortion of the CVQ encoding/decoding process becomes:

$$D = \sum_{m=1}^{M}\sum_{k=1}^{K}\iint p(\vec{x}, \vec{y}, \hat{\vec{x}}_m, \hat{\vec{y}}_{m,k}) d(\vec{y}, \hat{\vec{y}}_{m,k}) d\vec{x} d\vec{y} \tag{8.14}$$

The joint probability $p(\vec{x}, \vec{y}, \hat{\vec{x}}_m, \hat{\vec{y}}_{m,k})$ can be analyzed to

$$p(\vec{x}, \vec{y}, \hat{\vec{x}}_m, \hat{\vec{y}}_{m,k}) = p(\vec{x}, \vec{y}) p(\hat{\vec{x}}_m | \vec{x}, \vec{y}) p(\hat{\vec{y}}_{m,k} | \hat{\vec{x}}_m, \vec{y}, \vec{x})$$

using the Bayes rule. The latter expression can be simplified with two CVQ-related assumptions. The first assumption is that the decoder cannot have knowledge of $\vec{y}$, therefore $\hat{\vec{x}}_m$ is conditionally independent of $\vec{y}$: $p(\hat{\vec{x}}_m | \vec{x}, \vec{y}) \equiv p(\hat{\vec{x}}_m | \vec{x})$. The second assumption is that $\hat{\vec{y}}_{m,k}$ is conditionally independent of $\vec{x}$: $p(\hat{\vec{y}}_{m,k} | \hat{\vec{x}}_m, \vec{x}, \vec{y}) \equiv p(\hat{\vec{y}}_{m,k} | \hat{\vec{x}}_m, \vec{y})$ stating the piecewise mapping nature of the CVQ model; that no higher than first order local statistics are taken into account when mapping a X-space region to K Y-space regions. Using these two assumptions we conclude that:

$$D = \iint p(\vec{x}, \vec{y}) \sum_{m=1}^{M} p(\hat{\vec{x}}_m | \vec{x}) \sum_{k=1}^{K} p(\hat{\vec{y}}_{m,k} | \hat{\vec{x}}_m, \vec{y}) d(\vec{y}, \hat{\vec{y}}_{m,k}) d\vec{x} d\vec{y}.$$

If the number of samples $[\vec{x}_n, \vec{y}_n]$, $n = 1, 2, ..., N$ is large enough then the law of big numbers states that D can be approximated by:

$$D \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{M} p(\hat{\vec{x}}_m | \vec{x}_n) \sum_{k=1}^{K} p(\hat{\vec{y}}_{m,k} | \hat{\vec{x}}_m, \vec{y}_n) d(\vec{y}_n, \hat{\vec{y}}_{m,k}) \tag{8.15}$$

The conditional probability $p(\hat{\vec{x}}_m | \vec{x}_n)$ is the association probability relating the input vector $\vec{x}_n$ with codevector $\hat{\vec{x}}_m$, while the association probability $p(\hat{\vec{y}}_{m,k} | \hat{\vec{x}}_m, \vec{y}_n)$ relates the output vector $\vec{y}_n$ with the codevector $\hat{\vec{y}}_{m,k}$ of the m-th subcodebook of C_y. The conditional dependence of $\hat{\vec{y}}_{m,k}$ with $\hat{x}_m$ states that $\hat{\vec{y}}_{m,k}$ belongs to the m-th subcodebook of C_y. Although CVQ problem considers hard association probabilities taking

values in $\{0, 1\}$, the distortion formula (8.15) does not explicitly impose regular partitions. Therefore minimization of D can also be made with non-regular partitions, i.e. Gaussians, in X-space and/or Y-space.

The minimization of D is a hard problem, but the complexity can be reduced if it is broken into several easier subproblems: first compute a VQ of X-space and then minimize D. Since the partitioning of X-space determines the association probabilities $p(\hat{\vec{x}}_m|\vec{x}_n)$ and the codevectors $\hat{\vec{x}}_m$, the minimization problem breaks into a series of M typical weighted VQ minimization subproblems D_m:

$$\begin{aligned} D &\approx \sum_{m=1}^{M}[\frac{1}{N}\sum_{n=1}^{N} p(\hat{\vec{x}}_m|\vec{x}_n)\sum_{k=1}^{K} p(\hat{\vec{y}}_{m,k}|\hat{\vec{x}}_m, \vec{y}_n)d(\vec{y}_n, \hat{\vec{y}}_{m,k})] \\ &= \sum_{m=1}^{M} D_m \end{aligned}$$

Furthermore, with hard association probabilities each of the M minimization subproblems D_m operates in a subset of Y-space vectors providing therefore a significant computational advantage.

The resulting algorithm for hard association probabilities is:

- compute a VQ of X-space (M codevectors)
- for every $\hat{\vec{x}}_m \in C_x$:
- find the Y-space vectors corresponding to the X-space vectors that are nearest to $\hat{\vec{x}}_m$.
- perform a VQ on these Y-space vectors (K codevectors) to compute the m-th Y-space subcodebook

At the case where $K = 1$, the CVQ problem is similar to the GVQ (Generalized VQ) [170] problem, and the proposed solution is reduced to the NLIVQ (Non-Linear Interpolative VQ) [171] solution of GVQ. CVQ has also been used in [158]. Note, however, that in [158] the Y-space codebooks are taken from a Y-space partitioning that is trained independently of the X-space codebooks. This solution is not consistent with (8.15) where it is clearly shown that the Y-space codewords depend directly on the X-space partition and not via a precomputed partitioning of Y-space.

8.3 Estimation

In some applications like Speech Spectrum Expansion and VoIP packet loss concealment, the lost information Y is usually estimated from the available information X. The performance of the estimation is not always adequate in terms of subjective quality. CSI can overcome this limitation by providing an "enhanced" estimation at the

cost of a few extra bits. A comparison between CSI and estimation is then necessary to indicate the practical performance gain when this strategy is adopted.

For this purpose, we focus on three memoryless mapping estimators; Linear Prediction, a simple VQ mapping called NLIVQ (Non Linear Interpolative Vector Quantization) [171] and GMM-based estimation which will be referred to as GMM Conversion Function [172], [160]. The linear estimator provides a well-known baseline because it corresponds to the optimal linear relationship between the two spaces. The NLIVQ estimator provides useful insight as a special CVQ case (CVQ with $K = 1$). The GMM Conversion Function is a robust state-of-the-art estimator able to handle complex input-output space relationships.

8.3.1 Linear Estimation

In Linear Estimation the estimated $\hat{\vec{y}}_t$ is a linear combination of the available information: $\hat{\vec{y}}_t = A\vec{x}_t$. The linear estimation $\hat{\vec{y}}_t$ is computed according to the formulae:

$$\begin{aligned}
\hat{\vec{y}}_t &= E\{\vec{y}_t\} + \Sigma_{yx}\Sigma_{xx}^{-1}\left(\vec{x}_t - E\{\vec{x}_t\}\right), \\
\Sigma_{yx} &= \frac{1}{N}\sum_{t=1}^{N}\left(\vec{y}_t - E\{\vec{y}_t\}\right)\left(\vec{x}_t - E\{\vec{x}_t\}\right)^T, \\
\Sigma_{xx} &= \frac{1}{N}\sum_{t=1}^{N}\left(\vec{x}_t - E\{\vec{x}_t\}\right)\left(\vec{x}_t - E\{\vec{x}_t\}\right)^T
\end{aligned}$$

where N is the number of training set vectors and $E\{\cdot\}$ denotes the expectation operator. When the past is used to estimate the future, linear estimation is referred to as *linear prediction* [55] and it is commonly used in Predictive Vector Quantization [173].

8.3.2 NLIVQ

The NLIVQ method [171] uses two equal-sized codebooks, one for X-space codevectors and one for Y-space codevectors. The X-space vector is classified to the nearest X-space codevector which is mapped to one Y-space codevector. The X-space codebook is constructed by a variant of the well known binary split LBG VQ algorithm. The Y-space codebook is constructed from the means of Y-space vectors corresponding to X-space vectors that are nearest to the linked X-space codevector. NLIVQ is essentially the same to the CVQ method proposed in Section 8.2 when $K = 1$.

8.3.3 GMM Conversion Function

The GMMCF estimator uses an experts-and-gates regression function to "convert" the narrow-band vectors to the wideband vectors. Both input and output spaces are

modelled through GMM. The GMM conversion function is defined by:

$$\hat{\vec{y}} = \sum_{m=1}^{M} p(\omega_m|\vec{x})[\vec{y}_m + \Sigma_{yx}^m(\Sigma_{xx}^m)^{-1}(\vec{x} - \vec{x}_m)] \tag{8.16}$$

where $\vec{x}$ is the input vector associated with X-space, $\hat{\vec{y}}$ the estimation of $\vec{y}$, $\vec{x}_m$ and $\vec{y}_m$ denote the centroids of the m-th Gaussian of X-space and Y-space respectively, and Σ_{xx}^m is the covariance matrix of the m-th X-space Gaussian, Σ_{yx}^m is the cross-covariance matrix that relates the m-th Gaussians of X-space and Y-space, and ω_m denotes the m-th class of X-space. Finally, $p(\omega_m|\vec{x})$ is the gating probability given by:

$$p(\omega_m|\vec{x}) = \frac{p(\omega_m)|\Sigma_{xx}^m|^{-0.5} e^{-0.5(\vec{x}-\vec{x}_m)^T(\Sigma_{xx}^m)^{-1}(\vec{x}-\vec{x}_m)}}{\sum_{n=1}^{M} p(\omega_n)|\Sigma_{xx}^n|^{-0.5} e^{-0.5(\vec{x}-\vec{x}_n)^T(\Sigma_{xx}^n)^{-1}(\vec{x}-\vec{x}_n)}} \tag{8.17}$$

The learning process for the GMM-based estimation function comprises of two stages. In the first stage a GMM of the X-space is estimated via the standard EM algorithm, while in the second stage the Y-space means $\vec{y}_m$ and the matrices Σ_{yx}^m are computed using a least squares criterion [172]. For the experiments we used diagonal covariance matrices Σ_{xx}^m and full cross-covariance matrices Σ_{yx}^m.

8.4 CVQ of Lost Spectral Envelopes for VoIP

Speech signals contain considerable temporal correlations. These correlations can be used to tackle the packet loss problem in VoIP. For example, the LSF parameters of adjacent frames are highly correlated and this has been successfully used in modern codecs for Packet Loss Concealment (PLC) [54]. Waveform substitution PLC algorithms try to reconstruct the lost speech giving emphasis to the continuity of the speech waveform [126]. However, waveform substitution techniques do not ensure the continuity of the sinusoidal tracks nor phase coherency. These desirable properties can be provided by sinusoidal PLC schemes [49] which outperform waveform PLC schemes [126]. Sinusoidal PLC schemes require the knowledge of the spectral envelope(s) of the lost speech frame(s). The lost spectral envelopes can be recovered with a repetition scheme or with more sophisticated estimators [166], [167].

The performance of the estimators is bounded by the mutual information and the structure of the underlying probability space. To overcome these problems Forward Error Correction (FEC) techniques have been proposed [35]. These algorithms require full repetition of the information for each packet consuming, however, bandwidth (by doubling the bit-rate of a code.) CSI can be used to provide an adequate reconstruction of the lost spectral envelopes with minimal extra bandwidth. More specifically, past and future spectral envelopes (contained in the jitter buffer) can be used as side information for encoding the lost spectral envelope(s). In [1](pg. 158), a deterministic frame-fill technique has been used to increase the temporal resolution of coarsely sampled (every 30ms) spectral envelopes. CVQ is the stochastic counterpart

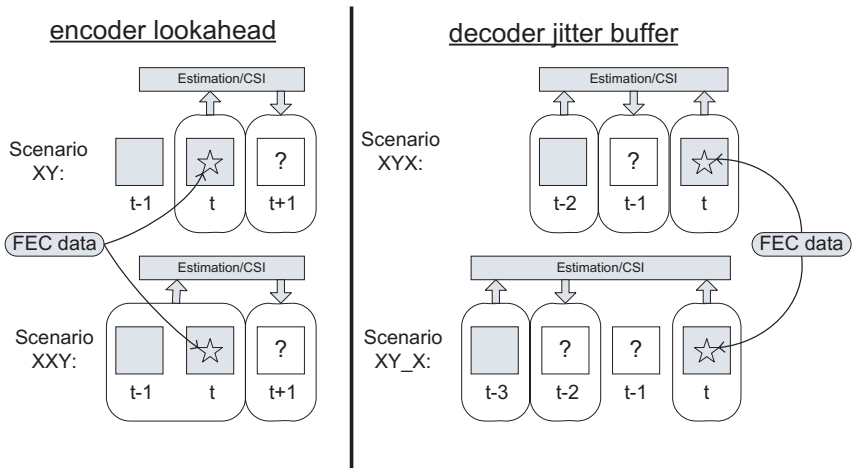


Figure 8.4 The 4 examined scenarios of lost/received packets using a 0-2 packet jitter buffer. The boxes indicate lost/received packets. A lost packet (questionmark) is estimated or CSI encoded using some of it's neighboring packets. In each scenario, the CSI data -when needed- is stored in the packets with the star.

of this frame-fill technique and it is capable of handling the complicated correlations between the received and the lost spectral envelopes.

A typical jitter buffer usually contains 1-2 packets (20-40 ms) of speech. With a jitter buffer of 2 packets, CVQ can be used to effectively handle single and double packets losses. We will focus on the narrow-band spectral envelopes, typically encoded with 10 LSFs per frame, assuming that each packet contains one spectral envelope. Note, however that CVQ can be also be used for other parameters like pitch and gain.

8.4.1 Recovery Scenarios

Let $\vec{u}_t$, $t = \{1, 2, ...\}$ be the sequence of transmitted LSF vectors. We assume that each packet contains 1 LSF vector. We further assume that the decoder has a jitter buffer of 1-2 packets and keeps a history of 1-2 packets. The idea is to use the information in the received packets to recover the information of a lost packet. Clearly, there are many possible combinations of lost/received packets that could be examined. Four possible scenarios, depicted in Figure 8.4, were selected. Incoming LSF vectors/packets are drawn as boxes. Lost packets contain a questionmark. CSI data are stored in each packet. The CSI data that is going to be used in each scenario under study, is presented by a star inside the boxes. The box with the star is always the last *received* packet $\vec{u}_t$. The two leftmost scenarios require 1 packet lookahead at the encoder, while the rightmost scenarios require a jitter buffer of 1-2 packets.

In scenarios XY, XXY, the LSF vector $\vec{u}_{t+1}$ is lost and one (for XY) or two (for

XXY) previous LSF vectors are used as side information. Using more than two past spectral envelopes for estimation does not enhance the estimation performance, as shown in [167]. Scenario XYX considers the case when the current LSF vector is lost, while the next and the previous vectors are received and used as input space information. Scenario XY_X is the case when two consecutive LSF vectors are lost and we wish to recover $\vec{u}_{t-2}$ using the information carried in $\vec{u}_{t-3}$,$\vec{u}_t$.

The other lost vector $\vec{u}_{t-1}$, can be recovered from the reconstructed $\hat{\vec{u}}_{t-2}$ and the received $\vec{u}_t$ applying a technique used in scenario XYX. This way, a decoder can use the XYX CSI bitstream to recover from single packet losses and both XYX, XY_X CSI bitstreams to recover from double packet losses, without retransmitting redundant information for $\vec{u}_{t-1}$. Table 8.1 shows a brief description of the 4 scenarios. For each scenario, input space X denotes the known (side) information and output space Y denotes the lost information that is going to be reconstructed.

Scenario	packet delay	Input Space	Output Space
XY	1 (lookahead)	$\vec{x}_t = [\vec{u}_t]$	$\vec{y}_t = [\vec{u}_{t+1}]$
XXY	1 (lookahead)	$\vec{x}_t = [\vec{u}_{t-1}\vec{u}_t]$	$\vec{y}_t = [\vec{u}_{t+1}]$
XYX	1 (jitter buffer)	$\vec{x}_t = [\vec{u}_{t-2}\vec{u}_t]$	$\vec{y}_t = [\vec{u}_{t-1}]$
XY_X	2 (jitter buffer)	$\vec{x}_t = [\vec{u}_{t-3}\vec{u}_t]$	$\vec{y}_t = [\vec{u}_{t-2}]$

Table 8.1 Brief description of the 4 scenarios. Note that the packet delay in the first two scenarios refers to the case of transmitting FEC data.

8.4.2 Practical CSI

The simplest form of CSI is residual coding. Let $\hat{\vec{y}}_t$ be the estimation of $\vec{y}_t$. Residual coding uses a form of VQ to encode $\vec{\epsilon}_t = \vec{y}_t - \hat{\vec{y}}_t$. In literature, residual coding is typically made using Linear Estimation [173].

In this section we suggest to use the *CF* estimator for residual coding. The *CF* estimator capability of modelling complex non-linear relationships between X and Y, provides a residual $\vec{\epsilon}_t$ that is more whitened, compared to the *LE* residual. In our knowledge, until now, nobody has used GMM-based estimators like *CF* for residual coding of LSFs.

Even if the (unknown) *optimal* estimator was used, residual coding may not be able to benefit from all the mutual information between X and Y. Our measurements indicate that in scenario XY, the mutual information between the side information $\vec{x}_t$ and *CF* estimation residual $\vec{\epsilon}_{CF,t}$, is 2.51 bits, while the mutual information between $\vec{x}_t$ and $\vec{y}_t$ is 5.85 bits. The mutual information between the *LE* estimation residual and $\vec{x}_t$ is measured to be 2.82 bits. In other words, the *CF* estimation residual has nearly 43% of the initial mutual information between $\vec{x}_t$ and $\vec{y}_t$. Note also, that *CF* residual has less mutual information that *LE* residual, indicating that *CF* provides a better estimation than *LE*. Similar results were also obtained for scenarios XXY,

XYX, XY_X. The mutual information measurements were conducted using GMM with diagonal covariance matrices, 1024 Gaussians and 10^6 samples for the Monte Carlo integration.

We attempt to gain from the mutual information between the estimation residual and the side information, by using CVQ presented in section 8.2. CVQ will be used to encode the estimation residual, and not $\vec{y}_t$. When CVQ is used to directly encode $\vec{y}_t$ the results were worse than the results obtained from a simple VQ of the linear estimation (LE) residual. However, as the number of X-space classes M increases from 32 to 512, the results were improving, indicating that a much higher M is required for a proper modeling of the input-output space relationship. The removal of a simple rotational relationship between Y-space and X-space by LE was enough to let CVQ benefit from the (remaining) mutual information.

8.4.3 Experiments

For the scenarios presented in Section 8.4.1, two modes will be evaluated for the recovery of lost LSFs:

- estimation mode (no data transmission), using the following estimators:
  - Linear Estimation (**LE**)
  - GMM Conversion Function (**CF**)
- CSI mode (with data transmission), using the following methods:
  - VQ of the *LE* Residual (**VQLE**)
  - VQ of *CF* Residual (**VQCF**)
  - CVQ coding of *LE* residual (**CVQLE**)
  - CVQ coding of *CF* residual (**CVQCF**)

The experiments were conducted using the whole training set of TIMIT database for training and the whole testing set of TIMIT for testing. A sequence of LSF vectors (10 LSFs/frame) was extracted using analysis frames of 25ms at a rate of 50 frames/sec (5 ms overlap). For each scenario, all available X-space and Y-space features were collected from the LSF sequence, excluding silent frames. The AR filter was computed from the full narrowband (0-4 kHz) signal with the autocorrelation method using preemphasis ($\mu = 0.95$). The Spectral Distortion measure that is used is given by:

$$\mathcal{D}(X_t, \tilde{X}_t) = \frac{1}{\pi} \int_0^{\pi} \left(20 \log_{10} \frac{|X_t(e^{j\omega})|}{|\tilde{X}_t(e^{j\omega})|} \right)^2 d\omega \tag{8.18}$$

where $|X_t(e^{j\omega})|$,$|\tilde{X}_t(e^{j\omega})|$ is the original spectrum and the reconstructed spectrum respectively. Simple averaging was used for the evaluation over the test-set.

