Error Resilient Multiple Description Coding

Rui Ma

Department of Electrical & Computer Engineering
McGill University
Montreal, Quebec, Canada

November 2009

A dissertation submitted to McGill University in partial fulfillment of the requirements for the degree of Ph.D.

2009/11/19

Abstract

In order to combat channel failures in data communications, multiple description coding (MDC) produces two or more equally important bitstreams or descriptions, and transmits them independently over erasure channels. If only one description is correctly received, a coarse copy of the source is obtained. The more descriptions correctly received, the finer the accomplished quality. When all descriptions are correctly received, the transmitted signal can be completely reconstructed.

In this work, we apply MDC to accommodate multimedia transmissions over hybrid wireline-wireless networks, which require low delay and high robustness against both packet losses and bit errors. In addition to the classical MDC channel model, i.e., on/off channels, we study channels that are also suffering from bit errors. Based on this channel model, we design what we call ERMDC or error resilient multiple description coding.

The proposed ERMDC encoder maximizes the Hamming distance between used codewords in MDC, so as to make as many errors as possible detectable at the decoder. In order to reduce the reconstruction distortion, the proposed ERMDC decoder can detect binary transmission errors and estimates their output values in two means: (i) one is MSE-optimal, but requires information about channel conditions; (ii) the other is suboptimal, but does not require channel conditions. The ERMDC achieves graceful performance degradation associated with BERs, and outperforms classic MDC when meeting with both packet losses and bit errors.

In order to avoid long time of design optimization, simplified index assignment (IA) algorithm for easy ERMDC encoder design is developed. This algorithm obtains "close-to-optimal" solutions as well as low computational complexity. Furthermore, this IA algorithm can be extended to embedded coding in progressive transmissions.

Moreover, we study performance of the ERMDC over Rayleigh fading channels by utilizing modulated signals as inputs. We also discuss usages of the ERMDC and its system-level performance over channels with both packet losses and bit errors. Experimental results show that, in general, the ERMDC system outperforms classic MDC systems.

Résumé

Le codage à descriptions multiples (MDC) vise à combattre les effets néfastes des défaillances du canal de transmission; à cette fin, il produit deux (ou plusieurs) flux binaires ou descriptions d'égale importance, qui sont ensuite transmis indépendamment sur des canaux à effacement. Si seulement une des descriptions est reçue correctement, une copie grossière de la source est alors obtenue. Plus le nombre de descriptions reçues correctement augmente, plus la qualité de reproduction augmente. Lorsque toutes les descriptions sont reçues correctement, le signal transmis peut être reconstruit complètement.

Dans le présent travail, nous appliquons le MDC au cas de la transmission de multimédia sur des canaux hybrides filaire/sans-fil, qui requiert d'atteindre un délai faible et une grande robustesse vis-à-vis des pertes de paquets et des erreurs binaires. Au-delà du modèle classique de canal MDC (de type "on/off"), nous étudions des canaux qui créent des erreurs individuelles sur les bits transmis. En se basant sur ce modèle de canal, nous concevons ce que nous appelons ERMDC, pour *codage à descriptions multiples résistant aux erreurs*.

Le codeur ERMDC proposé ici maximise la distance de Hamming entre les mots-codes du MDC, de manière à permettre au décodeur de détecter autant d'erreurs que possible. Afin de réduire la distorsion à la reconstruction, le décodeur ERMDC proposé ici a la capacité de détecter les erreurs de transmission binaires, et peut estimer les échantillons à reconstruire de deux façons : (i) l'une est optimale au sens de la distorsion quadratique moyenne, mais requiert la connaissance d'informations à propos de l'état du canal; (ii) l'autre est sous-optimale, mais ne nécessite pas cette connaissance. Le système ERMDC permet d'obtenir une dégradation graduelle de performance en fonction du taux d'erreur binaire (BER), et offre des performances supérieures au MDC classique dans le cas où les pertes de paquets et les erreurs sur les bits sont considérées conjointement.

Afin d'éviter des temps de conception trop longs liés à des optimisations complexes, un algorithme d'assignation d'indices (IA) simplifié est développé pour simplifier la conception de l'encodeur ERMDC. Cet algorithme obtient des solutions proches de l'optimum, à un coût de complexité réduit. Qui plus est, cet algorithme d'IA est étendu au cas du codage progressif (*embedded*) pour la transmission progressive.

Finalement, nous étudions la performance du système ERMDC sur des canaux à évanouissement de Rayleigh, en utilisant des signaux modulés en entrée. Nous discutons aussi de la performance au niveau système de ERMDC sur des canaux à pertes de paquets

et erreurs binaires. Les résultats expérimentaux confirment, en général, que le système ERMDC offre des performances supérieures aux systèmes MDC classiques.

Acknowledgments

First, I would like to thank my parents and sisters for their unselfish support all the time. Without their love, this work would be impossible.

I am very grateful to my supervisor, Professor Fabrice Labeau, for his academic guidance, productive discussion, generous support and patience. I would also like to thank Professor Peters Kabal, Jeremy Cooperstock for serving on my committee.

I really appreciate Ramdas Satyan's valuable suggestions on my dissertation. I cherish every minute with my friends and lab mates at McGill. Some of them are as follows: Wei Chu, Qipeng Gong, Ramdas Satyan, Sunday Nyamweno, Aarthi Reddy, Suning Jiang, Xiaohong Huang, Alexander Wyglinski, Karim Ali, Tania Leppert, Yongteng Ma, Zhengyan Shi, Suqun Fan and Qing Shi.

Contents

List of Figures

List of Tables

List of Acronyms

ARQ	Automatic Repeat reQuest
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BET	Bit Error Tolerance
BPSK	Binary phase-shift keying
BSC	Binary Symmetric Channel
COQ	Channel-optimized quantizer
COSQ	Channel Optimized Scalar Quantizer
COVQ	Channel Optimized Vector Quantizer
CRC	Cyclic Redundancy Check
bpss	bits per source sample
EBCOT	Embedded Block Coding with Optimal Truncation
EEP	Equal Error Protection
ERMDC	Error Resilient Multiple Description Coding
EZW	Embedded Zerotree Wavelet
FEC	Forward Error Correction
GA	Genetic Algorithm
IA	Index Assignment
i.i.d.	independent and identically distributed
IPTV	Internet Protocol Television
JPEG	Joint Photographic Experts Group
KLT	Karhunen-Loéve Transform
LSB	Least Significant Bit
MD	Multiple Description

MDC	Multiple Description Coding
MDCT	Multiple Description Correlating Transform
MDSQ	Multiple Description Scalar Quantizer
MDVQ	Multiple Description Vector Quantizer
MIMO	Multiple Input/Multiple Output
ML	Modified Linear
MLIA	Modified Linear Index Assignment
MN	Modified Nested
MNIA	Modified Nested Index Assignment
MSB	Most Significant Bit
MSE	Mean Square Error
OFDM	Orthogonal Frequency Division Multiplexing
PER	Packet Error Rate
PLR	Packet Loss Rate
PTSQ	Polyphase Transform and Selective Quantization
pdf	probability density function
RS	Reed-Solomon
RVQ	Robust Vector Quantizer
SD	Single Description
SDSQ	Single Description Scalar Quantizer
SER	Symbol Error Rate
SDR	Signal-to-Distortion Ratio
SNR	Signal-to-Noise Ratio
SPIHT	Set Partitioning in Hierarchical Trees
TCP	Transmission Control Protocol
UEP	Unequal Error Protection
VLC	Variable Length Coding
VoIP	Voice over Internet Protocol

List of Notations

A_l	the partition interval associated with l
A_{ij}	the partition interval associated with (i, j)
c_l	the reproduction level associated with l
c_{ij}	the reproduction level associated with (i, j)
$c_i^{(1)}$	the side reproduction level associated with i in the first description
$c_j^{(2)}$	the side reproduction level associated with j in the second description
D_0	the central distortion
D_1	the side distortion associated with the first description
D_2	the side distortion associated with the second description
D_s	the average side distortion of two descriptions
$f_X(x)$	the pdf of the source X
g_0	the central decoder
g_1	the side decoder associated with the first description
g_2	the side decoder associated with the second description
h_{min}	the minimum Hamming distance between MD index pairs
(i, j)	the multiple description coding index pair, where i and j are two descriptions of x respectively.
$\mathcal{I}$	the set of MDSQ indices i
$\mathcal{J}$	the set of MDSQ indices j
l	the single description scalar quantizer index
L	the number of indices l
$\mathcal{L}$	the set of indices l
M	the number of side quantization levels
q_0	the central encoder
q_1	the side encoder associated with the first description

q_2	the side encoder associated with the second description
R	the total code rate
R_0	the code rate of information, or the central code rate
R_s	the side code rate
$s^{(m)}(n)$	the spread associated with the MD index value n of the m-th description
$\bar{s}^{(m)}$	the average spread associated with the m-th description
$\bar{s}$	the overall average spread
s_{dif}	the difference between s_{max} and s_{min}
s_{min}	the minimum spread of all row and column spreads
s_{max}	the maximum spread of all row and column spreads
μ	the mean of the source signals
x	the source sample
X	the random variables of the source
$\mathcal{X}^{(0)}$	the central codebook
$\mathcal{X}^{(1)}$	the side codebook associated with the first description
$\mathcal{X}^{(2)}$	the side codebook associated with the second description

Chapter 1

Introduction

As required by today's telecommunications subscribers, emerging communications devices are not only telephones, but also data and video terminals. They connect with service providers via wireline and wireless connections. Service providers are trying to more efficiently support such users' requirements in means of reducing communication cost and improving the end users' experience. As a result, wireline-wireless convergence has become an important trend in the telecommunications industry [1].