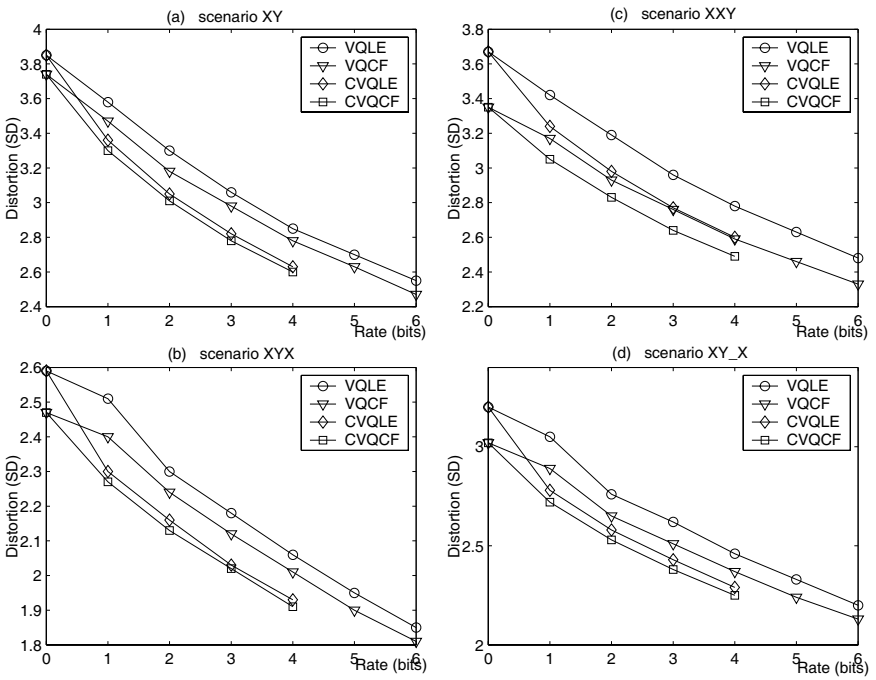


Figure 8.5 CSI Rate-Distortion curves for each scenario and each CSI method. Note that *CVQLE* and *CVQCF* uses M=256 X-space classes.

The linear system that has to be solved for *CF* training [172], is ill-conditioned in scenarios XXY, XYX, XY_X, where X-space has 20 dimensions. A dimensionality reduction via PCA (Principal Component Analysis) to the 18 strongest dimensions was used to avoid the ill-conditioning. This indicates the existence of redundancy in X-space. In all scenarios, the *CF* estimator had 128 Gaussians, and CVQ had M=256 input space codevectors. The size of the Y-space codebook in CVQ was constrained to have at most 4096 vectors. Therefore, $K = \{2^1, 2^2, 2^3, 2^4\}$ $Y|X$-space classes.

8.4.4 Results

The experiment results are shown in Figure 8.5. Rate-Distortion measurements are plotted for each scenario and each CSI method (*VQLE*, *VQCF*, *CVQLE*, *CVQCF*). Since each CSI method is associated with an estimation method, it is convenient to represent the estimator performance as the performance of the corresponding CSI scheme at the rate of 0 bits/frame (no FEC transmission). This allows a direct comparison of CSI techniques and estimation methods, in terms of distortion.

Regarding estimation methods, *CF* outperforms *LE* in all scenarios, especially in scenario XXY, where 3.35 dB were obtained. These results are similar to those presented in [167]. Having the performance of *CF* in scenario XXY as a reference,

jitter buffer provides an improvement of 0.35-0.90 dB when *CF* estimation is used.

In all scenarios, distortion can be significantly reduced with a few bits. Regarding CSI techniques, it is clearly seen that VQ-based residual coding can benefit from a better estimator, i.e. *VQCF* outperforms the widely used *VQLE* at least 0.5 bit, while in scenario XXY the gain is greater than 1 bit. The clear advantage of *VQCF* over *VQLE* in scenario XXY suggests using a "predictive" VQ technique based on *CF* estimation for "transparent" residual coding of LSFs [173]. On the contrary, CVQ-based residual coding is less dependent on the estimator and provides similar performance for both estimators in all scenarios except XXY. Furthermore, CVQ always benefits from the available mutual information between the residual and the side information, providing an improvement of 1 bit over the widely used *VQLE*, and a gain of 0.75-1 bit over *VQCF*.

For single packet losses, just 4 bits/frame of FEC data encoded with *CVQLE* provide a 42% distortion reduction (-1.42 dB) over the best "predictive" estimation (3.35 dB using *CF* in scenario XXY), and a 21% distortion reduction (-0.54 dB) over *CF* estimation in scenario XYX. For double packet losses, Figure 8.5d shows only the distortion from the reconstruction of the first lost vector. Note, that the recovery of the second lost vector is made from the *reconstructed* first vector as stated in Section 8.4.1. However, our measurements showed that when this *cascaded* form of CSI recovery is made, the second lost vector is reconstructed with less distortion than the first lost vector. Therefore, double packet losses can be recovered with at least 25% distortion reduction (-0.75 dB) over *CF* estimation, and at least 32% distortion reduction (-1.10 dB) over the best "predictive" estimation, using *only* 4 additional bits.

Since both CVQ-based methods have the same performance, and *CVQLE* is more simple than *CVQCF*, we chose *CVQLE* for informal subjective testing, assuming a 2 frame (40ms) jitter buffer, and using 4 bits/frame of FEC data for scenario XYX and 4 bits/fra-me for scenario XY_X. These 8 bits/frame of FEC data were used to recover from 1 or 2 packet losses as stated in Section 8.4.1. LSFs were computed from the speech signal, according to Section 8.4.3, and speech was inverse filtered using the original AR filter parameters. An amount of 5%-25% losses was introduced to the LSF vector sequence, constricted to generate either 1 or 2 sequential losses. The proposed *CVQLE* methods were compared to simple interpolation. Speech signal was then synthesized from the original excitation and the reconstructed LSFs. Informal listening tests showed that envelope related artifacts were fewer and milder with *CVQLE*.

The results from the reported subjective tests show that artifacts related to spectral envelope distortions can be efficiently removed based on the proposed approach. For speech codecs that rely explicitly on the use of an excitation signal (e.g., CELP-based coders), additional tests should be conducted including the coding of the excitation signal. Obviously, in this case a deterioration of the obtained quality is expected. On the other hand, the spectral envelope information is very important for the quality of the reconstructed signal for speech coders based on the sinusoidal

representation [1] where the excitation signal is obtained through a phase model that is based on the spectral envelope information.

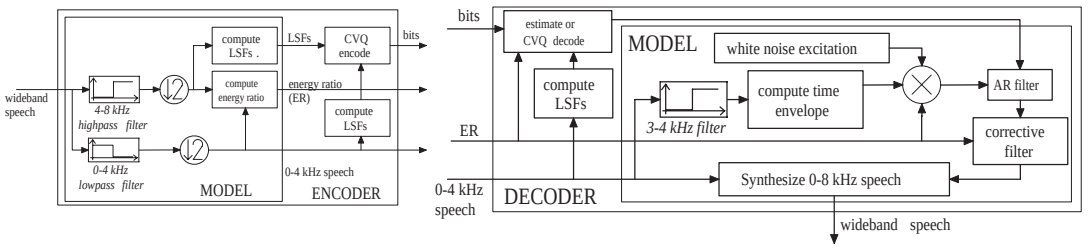
8.5 Speech Spectrum Expansion

The problem of BandWidth Expansion (BWE) has gained attention as a cost effective way to enhance narrow-band speech into wideband. The main assumption is that narrow-band (NB) speech contains enough information for the reconstruction of the missing highband (HB) frequencies. Another assumption is that the listener does not need an exact reconstruction of the lost frequencies but a perceptually valid one. Consequently, many researchers try to estimate the lost information from the transmitted information [158], [159], [160], [161], [164]. Narrow-band features, like spectral envelopes under several parameterizations, pitch, voicing, zero-crossing etc, have been extracted from the narrow-band speech signal and used for the estimation of a highband features. The highband is then reconstructed from these features, usually an LSF (Line Spectrum Frequencies) spectral envelope [29] and a gain parameter. The highband excitation is often an altered form of the narrow-band excitation [161] or modulated white noise [174]. Reconstructed speech suffers from artifacts like whistling sounds and crispy sounds whose nature is associated with the employed excitation. These artifacts disappear if the highband LSF are encoded with a few bits. However, the distortion at which this happens is significantly lower that the distortion resulting from the estimation. Therefore, it seems that a high quality reconstruction of the highband cannot be based solely on estimation.

This section investigates the expansion of the narrowband speech spectrum to wideband and proposes a split-band approach to wideband speech coding similar to the one proposed in [174]. The focus is given to the highband spectral envelope and CVQ is used to benefit from the correlations between the two bands. The proposed SSE system demonstrates that the bit-rate requirements for high quality bandwidth expansion are actually very low and it constitutes the basis for a wideband speech codec that needs only slightly more bit-rate than the corresponding narrowband speech codec.

8.5.1 The Expansion System

The expansion system consists of a highband encoder and a highband decoder. The encoder (Figure 8.6a) receives the wideband speech signal and splits it in two bands, the 0-4 kHz narrowband and the 4-8 kHz highband. The narrowband signal is treated independently of the highband and it may be quantized, while the highband is modelled and coded for the transmission. The highband model consists of the energy ratio (in decibel) between the two bands and 10 LSF parameters describing a 10-th order AR filter. The LSF parameterization is preferred for it's useful coding and stability properties [1]. At the decoder (Figure 8.6a), the wideband signal is resynthesized from



(a) SSE encoder **(b)** SSE decoder

Figure 8.6 The Speech Spectrum Expansion system.

the transmitted narrowband signal and the reconstructed highband signal, according to the energy ratio and the highband LSF.

The encoder (Figure 8.6a) computes the highband parameters from the sub-sampled highband. The sub-sampling flips the highband spectrum into the 0-4 kHz spectrum. The AR polynomial is computed using the autocorrelation method and it is transformed to LSF parameters. The highband LSF describe the flipped 4-8 kHz spectrum and can be flipped back using the following formula [174]:

$$LSF(i)_{flip} = \pi - LSF(P - i + 1),$$

where $P = 10$ is the order of the AR filter. The flipped highband LSFs are then CVQ-encoded using the narrowband LSF and $log_2(K)$ bits. The decoder (Figure 8.6b) computes the narrowband LSF and uses the transmitted CVQ information to decode the highband LSF or drops the CVQ bits and estimates the highband LSF with one of the estimators seen in Section 8.3.

The highband signal is reconstructed by exciting the 10-th order AR filter with modulated white noise. The modulation is done with the time envelope of the 3-4 kHz transmitted narrowband signal. The synthesis is not done via OLA (OverLap Add), since OLA in the case of noise synthesis -our case- may introduce audible fluctuations [175], but with a variable lattice filter and sample by sample interpolation of the reflection coefficients.

If the highband envelope is excited with unmodulated white gaussian noise, the reconstructed wideband speech contains an unnatural noisy sound. Modulating with the time envelope removes this artificial sound and provides a high quality highband. The time envelope gives a pitch dependent temporal structure and thus a phase information to the white noise. The noise is better integrated when the noise bursts are concentrated around pitch closure instants [163].

The time envelope is computed by filtering the absolute value of the 3-4 kHz speech signal with a simple lowpass filter of 300 Hz cutoff frequency. When the highband spectral envelope is well estimated or coded, the modulation produces high quality wideband speech. To the contrary, highband envelope errors tend to be amplified due to errors in the excitation signal. This is caused by rapid amplitude variations of

the time envelope, mainly in unvoiced parts of speech. To cope with this, we follow a strategy similar to [174] and filter the time envelope with a lowpass variable filter controlled by a simple voicing criterion, based on the energy ratio.

8.5.2 Objective Evaluation

We conducted several experiments to evaluate the quality of the reconstruction of highband spectral envelopes using the previously presented estimators, CVQ and simple VQ. All experiments were conducted using the TIMIT database. LSF parameterization was used for representing the spectral envelopes in the low and in the high band using 14 and 10 size vectors, respectively. Each experiment involves the use of approximately 730,000 LSF vectors for training and about 270,000 LSF vectors for testing, while frames considered as silence were excluded from the training or the testing corpus. A pre-emphasis filter with $\mu = 0.95$ was applied on the narrow-band signal. The length of the analysis window was set to $30ms$. Voicing decisions -when needed- were made according to the energy ratio between the narrowband and the highband. As an objective metric, we used the symmetric Kullback Leibler (SKL) distance given by:

$$d_{SKL}(P,Q) = \frac{1}{2\pi}\int_0^{2\pi}(P(\theta)-Q(\theta))\log\frac{P(\theta)}{Q(\theta)}d\theta \tag{8.19}$$

where $P(\theta)$,$Q(\theta)$ are the two power-normalized spectral envelopes. The SKL distance can also be seen as a weighted formant distance [176] and it seems to reflect the perceptual differences between AR spectra [177]. The SKL distance was chosen as an alternative to spectral distortion.

Figure 8.7 depicts the mean SKL distance of the presented estimators. The horizontal axis refers to the number of X-space classes used by the estimator. For example, the NLIVQ estimator has been tested for 16, 32, ..., 2048, 4096 classes, while the GMMCF estimator has been tested for 128 classes. Accordingly, a multiple estimator system with 2 GMMCF estimators (one for voiced frames and one for unvoiced frames) had 2*128=256 classes, and a voiced/semivoiced/unvoiced system had 384 classes. Results from the NLIVQ estimator are linked with a line to indicate the convergence of the estimator. The horizontal dotted line shows the mean SKL distance achieved when the highband is encoded with just 1 bit. From this figure, it is worthwhile to note that even the best estimator cannot provide 1 bit regarding the highband spectral envelope.

The performance of CVQ for 1,2,3 and 4 bits/frame and 128 classes for the X-space is shown in Figure 8.8, where we have also included the performance of simple Y-space VQ with 1...5 bits, and the performance of the previously mentioned estimators. Clearly, CVQ outperforms VQ. Notice that CVQ benefits more from the mutual information, as the number of bits, $\log_2(K)$, is increasing [1]. For CVQ with 1 bit/frame

[1] K is the size of each linked subcodebook

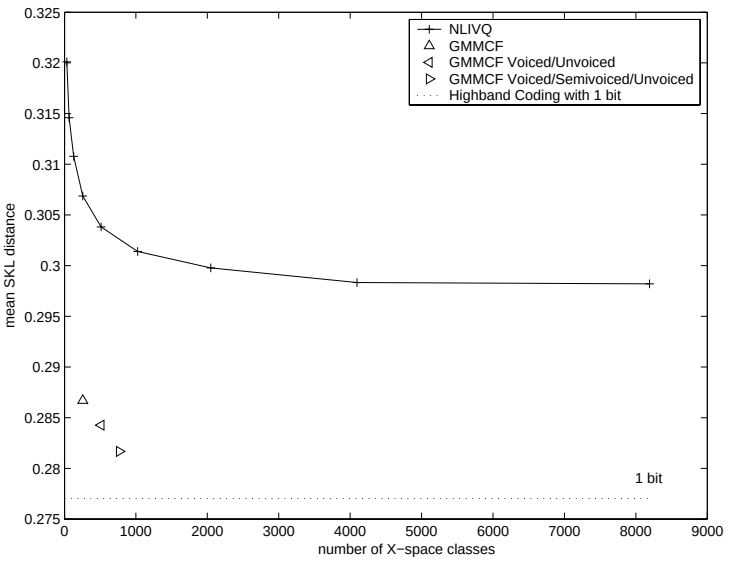


Figure 8.7 The performance (SKL mean distance) of a NLIVQ estimator and three GMMCF based estimators, in comparison with the SKL distortion of a simple highband VQ with 1 bit

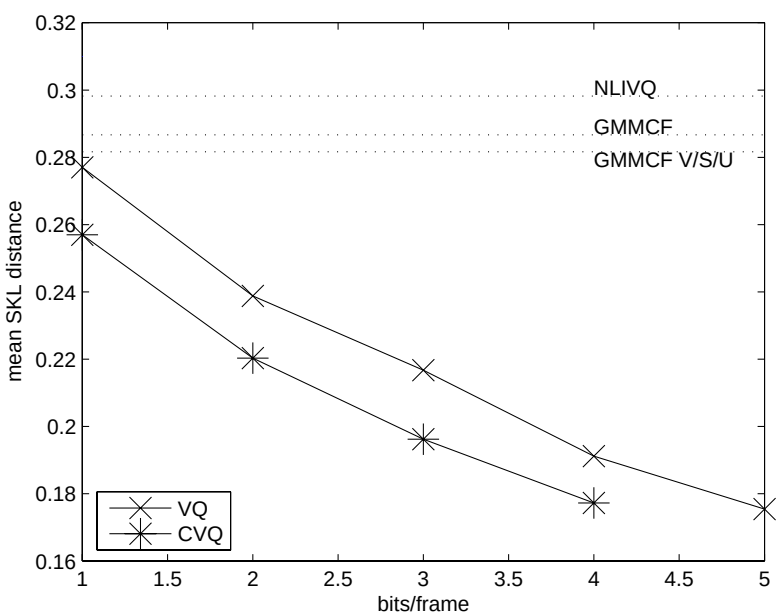


Figure 8.8 The performance of CVQ with 128 X-space classes, in comparison with the SKL distortion of a simple highband VQ with 1,2,3,4,5 bits. The performance of the estimators is indicated with horizontal lines.

the distortion is slightly below the distortion of VQ with the same rate. It is a slight improvement compared to the performance of the best estimator (nearly 1 bit/frame), but it is much better than the performance of the NLIVQ estimator. Note that the best estimator has extra voicing information and uses second order local statistics (covariances) to perform the mapping between X-space and Y-space. Therefore, CVQ can be directly compared with NLIVQ which is a special case of CVQ ($K = 1$). As coding rate $\mathcal{R}_{y|x} = \log_2 K$ increases, CVQ gains approximately 1 bit from the available mutual information, in terms of the SKL-based distortion. In relative terms, CVQ offers a 20% improvement over simple VQ.

8.5.3 Subjective Evaluation

The described speech spectrum expansion system was subjectively evaluated for the three following cases:

- original highband LSFs.
- estimated highband LSFs by NLIVQ with 128 classes.
- estimated/coded highband LSFs by the proposed method with 134 bits/sec.

The first two subjective tests were conducted in order to determine an upper and lower bound, respectively, for the subjective quality of the proposed CVQ-based method. For the third test we used 128 X-space classes and 4 bits/frame (1-to-16 mapping). Since the evolution of highband envelopes in time is relatively slow, and the human ear is insensitive to errors in these higher frequencies, a frame rate of 33.3 frames/sec (thus 134 bits/sec) was found to be sufficient. Note that all tests use the original energy ratio.

The evaluation was conducted with a DCR (Degradation Category Rating) test [1]. The subjects were presented with the *original wideband signal* and the reconstructed wideband signal, and were asked to vote the degradation of the latter according to the former. The DCR scale is shown in Table 8.2.