Multimedia communications, especially, voice, audio, image and video transmissions, become important applications over the Internet. For example, a study predicted that the market for IPTV (Internet Protocol Television) services would rise from $779.2 million in 2006 to $26.3 billion in 2011 [2]. A report showed that the number of worldwide residential VoIP (Voice over Internet Protocol) subscribers reached over 75 million in 2007, 60% more than the number in 2006 [3].

However, many technical challenges exist in converging wireline and wireless networks. For example, channel characteristics of wireline and wireless networks are different: the main reason of channel failures over wireline networks is packet losses, because of congestion control and/or buffer overflow at edges between high and low-speed networks; on the other hand, channel failures and bit errors over wireless networks are usually caused by noisy channels. Normally, different error resilient methods are applied to deal with different channel characteristics. Instead, over hybrid wireline-wireless networks or wireless packet networks, an end-to-end solution is needed to provide robust transmissions with high efficiency and flexibility.

1.1 Why is multiple description coding used?

The quality of radio signals is influenced by various factors, such as other radio signals, obstacles, fading channels, multipath propagation and so on. Thus, wireless communication is suffering from error-prone channels. Techniques such as retransmission and error control coding or channel coding are developed to combat transmission errors [4].

Retransmission, well-known as Automatic Repeat reQuest (ARQ) in TCP, can provide error-free transmission, if an unlimited number of retransmissions are allowed. However, the low-delay requirement of multimedia communications prevents this from happening.

By adding redundancy, channel coding, also known as forward error correction (FEC) coding, works well for data file transmission. The added redundancy provides the capability of correcting transmission errors. Since different components of multimedia signals are not equally important, losses of more important signal components may result in significant performance degradation; losses of less important signal components may result in less or even invisible performance degradation. It is preferable to transmit more important information with higher reliability under the constraint of a limited bandwidth. Therefore, layered coding [5], [6], progressive transmission [7]–[10] and unequal error protection (UEP) [11]–[14] have been developed. In layered coding, more important signals are transmitted over more reliable channels. In progressive transmission, more important signals are transmitted earlier. UEP protects signals according to their importance: it provides more protection for more important signals, less protection for less important signals.

In a layered coding system, the source is encoded into several bitstream layers. There are two types of layers: base layer and enhancement layer(s). In general, enhancement layers can only be encoded/decoded based on encoding/decoding base layers that are more important.

In a progressive transmission system, the source is encoded into several packets. The receiver reconstructs the transmitted signals as packets arrive. The quality of reconstructed signals improves steadily with the number of consecutive packets received. The order of packets is so critical that the quality of reconstructed signals is only determined by the packets received before the missed one. For example, imagine that the source is encapsulated in seven packets. If six packets except the third one are received, the quality is only determined by the first two packets. Although the last four packets are received, they cannot be used, because they need the third packet to be received before decoding can

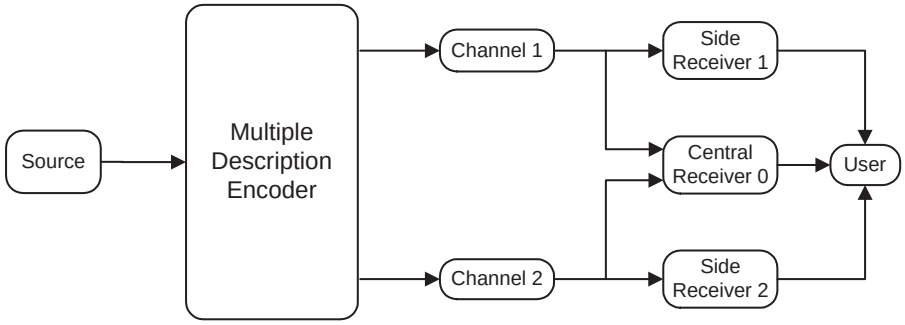


Fig. 1.1 An MDC system with two channels and three receivers.

proceed.

In both layered coding and progressive transmission systems, the quality of reconstructed signals depends on layers or packets that are more important than the lost one; on the other hand, the received information that is less important than the lost one is not usable. Therefore, multiple description coding (MDC), which will be described in details in the next section, is developed, so that all received information is usable, no matter which parts of information are lost [15].

1.2 Multiple Description Coding

Multiple description (MD) coding produces two or more equally important bitstreams, called *descriptions*, from a single information source. Descriptions are transmitted independently over erasure channels. At the receiver end, if all descriptions are received, the transmitted signals are reconstructed. If only one description is correctly received, a coarse copy of the transmitted signals with acceptable quality in the sense of distortion is achieved. The more descriptions correctly received, the finer the accomplished quality.

A typical MD coding (MDC) system with two channels and three receivers is demonstrated in Fig. 1.1. The distortion of the signals reconstructed from all descriptions is called the *central distortion*, which can be achieved by a single description encoder at the *central bit rate*. The *side distortion* is the distortion of the signals reconstructed from one or several, but not all, descriptions. In the MDC system with two channels and three receivers, if both channels work well, *Central receiver 0* attempts to recover the source from the two correctly received descriptions. Ideally, the central distortion is only caused by source coding, such as quantization. If one channel fails, the corresponding *side decoder 1* or *2*

can only recover part of the transmitted signals from the only received description. The resulting side distortion is much higher than the central distortion. The bit rate associated with each description is called the *side bit rate*.

Two descriptions can be generated in simple ways, for example:

- Sending two duplicated copies of the source signal. The side distortion is as low as the central distortion; however, the side bit rate is the same as the central bit rate.
- Sending one half of the source signal over one channel. The side bit rate is one half of the central bit rate; nevertheless, the side distortion is much higher than the central distortion.

Therefore, the target of the MDC problem is to find a best tradeoff between the central distortion and the side distortion at the cost of a modest side bit rate, which is usually smaller than the central bit rate. These kinds of problems were studied first by Ozarow [16], El Gamal and Cover [17]. As a typical MDC system, the multiple description scalar quantizer (MDSQ) was developed by Vaishampayan [18] to decompose source samples into two descriptions.

1.3 Motivation and Contributions

In this section, the motivation and contributions of this work are respectively summarized.

1.3.1 Motivation

In a typical MDC system, all channels are modeled as on/off characteristic. That is to say, descriptions, which are encapsulated in packets over packet networks, are either received correctly or lost completely [18]–[21]. Since nowadays most data networks are transmitting packets, here, we consider channel failures equivalent to packet losses.

In wireless communications, descriptions transmitted over radio channels suffer from bit errors. Even though FEC codes are regularly applied to protect descriptions from bit errors, descriptions after decoding FEC codes may still contain bit errors, as the result of the fact that the amount of errors exceeds the error correcting capability of the applied FEC codes. In a typical MDC system, such erroneous descriptions are usually dropped, although residual information is contained in these descriptions. Discarding erroneous descriptions

or packets [18], [19] results in significant degradation in the sense of rate-distortion performance, because a large performance gap exists between the central distortion and the side distortion.

Therefore, we propose an enhanced MDC decoder to utilize residual information in erroneous descriptions so as to achieve graceful performance degradation [22]. In order to combat channel failure or packet losses, redundancy is usually added to generate multiple descriptions. More redundancy provides higher robustness. The enhanced MDC decoder exploits the inherent redundancy and dependency among multiple descriptions to detect transmission errors, and estimate output values so as to reduce the reconstruction distortions.

Since in the design of the traditional MDC encoder, index assignment (IA) is achieved based on naturally labeled indices, which means that a smaller index represents a smaller quantity, one-bit errors may not be detected at the decoder. A robust IA algorithm is proposed by us to enlarge the Hamming distance between any two MD index pairs, so that one or more bit errors can be found more easily [23]. Consequently, the reconstruction distortion can be decreased further, especially, when this encoder works with the enhanced MDC decoder.

This pair of encoder and decoder constitutes what we will call the error resilient multiple description coding (ERMDC) system [24]. The objective of the ERMDC is to design a pair of encoder and decoder that attempts to detect as many bit errors as possible with inherent redundancy among multiple descriptions so as to minimize the distortion introduced by bit errors. Because it is very difficult to obtain an optimal IA scheme according to multiple criteria, such as rate distortion and robustness, a heuristic algorithm (in this case, the genetic algorithm (GA)) is utilized to find a "close-to-optimum" solution in a reasonable time. Experimental results show that the ERMDC not only achieves graceful performance degradation, but also effectively improves the performance in the presence of bit errors. In addition, performance of the ERMDC is tested over slow Rayleigh fading channels [25]. With the help of the ERMDC encoder, by utilizing received analog signals as soft inputs, performance of the ERMDC exceeds performance of other MDC methods.

As mentioned above, the GA is applied in the ERMDC encoder to obtain a "close-to-optimum" IA scheme. However, the computational complexity of this algorithm increases exponentially, as the search space, i.e., the bit rate of the information source, grows. We develop a fast robust IA algorithm for the ERMDC to add one-bit redundancy [26]. This

fast IA algorithm achieves similar performance to the GA at low bit rates, and even better performance at high bit rates in a limited design time. It is further generalized to add any arbitrary number of redundancy bits in the ERMDC [27]. Compared with existing methods, the ERMDC obtained by using this generalized fast IA scheme accomplishes similar robustness against packet losses, and higher robustness against bit errors. Moreover, because of its high flexibility and low computational complexity, the procedures of encoding/decoding two descriptions and detecting transmission errors only introduce fixed and short processing delays, and the added redundancy can be easily adapted to channel conditions.

In order to compare performance of the proposed ERMDC system with other error resilient techniques, such as the classical MDC and FEC-based techniques, we apply a three-state Markov chain to model the scenario where both packet losses and bit errors exist [28]. Experimental results show that the ERMDC system provides higher robustness against both packet losses and bit errors, and more consistent performance. Furthermore, we derive the optimal redundancy to minimize the average reconstruction distortion at a given packet loss rate (PLR) and bit error rate (BER) for the classical MDC system. The behaviour of the classical MDC system in the presence of both packet losses and bit errors is also analyzed. Even though this method is not always suitable for the ERMDC system, it provides a good reference.

1.3.2 Contributions

The main contributions of this dissertation are summarized as follows:

- The ERMDC decoder: The ERMDC decoder estimates outputs of detected transmission errors so as to minimize the reconstruction distortion in the presence of bit errors. As a result, it achieves graceful performance degradation in the rate-distortion sense.

- The ERMDC encoder: The ERMDC encoder enlarges the minimum Hamming distance between MD index pairs, so that more transmission errors can be detected, and the reconstruction distortion can be reduced further. Consequently, the distortion obtained by the ERMDC decoder is further reduced. In the design of the ERMDC encoder, exhaustive search and the GA are applied.

- A generalized fast IA algorithm for the ERMDC encoder: A generalized fast IA algorithm for the ERMDC encoder is proposed to reduce the design complexity of the ERMDC encoder. Compared with exhaustive search and the GA used before, in addition to significantly lower computational complexity, this fast IA algorithm achieves similar rate-distortion performance at low bit rates, and even better performance at high bit rates in a limited design time.

- The soft inputs ERMDC for Rayleigh fading channels: The ERMDC encoder-decoder pair utilizes modulated inputs received from noisy wireless channels, such as Rayleigh fading channels, to decrease the reconstruction distortion.

- The system-level evaluation of the ERMDC system: In the presence of both packet losses and bit errors, the ERMDC system outperforms other existing robust transmission systems in the rate-distortion sense by utilizing redundancy in both MDC and FEC. In order to conduct this system-level evaluation, a three-state Markov chain is proposed to model channels where both packet losses and bit errors exist.

- The optimal redundancy of the ERMDC system: The optimal redundancy is derived for the ERMDC system to achieve the lowest average reconstruction distortion at a given PLR and BER.

In addition, this work has been partially published in a series of papers:

- R. Ma and F. Labeau, "Enhanced multiple description decoder for error-prone channels," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2006, pp. 805–808.

- R. Ma and F. Labeau, "Robust index assignment for MDSQ encoder over noisy channels," in *Proc. IEEE Int. Workshop Multimedia Signal Processing*, Oct. 2006, pp. 286–290.

- R. Ma and F. Labeau, "Soft input error resilient multiple description coding for Rayleigh fading channels," in *Proc. IEEE Int. Conf. Multimedia Expo*, July 2007, pp. 1147–1150.

- R. Ma and F. Labeau, "Error-resilient multiple description coding," *IEEE Trans. Signal Processing*, vol. 56, no. 8, pp. 3996–4007, Aug. 2008.

- R. Ma and F. Labeau, "Fast index assignment for robust multiple description coding," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2008, pp. 2052–2055.
- R. Ma and F. Labeau, "Generalized fast index assignment for robust multiple description scalar quantizers," in *Proc. Asilomar Conf. Signals, Syst., Computers*, Oct. 2008, pp. 1287–1291.
- R. Ma and F. Labeau, "End-to-end performance of robust multiple description scalar quantizer," in *Proc. IEEE Veh. Technol. Conf.*, Apr. 2009.

1.4 Organization

In Chapter 2, existing work related to this dissertation is reviewed. As a typical joint source-channel coding technique, the noisy channel quantizer is briefly reviewed. Robust quantizers and channel-optimized quantizers are two classes of noisy channel quantizers. The robust quantizer is later used to develop the ERMDC. Next, after introducing basic concepts, methodologies and applications of MDC, the MDSQ is described in details, because it is not only a classical MDC system, but is also used as a foundation for our work. Thereafter, FEC-based MDC techniques are presented concisely.

In Chapter 3, the ERMDC as a whole is briefly introduced, followed by detailed description of the ERMDC decoder. Optimal and suboptimal ERMDC decoders are derived. Experimental results indicate that the ERMDC decoder achieves graceful performance degradation. The ERMDC decoder effectively improves the performance against isolated and burst bit errors. In addition, the suboptimal decoder with low computational complexity does not require knowledge of channel conditions, and provides consistent performance in various circumstances.

Design of the ERMDC encoder is elaborated in Chapter 4. The general design procedure is given in Section 4.2. In order to enlarge the Hamming distance between any two MD index pairs, the exhaustive search is used to find qualified MD index pairs. Then, the GA is exploited to assign indices so as to achieve as low side distortion as possible. Experimental results show that with the help of the ERMDC encoder, the performance achieved by the ERMDC decoder is further improved. At low bit rate, the GA achieves better performance than the traditional IA applied by the MDSQ.

However, as the bit rate of the information source grows, the computational complexity

of assigning indices by using the GA increases significantly. Therefore, in Chapter 5, a generalized fast robust IA algorithm is proposed. This IA algorithm is described in details for different numbers of bits of redundancy. The optimal IA scheme is also provided. Experimental results show that this fast IA scheme not only substantially reduces the computational complexity, but also provides robustness that is similar to the schemes obtained by the GA, and better than other existing methods. In addition, the average side distortion is computed through numerical solutions for arbitrary source signals in high rate systems. The analytical results for uniform distributed source signals are also derived. Thereafter, the fast IA algorithm is extended to embedded coding.

In Chapter 6, we attempt to test the proposed ERMDC in different scenarios. First, the ERMDC is applied in wireless communications over slow Rayleigh fading channels. By utilizing received analog signals as soft inputs, the ERMDC outperforms previous methods. Second, the ERMDC system is tested in the presence of both packet losses and bit errors. Combining with FEC codes, the ERMDC system provides higher and more consistent robustness against packet losses and bit errors. Third, the optimal redundancy of the classic MDC system is derived, and regarded as a good reference for the ERMDC system. In addition, the behaviour of the classic MDC system in the presence of packet losses and bit errors is analyzed.

Finally, we conclude this work, summarize the main contributions, and discuss future work in Chapter 7.

Chapter 2

Joint Source-Channel Coding for Robust Transmissions

Shannon's separation theorem states that source coding (compression) and channel coding (error protection) can proceed individually and optimally [29], [30]. However, the optimality can only be ensured in the case of asymptotically long block lengths of data, which is usually not practical. In addition, multimedia communications have been widely applied over wireline and wireless networks. Thus, joint source-channel coding has attracted substantial interest in the area of robust multimedia transmission over wireline and wireless networks.

In this chapter, we will review existing research related to our work. First, we introduce the typical techniques of designing noisy channel quantizers for one single description. Secondly, the basic idea and the existing works of the multiple description coding (MDC) are briefly reviewed. Thereafter, a short review of the MDC applied over noisy channels is provided. Finally, as a supplement, FEC-based packet-loss recovery techniques are reviewed.

2.1 Noisy Channel Quantizers

As a typical technique of source coding, quantizers, such as scalar and vector quantizers, approximate infinite real values to a relatively small number of integer values. In this work, we only consider scalar quantizers, even though the proposed ideas can be extended to vector quantizers.

A data compression system removes, to some possible extent, the redundancy in the

source, and keeps useful (non-redundant) information so as to decrease the data rate. The removal of redundancy introduces higher sensitivity to the transmission noise or the storage device errors. These kinds of noise (generally referred to as the channel noise) may incur significant degradations in the performance of the compression algorithm. Therefore, noisy channel quantizers are developed against noisy channels.