All the tests were conducted with PHILLIPS SBC-HP800 headphones and a SoundBlaster Extigy sound card, in an office environment. Listeners were initially presented with written instructions and one example for each of the gradings: 5,3,1. Each listener received a different randomization of the stimuli. The initiation of each stimuli was made automatically, but the listener also had the option to reinitiate the stimuli by clicking a button. A short tone preceded the stimuli initiation to prepare the listener for the initiation. The listener voted by clicking a button, and a new stimuli was presented to him. All stimuli were energy normalized to the same acoustic level.

For the first two tests, 29 listeners participated, and they were asked to vote for 41 utterances from test set speakers; 14 utterances for the NLIVQ estimator, 14 utterances using the original LSFs, a null set of 5 stimuli used to check the bias of

the listener [1], and 4 stimuli repeated for each test, to check if the listener had a consistent opinion. A few extreme cases of outlier listeners who obviously failed to the null set and to the repeated stimuli set, were excluded. The test for the CVQ scheme was conducted with 19 listeners, using 16 utterances from the test set, 4 repeated utterances and 5 null set utterances, under the very same conditions.

Table 8.3 states that using a synthetic excitation and the original LSFs produces a high quality wideband speech, almost indistinguishable from the original. The NLIVQ estimator did not perform well, as expected. The proposed scheme gets a very good DCR score which is close to the score obtained using the original LSFs. This shows that the highband envelope is well represented by only 134 bits/sec.

All the experiments in this section were conducted using the TIMIT database training set (738431 samples) for training, and the TIMIT test set (271366 samples) for testing, preemphasis at the narrowband, 30ms windows, 14 LSFs for the X-space, and 10 LSFs for the Y-space.

The proposed method can be used to construct a highband expansion coder. We found that it suffices to quantize the energy ratio with 6 bits and use 4 bit CVQ for the highband spectral envelopes. For a frame rate of 50 frames/sec, the highband expansion coder requires 0.5 kbps, while for a frame rate of 100 frames/sec the coder requires 1 kbps. The coder is evaluated in Chapter 9 where it is applied on quantized narrowband speech.

Description	Rating
Degradation is not perceived	5
Degradation is perceived but not annoying	4
Degradation is slightly annoying	3
Degradation is annoying	2
Degradation is very annoying	1

Table 8.2 DCR test scale.

Estimator	DCR score(95% CI)
NLIVQ estimator with 128 classes	3.59 (0.23)
CVQ with 4 bits/frame	4.41 (0.20)
ORIGINAL highband envelope	4.67 (0.15)

Table 8.3 Average DCR score (and 95% Confidence Interval) using the original wideband signal as reference.

Chapter 9

Harmonic Coding of Speech for VoIP

This chapter presents a narrowband speech codec, referred to as HMC (Harmonic Model Codec) that is based on the Harmonic-Model representation. The codec summarizes some of the results presented in the previous chapters to demonstrate the potential of Harmonic Models for speech coding. Each frame is encoded independently of the other frames, in order to recover instantaneously from a packet loss.

Two versions of the codec are developed, differing solely on the employed quantizers for the spectral envelope and the harmonic phases. The first version is referred to as HMC-SD (Harmonic Model Codec - Single Description) and it is a variable-rate single description codec that requires about 12.9-14.2 kbps on average with speech quality that is equivalent to iLBC. The second version, HMC-MD (Harmonic Model Codec - Multiple Description), is a multiple description codec that generates two descriptions for each frame using about 21 kbps on average. The codecs show to be robust upon packet losses.

9.1 Harmonic Model Analysis/Synthesis procedure

The codec receives the narrowband speech signal $x[n]$ at a sampling rate of 8000 samples/sec. The signal is analyzed with fixed 20 ms frames (160 samples) every 10 ms (80 samples) using a Hanning window. The synthesis is made using simple OLA (OverLap-Add) techniques. The analysis/synthesis OLA buffers are schematically shown in Figure 9.1. The encoder has an algorithmic delay of 15 ms: 10 ms for the analysis/synthesis window plus another 5 ms for the 30 ms window that is used for pitch detection purposes. The decoder holds a jitter buffer of future speech frames which are used for PLC purposes upon a signaled packet-loss. Each frame is encoded/decoded independently of the other frames. The PLC unit is used only when the current frame (the frame that describes the 10 ms of speech that should be send to the playout device) is not received. In that case, it synthesizes the current 10 ms of

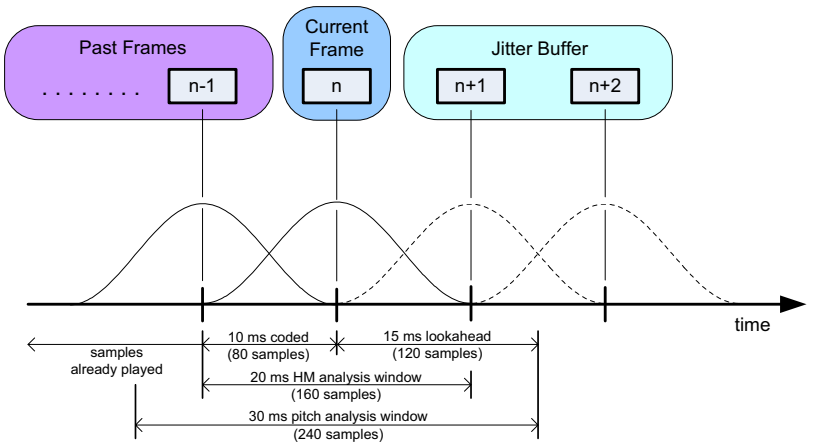


Figure 9.1 Analysis/Synthesis OLA buffers for the HMC codec.

speech using interpolation or extrapolation according to the PLC algorithm described in Chapter 5, and suitably updates the codec's internal states and OLA buffers. The encoding and the decoding procedure of a single frame is shown in Figure 9.3.

9.1.1 HMC Encoder

The encoder receives the narrowband speech signal $x[n]$ sampled with 8000 samples/sec and encodes each frame independently of the others. A schematic representation is shown in Figure 9.3a.

Pitch Detection and Voicing Detection

The Analysis-by-Synthesis pitch detection algorithm described in Section 2.5.2 is then used to determine the following parameters:

e**:** the energy of the frame (in decibel)

f_0**:** the pitch value in Hz.

VU**:** binary Voiced/Unvoiced decision

P_v**:** probability of voicing

The voicing decision VU is made according to the rules presented in Section 2.5.3 and uses the normalized SNR_{norm}, the zero-crossings ζ and the SNR when the amplitudes are sampled from a 16-th order RCC cepstral envelope that fits the harmonic amplitudes. The probability of voicing is computed according to the following equation:

$$P_v(SNR_{norm}) = \begin{cases} 0, & SNR_{norm} < 5 \\ \frac{SNR_{norm}-5}{5}, & 5 \geq SNR_{norm} \geq 10 \\ 1, & SNR_{norm} > 10 \end{cases} \tag{9.1}$$

The voicing decision VU is used for analysis/synthesis, while P_v is used solely for PLC. Note that the VU decision is biased towards declaring unvoiced frames as voiced, because quantizing a unvoiced frame as a voiced has no perceptual impact in HMC, while the opposite does. However, such a voicing decision cannot be used for PLC purposes. A *soft* voicing criterion, the voicing probability, is more suitable for PLC. In unvoiced frames, the pitch is set to a fixed value of $f_0 = 100$ Hz. We used the pitch detection algorithm that was proposed in Section 2.5. The pitch detection algorithm needs a lookahead of 15 ms for the initial "coarse search" step on the 30 ms speech frame.

The speech frames are classified in four categories which effect the quantization procedure:

silent: when the energy e of the frame is below -70 dB.

unvoiced: when VU=*unvoiced*. The pitch f_0 and the phase information ψ_k, τ are not encoded for these frames because unvoiced frames are reconstructed using random phases and fixed pitch $f_0 = 100$ Hz.

transitional: when VU=*voiced* and $P(SNR_{norm}) = 0$ (that is $SNR_{norm} \leq$5 dB).

voiced: when VU=*voiced* and $P(SNR_{norm}) > 0$ (that is $SNR_{norm} >$5 dB).

Harmonic Model analysis

The harmonic amplitudes and phases can be determined by solving the least-squares linear system described in Section 2.2. Only the harmonics below 3700 Hz are estimated. The output of the HM analysis unit is the following parameters:

A_k**:** K harmonic amplitudes

ϕ_k**:** K harmonic phases

Cepstral Envelope extraction

A 20-th order real cepstral envelope is computed from the harmonic amplitudes A_k, according to Section 2.4. The cepstral envelope is computed using Bark-scale with regularization constant $\lambda = 0.002$. The following parameters result from the Cepstral Envelope extraction unit:

RCC**:** 20 Real Cepstrum Coefficients describing the spectral envelope

Dispersion Phase extraction

Since the phases of unvoiced frames are not quantized, the dispersion phase is extracted only for transitional and voiced frames. The RCC cepstral envelope is sampled at the harmonics of f_0 to yield the phase response of the spectral envelope $\angle H_s(kf_0)$:

$$\angle H_s(kf_0) = -2\sum_{p=1}^{20} c_p \sin(2\pi k f_0 p). \tag{9.2}$$

The harmonic phases ϕ_k, $k = 1, ..., K$ are then decomposed to the linear phase term $k\frac{2\pi f_0}{F_s}\tau$ and the dispersion phase term ψ_k:

$$\phi_k = k\frac{2\pi f_0}{F_s}\tau + \angle H_s(kf_0) + \psi_k, \tag{9.3}$$

where F_s is the sampling rate (8000 samples/sec). The two phase terms are estimated according to the phase decomposition procedure described in Section 4.2. The translation term τ is between $[0, T_0]$, where T_0 is the fundamental period, and it is estimated with an accuracy of 7 bits. Summarizing, the output of the dispersion phase unit are the following parameters:

τ: the translation parameter

ψ_k: K dispersion phases

Encode Scalar Parameters

The parameters f_0, e, τ, VU, P_v are encoded with scalar quantizers. The pitch f_0 is quantized to $\hat{f}_0$ with 8 bits using a codebook trained with pitch samples from TIMIT train-set. The same train-set was used to compute a codebook that quantizes energy e with 8 bits in the log-domain (in decibel). The translation parameter τ uses 7-bit uniform quantization in $[0, \frac{8000}{\hat{f}_0}]$. The VU decision and the voicing probability P_v are jointly encoded with 3 bits according to the Table 9.1. The pitch f_0 and the translation term τ is not encoded for unvoiced frames.

bits	VU	P_v	frame type
000	U	0	unvoiced frame
001	V	0	transitional frame
010	V	0.1	voiced frame
011	V	0.3	voiced frame
100	V	0.5	voiced frame
101	V	0.7	voiced frame
110	V	0.9	voiced frame
111	V	1.0	voiced frame

Table 9.1 Encoding of voiced/unvoiced decision VU, voicing probability P_v and frame classification using 3 bits.

Encode RCC and Dispersion phases

The 20 RCC parameters and the dispersion phases are encoded using single description quantizers for the HMC-SD codec and multiple description quantizers for the HMC-MD codec. The quantizers of the RCC parameters were based on GMM while the quantizers of the dispersion phase parameters were based on WGMM. Different rates are used for unvoiced, transitional and voiced frames. The tradeoff's between quality and rate are examined in Sections 9.2 and 9.3.

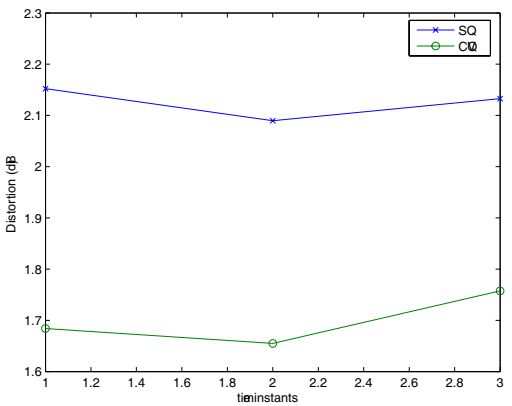
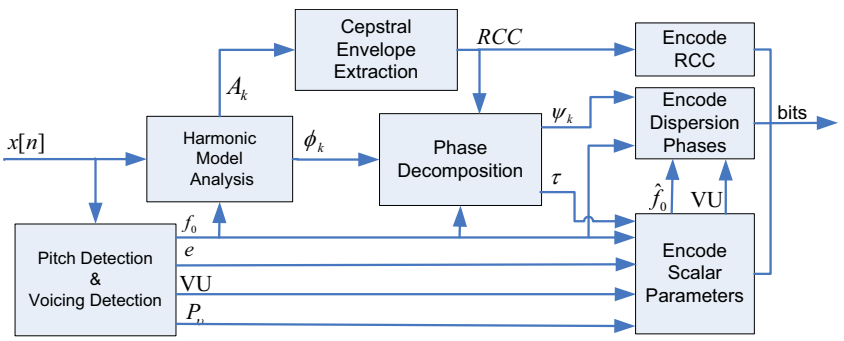


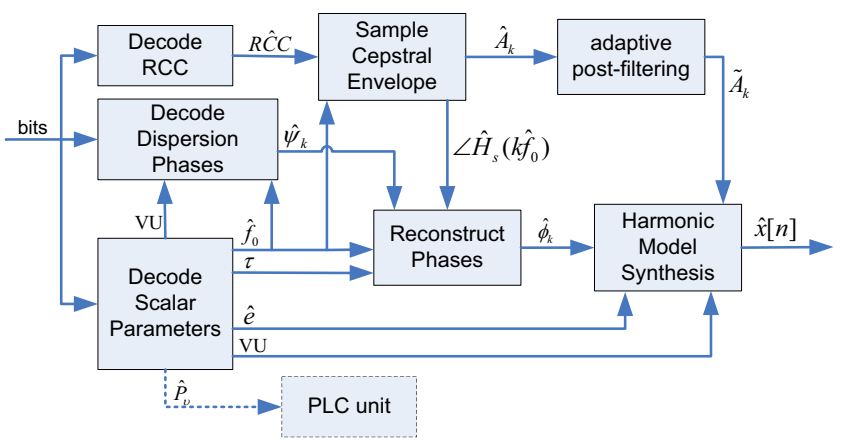
Figure 9.2 MSE for backwards predictive quantization of energy using scalar quantization (SQ) and scalar CVQ of the linear prediction residual. Time instant i refers to the i-th past frame that is recursively quantized.

Side Information for PLC

The performance of the PLC unit can be enhanced by introducing a small *corrective bitstream* with the necessary side information. We found that the most important parameter for the PLC unit is the frame energy, and so we propose a backwards predictive quantization scheme for the energy e. Therefore, for each transmitted packet (that may contain more than one speech frames) we quantize the energy of the previous speech frames using the last received frame energy as side information. This is made in an recursive manner: the energy of the $(n-1)$-th frame is quantized using the energy of the n-th frame as side information. Then, the energy of the $(n-2)$-th frame is quantized conditioned on the energy of the $(n-1)$-th frame, and so on. The procedure is repeated until we quantize a predefined number of past frame energies. This information is useful when packets are lost at the boundaries of spoken words, where interpolation of the energy may lead to unnaturally lengthened beginnings (i.e. onset frames) or endings. For a jitter-buffer of four frames, it makes sense to quantize 4 past energies, so that the reconstruction upon interpolation is made according to the recovered energy of the frames. The backwards predictive quantization is made using the scalar version of CVQ for the quantization of the prediction residual at 3 bits/frame. A total of 64 X-space classes are used for CVQ. Figure 9.2 shows the average quantization error of the energy when the prediction residual is quantized with scalar quantization (SQ) and with scalar CVQ. CVQ gains about 0.4 dB in distortion over SQ. Note that the energy is expressed in decibel, while the distortion is not. Assuming that we put two frames in one packet, the corrective bitstream requires 600 bits/sec.



(a) Encoder



(b) Decoder

Figure 9.3 Harmonic-Model Codec.

9.1.2 HMC Decoder

The decoder receives the bitstream and decodes each frame independently of the previous. The decoding is made according to the type of the frame (unvoiced, transitional, or voiced). A schematic representation is shown in Figure 9.3b. The following notation is used:

$\hat{e}$**:** quantized frame energy

$\hat{f}_0$**:** quantized pitch (in Hz)

τ**:** translation parameter (linear phase component)

VU**:** voiced/unvoiced decision

$R\hat{C}C$**:** quantized RCC parameters (20 dimensions)

K : the number of harmonics $K = \lfloor \frac{3700}{\hat{f}_0} \rfloor$.

$R\hat{C}C$**:** the quantized cepstral envelope

$\hat{c}_p$**:** the p-th coefficient of $R\hat{C}C$ (p=1,...,20)

$\hat{A}_k$**:** the K quantized harmonic amplitudes

$\tilde{A}_k$**:** the K quantized harmonic amplitudes after post-processing

$\hat{\psi}_k$**:** the K quantized dispersion phases

$\hat{\phi}_k$**:** the K reconstructed harmonic phases

$\angle\hat{H}_s(k\hat{f}_0)$**:** the phase response of the quantized spectral envelope $\hat{H}_s(\cdot)$ at the harmonics.

Silent and Unvoiced Frames

The frame is considered to be "silent" if the quantized energy $\hat{e}$ is below -70 dB. Silent and unvoiced non-silent frames are reconstructed in a similar manner. The spectrum is reconstructed with harmonics of fundamental frequency $\hat{f}_0 = 100$ Hz, up to 3700 Hz. The amplitudes $\hat{A}_k$ are sampled from the quantized RCC cepstral envelope according to the following formula (see eq. (2.20)):

$$\hat{A}_k = exp\left(2\sum_{p=1}^{20}\hat{c}_p\cos(2\pi k\hat{f}_0 p)\right), \tag{9.4}$$

where $\hat{c}_p$, $p = 1, ..., 20$ is the p-th cepstral coefficient. The harmonic phases $\hat{\phi}_k$ are set to be random, uniformly distributed in $(-\pi, \pi]$. Prior to synthesis, the amplitudes

$\hat{A}_k$ are post-filtered at the frequency domain, using the post-filtering technique of Section 2.4.2. The synthesis is made with the post-filtered amplitudes $\tilde{A}_k$.

Transitional and Voiced Frames

Transitional and voiced frames are reconstructed the same way, but at different bit rates. The quantized RCC cepstral envelope is sampled at the harmonics of $\hat{f}_0$ to get the amplitudes $\hat{A}_k$ (see eq. (9.4)) and the phase responce $\angle\hat{H}_s(k\hat{f}_0)$ of the minimum phase cepstral envelope (see eq. (9.2)).

The quantized dispersion phases $\hat{\psi}_k$ and the translation parameter τ are composed to yield the harmonic phases $\hat{\phi}_k$ according to equation (9.3). The harmonic amplitudes are post-filtered prior to synthesis as made for the silent and the unvoiced frames, using the adaptive post-filtering method presented in Section 2.4.2. The post-filtering reduces a slight "loss of presence" effect that the synthesized speech has. Note that the quantized voicing probability $\hat{P}_v$ is used only by the PLC algorithm, in the case of a packet loss.

Packet Loss Concealment

Packet losses are handled by the PLC algorithm proposed in Chapter 5. The algorithm uses the jitter buffer to perform interpolation when a future speech frame is available and extrapolation when the jitter buffer is empty. The harmonics are classified as voiced and unvoiced and a different type of synthesis is used for each case. When interpolation is used, the harmonics of the start-frame are linked to the harmonics of the end-frame via a pair-matching or a death-birth procedure. The paired sinusoids, which may not be harmonically related anymore, are synthesized according to their voicing state at the start-frame and the end-frame. The interpolation between two voiced harmonics is made using a cubic phase model. The PLC can also be assisted by a small corrective bitstream that holds energy information regarding past frames, as it is described in Section 9.1.1.