Generally speaking, design methods of quantizers against noisy channels, called *noisy channel quantizers*, fall into two categories:

1. **Robust quantizer** [31]– [33]: A robust quantizer encoder is trained for an error-free channel, and designed to be robust against noisy channels by assigning appropriate indices to the code words so as to minimize the distortion introduced by transmission errors. The robust quantizer is usually used under the circumstances where codebook training time is of primary concern.

2. **Channel-optimized quantizer (COQ)** [34] [35]: The encoder-decoder pair in a COQ system is jointly trained according to a given channel. The COQ is more suitable when higher reliability is needed.

Compared with the robust quantizer, the COQ requires more computations and provides lower performance when channel conditions are good, or when its design parameters are not matched to actual channel conditions. Thus, we utilize the robust quantizer technique to design the ERMDC encoder.

Developing a robust index assignment (IA) scheme is the critical step of designing a noisy channel quantizer. In IA, quantizer outputs are one-to-one mapped to a series of indices, which usually are integers. For a noisy channel quantizer, the overall process of IA was described [36] in three stages:

(i) an encoder mapping,

(ii) a channel index mapping, and

(iii) a decoder mapping.

The encoder mapping is the output of a quantizer. Generally, the encoder mapping is based on natural binary representation, e.g., *2* is coded as *010*. The channel index mapping, which is mainly discussed in this work, attempts to re-order the encoder mapping for robust transmission. In this work, unless otherwise specified, IA refers to the channel index

mapping. The decoder mapping is the inverse function of the encoder and channel index mapping.

Index Assignment is a class of combinatorial optimization problems that are usually referred to as the class of *NP*-complete problems [37]. Heuristic algorithms are widely used to find a "good" or "close-to-optimal" solution for these problems, since the complexity of exhaustive search becomes prohibitively high when the problem size is large. Most heuristic algorithms perform a deterministic search in a set of admissible configurations, and often terminate in a local minimum. Typical IA approaches in noisy channel quantizers are:

1. Binary Switching Algorithm (BSA) [31], [32]
2. Simulated Annealing [21], [35]
3. Genetic Algorithm [33]

Experimental results indicated that the genetic algorithm found more accurate and more consistent solutions than the BSA and the simulated annealing [33]. Therefore, we will use the genetic algorithm in this work.

In the following, we are going to give more details about noisy channel quantizers.

2.1.1 Design Procedure of Noisy Channel Quantizers

Zeger and Gersho presented a typical design procedure of a noisy channel quantizer, called a *generalized Lloyd-Max algorithm* [38]. They proposed an algorithm to design a vector quantizer to find a quantizer with minimum expected distortion between an input vector and a decoded (or quantized) output vector. Even though this algorithm was originally proposed for a vector quantizer, it can be easily applied to other types of quantizers, such as scalar quantizers and the COQ.

Let $\mathcal{I}$ be the set of possible indices. A *noisy channel quantizer* Q, illustrated in Fig. 2.1, is defined as

$$Q = q \circ \Pi \circ \tau \circ \Pi^{-1} \circ q^{-1} \tag{2.1}$$

where $\tau : \mathcal{I} \to \mathcal{I}$ is a memoryless noisy channel index mapping, and $\Pi : \mathcal{I} \to \mathcal{I}$ is a one-to-one function that permutes the assignment of indices to code vector. Π^{-1} "unpermutes" the IA at the receiver end. For a binary channel, the noise function τ can be represented by

$$\tau(i) = i \oplus n_0 \quad (i \in \mathcal{I}) \tag{2.2}$$

Table 2.1 Generalized Lloyd-Max algorithm for noisy channel quantizers.

Step 1:	Pick an initial codebook and IA function.
Step 2:	Partition the training set optimally by using code vectors and channel conditions. Compute centroids.
Step 3:	Find an optimal IA function for the current codebook and partition.
Step 4:	Find the best codebook from centroids and channel probabilities. Go to **Step 2**.

where n_0 is the channel noise, and the operation $\oplus$ is the modulo-2 addition. Therefore, the

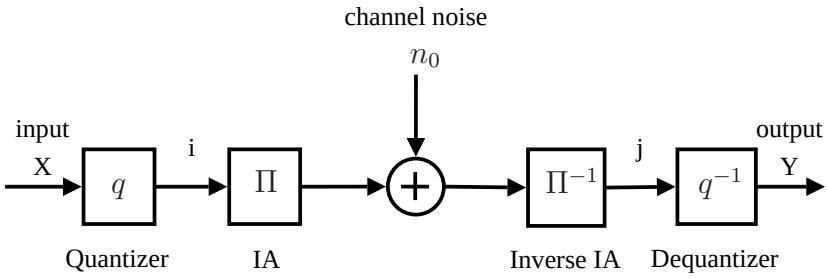


Fig. 2.1 A noisy channel quantizer Q.

goal of finding the optimal quantizer is equivalent to finding a quantizer which minimizes the average distortion

$$D \triangleq E[d(X, Q(X))]. \tag{2.3}$$

where $d(\cdot, \cdot)$ denotes the distortion between the input source X and the output $Q(X)$ at the receiver end. Minimizing D requires the joint optimum of the codebook, the partition, and the IA function Π. When $n_0 = 0$, this is a robust quantizer; when $n_0 \neq 0$, it is a COQ.

Generalized Lloyd-Max algorithm is applied to find the optimal noisy channel quantizer Q. This leads to iterative quantizer "design loops" with monotonically reducing distortions. The design algorithm is described in Table 2.1 and illustrated in Fig. 2.2. In each step of Steps 2–4 of Table 2.1, the other two factors other than the objective are assumed fixed.

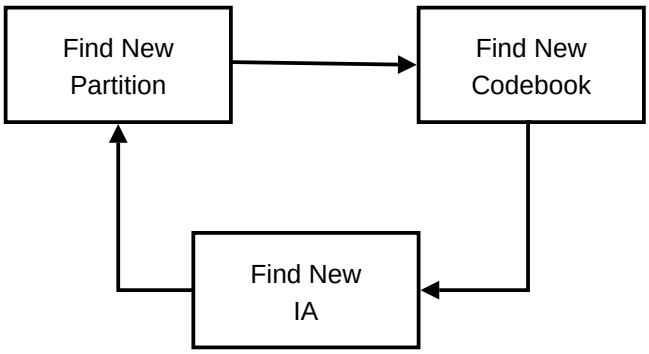


Fig. 2.2 Design loop for noisy channel quantizer.

2.1.2 Robust Quantizers

A robust quantizer encoder is trained for an error-free channel and designed to be robust against noisy channels by assigning appropriate indices to code words. The robust quantizer is usually used under the circumstances where codebook training time is of primary concern.

Zeger and Gersho developed an algorithm of locally optimal pseudo-Gray coding to reduce the expected distortion caused by channel bit errors [31], [32]. Gray code is developed to minimize the Hamming distance between neighbour codewords. By applying Gray coding, zero redundancy error control was developed by optimizing IA to minimize the average signal distortion. This technique was applied in low-bit-rate (4.8 kb/s) speech transmission [39], in which few redundancy bits could be allocated for error protection. An important observation was that there always existed some reordering of a given codebook with a nonnegative decrease in overall average distortion in the quantizer system. Therefore, an improvement could always be achievable in any quantizer system by using optimal pseudo-Gray coding.

Anti-Gray coding was proposed by Kou et al. to encode code vectors as far as possible in the sense of Hamming distance [40]. Consequently, at the receiver end, more erroneous code vectors were detected, and associated output values were approximated so as to achieve lower distortion of the reconstructed signals. This implies that some signal property, such as smoothness, can be used to detect errors.

Chang et al. studied parametric modeling of noisy channel by using an N-transition-state Markov channel model and IA to achieve a vector quantizer with high robustness against channel errors [33]. A genetic algorithm was applied to find a suboptimal IA to minimize the distortion introduced by bit errors. The experimental results showed that

the genetic algorithm helped to find a more accurate and more consistent solution than the BSA and the simulated annealing.

2.1.3 Channel-Optimized Quantizer

The encoder-decoder pair in a COQ system is jointly trained according to a given channel with a certain bit error rate (BER). The COQ is more suitable when transmission is needed over a channel with a high BER.

As a typical COQ, a channel-optimized scalar quantizer (COSQ) was developed by Farvardin and Vaishampayan [34]. Compared with the robust quantizer, the improvements achieved by the COSQ were obvious at high BERs. This method was extended to design a channel-optimized vector quantizer (COVQ) [35]. In addition to further study on Gauss-Markov sources, they provided a geometric structure of the COVQ [36]. As a step of this method, the simulated annealing was applied to find a "good" solution for IA. The experimental results showed that a COVQ could obtain higher robustness than a robust vector quantizer according to the actual channel characteristics. The COVQ was designed in terms of k-dimension and one bit per source sample (bpss). For the purpose of comparison, a simple method of unequal error protection (UEP) was applied to improve the robustness of a vector quantizer against noisy channel. In UEP, the first bit of each k bits in a code vector was replaced by the second bit. The resulting vector quantizer was equivalent to a vector quantizer with $\frac{k-1}{k}$ bpss. When $k = 8$, the performance of the COVQ was close to and even higher than that of the vector quantizer with UEP. Furthermore, a COVQ did not bring extra complexity and delay compared with a conventional source encoder followed by a channel code.

However, since the COQ is trained for certain channel conditions, its design usually requires more computations than that of the robust quantizer. Moreover, it provides lower performance when channel conditions are good, or when its design parameters are not matched to actual channel conditions.

2.2 Multiple Description Coding

Multiple description coding (MDC) was originally proposed to provide protection from channel failure for multimedia transmissions, in which some distortion at the receiver is tolerable [16], [17], [41]. Later on, MDC has been widely studied as a packet-loss recovery

technique in speech, image and video transmissions over erasure networks [12], [42]–[66].

2.2.1 Introduction

In multiple description coding, the source signals are decomposed into two or more highly correlated descriptions, and, then, sent over independent erasure channels. At the receiver, if all descriptions are correctly received, a fine copy of the transmitted signals is reconstructed. Otherwise, if just a portion of descriptions is correctly received, a coarse copy can be obtained. The more descriptions are correctly received, the higher fidelity of the reconstructed signals is.

Multiple description coding provides protection from channel failure or packet losses at the cost of higher coding rate, since some redundancy is added. The challenge in designing a multiple description encoder is to obtain a good tradeoff between the coding gain and the reconstruction quality from one description.

In general, methods of generating multiple descriptions fall into two categories:

- source coding: In techniques based on source coding, the source data are partitioned into several sets, and then compressed independently to obtain descriptions. It involves two typical families:
  - MD quantization, which is applied in our work;
  - MD correlating transforms (MDCT).

  These techniques are briefly reviewed in this section.
- channel coding: When channel coding is utilized to generate multiple descriptions, more powerful FEC codes are applied to protect more important information. More details will be described in Section 2.4.

Multiple description quantization

Techniques of MD quantization, such as MD scalar quantizer (MDSQ) and MD vector quantizer, are usually applied for memoryless sources. The MDSQ is flexible to choose the relative importance of the central distortion and each side distortion by index assignment (IA) [18], [67]. The basic idea of the MDSQ is easy to extend to the MD vector quantizer. Various techniques, such as MD lattice vector quantization and MD trellis-coded quantization, were developed to reduce the difficulty of the IA problem [68]–[71].

The MDSQ was applied to sources with memory in multiple description transform coding by Batllo and Vaishampayan [72]. Sources are transformed into independent coefficients by using the Karhunen-Loéve transform (KLT). Transform coefficients are further decomposed into multiple descriptions by using the MDSQ. The asymptotic analysis of the performance is provided. It is inferred that the optimal MD coding for the transform coefficients is equivalent to the optimal MD coding for the original source. Thus, dependency between transform coefficients is produced and used to estimate transform coefficients, when a description is lost. This method is further applied in audio, image and video coding (See details in Section 2.2.3).

Multiple description correlating transforms

In the MDCT, transforms are applied to generate correlated multiple descriptions. Goyal et al. utilized a tight frame expansion to generate multiple descriptions [46]. It is similar to block channel coding, except that it is applied on analogue signals before quantization.

Furthermore, Goyal et al. proposed an algorithm to generate two or more descriptions by using square, linear transforms and scalar quantization. If one description or more is lost, the output levels are estimated by using the statistical correlation introduced by the transform [73]. This algorithm was further generalized and applied for image and audio coding [74].

Similar to the algorithm proposed by Goyal et al. [74], Wang et al. developed pairwise correlating transform, by which controlled dependency among the transformed coefficients was set up [75]. After deriving the optimal redundancy allocation among a given set of pairs and the optimal pairwise strategy, the authors applied this transform to image coding.

MDC over noisy channels

MD techniques based on channel coding utilize forward error control (FEC) codes to produce multiple descriptions, and recover lost description(s) [12], [13], [48], [53], [76].

Since our work is based on the MDSQ, we will focus on the existing work based on the MDSQ. A more thorough survey was provided by Goyal [15]. In the following, we will go through the MDSQ in details, and use it as the example to explain the MD problem.

2.2.2 The MD Rate Distortion Region

In the MDC system with two channels and three receivers, illustrated in Fig. 1.1, source samples at the bit rate R_0 bpss are split into two descriptions, which are encoded at rates R_1 bpss and R_2 bpss respectively. In order to reconstruct the original signal with acceptable quality from each description at the receiver, two descriptions are required to be correlated. Thus, source coding efficiency (in the rate-distortion sense) is reduced and redundancy is increased. The tradeoff between central and side distortions is constrained by the correlation between the descriptions. The theoretical framework in the sense of rate distortion was discussed in [16], [17], [41].

The *central distortion* $D_0(R_0)$ represents the distortion achieved with both descriptions. *Side distortions* $D_1(R_1)$ and $D_2(R_2)$ represent the average distortion achieved respectively by one individual description. Therefore, for the two-channel case, the MD rate distortion region is defined by the set of achievable 5-tuples $(R_1, R_2, D_0, D_1, D_2)$. $R_s = \frac{R_1+R_2}{2}$ denotes the average side bitrate in bpss. $D_s = \frac{D_1+D_2}{2}$ denotes the average side distortion. In our work, we only study the balanced case, where $R_s = R_1 = R_2$ and $D_s = D_1 = D_2$.

For an independent and identically distributed (i.i.d.) Gaussian random source X with variance σ^2, the MD region $(R_1, R_2, D_0, D_1, D_2)$ satisfies

$$D_i \geq \sigma^2 \, 2^{-2R_i}, \quad \text{for } i = 1, 2, \tag{2.4}$$

$$D_0 \geq \sigma^2 \, 2^{-2(R_1+R_2)} \cdot \gamma_D(R_1, R_2, D_1, D_2), \tag{2.5}$$

where $\gamma_D = 1$, if $D_1 + D_2 > \sigma^2 + D_0$; and

$$\gamma_D = \frac{1}{1 - \left[\sqrt{(1-D_1)(1-D_2)} - \sqrt{D_1 D_2 - 2^{-2(R_1+R_2)}}\right]^2}, \quad \text{otherwise.} \tag{2.6}$$

In the balanced case, under the assumptions $R_s = R_1 = R_2 >> 1$ and $D_1 = D_2$, there exist encoders, for which D_0 and D_s reduce no faster than $2^{-2R_s(1+\alpha)}$ and $2^{-2R_s(1-\alpha)}$ respectively, where $0 < \alpha < 1$. Letting α close to 1 minimizes the redundancy and the central distortion decreases at the cost of greater side distortion. Letting α close to 0 maximizes redundancy with the extreme case that duplicates information over the two descriptions.

2.2.3 Applications on Image Coding

The MDC has been widely studied in areas of image communications, which is a potential application of this work. Transmitted over packet networks, images are usually split and encapsulated into packets. Channel failures in conventional MDC systems are generally treated as packet losses. The objective of MDC systems in image communications is to obtain acceptable reconstruction quality with receiving some packets, and the finest quality with all packets. In addition, conventional MDC approaches, which were developed originally based on independent signals, are also modified to adapt to the state-of-the-art techniques in image compression, such as embedded subband coding and progressive transmissions. A short review is given here as reference for the future work.

Although there is a lot of existing work to achieve MD image coding, a typical system of MD image coding can be summarized and illustrated in Fig. 2.3. In MD image coding, signals of an image are first converted to less correlated coefficients by transform coding, such as discrete cosine transform and wavelet transform. An MD encoder splits coefficients into two or more correlated descriptions, followed by entropy coding, such as arithmetic coding and Huffman coding. After adding FEC codes, packets are transmitted through error-prone channels. At the receiver end, transmission errors are detected and possibly corrected by utilizing the error correcting capability of the applied channel codes. After entropy decoding, an MD decoder attempts to reconstruct transform coefficients from received descriptions. The reconstructed coefficients are inverse transformed into an image.

Wavelet subband coding and embedded progressive coding are widely utilized in current standards and techniques of image compression, such as EZW [7], SPIHT [8] and JPEG2000 [10], [77]. Techniques developed based on MD quantization and MDCT have been adapted to these techniques to improve the robustness of image transmissions over noisy channels.