9.2 HMC-SD: Single Description Quantization

This section describes the single description version of HMC (HMC-SD). GMM-based quantization is used to encode the RCC cepstral envelope, while the phases are encoded with the WGMM-based quantization scheme that is proposed in Section 4.6. Both schemes are trained with the TIMIT training set using a relatively low number of components; 16 Gaussians for the RCC and 32 Wrapped Gaussians for the phases. The WGMM-based quantization was made using PCF-based wrapped Gaussian quantizers and the corresponding empirical bit-allocation algorithm, presented in Section 4.5.2.

Three different instances of the HMC-SD are evaluated. The three instances differ solely on the allocation of bits to the RCC and phase quantizers, for each frame class (silent, unvoiced, transitional, voiced). At all instances, the frame refresh rate is 100 Hz, corresponding to one frame every 10 ms. The bit-allocation for the three codec instances HMC-SDa, HMC-SDb, HMC-SDc and for each frame type is

summarized in Table 9.2. An evaluation of the average rate is made using 64 male utterances and 64 female utterances from TIMIT test-set. The average rates per utterance are presented in Table 9.2, along with standard deviations.

CODEC	HMC-SDa			HMC-SDb			HMC-SDc		
Parameters	RCC	ψ_k^{low}	ψ_k^{high}	RCC	ψ_k^{low}	ψ_k^{high}	RCC	ψ_k^{low}	ψ_k^{high}
Silent	20	0	0	20	0	0	20	0	0
Unvoiced	60	0	0	60	0	0	50	0	0
Transitional	50	70	30	50	60	20	50	50	17
Voiced	60	70	30	60	60	20	50	50	17
Average (all)	14191 (1076)			12926 (898)			11337 (763)		
Average (males)	14698 (1074)			13280 (927)			11638 (782)		
Average (females)	13684 (811)			12572 (715)			11032 (612)		

Table 9.2 Bit-allocation for the RCC parameters, the lower frequency phases ψ_k^{low} and the high frequency phases ψ_k^{high}, for three HMC-SD instances. The average rates for all test-set utterances, male and female speakers are also included along with the standard deviation.

The codecs are objectively evaluated in terms of PESQ-MOS (Perceptual Evaluation Subjective Quality - Mean Opinion Score) [5]. PESQ is an ITU standardized algorithm that predicts the MOS (Mean Opinion Score) of the quantized speech. The MOS score is the sample average of the so-called ACR (Absolute Category Rating) scale ([1] pg. 476) which measures the subjective quality of speech according to Table 9.3. The evaluation was made for the following cases:

AS: Harmonic Model Analysis/Synthesis

AS-RCC: Harmonic Model Analysis/Synthesis with amplitudes derived from a 20-th order RCC cepstral envelope. This system is a version of HMC-SD with unquantized parameters.

iLBC: internet Low Bit-Rate Codec (20 ms version, 15.2 kbps, fixed rate)

HMC-SDa: codec instance A (14.2 kbps, variable rate)

HMC-SDb: codec instance B (12.9 kbps, variable rate)

HMC-SDc: codec instance C (11.3 kbps, variable rate)

The proposed HMC-SD codecs are compared to iLBC [40], a narrowband codec that also encodes speech in independent packets which contain 20 ms or 30 ms of speech. The 20 ms version of iLBC requires a fixed-rate of 15.2 kbps, while the 30 ms version requires 13.33 kbps. The latter codec is chosen as a plausible competitive choice to HMC-SD because it operates at similar rates, it also has the packet independence property and it is narrowband.

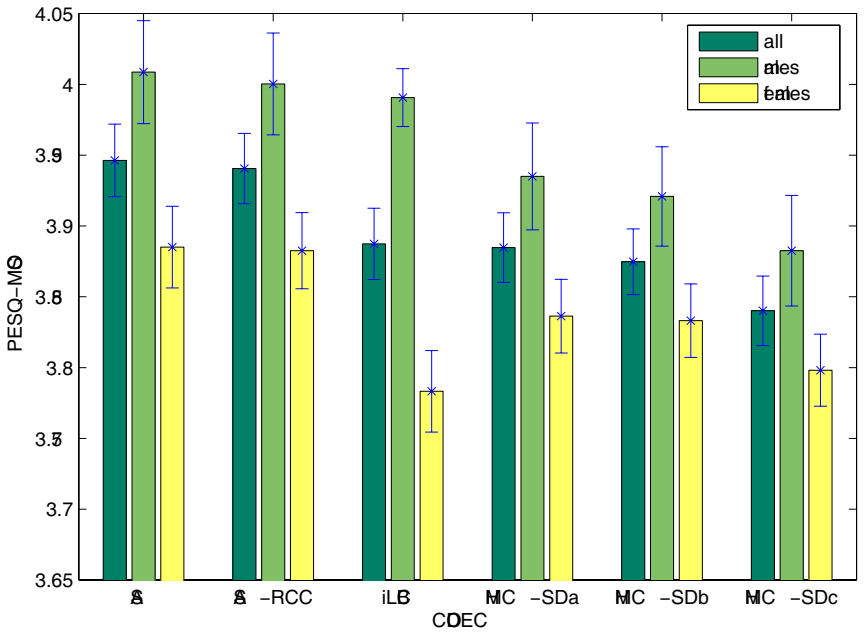


Figure 9.4 PESQ-MOS evaluation (mean and 95% confidence interval) of the Single Description HM codec, iLBC and the analysis/synthesis systems.

The measurements were made with the same 64 male utterances and 64 female utterances from TIMIT test-set used for the evaluation of the average rate. The results are plotted in Figure 9.4. We can observe that the AS system delivers high quality speech of PESQ-MOS equal to 3.95 and that sampling the harmonic amplitudes from the RCC cepstral envelope does not effect the PESQ-MOS. All codecs perform better for male speakers than for female speakers, while the deviation between the two genders is higher for iLBC. The HMC-SDa and HMC-SDb have similar performance to iLBC. The low bitrate instant HMC-SDc is worse than all other cases.

Description	ACR Rating
Excellent	5
Good	4
Fair	3
Poor	2
Bad	1

Table 9.3 Absolute Category Rating (ACR) scale of subjective speech quality.

9.3 HMC-MD: Multiple Description Quantization

This section describes the multiple description version of HMC (HMC-MD). The only difference of HMC-MD from HMC-SD is that the former encodes the RCC envelopes and the dispersion phases with multiple description quantizers. The other parameters are quantized as in HMC-SD with single description quantizers and they are repeated to each side description packet. Furthermore, HMC-MD uses the same GMM and WGMM that were used in HMC-SD. In MDC-MD, the RCC envelopes are quantized with the GMM-based MDC quantization algorithm (GMM-MDSQ$_{TC}$) described in Section 6.3, while the dispersion phases are quantized with the WGMM-based MDC scheme proposed in Section 6.4. The RCC quantizer GMM-MDSQ$_{TC}$ was restricted to a maximum of 4 bits/dimension per side description in order to limit the number of precomputed MDSQ codebooks. Alternatively, similar central/side distortion tradeoffs can be obtained with the GMM-MDTC scheme that is proposed in Section 7.2. The GMM-MDSQ$_{TC}$ was chosen for two reasons: first, it adapts instantly to different rates, second, the precomputed MDSQ codebooks are shared with the WGMM-based phase quantization scheme.

Two different instances of the HMC-MD are evaluated. The instances differ solely on the allocation of bits to the RCC and phase quantizers. As in HMC-SD, the frame refresh rate is 100 Hz, corresponding to one frame every 10 ms. Each frame is then encoded in two descriptions, yielding a total rate of 200 descriptions/sec. The bit-allocation for the two HMC-MD instances HMC-MDa, HMC-MDb and for each frame type is summarized in Table 9.2. The rates in Table 9.2 correspond to the rate allocated for both descriptions. An evaluation of the average rate is made using the same test-set as HMC-SD (64 male and 64 female utterances). The average rates per utterance are presented in Table 9.4, along with standard deviations.

CODEC	HMC-MDa			HMC-MDb		
Parameters	RCC	ψ_k^{low}	ψ_k^{high}	RCC	ψ_k^{low}	ψ_k^{high}
Silent	30	0	0	30	0	0
Unvoiced	90	0	0	80	0	0
Transitional	70	90	40	60	80	30
Voiced	90	90	40	80	80	30
Average (all)	20823 (1495)			18726 (1266)		
Average (males)	21484 (1511)			19231 (1308)		
Average (females)	20151 (1144)			18213 (995)		

Table 9.4 Bit-allocation for the RCC parameters, the lower frequency phases ψ_k^{low} and the high frequency phases ψ_k^{high}, for two HMC-MD instances. The average rates for all test-set utterances, male and female speakers are also included along with the standard deviation.

The evaluation of a multiple description codec is not as straight-forward as the

evaluation of a single description codec. The quality of quantized speech depends on the total rate allocated to each description, on the central/side description tradeoff, and on the channel conditions. We conducted some experiments to investigate the behavior of the two proposed multiple description codecs. The evaluation is made in terms of PESQ-MOS. Three different cases are examined for each codec:

Central Description: both descriptions are received for each frame.

Side Description 1: only the first side description is received for each frame.

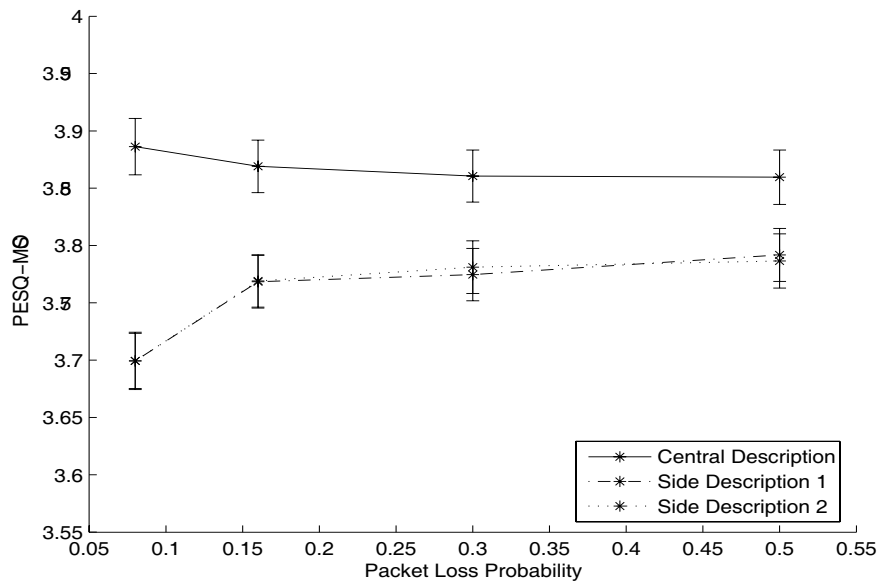
Side Description 2: only the second side description is received for each frame.

The first case provides an upper bound to the quality that can be achieved by perfect channel conditions. When the channels introduce packet losses, a large percentage of frames will be reconstructed with only one side description. For example, if we assume that the descriptions are routed through two identical independent symmetric channels with 10% packet losses (packet loss probability $\rho = 0.1$), then 1% ($\rho^2 = 0.01$) of the frames will be totally lost, 18% ($2\rho(1-\rho) = 0.18$) of the frames will be recovered from a single side description, and 81% ($(1-\rho)^2 = 0.81$) of the frames will be recovered from both descriptions. For 20% packet losses, 32% of the frames will be recovered from one description only. At any case, speech is reconstructed from a mixture of high-quality central description frames and lower-quality side description frames. In fact, even for moderate loss rates, a considerable portion of frames will be reconstructed from one description only. Therefore, it is interesting to evaluate the quality of speech that a single description provides as well as the quality obtained by the central description.

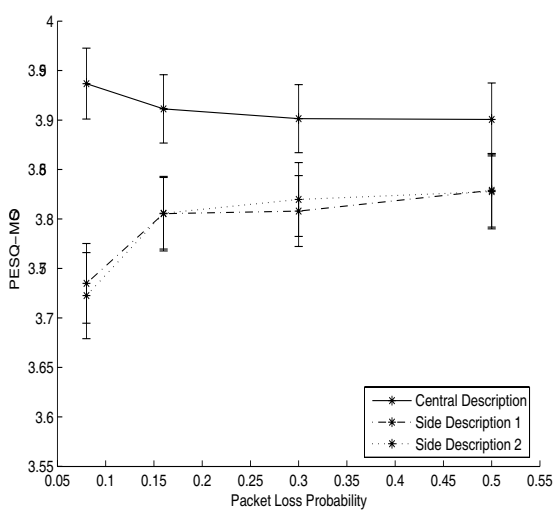
The operation of HMC-MD is controlled by the central/side description tradeoff that is selected for the operation of the GMM-based and the WGMM-based multiple description quantizers. As we saw in Chapters 6 and 7, the central/side description tradeoff points are controlled by the packet loss probability ρ that was used to train the MDC quantizer. The two instances HMC-MDa and HMC-MDb are evaluated for the central/side description tradeoffs that correspond to packet loss probabilities $\rho = \{0.08, 0.16, 0.3, 0.5\}$. Note, however, that the MDC quantizers are train to be optimal in MSE sense which is not necessarily optimal in a perceptual sense. Therefore, the tradeoffs obtained in $\rho = \{0.08, 0.16, 0.3, 0.5\}$ should be interpreted as possible *operating states* of the HMC-MD codec and not as the actual loss probabilities for which the codec is trained to perform optimally.

The results are depicted in Figures 9.5 and 9.6. We can observe that the PESQ-MOS difference between the central and the side reconstructions at $\rho = 0.08$ is 0.186 for the HMC-MDa codec and 0.235 for the (lower rate) HMC-MDb. At higher correlations, $\rho = 0.5$ the same difference is 0.09 for both codecs. Therefore, the operating state of the HMC-MD codec specifies the gain that we have when we receive both descriptions instead of one. Furthermore, the higher rate HMC-MDa is about 0.04 units better than HMC-MDb at all operating states, while it needs 2 kbps more on average.

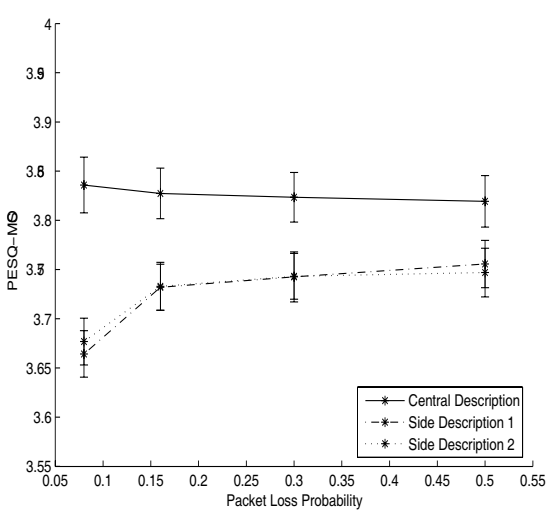
The purpose of these experiments is to demonstrate some of the central/side description quality tradeoffs provided by HMC-MD. The HMC-MD codec provides a flexible mechanism to control the quality of the central reconstruction over the quality of the side reconstructions. However, it is up to the VoIP system engineer to set the several control parameters that govern the relationship between the rate, the redundancy and the quality for the measured channel conditions.



(a) HMC-MDa: all speakers

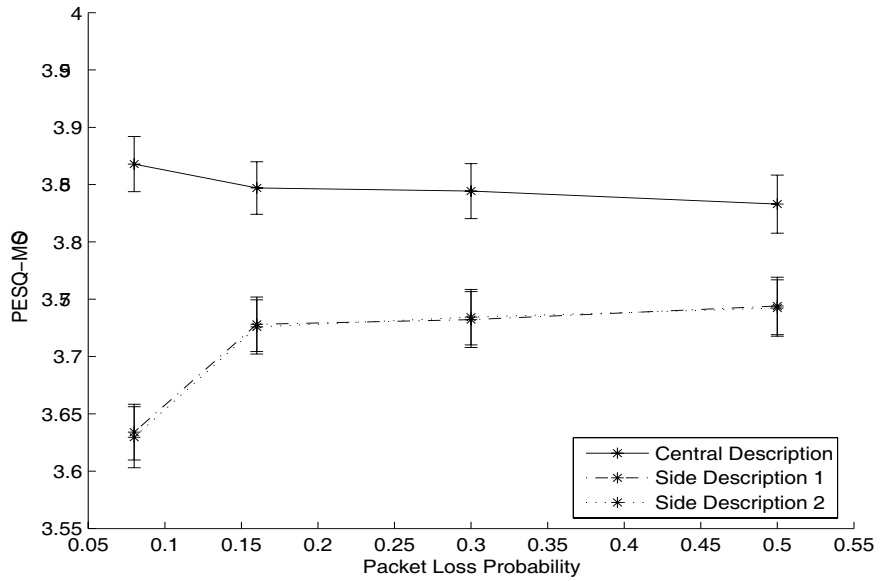


(b) HMC-MDa: males

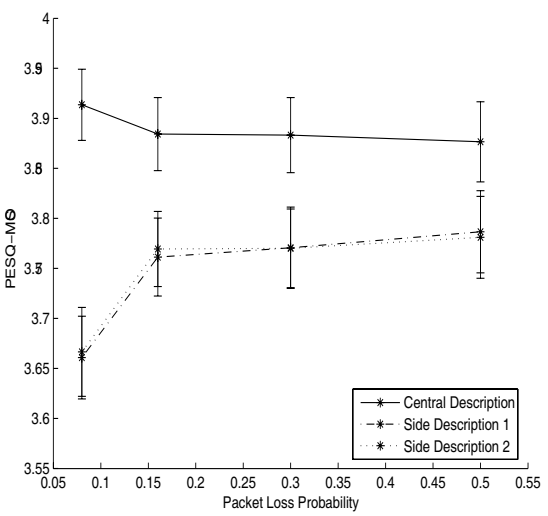


(c) HMC-MDa: females

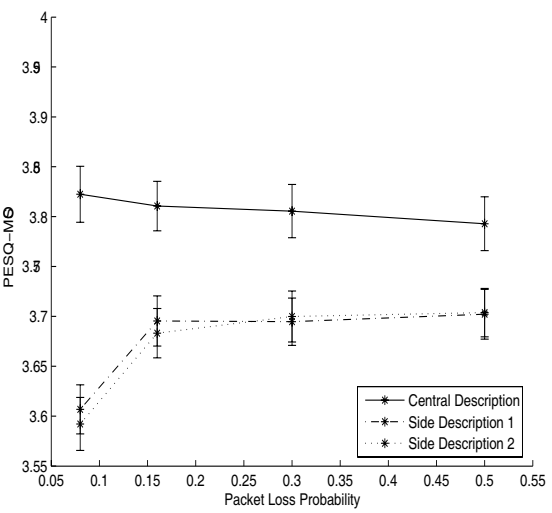
Figure 9.5 HMC-MDa codec Central and Side Description PESQ-MOS ratings (and confidence intervals) for loss probabilities 0.08, 0.16, 0.3 and 0.5.



(a) HMC-MDb: all speakers



(b) HMC-MDb: males



(c) HMC-MDb: females

Figure 9.6 HMC-MDa codec Central and Side Description PESQ-MOS ratings (and confidence intervals) for loss probabilities 0.08, 0.16, 0.3 and 0.5.

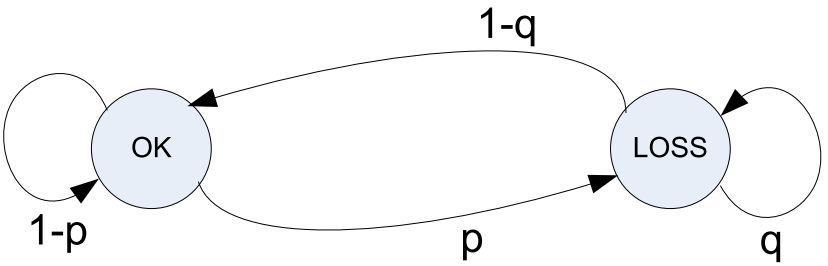


Figure 9.7 State diagram and transition probabilities for the Gilbert-Elliot model.

9.4 Subjective Evaluations

9.4.1 Quality of Quantization

The subjective quality of the HMC codecs is evaluated via a DCR (Degradation Category Rating) test. The listeners were presented with two stimuli, the original and the encoded speech signal, and were asked to evaluate the degradation of the perceptual quality that the quantization process introduced to the encoded signal. The degradation was graded according to the DCR scale, which is presented in Table 8.2. A total of 16 listeners participated on this test. The samples were randomly drawn from a small database that was constructed from 15 male and 15 female utterances from TIMIT test-set. Five codecs were evaluated: four HMC-based codecs HMC-SDa, HMC-SDb, HMC-MDa, HMC-MDb and iLBC. The MDC codecs were evaluated at the maximum correlation point $\rho = 0.5$. Each codec was graded about 80 times. All signals were lowpass-filtered and decimated to an 8 kHz sampling rate. The results are shown in Figure 9.8. The first three codecs, HMC-SDa, HMC-SDb and HMC-MDa have similar DCR ratings to iLBC (around 4.13). This observation is consistent with the PESQ-MOS evaluation made over these codecs. On the contrary, the fourth codec, HMC-MDb has a much lower DCR score of 3.89 although it uses only 2 kbps less than HMC-MDa.

Description	Rating
Degradation is not perceived	5
Degradation is perceived but not annoying	4
Degradation is slightly annoying	3
Degradation is annoying	2
Degradation is very annoying	1

Table 9.5 DCR test scale.

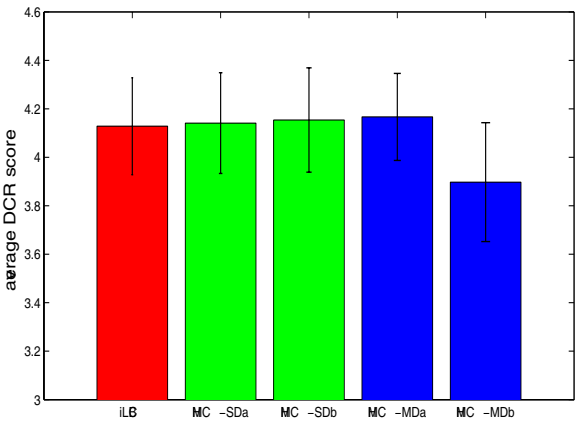


Figure 9.8 Subjective evaluation (mean and 95% confidence interval) of the HMC codecs and iLBC according to the DCR test.

9.4.2 Robustness to Packet Losses

This section makes a subjective evaluation of the robustness of HMC-SD and HMC-MD codecs to packet losses. The speech is quantized, transmitted through one or two simulated channels and reconstructed at the decoder using PLC. As a baseline, the iLBC codec is evaluated under the same conditions. The packets of the single description codecs iLBC, HMC-SD, are transmitted through a single channel, while the packets of the multiple description codec HMC-MD are transmitted through two independent channels.

The channels introduce packet losses according to the Gilbert-Elliot (GE) model. The Gilbert-Elliot model is a two state Markov model; the first state represents the case where the packet is received while the second state represents the case where the packet is lost. There are two degrees of freedom: the probability p of losing a packet given that the previous packet is received and the probability q of losing a packet given that the previous packet is lost. Figure 9.7 depicts the state diagram for the Gilbert-Elliot model. The GE model provides a easy way to simulate bursty losses in packet networks and reduces to a simple Bernoulli model when $p = q$. Although the behavior of IP networks is well approximated by high order Markov Models [21], the Gilbert-Elliot model provides a reasonably good approximation and it's simplicity make it a plausible choice for the study of the behavior of speech codecs under packet losses [23], [178]. In the experiments of this section we set the probability q of loosing a packet after a packet loss to be twice the probability p of loosing a packet when the previous packet is received. This constrain reduces the number of free parameters to one and provides a bursty nature to the packet loss process.

The single description codecs iLBC, HMC-SD are evaluated for packet losses of 5%, 10%, 20% and 30%, while the multiple description codec HMC-MD is evaluated for higher packet loss probabilites of 20%, 30%, 40%, 50%. We assume that each packet

of HMC-SD contains two frames (20 ms) and that each packet of HMC-MD contains two side descriptions from two consecutive speech frames. Therefore, there are 50 packets/sec for HMC-SD and 50 packets/sec per channel for HMC-MD. Respectively, the iLBC codec is evaluated using the 50 packets/sec mode (20 ms frames). A four-frame (40 ms) jitter buffer was used in both HMC-based codecs. The HMC-SD codec is enhanced with a 600 bps corrective bitstream that performs backwards predictive quantization of the energy of the previous frames according to Section 9.1.2. The HMC-SDa (14.2+0.6=14.8 kbps) and HMC-MDa (20.8 kbps) versions of the codecs were used for the experiments. HMC-MDa was tuned to operate at the highest correlation state (corresponding to a packet loss probability $\rho = 0.5$).

The subjective evaluation is made with a DCR (Degradation Category Rating) test. The listeners were presented with the quantized speech signal without packet losses and the reconstructed quantized speech signal with packet losses and voted the degradation of the latter compared to the former, according to the DCR scale (Table 9.5). Therefore, the experiments focus on the robustness of the codec to packet losses.

Each listener was presented with a stimuli that was randomly drawn from a database of stimuli. The database was constructed from 9 male and 9 female utterances from TIMIT database. Each utterance was encoded with iLBC, HMC-SDa, HMC-MDa and decoded with no packet losses and with 5%, 10%, 20%, 30% packet losses for the single description case and 20%, 30%, 40%, 50% packet losses for the multiple description case. Each utterance was decoded three times with different packet losses for each packet loss condition and for each codec. Therefore, the database consisted of 18*4*3*3 = 648 stimuli. Each listener was presented with 3 stimuli per channel condition and codec; a total of 3*3*4=36 stimuli. Twenty listeners participated to the experiment, therefore each codec at each channel condition received 60 votes. The experiments were made using Phillips SBC-HP800 headphones and a SoundBlaster Extigy soundcard.

The average DCR score from the experiments is shown in Figure 9.9, along with the corresponding confidence intervals. We can observe that the HMC-SDa outperforms iLBC at all evaluated channel conditions and that iLBC suffers a rapid degradation for 20% packet losses. In contrast, packet losses of 20% yield a degradation that is nearly not annoying for HMC-SDa. But HMC-SDa also degrades to a DCR score less than three 3 at 30% packet losses. However, the HMC-MD codec is capable of dealing with 30% packet losses where it provides a degradation that is perceived but not annoying. At the very high loss rates of 50%, HMC-MDa is capable of accepting 50% packet losses with slightly annoying degradation. Further insight is provided in Figure 9.10 where the distribution of the votes for each of the three codecs and each packet loss condition is shown.

The presented results demonstrate the effectiveness of the sinusoidal PLC scheme presented in Section 5. Sinusoidal PLC seems to work well for packet losses up to 20%, but it is not enough for 30% packet losses and redundancy is needed to cope with the increased losses. The redundancy can be introduced via an MDC framework

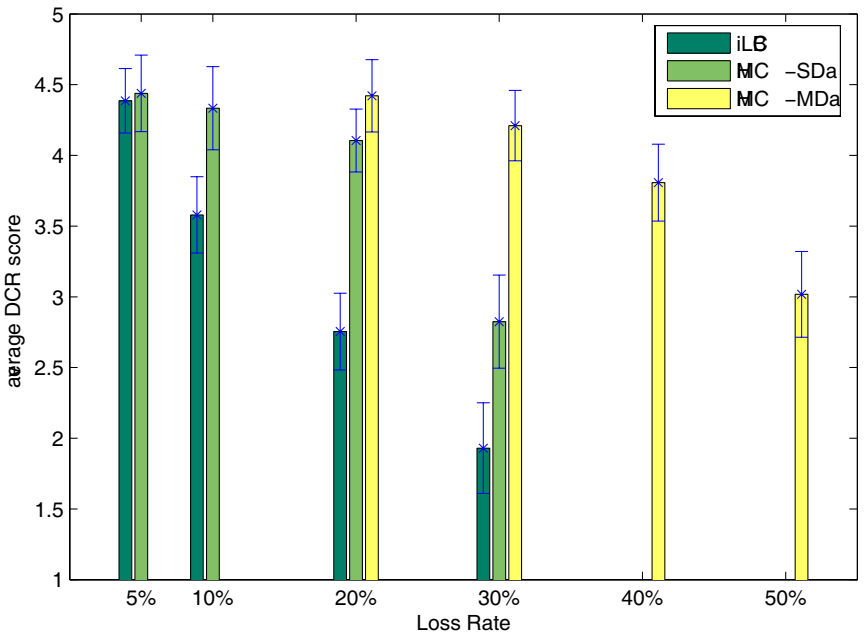
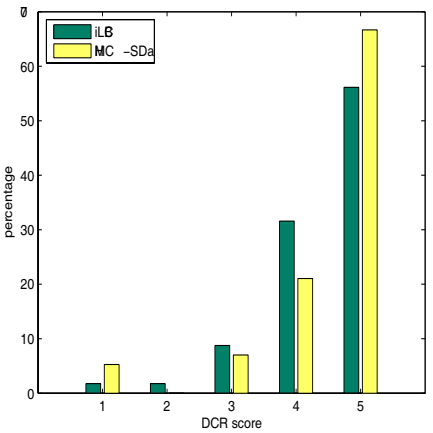
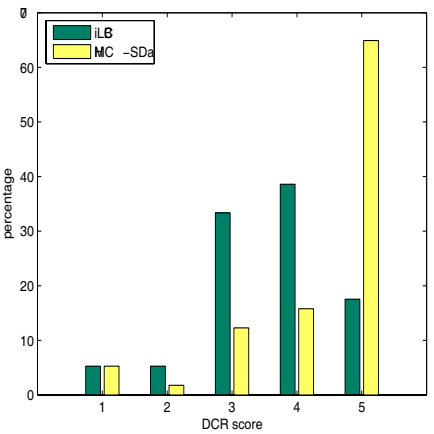


Figure 9.9 Average DCR score and 95% confidence intervals of iLBC, HMC-SDa and HMC-MDa codecs for several loss rates.

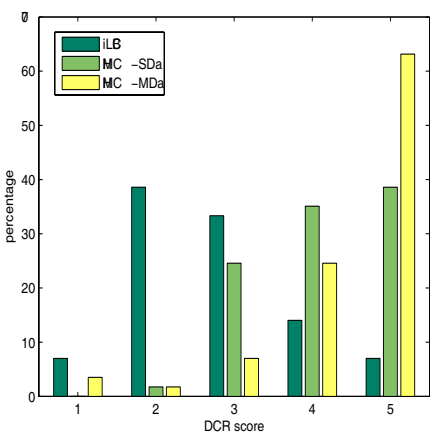
and the corresponding HMC-MDa codec is capable of dealing with higher losses of 30% and 40%. The increased robustness of the MDC codec can be attributed to the redundancy that is introduced and to the independence between the two channels that makes the loss of any information regarding a speech frame less probable.



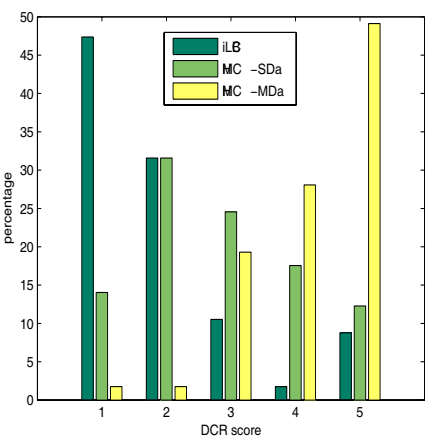
(a) 5% packet loss



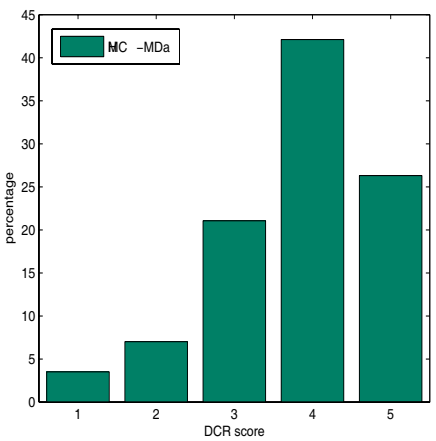
(b) 10% packet loss



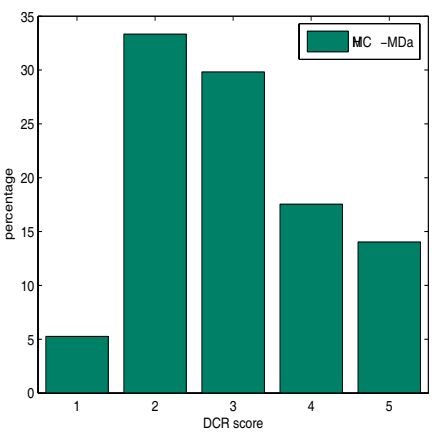
(c) 20% packet loss



(d) 30% packet loss



(e) 40% packet loss



(f) 50% packet loss

Figure 9.10 Distribution of votes for the iLBC, HMC-SDa and HMC-MDa codecs in the DCR scale.

Chapter 10

Discussion and Future Research Directions

10.1 Speech Coding for VoIP

This thesis presents a framework for the development of sinusoidal speech codecs suitable for VoIP. The effectiveness of the framework is demonstrated in two proof-of-concept codecs, a single description and a multiple description harmonic codec. A possible usage scenario would be to use the 14.2+0.6=14.8 kbps single description codec for packet losses up to 20% and the 20.8 kbps multiple description codec for higher packet losses of 30% to 40%. Obviously, the codecs are not as efficient as CELP codecs under no packet losses, but they can accept considerably more packet losses. Would it be better to take an efficient CELP codec and add channel coding? For example, the presented single description harmonic codec demonstrates a DCR-score about 4.0 for 20% packet losses at a rate of ≈15 kbps. The rate is 187.5% the rate of the 8 kbps G.729 codec, but at 20% packet losses, even if we transmit each packet of G.729 twice (with 16 kbps), the MOS score will go below 3 ("fair") [35] where a conversation cannot be held. A better strategy could be to use the low rate 5.3 kbps G.723.1 with a strong Reed-Solomon channel code at triple rate. Finding the optimal choice requires extensive experiments. This brings us back to the dilemma posed in the introduction of this thesis: efficient source coding with channel coding or redundant source coding with fine-tuned PLC.

This thesis follows the second option and allows redundant source coding for better PLC. The redundancy in the single description harmonic coder is the redundancy that naturally exists between frames. The 21 kbps multiple description harmonic coder introduces additional redundancy within the MDC context. Therefore, comparing these codecs with standardized codecs at 0% packet losses is unfair because both harmonic codecs perform joint source/channel coding of speech. However, they are not ideal joint source/channel speech codecs. Such a codec would operate like a predictive codec at 0% packet losses and would gradually become a packet independent codec at higher packet losses.

The potential of Harmonic Codecs is not fully exploited in this thesis and many improvements can be made. For example, when two or more frames are placed in a single packet, the first frame can be encoded independently of the previous but the following frames can be encoded predictively. There is an extensive literature on predictive coding schemes for spectral envelopes. Gain and pitch can also be encoded predictively. Finally, the dispersion phases can be encoded differentially using WGMM-based quantization for the difference between the dispersion phases of successive frames.

Another source of improvement is the incorporation of perceptual weighting to the bit-allocation procedure in dispersion phase quantization using one of the psychoacoustic models that are widely used in audio coding. Furthermore, the quantization process of the dispersion phases can be made with respect to a CELP-like analysis-by-synthesis weighted SNR criterion that shapes the quantization noise. Finally, the PCF-based quantizers for wrapped Gaussians can be extended to MDC in order to improve WGMM-based MDC. However, it is not obvious how to incorporate perceptual weighting and quantization noise shaping in a MDC codec.

Not all frames are of equal significance. The frames located in stationary parts of the speech signal can easily be concealed with high quality. The frames located in non-stationary parts usually result in an audible degradation after concealment. Therefore, these frames should be protected more with multiple description schemes if necessary. An adaptive sinusoidal codec could use MDC for the perceptually significant frames at packet losses around 15% and then gradually extend MDC to all frames at higher loss rates. This would operate like a gradual rate/robustness tradeoff mechanism.

Concluding, we suggest that the single description Harmonic Model Codec can be used in *small footprint Text-To-Speech* (TTS) systems. Today's high-quality TTS systems use a large corpus database that contains many speech segments and synthesize the spoken sentences by selecting an appropriate sequence of these segments. The segments are usually stored in a sinusoidal parametric form similar to the Harmonic Model used in this thesis [179], [76] and the ideas presented here can also be applied to the quantization of the corpus.

10.2 Speech Analysis

Harmonic Models are widely used in speech processing as a versatile tool for speech modeling. They are used for high quality modifications of the speech signal, like time-scaling and pitch-scaling and many commercial Text-To-Speech systems are based on harmonic models to concatenate small fragments of speech in order to synthesize a spoken sentence. From the signal processing point of view, harmonic models are a parametric representation of the signal that is lossy in the sense that it usually contains only a subspace of the signal (the set of harmonic sinusoids cannot perfectly reconstruct the waveform). However, analysis/synthesis using harmonic models pro-

vides speech nearly indistinguishable from the original, indicating that the discarded information is not perceptually important. Therefore, HM can be used as a frequency domain parameterization of the speech signal. In that aspect, HM offers the ability of performing complicated signal processing tasks on a parameterized frequency domain.

The properties of amplitude information in speech signals are well understood and there are numerous publications on the subject. A lot of work has also been made on the phase information of the speech signal. Unfortunately, handling phase information has an intrinsic difficulty arising from the fact that phase is defined on a modulo-2π space. This thesis makes a contribution to the stochastic modeling of phases: it proposes WGMM a stochastic model that is defined over modulo-2π spaces.

Although WGMM is successfully used for quantization purposes, its importance may extend beyond speech coding. The spectral envelope contains information regarding the formants of speech. The same information is also contained in the phase response of the spectral envelope because the latter is a minimum phase envelope. The dispersion phases are the harmonic phases after the removal of the minimum phase response of the spectral envelope and a linear phase term that aligns the signal to a reference point within the glottal cycle. Therefore, we can expect that dispersion phases contain mainly non-formant phase information regarding the speech signal in the sense that some formant-preserving information is already removed via the spectral envelope. *WGMM can be a useful tool for the exploration of the properties of this source of information.* WGMM can be incorporated in Hidden Markov Models and other stochastic models of speech. The unanswered question is what type of information do dispersion phases hold and where can we use it.

Bibliography

[1] Bastiaan W. Kleijn and K. K. Paliwal, *Speech Coding and Synthesis*, Elsevier, 1995.