Image coding based on MD quantization

First, MD quantizers are utilized to decompose transform coefficients into multiple descriptions. Srinivasan and Chellappa tried to generate optimal multiple descriptions based on the classification of subbands [44]. This classification scheme divides each subband into several classes by minimizing the resulting central distortion. Coefficients in different

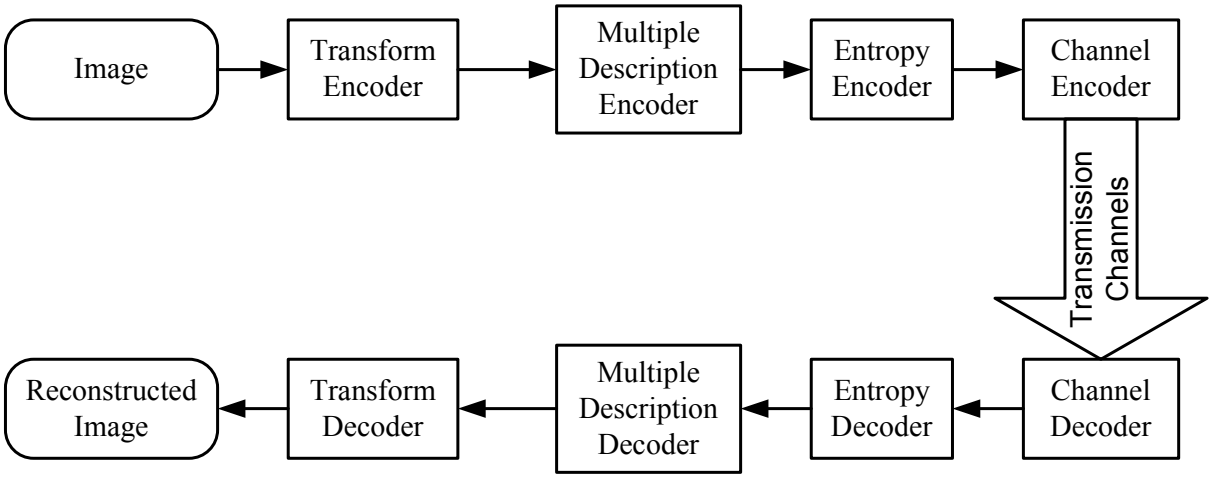


Fig. 2.3 A typical system of multiple description image coding.

classes are optimally decomposed into two descriptions by using appropriate MDSQs under constraints of rates and the average side distortion.

Servetto et al. utilized the MDSQ to decompose a wavelet subband coded image into two descriptions, each of which were entropy encoded [51]. An algorithm based on dynamic programming was developed to find an approximate optimal solution in the sense of rate-distortion function. Furthermore, this algorithm of MD image coding was extended to generate arbitrary number of descriptions by applying repetition codes $(n, 1, n)$ on two descriptions of low frequency subband coefficients. Consequently, the quality of the reconstructed image depended on the number of received packets, instead of exactly which packets were received. Experimental results showed that in the case of two descriptions, the proposed algorithm outperformed MD image coders of Wang et al. [75], and of Srinivasan and Chellappa [44]; in the case of many descriptions, it outperformed MD image coders of Goyal et al. [45], and of Jiang and Ortega [47].

The original MD quantizers have also been modified to accommodate to embedded image coding. Guionnet et al. developed two embedded multiple description coding algorithms for progressive image transmissions [54]. Embedded index assignment was developed based on the uniform MDSQ in the first algorithm. The resulting indices were compressed by EBCOT-based bit-plane coding to achieve progressive bitstreams. The second method was based on a combination of subband polyphase decomposition of wavelet coefficients and selective quantization of polyphase components. It was followed by JPEG2000 or SPIHT

coding to accomplish progressivity. At low bit rates and high redundancy, the first one achieved better performance than the second one.

Gavrilescu proposed another type of embedded scalar quantizer, named the embedded MDSQ [78], [79], based on quad tree coding [80]. This algorithm was generalized to M channels. The experimental results showed that the embedded MDSQ outperformed the uniform MDSQ.

Coarser quantizers have been applied to generate redundancy as well. Jiang and Ortega attempted to achieve MDC by using polyphase transform and selective quantization (PTSQ) [47]. Redundancy in the PTSQ was generated by using a lower resolution scalar or vector quantizer at a given bit rate. The authors also derived the optimal bit allocation to achieve minimum central distortion and average side distortion for a given total coding rate. Thereafter, the PTSQ was applied to transmit SPIHT-encoded images over erasure channels. Furthermore, the PTSQ was extended to generate arbitrary number of descriptions by Sagetong and Ortega [49].

An advantage of the PTSQ is its simplicity and flexibility, so that it is widely utilized in progressive image transmissions. Some variants of the PTSQ are developed for various image coding, such as EZW, SPIHT and JPEG2000. An MD construction algorithm for SPIHT was proposed by Miguel et al. [81]. By combining differently important bitstreams from various trees into one description, each description provided more or less information of other trees. Eom and Kim modified the thresholds of EZW to the powers of 1/4, instead of 1/2 in the original EZW algorithm, so as to form two descriptions [60]. The PTSQ was also applied in generating multiple descriptions in JPEG2000. In the environment of noisy channels, Pereira et al. tried to optimally allocate bit rates among subband blocks so as to achieve minimal central distortion with constraints of a certain side distortion and BER [56]. Tillo and Olmo [62] utilized coarser codeblock truncation with lower bit rates to generate redundancy, instead of coarser quantization.

Furthermore, the PTSQ has been combined with packetization in robust image transmissions [64]–[66]. Since the lowest frequency subband in wavelet transformed images provides most energy, the PTSQ-based techniques are applied to provide more protection for coefficients from this subband. Coefficients of higher frequency subbands are separately packetized. Consequently, each packet consists of information from all subbands.

Image coding based on MDCT

The multiple description correlating transforms are applied on image coding. Goyal et al. applied square correlating transforms [73] and overcomplete frame expansions [46] to deal with images [45]. Wang et al. utilized pairwise correlating transforms to image coding as well [75].

In addition, other types of MDC techniques have been developed for image coding. Ashwin et al. utilized correlation among wavelet coefficients to generate two descriptions and achieve better concealment performance [55]. The correlation matrix of wavelet coefficients is produced by the encoder and transmitted. Based on the correlation matrix, the lost wavelet coefficients are estimated.

2.2.4 Applications on Video Coding

The MDC is also used in video transmissions, such as work by Puri et al. [53] and Goshi et al. [14]. Wang et al. gave a broad review on MD video delivery [63]. A particular challenge in MD video transmissions is how to deal with motion compensation.

2.2.5 Summary

As a joint source-channel coding approach, the MDC was originally developed to provide high fidelity over error-free channels as well as high robustness against channel failures. For the Gaussian random source, the optimal MDC was theoretically discussed in the rate-distortion sense. The MDC has been widely studied in image and video transmissions. MD quantization and MD correlating transforms are two typical methods to generate multiple descriptions. Usually, however, the MDC is individually designed for specific kinds of signals, such as images and videos.

In this work, we will focus on MD quantization and extend the MDC to combat not only channel failures but also bit errors.

2.3 The Multiple Description Scalar Quantizer

The multiple description scalar quantizer (MDSQ) was developed for decomposing the quantized source samples into two correlated descriptions, which are sent over two independent channels [18]. The MDSQ was refined further under the constraints of equal entropy

over two balanced channels, i.e., the *entropy-constrained MDSQ* [67]. In the entropy-constrained MDSQ, entropy coding, such as Huffman coding, was applied after MDSQ to achieve higher compression efficiency.

The model of the MDSQ system is the same as the typical MDC system, shown in Fig. 1.1. It consists of two channels, which may fail at any time. Channel conditions are available at the decoder, but not at the encoder. The encoder of an MDSQ sends data over each channel at R_s bpss. The decoder reconstructs the signals received from the channel(s) working correctly. The objective of designing a pair of encoder and decoder is to minimize the average distortion when both channels work, subject to the constraint on the average distortion when only one channel works.

2.3.1 Problem Formulation

Let $x \in X$ be the output of a discrete-time, continuous alphabet source without memory. $f_X(x)$ is the pdf of the source X. Let each real-valued sample x be quantized by an L-level single description scalar quantizer (SDSQ) to an index l through the quantizer function $q_0(x) = l$. $\mathcal{L}$ denotes the set of the SDSQ indices l, i.e., $\mathcal{L} = \{0, 1, \cdots, L-1\}$. The quantization partition A_l associated with l is given by $A_l = \left[x_l^L, x_l^H\right)$. The reproduction level c_l associated with l is chosen as the centroid of the corresponding A_l:

$$c_l \stackrel{\triangle}{=} q_0^{-1}(x) \stackrel{\triangle}{=} \int_{A_l} x f_X(x) dx.$$

Each SDSQ index $l \in \mathcal{L}$ is then mapped to a pair of MDSQ indices (i, j) through an IA function $a(l)$: $(i, j) = a(l)$. MDSQ indices i and j are two descriptions of the source sample x transmitted over independent channels. $\mathcal{I}$ denotes the set of MDSQ indices i:

$$\mathcal{I} \stackrel{\triangle}{=} \{i \in \mathbb{N} : \exists l \in \mathcal{L}, j \in \mathbb{N} : (i, j) = a(l)\},$$

and $\mathcal{J}$ denotes the set of MDSQ indices j:

$$\mathcal{J} \stackrel{\triangle}{=} \{j \in \mathbb{N} : \exists l \in \mathcal{L}, i \in \mathbb{N} : (i, j) = a(l)\}.$$

When i and j are represented in R_s bpss respectively, $i, j \in \{0, 1, \cdots, M-1\}$, where $M = 2^{R_s}$, and $M \leq L$. The IA function $a(l)$ can be represented in mapping a cell of a

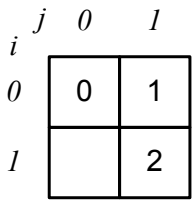


Fig. 2.4 SDSQ indices 0, 1, 2 are mapped to MDSQ index pairs (i, j).

matrix to a row and column pair, as illustrated in Fig. 2.4. SDSQ indices l appear in the matrix, whereas MDSQ indices i and j are the row and column numbers, respectively. The two components of the mapping are referred to $i = a^{(1)}(l)$ and $j = a^{(2)}(l)$, respectively. At the decoder, the inverse mapping, which can be deduced, e.g., from Fig. 2.4, is denoted by $l = a^{-1}(i, j)$.