[2] W.D. Voiers, "Diagnostic acceptability measure for speech communication systems," in *ICASSP*, 1977, pp. 204–207.

[3] W.D. Voiers, "Diagnostic evaluation of speech intelligibility," in *Speech Intelligibility and Speaker Recognition*, M. Hawley, Ed. Hutchinson and Ross, Stroudsburg, 1977.

[4] ITU-T Recomendation BS.1534 (06/01), "Method for the subjective assessment of intermediate quality level of coding systems," 2001.

[5] ITU-T, "Perceptual evaluation of speech quality assessment of narrowband telephone networks and speech codecs," February 2001 2001.

[6] S. Voran, "Listener ratings of speech passbands," in *IEEE Workshop on Speech Coding*, 1997, pp. 81–82.

[7] ITU-T Recomendation Q.23 (11/88), "Technical features of push-button telephone sets," 1988.

[8] Friedhelm Hillebrand, *GSM and UMTS: The Creation of Global Mobile Communications*, John Wiley and Sons, 2001.

[9] Kamran Etemad, *CDMA2000 Evolution: System Concepts and Design Principles*, Wiley and Sons, 2004.

[10] C. William Hardy, *VoIP Service Quality: Measuring and Evaluating Packet-Switched Networks*, McGraw-Hill, 2003.

[11] Martin Rainer and Steffan Gustafsson, "The echo shaping approach to acoustic echo control," *Speech Communication*, vol. 20, no. 3-4, pp. 181–190, 1996.

[12] M. Harteneck and R.W. Stewart, "Acoustic echo cancelation using a pseudo-linear regression and qr-decomposition," in *Proc. ISCAS*, 1996.

[13] I. P. Sound Global, "NetEQ: advanced jitter buffer and packet loss concealment module," 2006, (TM).

[14] Y.J. Liang, N. Farber, and B. Girod, "Adaptive playout scheduling using time-scale modification in packet voice communications," in *ICASSP*, 2001, vol. 3.

[15] R. Rodbro and Soren Holdt Jensen, "Time-scaling of sinusoids for intelligent jitter buffer in packet-based telephony," in *IEEE Workshop on Speech Coding*. 2002.

[16] Miroslaw Narbuttm, Andrew Kelly, Liam Murphy, and Philippe Perry, "Adaptive VoIP playout scheduling: Assessing user satisfaction," *IEEE Internet Computing*, vol. 9, pp. 28–34, 2005.

[17] ITU-T Recomendation G.114 (05/03), "One way transmission time," 2003.

[18] P.T. Brady, "Effects of transmission delay on conversational behavior on echo-free telephone circuits," *The Bell System Technical Journal*, pp. 115–134, 1971.

[19] S. Dimolitsas and J. Phipps, "Experimental quantification of voice transmission quality of mobile-satellite personal communication systems," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 2, 1995.

[20] J. Bolot, "Characterizing end-to-end packet delay and loss in the internet," *Journal of High-Speed Networks*, vol. 2, no. 3, pp. 305–323, 1993.

[21] H. Sanneck and G. Carle, "A framework Model for packet loss metrics based on loss runlengths," in *SPIE/ACM SIGMM Multimedia Computing and Networking Conference 2000 (MMCN 2000)*, 2000, pp. 177–187.

[22] M. Yajnik, S. Moon, J. Kurose, and D. Townsley, "Measurement and Modelling of the temporal dependence in packet loss," in *Proc, IEEE Computer Communications*, 1999, vol. 1, pp. 345–352.

[23] Moo Young Kim and W. B. Kleijn, "Rate-Distortion comparisons between FEC and MDC based on Gilbert channel Model," in *IEEE Int. Conf. on Networks (ICON)*, 2003.

[24] Wenyu Jiang and Henning Schulzrinne, "Comparisons of FEC and codec robustness on VoIP quality and bandwidth efficiency," in *ICN*, 2002.

[25] ITU-T Recomendation G.107 (12/02), "The E-Model, a computational Model for use in transmission planning," 2003.

[26] ITU-T Recomendation G.711 (11/88), "Pulse Code Modulation (PCM) of voice frequencies," 1989.

[27] ITU-T Recommendation G.726, "40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)," 1990.

[28] R. Deller Jr John H. L. Hansen John G. Proakis John, *Discrete-Time Processing of Speech Signals*, 1999, September.

[29] A. M. Kondoz, *Digital Speech: Coding for Low Bit Rate Communication Systems*, John Wiley and Sons, 2004.

[30] ITU-T Recomendation G.723.1, "Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s," 1996.

[31] ITU-T Recommendation G.729 (03/96), "Coding of speech at 8 kbits/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)," .

[32] K. Jarvinen, J. Vainio, P. Kapanen, T. Honkanen, P. Haauisto, R. Salami, C. Laflamme, and J. Adoul, "GSM Enhanced Full Rate Speech Codec," in *ICASSP*, 1997, vol. 2.

[33] 3GPP Recomendation TS 26.071, "AMR speech codec, general description," 1999.

[34] B. Bessette, R. Salami, R. Lefebvre, M. Jelinek, J. Rotola-Pukkila, J. Vainio, H. Mikkola, and K. Jarvinen, "The Adaptive Multi-Rate Wideband Speech Codec (AMR-WB)," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 8, pp. 620–636, 2002.

[35] Lefebvre Roch, Gournay Philippe, and Salami Redwan, "A study of design compromises for speech coders in packet networks," in *IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Montreal, Canada, 2004.

[36] Ingemar Johansson, Tomas Frankkila, and Per Synnergren, "Bandwidth efficient AMR operation for VoIP," in *IEEE Workshop Proceedings Speech Coding*, 2002, pp. 150–152.

[37] Mohamed Chibani, Philippe Gournay, and Roch Lefebvre, "Increasing the robustness of CELP-based coders by constrained optimization," in *ICASSP*, 2005.

[38] Mahamed Chibani, Roch Lefebvre, and Philippe Gournay, "Resynchronization of the adaptive codebook in a constrained CELP codec after a frame erasure.," in *ICASSP*, Toulouse, France, 2006.

[39] Gournay Philippe, Rouseau Francois, and Lefebvre Roch, "Improved packet loss recovery using late frames for prediction-based speech coders," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Hong Kong, 2003.

[40] S. V. Andersen, W. B. Kleijn, R. Hagen, J.Linden, M. N. Murthi, and J. Skoglund, "iLBC - a linear predictive coder with robustness to packet losses," in *IEEE Workshop on Speech Coding*, Tsukuba, Ibaraki, Japan, 2002.

[41] Global I. P. Sound, "GIPS RCU: Robustness enhancement of low bit-rate telephony applications," 2005.

[42] US Department of Defence Voice Processing Consortium, "DDVPC. LPC-10e speech coding standard. Technical Report FS-1015," Nov. 1984.

[43] D.W. Griffin and J.S. Lim, "Multi-Band Excitation vocoder.," in *ICASSP*, 1988, vol. 36, pp. 1223–1235.

[44] Robert Rudolph and Eddie Yu, "IMBE and AMBE speech compression," in *International IC'99*, 1999.

[45] Thomas F.Quatieri, *Discrete-Time Speech Signal Processing: Principles and Practice*, Prentice Hall, 2001.

[46] Yingbo Jiang and Vladimir Cuperman, "Encoding prototype waveforms using a phase codebook," in *IEEE Workshop on Speech Coding for Telecommunications*, 1995.

[47] Oded Gottesman, "Dispersion phase vector quantization for enhancement of waveform interpolative coder," in *ICASSP*, 1999, vol. 1, pp. 269–272.

[48] Oded Gottesman, "Enhanced waveform interpolative coding at low bit-rate," *IEEE Transactions Speech and Audio Processing*, vol. 9, no. 8, pp. 786–798, 2001.

[49] J. Lindblom and P. Hedelin, "Packet loss concealment based on sinusoidal Modeling," in *Proc. IEEE Workshop on Speech Coding*, Orlando USA, 2002, vol. 1, pp. 173–176.

[50] Yannis Stylianou, *Harmonic plus Noise Models for Speech, combined with Statistical Methods for Speech and Speaker Modification*, Ph.D. thesis, Ecole Nationale. Sup'erieure des Telecommunications, Paris, France, 1998.

[51] A. Rodbro Christoffer, G. Christensen Mads, Andersen Soren Vang, and Jensen Soren Holdt, "Compressed domain packet loss concealment of sinusoidally coded speech," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Hong Kong, 2003.

[52] E. Orozco, S. Villette, and A. M. Kondoz, "Multiple description coding for voice-over-IP using sinusoidal speech coding," in *ICASSP*, Toulouse, 2006.

[53] Sassan Ahmadi and Andreas Spanias, "A new phase model for sinusoidal coding of speech signals," in *5-th IEEE Mediterranean Conference on Control and Systems*, Paphos, Cyprus, 1997, pp. 21–23.

[54] Jonas Lindblom, "A sinusoidal voice over packet coder tailored for the frame-erasure channel," *IEEE Trans. Speech Audio Processing*, 2004.

[55] Allen Gersho and Robert M. Gray, *Vector Quantization and Signal Compression*, Kluwer, 1992.

[56] C.M. Garrido, M. N. Murthi, and S.V. Andersen, "On variable rate frame independent predictive speech coding: Re-engineering iLBC," in *ICASSP*, Toulouse, 2006, vol. 1.

[57] L. Rizzo, "Effective erasure codes for reliable computer communication protocols," *ACM Computer Communication Review*, vol. 27, no. 2, pp. 24–36, 1997.

[58] Wenyu Jiang and Henning Schulzrinne, "Comparison and optimization of packet loss repair methods on VoIP preceived quality under bursty loss," in *12th International Workshop on Network and Operating System support for digital audio and video*, Miami,Florida, 2002, pp. 73–81.

[59] T. Cover and J. Thomas, *Elements of Information theory*, New York: Wiley, 1991.

[60] Jongtae Lim, *Joint and Tandem Source-Channel Coding with complexity and delay constraints*, Ph.D. thesis, University of Michigan, 2001.

[61] Vivek K. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Processing Magazine*, vol. 18, pp. 74–93, September 2001.

[62] Kai Clüver and Thomas Sikora, "Multiple-description coding of logarithmic PCM," in *EUSIPCO*, Antalya, Turkey, Sept. 2005.

[63] A. Ingle and V.A. Vaishampayan, "DPCM system design for diversity systems with applications to packetized speech.," *IEEE Trans. on Speech and Audio Processing*, , no. 1, pp. 48–58, 1995.

[64] R. Singh and A. Ortega, "Erasure recovery in predictive coding environments using multiple description coding," in *Proc. of MMSP*, 1999.

[65] Wenyu Jiang and A. Ortega, "Multiple description coding via polyphase transform," in *Proc. of VCIP*, 1999.

[66] Y.J. Liang, E.G Steibach, and B. Girod, "Multi-stream voice over IP using packet path diversity," in *IEEE Fourth Workshop on Multimedia Signal Processing*, Cannes, France, 2001.

[67] Cheng-Chieh Lee, "Diversity control among multiple coders: a simple approach tomultiple descriptions," in *IEEE Workshop on Speech Coding, 2000.*, Delavan, WI, USA, 2000.

[68] Xin Zhong and Biing-Hwang Juang, "Multiple description speech coding with diversities," in *ICASSP*, 2002, vol. 1, pp. 177–180.

[69] Anand Anandakumar, Alan McCree, and Vishu Viswanathan, "Efficient CELP-based diversity schemes for VoIP," in *ICASSP*, Istanbul, Turkey, 2000, vol. 6, pp. 3682–3685.

[70] H. Dong, I.D. Chakares, A. Gersho, E. Belding-Royer, U. Madhow, and J.D. Gibson, "Speech coding for mobile ad-hoc networks," in *Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, Santa Barbara, 2003, vol. 1, pp. 280–284.

[71] B.W. Wah and Dong Ling, "LSP-based multiple-description coding for real-time low bit-rate voice over IP," *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 167–178, 2005.

[72] G. Kubin and W.B. Kleijn, "Multiple-description coding (mdc) of speech with an invertible auditory model," in *Speech Coding Proceedings, 1999 IEEE Workshop on*, 1999, pp. 81–83.

[73] Christian Feldbauer, Gernot Kubin, and W. Bastiaan Kleijn, "Anthropomorphic coding of speech and audio: A model inversion approach," *EURASIP Journal on Applied Signal Processing*, vol. 2005, no. 9, pp. 1334–1349, 2005, doi:10.1155/ASP.2005.1334.

[74] V.A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 821, 1993.

[75] R.J. McAulay and T.F. Quatieri, "Speech analysis-synthesis based on a sinusoidal representation," *IEEE Trans. Acoust., Speech and Signal Proc.*, 1986.

[76] Yannis Stylianou, "Applying the Harmonic-plus-Noise Model in concatenative speech synthesis," *IEEE Trans. on Speech and Audio Processing*, vol. 9, no. 1, pp. 21–29, 2001.

[77] Jonas Lindblom and Per Hedelin, "Packet loss concealment based on sinusoidal extrapolation," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, 2002, vol. 1, pp. 173–176.

[78] C.L. Lawson and R.J. Hanson, *Solving Least-Squares Problems*, Prentice Hall, 1974.

[79] Georg Heinig and Karla Rost, "New fast algorithms for Toeplitz-plus-Hankel matrices," *SIAM Journal Matrix Analysis and Applications.*, vol. 25, no. 3, pp. 842–857, 2004.

[80] Wim D'haes, *Automatic Estimation of Control Parameters for Music Synthesis Algorithms*, Ph.D. thesis, University of Antwerp, 2004.

[81] Yannis Stylianou, "A simple and fast way for generating a harmonic signal," *IEEE Signal Processing Letters*, vol. 7, no. 5, May 2000.

[82] O. Cappe, J. Laroche, and E. Moulines, "Regularized estimation of cepstrum envelope from discrete frequency points," October 1995.

[83] Hartmut Traunmuller, "Analytical expressions for the tonotopic sensory scale," *Journal Acoust. Soc. Am.*, vol. 88, pp. 97–100, 1990.

[84] Alain Cheveigne and Hideki Kawahara, "YIN, a fundamental frequency estimator for speech and music," *Journal Acoust. Soc. Am.*, vol. 111, no. 4, 2002.

[85] Wolfgang Hess, *Pitch Determination of Speech Signals: Algorithms and Devices*, Springer Series in Information Sciences, 2003.

[86] Chunyan Li, Vladimir Cuperman, and A. Gersho, "Robust closed-loop pitch estimation for harmonic coders by time scale modification," in *ICASSP*, 1999, vol. 1, pp. 257–260.

[87] S. Yeldener, Juan Carlos De Martin, and Vishu Viswanathan, "A mixed sinusoidally excited linear prediction coder at 4 kb/s and below," in *ICASSP*, 1998, vol. 2, pp. 589–592.

[88] D. Subramaniam Anand, *Gaussian Mixture Models in Compression and Communication*, Ph.D. thesis, University of California, 2003.

[89] J.A. Bucklew and N.C. Gallagher, "A note on the computation of optimal minimum mean-square error quantizers," *IEEE Trans. on Communications*, vol. 30, no. 1, pp. 298, 1982.

[90] T. Lookabough and R. Gray, "High-Resolution theory and the vector quantizer advantage," *IEEE Trans. Inform. Theory*, vol. 35, no. 5, pp. 1020–1033, 1989.

[91] S. Na and D. Neuhoff, "Bennet's integral for vector quantizers," *IEEE Trans. Inform. Theory*, vol. 41, no. 4, pp. 886–900, 1995.

[92] A. Gersho, "Asymptotically optimal block quantization," *IEEE Trans. Inform. Theory*, vol. 25, no. 4, pp. 373–380, 1979.

[93] Samuelsson Jonas, "Multidimensional companding quantization of the Gaussian source," *IEEE Trans. Inform. Theory*, vol. 49, no. 5, pp. 1343–1351, 2003.

[94] Samuelsson Jonas and Hedelin Per, "Recursive coding of spectrum parameters," *IEEE Transactions on Speech and Audio Processing*, vol. 9, 2001.

[95] G. J. McLachlan T. Krishnan, *The EM algorithm and Extensions*, Wiley, New York, 1997.

[96] K. K. Paliwal and Leigh Alsteris, "Usefulness of phase spectrum in human speech perception," in *EUROSPEECH*, 2003, pp. 2117–2120.

[97] Doh-Suk Kim, "On the perceptual irrelevant phase information in sinusoidal representation of speech," *IEEE Transactions Acoustics, Speech and Signal Processing*, vol. 9, no. 8, pp. 900–905, 2001.

[98] Li Liu, Jialong He, and Gunther Palm, "Effects of phase on the perception of intervocalic stop consonants," *Speech Communication*, vol. 22, no. 4, pp. 403–407, 1997.

[99] C. Gobl, *The voice source in speech communication*, Ph.D. thesis, KTH, 2003.

[100] N. Henrich, C. D' Alessandro, and B. Doval, "Glottal flow models: waveforms, spectra and physical measurements," in *Forum Acusticum*, Seville, Spain, 2002.

[101] Baris Bozkurt, *Zeros of the z-transform (ZZT) representation and group delay processing for the analysis of source and filter characteristics of speech signals*, Ph.D. thesis, Polytechnique de Mons, 2005.

[102] X.Q Sun, B.M.G. Cheetham, and W.T.K. Wong, "Spectral envelope and phase optimization for sinusoidal speech coding," in *IEEE Workshop on Speech Coding for Telecommunications*, Annapolis, USA, 1995, pp. 75–76.

[103] A.E. Rosenberg, "Effect of glottal pulse shape on the quality of natural vowels," *Journal Acoust. Soc. Am.*, vol. 49, no. 2, pp. 583–590, 1971.

[104] Sassan Ahmadi and Andreas Spanias, "A new sinusoidal phase Modelling algorithm," in *ICASSP*, Munich, Germany, 1997, vol. 3, pp. 1675–1678.

[105] Per Hedelin, "Phase compensation in all-pole speech analysis," in *ICASSP*, 1988, pp. 339–342.

[106] Xiaoqin Sun, Fabrice Plante, Barry M.G. Cheetham, and Kenneth W.T. Wong, "Phase Modelling of speech excitation for low bit-rate sinusoidal transform coding," in *ICASSP*, 1997, vol. 3, p. 1691.

[107] David L. Thomson, "Parametric Models of the magnitude/phase spectrum for harmonic speech coding," in *ICASSP*, 1988, vol. 1, pp. 378–381.

[108] Jorge S. Marques, Luis B. Almeida, and Jose M. Tribolet, "Harmonic coding at 4.8 kb/s," in *ICASSP*, 1990, vol. 1, pp. 17–20.

[109] Harald Pobloth and W. B. Kleijn, "Squared-error as a measure of perceived phase distortion," *Acoustical Society of America*, 2003.

[110] Doh-Suk Kim and Moo Young Kim, "On the perceptual weighting function for phase quantization of speech," in *IEEE Workshop on Speech Coding*, Delavan, WI, USA, 2000, pp. 62–64.

[111] K.V. Mardia, *Statistics of Directional Data*, Academic Press, 1972.

[112] Claus Bahlmann, "Directional features in online handwriting recognition," *Pattern Recognition*, vol. 39, 2006.

[113] Paris Smaragdis and Petros Boufounos, "Learning source trajectories using wrapped-phase Hidden Markov Models," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, 2005.

[114] Hajo Holzmann, Axel Munk, Max Suster, and Walter Zucchini, "Hidden Markov Models for circular and linear-circular time series," *Environmental and Ecological Statistics (to appear)*, 2006.

[115] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.

[116] Rave Harpaz and Robert Haralick, "The EM algorithm as a lower bound optimization technique," Tech. Rep., January 2006.

[117] Eric W. Weisstein, "Least Squares Fitting – Polynomial," 2006.

[118] M.D. Plumpe, T.F. Quatieri, and D.A. Reynolds, "Modeling of the glottal flow derivative waveform with applicationto speaker identification," *IEEE Transactions Speech and Audio Processing*, vol. 7, no. 5, pp. 569–586, 1999.

[119] Keiichi Tokuda, Heiga Zen, and Alan W. Black, "An HMM-based speech synthesis system applied to English," in *IEEE Speech Synthesis Workshop*, Santa Monica, California, 2002.

[120] L. Gavidia-Ceballos and J.H.L. Hansen, "Direct speech feature estimation using an iterative EM algorithm for vocal fold pathology detection," *IEEE Transactions on Biomedical Engineering*, vol. 43, no. 4, 1996.

[121] D. Chazan, R. Hoory, Z. Kons, D. Silberstein, and A. Sorin, "Reducing the footprint of the IBM trainable synthesis system," in *Proc.7th Int. Conf. Spoken Language Processing*, Denver, USA, 2002.

[122] Levent Tosun and Peter Kabal, "Dynamically adding redundancy for improved error concealment in packet voice coding," in *Proc. European Signal Processing Conference*, Antalya, Turkey, 2005.

[123] C. S. Xydeas and F. Zafeiropoulos, "Model-based packet loss concealment for AMR coders," in *ICASSP*, 2003.

[124] Moon-Keun Lee, Sung-Kyo Jung, Hong-Goo Kang, Yound-Cheol Park, and Dae-Hee Youn, "A packet loss concealment algorithm based on time-scale modification for CELP-type speech coders.," in *ICASSP*, 2003.

[125] B. Bessette, R. Salami, Roch Lefebvre, M. Jelinek, Rotola-Pukkila J., J. Vainio, H. Mikkola, and K. Jarvinen, "The adaptive multi-rate wideband codec (AMR-WB)," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 8, pp. 620–636, 2002.

[126] Appendix A ITU-T Recommendation G.711, "A high quality low-complexity algorithm for packet loss concealment with G.711," 1999.

[127] Appendix B ITU-T Recommendation G.711, "Packet loss concealment algorithm for use with ITU-T recommendation G.711," 2000.

[128] Goodman D.J., G.B. Lockhart, O.J. Wasem, and W.C.Wong, "Waveform substitution techniques for recovering missing speech segments in packet voice communications," *IEEE Transactions Acoustics, Speech and Signal Processing, ASSP*, vol. 34, pp. 1440–1448, 1986.

[129] Gunduzhan Emre and Momtahan Kathryn, "A linear prediction based packet loss concealment algorithm for PCM coded speech.," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 8, pp. 778–785, 2001.

[130] Maha Elsabrouty, Martin Bouchard, and Tyseer Aboulnasr, "Receiver-based packet loss concealment for Pulse Code Modulation (PCM G.711) coder," *Signal Processing*, vol. 84, pp. 663–667, 2004.

[131] Kazuhiro Kondo and Kiyoshi Nakagawa, "A speech packet loss concealment method using linear prediction.," *IEICE Transactions Information and Systems*, vol. E89-D, no. 2, pp. 806–813, 2006.

[132] D.J. Goodman, Kai Cluver, and Peter Noll, "Reconstruction of missing speech frames using sub-band excitation.," in *IEEE International Symposium on Time-Frequency and Time-Scale analysis.*, 1996, pp. 277–280.

[133] You-Li Chen and Bor-Sen Chen, "Model-based multirate representation of speech signals and it's application to recovery of missing speech packets.," *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 3, pp. 220–231, 1997.

[134] Lindblom Jonas and Per Hedelin, "Error protection and packet loss concealment based on a signal matched sinusoidal vocoder," in *ICASSP*, 2003.

[135] V.A. Vaishampayan and J. C. Batllo, "Asymptotic analysis of multiple description quantizers," *IEEE Trans. Inform. Theory*, vol. 44, no. 1, pp. 278–284, 1998.

[136] A. Gamal and T. M. Cover, "Achievable rates for multiple descriptions," *IEEE Trans. Inform. Theory*, vol. 28, pp. 851–857, 1982.

[137] L. Ozarow, "On a source coding problem with two channels and three receivers," *The Bell System Technical Journal*, vol. 59, pp. 1909–1921, 1980.

[138] Chao Tian and S.S. Hemami, "Universal multiple description scalar quantization: Analysis and design," *IEEE Trans. Inform. Theory*, vol. 50, no. 9, 2004.

[139] M. Flemming and E. Effros, "Generalized multiple description vector quantization," in *Proc. Data Compression*, 1999, pp. 3–12.

[140] Prashant Koulgi, Shankar L. Regunathan, and Kenneth Rose, "Multiple description quantization by deterministic annealing," *IEEE Trans. Inform. Theory*, vol. 49, no. 8, 2003.

[141] J.C. Batllo and V.A.Vaishampayan, "Asymptotic performance of multiple description transform codes," *IEEE Trans. Inform. Theory*, vol. 43, 1997.

[142] S. D. Servetto, V.A. Vaishampayan, and N. J. A. Sloane, "Multiple description lattice vector quantization," in *Proc. Data Compression Conference*, 1999.

[143] V.A. Vaishampayan, N. J. A. Sloane, and S. D. Servetto, "Multiple description quantization with lattice codebooks: design and analysis," *IEEE Trans. Inform. Theory*, vol. 47, no. 5, pp. 1718–1734, 2001.

[144] David Y. Zhao and W. Bastiaan Kleijn, "Multiple-description vector quantization using translated lattices with local optimization," in *Proceedings IEEE Global Telecommunications Conference*, Nov. 2004, vol. 1, pp. 41 – 45.

[145] Vivek K. Goyal and Jelena Kovacevic, "Generalized multiple description coding with correlating transforms," *IEEE Trans. Inform. Theory*, vol. 47, 2001.

[146] V. Goyal and J. Kovacevic, "Optimal multiple description transform of Gaussian vectors," in *IEEE Data Compression Conference*, 1998, pp. 388–397.

[147] Yao Wang, Michael T. Orchard, V.A. Vaishampayan, and A. Reibman, "Multiple description coding using pairwise correlating transforms," *IEEE Transactions on Image Coding*, vol. 10, no. 3, pp. 251–366, 2001.

[148] Vivek K. Goyal, Jelena Kovacevic, and Martin Vetterli, "Multiple description transform coding: Robustness to erasures using tight frame expansions.," in *ISIT*, Cambridge, MA USA, 1998.

[149] Vivek K. Goyal, Martin Vetterli, and Nguyen T. Thao, "Quantized overcomplete expansions in R^N: Analysis, synthesis and algorithms," *IEEE Trans. Inform. Theory*, vol. 44, no. 1, 1998.

[150] Philip A. Chou, Sanjeev Mehrotra, and A Wang, "Multiple description decoding of overcomplete expansions using projection onto convex sets.," in *IEEE Proc. Data Compression*, Snowbird, UT, 1999, pp. 72–81.

[151] Jonas Samuelsson and Jan Plasberg, "Multiple description coding based on Gaussian Mixture Models," *IEEE Signal Processing Letters*, June 2005.

[152] Michael T. Orchard, Y. Wang, V.A. Vaishampayan, and A. Reibman, "Redundancy Rate-Distortion analysis of multiple description coding using pairwise correlating transforms," in *Proc. IEEE Int. Conf. Image Proc.*, Santa Barbara, 1997, vol. 1, pp. 608–611.

[153] Y. Wang, A. Reibman, Michael T. Orchard, and Hamid Jafarkhani, "An improvement to multiple description transform coding," *IEEE Transactions Signal Processing*, vol. 50, no. 11, pp. 2843–2854, 2002.

[154] Sanjeev Mehrotra and Philip A. Chou, "On optimal frame expansions for multiple description quantization," in *ISIT*, Sorrento, Italy, 2000.

[155] V. Goyal, J. Kovacevic, and M. Vetterli, "Quantized frame expansions as source-channel codes for erasure channels," in *Proc. IEEE Data Compression Conference*, Snowbird, UT, USA, 1999, pp. 326–335.

[156] Robert M. Gray, "A new class of lower bounds to information rates of stationary sources via conditional rate-distortion functions," *IEEE Transactions on Information Theory*, vol. 19, 1973.

[157] T. Linder, R. Zamir, and K. Zeger, "On source coding with side information dependent distortion measures," *IEEE Trans. Inform. Theory*, 2000.

[158] J. Epps, *Wideband Extension of Narrowband Speech for Enhancement and Coding*, Ph.D. thesis, School of Electrical Engineering and Telecommunications, University of South Wales, 2000.

[159] Qian Yasheng and Peter Kabal, "Dual-mode wideband speech recovery from narrowband speech," in *ICASSP*, Montreal, Canada, 2004.

[160] K. Y. Park and H. S. Kim, "Narrowband to wideband conversion of speech using GMM-based transformation," in *Proc. ICASSP*, Istanbul, 2000.

[161] Peter Jax and Peter Vary, "Artificial bandwidth extension of speech signals using MMSE estimation based on a Hidden Markov Model," in *IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Hong Kong SAR, China, 2003, vol. 1.

[162] Peter Jax, *Enhancement of Bandlimited Speech Signals: Algorithms and Theoretical Bounds*, Ph.D. thesis, Institute of Communication Systems and Data Processing (IND), 2002.

[163] M. Nilsson, S. V. Andersen, and W. B. Kleijn, "Gaussian Mixture Model based mutual information estimation between frequency bands in speech," in *Proc. IEEE Int. Conf. Acoust. Speech Sign. Process*, Orlando USA, 2002.

[164] Yannis Agiomyrgiannakis and Yannis Stylianou, "Combined estimation/coding of highband spectral envelopes for speech spectrum expansion," in *ICASSP*, Montreal, Canada, 2004.

[165] So Stephen and Paliwal Kuldip, "Multi-frame GMM-based block quantization of line spectral frequencies for wideband speech coding," in *ICASSP*, Philadelphia, USA, 2005.

[166] R. Martin, C. Hoelper, and I. Wittke, "Estimation of missing LSF parameters using Gaussian mixture Models," in *Proc. IEEE Int. Conf. Acoust. Speech Sign. Process*, Salt Lake City, USA, 2001.

[167] Jonas Lindblom, Jonas Samuelsson, and Per Hedelin, "Model based spectrum prediction," in *IEEE Workshop on Speech Coding*, Delaway, USA, 2000.

[168] Robert M. Gray, *Source Coding Theory*, Kluwer Academic Publishers, 1990.

[169] R. Togneri, M. D. Alder, and Y. Attikiouzel, "Dimension and structure of the speech space," *IEE Proc.-I Communications, Speech and Vision*, vol. 139, no. 2, pp. 123–127, 1992.

[170] A. Rose, D. Rao, K. Miller, and A. Gersho, "A generalized VQ method for combined compression and estimation," in *In Proc IEEE Intern. Conf. on Acoustmics Speech and Sig. Proc.*, Atlanta, 1996, pp. 2032–2035.

[171] A. Gersho, "Optimal nonlinear interpolative vector quantization," *IEEE Trans. on Communications*, p. 1285, 1990.

[172] Yannis Stylianou, O. Cappe, and Moulines Eric, "Continuous probabilistic transform for voice conversion," *IEEE Trans. Speech Audio Processing*, 1998.

[173] J. Skoglund and J. Linden, "Predictive VQ for noisy channel spectrum coding: AR or MA?," in *Proc. ICASSP '97*, Munich, Germany, 1997, pp. 1351–1354.

[174] A. McCree, "A 14 kb/s wideband speech coder with a parametric highband Model," in *Proc IEEE Int. Conf. Acoust.*, Istanbul, 2000, pp. 1153–1156.

[175] Hanna Pierre and Desainte-Catherine Myriam, "Adapting the overlap-add method to the synthesis of noise," in *Proc. 5th Int. Conference on Digital Audio Effects (DAFx-02)*, Hamburg Germany, 2002.

[176] Veldhuis Raymond and Klabbers Esther, "On the computation of the Kullback-Leibler measure for spectral distances," *IEEE Transactions on speech and audio processing*, vol. 11, 2003.

[177] Yannis Stylianou and Ann K. Syrdal, "Perceptual and objective detection of discontinuities in concatenative speech synthesis," in *Proc. ICASSP*, 2001.

[178] Jonas Lindblom, *Coding Speech for Packet Networks*, Ph.D. thesis, Chalmers University Of Technology, 2003.

[179] D. Chazan, R. Hoory, Z. Kons, A. Sagi, S. Shechtman, and A. Sorin, "Small footprint concatenative text-to-speech synthesis system using complex spectral envelope Modeling," in *Eurospeech*, Lisbon, Portugal, 2005.

Appendix A

Expectation-Maximization for WGMM

A.1 Jensen's Inequality

If $f(.)$ is a *convex function* and x is a random variable then:

$$E\{f(x)\} \geq f\left(E\{x\}\right) \tag{A.1}$$

and if $f(.)$ is strictly convex, then equality implies that $x = E\{x\}$. Jensen's inequality is reversed if $f(.)$ is concave. Since $log(.)$ is concave, we have:

$$E\{\log(x)\} \leq \log\left(E\{x\}\right) \tag{A.2}$$

A.2 Optimization for the Expectation Step

We will maximize the function:

$$F = Q(\Omega_0, q_n(m,\vec{w})) + \sum_{n=1}^{N} \lambda_n \left(\sum_{m=1}^{M} \sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w}) - 1 \right) \tag{A.3}$$

$$= \sum_{n=1}^{N} \sum_{\vec{w}\in\mathbb{Z}^K} \sum_{m=1}^{M} q_n(m,\vec{w}) \ln\left(\frac{p(m,\vec{w},\vec{\theta}_n;\Omega)}{q_n(m,\vec{w})}\right) + \sum_{n=1}^{N} \lambda_n \left(\sum_{m=1}^{M} \sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w}) - 1 \right) \tag{A.4}$$

for $q_n(m,\vec{w})$. We equate the partial derivative with respect to $q_n(m,\vec{w}))$ to zero:

$$\frac{\partial F}{\partial q_n(m,\vec{w})} = \ln\Big(p(m,\vec{w},\vec{\theta};\Omega_0)\Big) - (1 + \ln\left(q_n(m,\vec{w})\right)) + \lambda_n = 0 \Rightarrow \tag{A.5}$$

$$q_n(m,\vec{w}) = e^{\lambda_n - 1} p(m,\vec{w},\vec{\theta};\Omega_0) \Rightarrow \tag{A.6}$$

$$\lambda_n = 1 - \ln\Big(p(\vec{\theta}_n,\Omega_0)\Big), \tag{A.7}$$

where we have summed equation (A.6) over $m,\vec{w}$ to obtain λ_n. Substituting λ_n to equation (A.5), we obtain the solution:

$$q_n(m,\vec{w}) = \frac{p(m,\vec{w},\vec{\theta}_n;\Omega_0)}{p(\vec{\theta}_n;\Omega_0)}. \tag{A.8}$$

A.3 Optimization for the Maximization Step

We will minimize $Q_2\left(\Omega, q_n(m,\vec{w})\right)$ under the constrain $\sum_{m=1}^{M}\alpha_m = 1$. Let

$$F = Q_2\left(\Omega, q_n(m,\vec{w})\right) + \lambda\left(\sum_{m=1}^{M}\alpha_m - 1\right) \Rightarrow$$
$$F = \sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})\ln\left(\alpha_m p(\vec{\theta}_n|m;\vec{w},\Omega)\right) + \lambda\left(\sum_{m=1}^{M}\alpha_m - 1\right). \tag{A.9}$$

First, we will solve for α_m:

$$\frac{\partial F}{\partial \alpha_m} = \sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})\frac{1}{\alpha_m} + \lambda = 0 \Rightarrow$$
$$\alpha_m = -\frac{1}{\lambda}\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w}) \overset{\text{sum over } m}{\Rightarrow} \lambda = -N$$

therefore

$$\alpha_m = \frac{1}{N}\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w}). \tag{A.10}$$

Then we will solve for $\vec{\mu}_m$, Σ_m. The natural logarithm of $p(\vec{w},\vec{\theta}_m|m;\Omega)$ (equation (4.10)) is:

$$\ln\left(p(\vec{w},\vec{\theta}_n|m;\Omega)\right) = -\frac{K}{2}\ln(2\pi) + \frac{1}{2}\ln\left|\Sigma_m^{-1}\right| - \frac{1}{2}(\vec{\theta}_n - \vec{\mu}_m - \vec{w}2\pi)^T\Sigma_m^{-1}(\vec{\theta}_n - \vec{\mu}_m - \vec{w}2\pi), \tag{A.11}$$

where $|.|$ denotes the determinant operation. Equating the derivative of F with respect to $\vec{\mu}_m$ to zero, we get

$$\frac{\partial F}{\partial \vec{\mu}_m} = 0 \Rightarrow \frac{\partial}{\partial \vec{\mu}_m}\left\{\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})\Sigma_m^{-1}(\vec{\theta}_n - \vec{\mu}_m - \vec{w}2\pi)\right\} = 0 \Rightarrow$$
$$\vec{\mu}_m = \frac{\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})\left(\vec{\theta}_n - \vec{w}2\pi\right)}{\sum_{n=1}^{N}\sum_{\vec{w}\in\mathbb{Z}^K} q_n(m,\vec{w})}. \tag{A.12}$$

Equating the derivative of F with respect to Σ_m^{-1} to zero, we get

$$\frac{\partial F}{\partial \Sigma_m^{-1}} = 0 \Rightarrow \frac{\partial}{\partial \Sigma_m^{-1}} \left\{ \sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w}) \left(\Sigma_m - (\vec{\theta}_n - \vec{\mu}_m - \vec{w}2\pi)(\vec{\theta}_n - \vec{\mu}_m - \vec{w}2\pi)^T \right) \right\} = 0,$$

therefore

$$\Sigma_m = \frac{\sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w})(\vec{\theta}_n - \vec{\mu}_m - \vec{w}2\pi)(\vec{\theta}_n - \vec{\mu}_m - \vec{w}2\pi)^T}{\sum_{n=1}^{N} \sum_{\vec{w} \in \mathbb{Z}^K} q_n(m, \vec{w})}. \tag{A.13}$$

where we have used the identities:

$$\frac{\partial}{\partial \Sigma_m^{-1}} \left\{ \vec{\theta}^T \Sigma_m^{-1} \vec{\theta} \right\} = \vec{\theta}\vec{\theta}^T \tag{A.14}$$

$$\frac{\partial}{\partial \Sigma_m^{-1}} \left\{ \ln \left| \Sigma_m^{-1} \right| \right\} = \Sigma_m^T = \Sigma_m \tag{A.15}$$

$$|\Sigma_m| = \frac{1}{|\Sigma_m^{-1}|} \tag{A.16}$$

$$\tag{A.17}$$

A.4 Update equations for a full EM step of a WGMM with diagonal covariance matrices

This section presents the equations that perform a full iteration of the EM algorithm. For computational purposes, the expectation step and the maximization step are intermixed.