The MDSQ maps the source sample x to the reconstruction levels c_l, $c_i^{(1)}$, and $c_j^{(2)}$. The values of c_l, $c_i^{(1)}$, and $c_j^{(2)}$ are accordingly taken from their codebooks, $\mathcal{X}^{(0)} = \{c_l,\ l \in \mathcal{L}\}$, $\mathcal{X}^{(1)} = \{c_i^{(1)},\ i \in \mathcal{I}\}$, and $\mathcal{X}^{(2)} = \{c_j^{(2)},\ j \in \mathcal{J}\}$. The MDSQ consists of one central encoder, q_0, two side encoders, q_1 and q_2, one central decoder, g_0, and two side decoders, g_1 and g_2:

$$q_0: \quad q_0(x) = l, \quad l \in \mathcal{L}, \tag{2.7}$$

$$q_1: \quad q_1(x) = i, \quad i \in \mathcal{I}, \tag{2.8}$$

$$q_2: \quad q_2(x) = j, \quad j \in \mathcal{J}, \tag{2.9}$$

and

$$g_0: \quad g_0(l) = c_l, \quad l \in \mathcal{L}, \tag{2.10}$$

$$g_1: \quad g_1(i) = c_i^{(1)}, \quad i \in \mathcal{I}, \tag{2.11}$$

$$g_2: \quad g_2(j) = c_j^{(2)}, \quad j \in \mathcal{J}, \tag{2.12}$$

where q_0, q_1 and q_2 produce indices l, i and j, respectively; g_0, g_1 and g_2 output the reconstruction levels corresponding to indices l, i, and j from the codebooks $\mathcal{X}^{(0)}$, $\mathcal{X}^{(1)}$, and $\mathcal{X}^{(2)}$, respectively. Let $\boldsymbol{q} = (q_0, q_1, q_2)$ be the encoder, $\boldsymbol{g} = (g_0, g_1, g_2)$ be the decoder. The central partition used by q_0 is denoted as $\mathcal{A} = \{A_l,\ l \in \mathcal{L}\}$ on $\mathbb{R}$, where $A_l =$

$\{x:\ q_0(x)=l\}$. Side partitions

$$A_i^{(1)} = \{x : \exists l \in \mathcal{L},\, q(x)=l,\ i=a^{(1)}(l)\} \tag{2.13}$$

and

$$A_j^{(2)} = \{x : \exists l \in \mathcal{L},\, q(x)=l,\ j=a^{(2)}(l)\} \tag{2.14}$$

satisfy conditions:

$$A_i^{(1)} = \bigcup_{j\in\mathcal{J}} A_{a^{-1}(i,j)}, \tag{2.15}$$

$$A_j^{(2)} = \bigcup_{i\in\mathcal{I}} A_{a^{-1}(i,j)}. \tag{2.16}$$

Because l is one-to-one mapped from (i,j) by $a^{-1}(i,j)$, where $i \in \mathcal{I}$ and $j \in \mathcal{J}$, for convenience of presentation, we use the two variables ij to represent the mapping function $a^{-1}(i,j)$. Therefore, two side partitions are written by

$$A_i^{(1)} = \bigcup_{j\in\mathcal{J}} A_{ij}\ , \tag{2.17}$$

$$A_j^{(2)} = \bigcup_{i\in\mathcal{I}} A_{ij}\ . \tag{2.18}$$

Then, the MDSQ is completely defined by $\mathcal{A}$, $\mathcal{X}^{(0)}$, $\mathcal{X}^{(1)}$, and $\mathcal{X}^{(2)}$.

Let random variables $\hat{X}^{(m)}$ represent the output of decoder g_m, $m=0,1,2$. Let $d(\cdot,\cdot)$ be the per-sample distortion between the source sample and the output of the central or side decoders. The average central and side distortions are given, respectively, by

$$D_0 = E\left\{d\left(X,\hat{X}^{(0)}\right)\right\} = \sum_{l\in\mathcal{L}} \int_{A_l} d\left(x,c_l\right) f_X(x)dx, \tag{2.19}$$

$$D_1 = E\left\{d\left(X,\hat{X}^{(1)}\right)\right\} = \sum_{i\in\mathcal{I}} \int_{A_i^{(1)}} d\left(x,c_i^{(1)}\right) f_X(x)dx, \tag{2.20}$$

$$D_2 = E\left\{d\left(X,\hat{X}^{(2)}\right)\right\} = \sum_{j\in\mathcal{J}} \int_{A_j^{(2)}} d\left(x,c_j^{(2)}\right) f_X(x)dx. \tag{2.21}$$

And the average side distortion is given by

$$D_s = \frac{D_1 + D_2}{2} \tag{2.22}$$

2.3.2 Optimum Encoder and Decoder

For given values M, $D_{1_{max}}$, and $D_{2_{max}}$, an MDSQ is said to be optimal if it minimizes D_0 subject to $D_1 \leq D_{1_{max}}$ and $D_2 \leq D_{2_{max}}$. The same as [18], in our work, only *balanced* descriptions are considered, where two equal rate descriptions are said to be balanced if they result in identical average distortions when being used individually.

The optimum encoder $\mathcal{A}^*$ and decoder $\boldsymbol{g}^*$ are obtained by taking the derivative of the Lagrangian function, $H(\mathcal{A}, \boldsymbol{g}, \lambda_1, \lambda_2)$, which is given by

$$H(\mathcal{A}, \boldsymbol{g}, \lambda_1, \lambda_2) = D_0 + \lambda_1(D_1 - D_{1_{max}}) + \lambda_2(D_2 - D_{2_{max}}), \tag{2.23}$$

where $\lambda_1 \geq 0$ and $\lambda_2 \geq 0$. Let $\mathcal{A}^*$ and $\boldsymbol{g}^*$ be such that $\mathcal{A}^*$ minimizes $H(\mathcal{A}, \boldsymbol{g}^*, \lambda_1, \lambda_2)$ for all $\mathcal{A}$, and $\boldsymbol{g}^*$ minimizes $H(\mathcal{A}^*, \boldsymbol{g}, \lambda_1, \lambda_2)$ for all $\boldsymbol{g}$. An iterative descent algorithm can be used to determine the optimal partition $\mathcal{A}^*$ and codebook $\boldsymbol{g}^*$.

By using square error distortion, (2.23) is simplified. Here, we omit the details of this derivation, which is available in [18], and only provide the results, which are used in our work to form balanced multiple descriptions adaptive to the source statistics.

Define

$$\alpha_l = c_l + \lambda_1 c_i^{(1)} + \lambda_2 c_j^{(2)}, \tag{2.24}$$

$$\beta_l = (c_l)^2 + \lambda_1 \left(c_i^{(1)}\right)^2 + \lambda_2 \left(c_j^{(2)}\right)^2, \tag{2.25}$$

and

$$\mathcal{L}_l^L = \{l' \in \mathcal{B} : \alpha_{l'} < \alpha_l\}, \tag{2.26}$$

$$\mathcal{L}_l^H = \{l' \in \mathcal{B} : \alpha_{l'} > \alpha_l\}, \tag{2.27}$$

where $\mathcal{B} = \mathcal{L} \setminus \{l\}$. Then the optimum central partition $\mathcal{A}^*$ is defined by the lower and

upper endpoints, t_l^L and t_l^H, of A_l:

$$A_l = \begin{cases} \emptyset, & \text{if} \quad t_l^L > t_l^H, \\ \left(t_l^L, t_l^H\right), & \text{if} \quad \mathcal{L}_l^L = \emptyset, \\ \left[t_l^L, t_l^H\right), & \text{otherwise}, \end{cases} \tag{2.28}$$

where

$$t_l^L = \begin{cases} -\infty, & \text{if} \quad \mathcal{L}_l^L = \emptyset, \\ \max\limits_{l' \in \mathcal{L}_l^L} \frac{\beta_l - \beta_{l'}}{2(\alpha_l - \alpha_{l'})}, & \text{otherwise;} \end{cases} \tag{2.29}$$

and

$$t_l^H = \begin{cases} \infty, & \text{if} \quad \mathcal{L}_l^H = \emptyset, \\ \min\limits_{l' \in \mathcal{L}_l^H} \frac{\beta_l - \beta_{l'}}{2(\alpha_l - \alpha_{l'})}, & \text{otherwise.} \end{cases} \tag{2.30}$$

Once the optimum partitions are obtained, the reconstruction levels are calculated as the centroids of all source samples falling into the same intervals. The optimum decoders are given by

$$g_1(i) = E\left\{\hat{X}^{(1)} \middle| \hat{i} = i\right\}, \quad i \in \mathcal{I}, \tag{2.31}$$

$$g_2(j) = E\left\{\hat{X}^{(2)} \middle| \hat{j} = j\right\}, \quad j \in \mathcal{J}, \tag{2.32}$$

$$g_0(l) = E\left\{\hat{X}^{(0)} \middle| \hat{l} = l\right\}, \quad l \in \mathcal{L}, \tag{2.33}$$

where $\hat{i}$ and $\hat{j}$ are the received indices corresponding to transmitted i and j, respectively; and $\hat{l} = a^{-1}(\hat{i}, \hat{j})$.