We will define some accessory variables that hold the information that is related

to the expectation step:

$$\delta_{k,m,n,w} = \frac{1}{\sqrt{2\pi\sigma_m^2(k)}} \exp\left(-\frac{\left(\vec{\theta}_n(k) - \vec{\mu}_m(k) - w2\pi\right)^2}{2\sigma_m^2(k)}\right) \tag{A.18}$$

$$\beta_{k,m,n} = \sum_{w\in\mathbb{Z}} \delta_{k,m,n,w} \tag{A.19}$$

$$\beta^{(\mu)}_{k,m,n} = \sum_{w\in\mathbb{Z}} \delta_{k,m,n,w}\left(\vec{\theta}_n(k) - w2\pi\right) \tag{A.20}$$

$$\beta^{(\sigma^2)}_{k,m,n} = \sum_{w\in\mathbb{Z}} \delta_{k,m,n,w}\left(\vec{\theta}_n(k) - \vec{\mu}_m(k) - w2\pi\right)^2 \tag{A.21}$$

$$\beta_{m,n} = \alpha_m \prod_{k=1}^{K} \beta_{k,m,n} \tag{A.22}$$

$$\beta_n = \sum_{m=1}^{M} \beta_{m,n} \tag{A.23}$$

Now, we define the following accessory parameters:

$$\omega_m = \sum_{n=1}^{N} \frac{\beta_{m,n}}{\beta_n} \tag{A.24}$$

$$\tag{A.25}$$

The update equations can then be written as:

$$a_m \leftarrow \frac{1}{N}\omega_m \tag{A.26}$$

$$\vec{\mu}_m(k) \leftarrow \frac{1}{\omega_m} \sum_{n=1}^{N} \frac{\beta_{m,n}}{\beta_{k,m,n}} \beta^{(\mu)}_{k,m,n} \tag{A.27}$$

$$\sigma_m^2(k) \leftarrow \frac{1}{\omega_m} \sum_{n=1}^{N} \frac{\beta_{m,n}}{\beta_{k,m,n}} \beta^{(\sigma^2)}_{k,m,n} \tag{A.28}$$

$$\tag{A.29}$$

Note that in practice the summation need not be made over the whole $\mathbb{Z}$; only ± 2 tilings are adequate.

Appendix B

Multiple Description Coding

B.1 Proof: Optimal MSE Reconstruction for the MDTC Side Decoders

Let $\hat{x}_1$ be the side reconstruction when only description $\hat{y}_1$ is received. Let $\tilde{y}_2$ be any estimation of the second description from the first description. Due to quantization and estimation noise, the joint vector $[\hat{y}_1^T; \tilde{y}_2^T]^T$ may not lay on the image of F. By reconstructing $\hat{x}_1$ using equation $x = F_1^T y_1 + F_2^T y_2$ we remove the component of $y = [y_1^T y_2^T]^T$ that is orthogonal to the image of F. In that case, the optimal MSE reconstruction $\hat{x}_1$ is the one that minimizes:

$$MSE = E_{y_2,y_1} \left\{ \|\hat{x}_1 - x\|_2 \,|y_1 \in Q_{\hat{y}1} \right\}$$

where the expectation is taken over y_2, y_1, and $Q_{\hat{y}1}$ is the quantization cell defined by $\hat{y}_1$. Taking the derivative and equating to zero, we get:

$$\hat{x}_1 = E_{y_2,y_1} \left\{ x \,|y_1 \in Q_{\hat{y}_1} \right\}$$

If we assume that the quantization noise $e_1 = y_1 - \hat{y}_1$, $y_1 \in Q_{\hat{y}_1}$ is zero mean, then

$$\begin{aligned}
&\hat{x}_1 = E_{y_2,e_1} \left\{ x \,|y_1 = \hat{y}_1 + e_1 \right\} \\
&= F_1^T \left(\hat{y}_1 + E_{e_1} \left\{ e_1 \right\} \right) + F_2^T E_{y_2,e_1} \left\{ y_2 \,|y_1 = \hat{y}_1 + e_1 \right\} \\
&= F_1^T \hat{y}_1 + F_2^T \Sigma_{y_2 y_1} \Sigma_{y_1 y_1}^{-1} \left(\hat{y}_1 + E_{e_1} \left\{ e_1 \right\} \right) \Rightarrow \\
&\hat{x}_1 = \left(F_1^T + F_2^T \Sigma_{y_2 y_1} \Sigma_{y_1 y_1}^{-1} \right) \hat{y}_1
\end{aligned}$$

where $\Sigma_{y_2 y_1}$ is the cross-covariance matrix between y_2 and y_1, and $\Sigma_{y_1 y_1}$ is the covariance matrix of y_1. Note that $\Sigma_{y_2 y_1} \Sigma_{y_1 y_1}^{-1}$ is the optimal MSE linear regression matrix of the regression $E_{y_2} \left\{ y_2 \,|y_1 \right\}$.

The case where only description $\hat{y}_2$ is received is treated in a similar manner.

B.2 MDC Computational Issues

This section compares GMM-MDSQ$_{TC}$ and GMM-MDTC in terms of complexity and storage requirements. The comparison is made using an optimized pseudo-code

implementation of both systems. Since both systems share an amount of source code, the comparison is made solely on the parts of the code that differ. The following notation is used in the pseudo-code:

M**:** number of Gaussian components on the GMM.

P**:** number of dimensions of the source.

$V_{x,m}^{T}$**:** a P-by-P matrix holding the eigenvectors of the m-th Gaussian component in it's columns.

$\mu_{x,m}$**:** a P-by-1 vector with the mean value of the m-th Gaussian component.

$\sigma_{m,p}$**:** the standard deviation of the p-th dimension of the m-th Gaussian component.

ρ**:** packet loss probability.

B.2.1 GMM-MDSQ$_{TC}$

The pseudo-code implementation of GMM-MDSQ$_{TC}$ is shown in Algorithm 1. The GMM-MDSQ$_{TC}$ scheme stores a set of MDSQ codebooks for each possible loss probability and side description rate. Let $C_{0,N,\rho}$, $C_{1,N,\rho}$, $C_{2,N,\rho}$ be the central and the two side codebooks trained for packet loss probability ρ and side description rate $\log_2(N)$. Let $I_{1,N,\rho}$, $I_{2,N,\rho}$ be the corresponding index assignment. The following notation is then used:

$N_q[\mathrm{m,p}]$**:** the number of side description quantization levels for the p-th dimension of the m-th Gaussian component.

$C_{0,N,\rho}[\mathrm{n}]$**:** the n-th entry of the MDSQ central codebook.

$C_{1,N,\rho}[\mathrm{n}]$**:** the n-th entry of the first MDSQ side codebook.

$C_{2,N,\rho}[\mathrm{n}]$**:** the n-th entry of the second MDSQ side codebook.

$I_{1,N,\rho}[\mathrm{n}]$**:** side codebook 1 index associated with the n-th central codebook entry.

$I_{2,N,\rho}[\mathrm{n}]$**:** side codebook 2 index associated with the n-th central codebook entry.

The computational requirements of GMM-MDSQ$_{TC}$ are:

complexity: $\sum_{p=1}^{P}\sum_{m=1}^{M}\left(4+10N_q[p,m]\right)$ flops.

storage: $\sum_{n=2}^{N_{q,max}}\left(4|C_{0,n,\rho}|+4|C_{1,n,\rho}|+4|C_{2,n,\rho}|+2n\right)=\sum_{n=2}^{N_{q,max}}\left(4|C_{0,n,\rho}|+10n\right)$ bytes of static memory plus MP bytes of dynamic memory.

where $N_{q,max}$ is the maximum size of the side description codebook and $|C_{0,n,\rho}|$, $|C_{1,n,\rho}|$, $|C_{2,n,\rho}|$ are the sizes of the corresponding MDSQ codebooks. Each codeword is assumed to be stored using a 4 byte float and each index in $I_{1,N,\rho}$, $I_{2,N,\rho}$ using 1 byte (unsigned character). The dynamic memory consists of MP unsigned characters needed for the side description bit allocation. The side description bit allocation is made every time there is a change in the channel conditions (resp. loss probability). Note that we do not measure the complexity nor the storage for the whole GMM-MDSQ$_{TC}$ system, but only for the part of GMM-MDSQ$_{TC}$ that differs from GMM-MDTC.

B.2.2 GMM-MDTC

The pseudo-code implementation of GMM-MDTC is given in Algorithm 2. The implementation is optimized for the case where each Gaussian component is MDTC encoded with 1 degree of freedom.

The algorithm uses a set of precomputed static data according to the following notation:

C_N**:** an 1-by-N matrix that holds a scalar codebook trained for $N(0,1)$ Gaussians with N codewords.

$\phi_{m,\rho}$: the frame expansion parameter (angle) for the m-th Gaussian component. The angle is trained for packet-loss probability ρ.

Furthermore, the algorithm updates the following dynamic data every time the channel condition changes:

$N_{q,1}$**:** an M-by-P matrix that holds the number of quantization levels assigned to the p-th dimension of the first description.

$N_{q,2}$**:** an M-by-P matrix that holds the number of quantization levels assigned to the p-th dimension of the second description.

COS**:** an M-by-1 matrix that holds $(\cos(\phi_{m,\rho}))^2$.

SIN**:** an M-by-1 matrix that holds $(\sin(\phi_{m,\rho}))^2$.

Note that all matrices needed in equations (7.7),(7.6) and (7.5) are diagonal and highly structured for the case of MDTC with 1 degree of freedom. This allows a significant number of optimizations to take place. A total of $2MP$ characters are needed to store the matrices $N_{q,1}$,$N_{q,2}$. Furthermore, the shuffling of the eigenvectors in $V_{x,m}^T$ is made so that the first $P_1 = \lfloor \frac{P}{2} \rfloor$ dimensions of the m-th Gaussian component require the same rate with the rest $P - P_1$ dimensions, by default.

The computational requirements of GMM-MDTC are:

complexity: $\sum_{p=1}^{P} \sum_{m=1}^{M} (7 + 3N_{q,1}[p,m] + 3N_{q,2}[p,m])$ flops.

Algorithm 1 GMM-MDSQ$_{TC}$ Encoder. The algorithm encodes the input source vector $\mathbf{x} \in \mathbb{R}^P$ to the index m' (the "best" Gaussian component) and the P-dimensional vector indices $\mathbf{i}_1$, $\mathbf{i}_2$ of the scalar MDSQ side description indices.

procedure $[m', \mathbf{i}'_1, \mathbf{i}'_2] \leftarrow$ GMM-MDSQ-TC_ENCODE($\mathbf{x}$)
  $D_{min} \leftarrow \infty$;
  $m' \leftarrow 0$; $\mathbf{i}'_1 \leftarrow \mathbf{0}$; $\mathbf{i}'_2 \leftarrow \mathbf{0}$;
  $\lambda \leftarrow \frac{\rho}{1-\rho}$;
  for $m \leftarrow 1, \cdots, M$ **do**
    ▷ Encode the source vector $\mathbf{x}$ according to the m-th Gaussian Component of the GMM
    $\mathbf{x}'_m \leftarrow V_{x,m}^T(\mathbf{x} - \mu_{x,m})$; ▷ Translate and Decorrelate the vector $\mathbf{x}$
    for $p \leftarrow 1, \cdots, P$ **do**
      $\mathbf{x}''_m[p] \leftarrow \left(\frac{1}{\sigma_{m,p}}\right) * \mathbf{x}'_m[p]$; ▷ Scale to $N(0,1)$ **(+1 flop)**
      $[\hat{\mathbf{x}}_0[p], \hat{\mathbf{x}}_1[p], \hat{\mathbf{x}}_2[p], \mathbf{i}_1[p], \mathbf{i}_2[p]] \leftarrow$ MDSQ_Encode($\mathbf{x}''_{\mathrm{m}}$[p], N$_{\mathrm{q}}$[m, p], ρ); ▷ MDSQ quantization **(+10$N_q[m,p]$ flops)**
      $\hat{\mathbf{x}}_0[p] \leftarrow \hat{\mathbf{x}}_0[p] * \sigma_{m,p}$; ▷ Scale back to $N(0, \sigma^2_{m,p})$ **(+1 flop)**.
      $\hat{\mathbf{x}}_1[p] \leftarrow \hat{\mathbf{x}}_1[p] * \sigma_{m,p}$; ▷ Scale back to $N(0, \sigma^2_{m,p})$ **(+1 flop)**
      $\hat{\mathbf{x}}_2[p] \leftarrow \hat{\mathbf{x}}_2[p] * \sigma_{m,p}$; ▷ Scale back to $N(0, \sigma^2_{m,p})$ **(+1 flop)**
    end for
    $D \leftarrow \|\mathbf{x}'_m - \hat{\mathbf{x}}_0\|^2 + \lambda\left(\|\mathbf{x}'_m - \hat{\mathbf{x}}_1\|^2 + \|\mathbf{x}'_m - \hat{\mathbf{x}}_2\|^2\right)$; ▷ Compute MDC distance
    if $D < D_{min}$ **then**
      $D_{min} \leftarrow D$; $m' \leftarrow m$; $\mathbf{i}'_1 \leftarrow \mathbf{i}_1$; $\mathbf{i}'_2 \leftarrow \mathbf{i}_2$;
    end if
  end for
end procedure

procedure $[\hat{x}_0, \hat{x}_1, \hat{x}_2, i_1, i_2] \leftarrow$ MDSQ_ENCODE(x,N,ρ)
  $\lambda \leftarrow \frac{\rho}{1-\rho}$;
  $D_{min} \leftarrow \infty$;
  for $n \leftarrow 1, \cdots, N$ **do**
    $d_0 \leftarrow (x - C_{0,N}[n])^2$; ▷ Central distortion, **(+2 flops)**
    $d_1 \leftarrow (x - C_{1,N}[I_{1,N}[n]])^2$; ▷ Side distortion 1, **(+2 flops)**
    $d_2 \leftarrow (x - C_{2,N}[I_{2,N}[n]])^2$; ▷ Side distortion 2, **(+2 flops)**
    $D \leftarrow d_0 + \lambda * (d_1 + d_2)$; ▷ MDC distance, **(+3 flops)**
    if $D < D_{min}$ **then** ▷ **(+1 flop)**
      $D_{min} \leftarrow D$;
      $n' \leftarrow n$;
    end if
  end for
  $\hat{x}_0 \leftarrow C_{0,N,\rho}[n']$;
  $\hat{x}_1 \leftarrow C_{1,N,\rho}[I_1[n']]$;
  $\hat{x}_2 \leftarrow C_{2,N,\rho}[I_2[n']]$;
end procedure

storage: $4M + 4\frac{(N_{max}+1)N_{max}}{2}$ bytes of static memory plus $2MP + 8M$ bytes of dynamic memory.

where N_{max} is the maximum size of the precomputed $N(0,1)$ codebooks. As discussed in the previous section, we do not measure the complexity nor the storage for the whole GMM-MDTC system but only for the part of GMM-MDTC that differs from GMM-MDSQ$_{TC}$.

Algorithm 2 GMM-MDTC Encoder. The algorithm encodes the input source vector $\mathbf{x} \in \mathbb{R}^P$ to the index m' (the "best" Gaussian component) and the P-dimensional vector indices $\mathbf{i}_1$, $\mathbf{i}_2$ of the scalar side description indices.

procedure $[m', \mathbf{i}'_1, \mathbf{i}'_2] \leftarrow$ GMM-MDTC_ENCODE($\mathbf{x}$)
  $D_{min} \leftarrow \infty$;
  $m' \leftarrow 0$; $\mathbf{i}'_1 \leftarrow \mathbf{0}$; $\mathbf{i}'_2 \leftarrow \mathbf{0}$;
  $\lambda \leftarrow \frac{\rho}{1-\rho}$;
  for $m \leftarrow 1, \cdots, M$ **do**
    ▷ Encode the source vector $\mathbf{x}$ according to the m-th Gaussian Component of the GMM
    $\mathbf{x}'_m \leftarrow V_{x,m}^T(\mathbf{x} - \mu_{x,m})$; ▷ Translate and Decorrelate the vector $\mathbf{x}$
    $w_c = COS[m]$;
    $w_s = SIN[m]$;
    for $p \leftarrow 1, \cdots, P_1$ **do**
      $y_1 \leftarrow \left(\frac{1}{\sigma_{m,p}}\right) \mathbf{x}'_m[p]$; ▷ (+1 **flop**)
      $y_2 \leftarrow -y_1$; ▷ (+1 **flop**)
      $[\hat{y}_1,\ \mathbf{i}_1[p]] \leftarrow Norm\_Encode\,(y_1, N_{q,1}[m,p])$; ▷ ($+3N_{q,1}[m,p]$ **flops**)
      $[\hat{y}_2,\ \mathbf{i}_2[p]] \leftarrow Norm\_Encode\,(y_2, N_{q,2}[m,p])$; ▷ ($+3N_{q,2}[m,p]$ **flops**)
      $\hat{y}_1 \leftarrow \hat{y}_1\sigma_{m,p}$; ▷ (+1 **flop**)
      $\hat{y}_2 \leftarrow \hat{y}_2\sigma_{m,p}$; ▷ (+1 **flop**)
      $\hat{\mathbf{x}}_0[p] \leftarrow w_c\hat{y}_1 - w_s\hat{y}_2$; ▷ (+3 **flops**)
      $\hat{\mathbf{x}}_1[p] \leftarrow \hat{y}_1$;
      $\hat{\mathbf{x}}_2[p] \leftarrow -\hat{y}_2$; ▷ (+1 **flop**)
    end for
    for $p \leftarrow P_1 + 1, \cdots, P$ **do**
      $y_1 \leftarrow \left(\frac{1}{\sigma_{m,p}}\right) \mathbf{x}'_m[p]$; ▷ (+1 **flop**)
      $y_2 \leftarrow y_1$;
      $[\hat{y}_1,\ \mathbf{i}_1[p]] \leftarrow Norm\_Encode\,(y_1, N_{q,1}[m,p])$; ▷ ($+3N_{q,1}[m,p]$ **flops**)
      $[\hat{y}_2,\ \mathbf{i}_2[p]] \leftarrow Norm\_Encode\,(y_2, N_{q,2}[m,p])$; ▷ ($+3N_{q,2}[m,p]$ **flops**)
      $\hat{y}_1 \leftarrow \hat{y}_1\sigma_{m,p}$; ▷ (+1 **flop**)
      $\hat{y}_2 \leftarrow \hat{y}_2\sigma_{m,p}$; ▷ (+1 **flop**)
      $\hat{\mathbf{x}}_0[p] \leftarrow w_c\hat{y}_1 + w_s\hat{y}_2$; ▷ (+3 **flops**)
      $\hat{\mathbf{x}}_1[p] \leftarrow \hat{y}_1$;
      $\hat{\mathbf{x}}_2[p] \leftarrow \hat{y}_2$;
    end for
    $D \leftarrow \|\mathbf{x}'_m - \hat{\mathbf{x}}_0\|^2 + \lambda\left(\|\mathbf{x}'_m - \hat{\mathbf{x}}_1\|^2 + \|\mathbf{x}'_m - \hat{\mathbf{x}}_2\|^2\right)$; ▷ Compute MDC distance
    if $D < D_{min}$ **then**
      $D_{min} \leftarrow D$; $m' \leftarrow m$; $\mathbf{i}'_1 \leftarrow \mathbf{i}_1$; $\mathbf{i}'_2 \leftarrow \mathbf{i}_2$;
    end if
  end for
end procedure

procedure $[\hat{x}, i] \leftarrow$ NORM_ENCODE(x,N)
  $D_{min} \leftarrow \infty$;
  for $n \leftarrow 1, \cdots, N$ **do**
    $D \leftarrow (x - C_N[n])^2$; ▷ **(+2 flops)**
    if $D < D_{min}$ **then** ▷ **(+1 flop)**
      $D_{min} \leftarrow D$;
      $n' \leftarrow n$;
    end if
  end for
  $i \leftarrow n'$;
  $\hat{x} \leftarrow C_N[n']$;
end procedure