For a given $\mathcal{L}$, if $\lambda_1 = \lambda_2 = 0$, the MDSQ is actually an L-level Lloyd-Max quantizer, that is to say, D_0 is minimized. In order to achieve balanced side distortions, i.e., $D_1 = D_2$, we set $\lambda_1 = \lambda_2 > 0$. If λ_1 and λ_2 are very small, the central distortion of the MDSQ is close to the optimal distortion obtained by a Lloyd-Max quantizer, but side distortions may not be close, i.e., $D_1 \not\approx D_2$. On the contrary, if λ_1 and λ_2 are large, it may achieve close side distortions at the cost of a higher central distortion. Thus, the MDSQ is a modified Lloyd-Max quantizer with considering balanced side distortions. It attempts to

Table 2.2 The design algorithm of the optimal MDSQ.

Step 1: SET iteration counter $n = 0$.

Select an IA function,

Select an initial central partition $\mathcal{A}^{(n)}$,

Select Lagrange multipliers $\lambda_1, \lambda_2 \geq 0$.

SET $H^{(n)} = \infty$.

SET the stopping threshold δ to a suitably small value.

Step 2: $n \longleftarrow n + 1$.

Determine the optimum decoder $\boldsymbol{g}^{(n)}$ for using (2.31)–(2.33) for fixed $\mathcal{A}^{(n-1)}$.

Step 3: Determine the optimum partition $\mathcal{A}^{(n)}$ for fixed decoder $\boldsymbol{g}^{(n)}$ according to (2.28)–(2.30).

Step 4: Compute the Lagrangian function, $H^{(n)}$, using (2.23).

Step 5: IF $\frac{H^{(n)} - H^{(n-1)}}{H^{(n)}} < \delta$
THEN STOP
ELSE GOTO **Step 2**.

find a good tradeoff between a "close-to-minimum" central distortion and well-balanced low side distortions.

2.3.3 Design Algorithm

The design algorithm of the optimal MDSQ is described in Table 2.2. This design procedure is actually a specific implementation of the generalized Lloyd-Max algorithm described in Table 2.1 and Fig. 2.2. Here, the IA scheme is given. At each iteration, the optimum decoder $\boldsymbol{g}$ is first calculated in terms of the current partition $\mathcal{A}$. Based on the refreshed decoder, a new partition $\mathcal{A}$ is obtained. Then the objective function is computed so as to judge if this function has converged; if so, the algorithm terminates.

2.3.4 The Index Assignment Problem

We use a simple example to demonstrate the IA problem. Assume that X is uniformly distributed over $(-2,\ 2)$ and that $R_0 = 2$ bpss is used in the SDSQ. Consider various MDSQ

designs illustrated in Fig. 2.5, Fig. 2.6, Table 2.3 and Table 2.4. Figure 2.5 gives four IA schemes: (a) $R_1 = R_2 = 2$ bpss, each SDSQ index l is duplicated as MDSQ indices i and j respectively; (b)(c)(d) $R_1 = R_2 = 1$ bpss. SDSQ indices l are filled in cells. IA matrices indicate mapping between SDSQ indices l and MDSQ index pairs $(i,\ j)$. Blank cells suggest that associated MDSQ index pairs are not mapped to SDSQ indices. For instance, in Fig. 2.6(a), the SDSQ index *0* is mapped to the MDSQ index pair $(0,\ 0)$; in the mean time, the MDSQ index pair $(0,\ 1)$ is not mapped to any SDSQ index. Figure 2.6 shows SDSQ partitions, reconstruction codebooks, indices l and MDSQ index pairs $(i,\ j)$, which correspond to the four IA schemes illustrated in Fig. 2.5.

Tables 2.3 and 2.4 shows side codebooks, central distortions and side distortions in mean square errors (MSE). Table 2.4 indicates that IA schemes (a)–(c) provide the same central distortions, which is lower than that of the IA scheme (d). The IA scheme (a) provides the lowest and balanced side distortions; however, the total bit rate $R_1 + R_2$ of two descriptions is two times as much as other schemes. Even though the total bit rate $R_1 + R_2$ and central distortions of IA schemes (b) and (c) are low, their side distortions are high and not balanced. Therefore, the IA scheme (d) provides a good tradeoff between the total bit rate $R_1 + R_2$, central and side distortions.

From this example, we infer that the achievable total bit rate $R_1 + R_2$, central and side distortions can be determined by the IA. It is easy to search for an optimal solution in this example; nevertheless, the computational complexity grows significantly, as the total bit rate $R_1 + R_2$ increases. The total number of all possible IA schemes is $L! \cdot \binom{M^2}{L} = \frac{M^2!}{(M^2-L)!}$ for given M and L. In this subsection, we introduce the IA algorithms developed by Vaishampayan [18]. In later chapters, we will propose robust IA algorithms for the MDSQ to provide protection against packet losses and bit errors.

Vaishampayan gave two IA schemes: modified nested (MN) and modified linear (ML) IA [18]. The MLIA showed better performance in the sense of rate-distortion. In order to encode the source at R_s bpss on each channel, L index pairs are selected from the main diagonal and the $2k$ diagonals closest to the main diagonal of an $M \times M$ IA matrix. Examples of MNIA and MLIA are illustrated in Figs. 2.7 and 2.8. Each IA implementation is defined by $(R_s,\ k,\ L)$, such as $MNIA(3, 1, 22)$ in Fig. 2.7(a) and $MLIA(3, 1, 21)$ in Fig. 2.8(a).

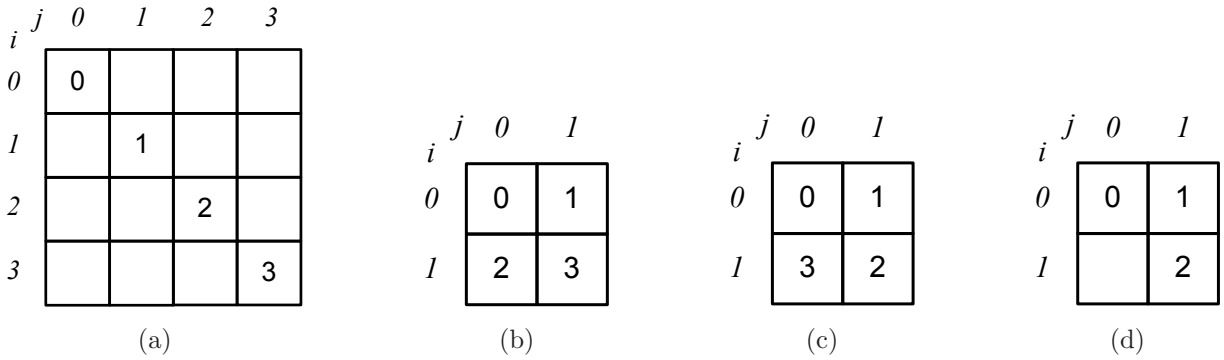


Fig. 2.5 Four index assignment schemes.

Table 2.3 Side codebooks associated with IA schemes illustrated in Fig. 2.5.

	$\hat{x}_0^{(1)}$	$\hat{x}_1^{(1)}$	$\hat{x}_2^{(1)}$	$\hat{x}_3^{(1)}$	$\hat{x}_0^{(2)}$	$\hat{x}_1^{(2)}$	$\hat{x}_2^{(2)}$	$\hat{x}_3^{(2)}$
(a)	-1.5	-0.5	0.5	1.5	-1.5	-0.5	0.5	1.5
(b)	-1	1	—	—	-0.5	0.5	—	—
(c)	-1	1	—	—	0	0	—	—
(d)	-2/3	4/3	—	—	-4/3	2/3	—	—

Table 2.4 Central and side distortions in MSE associated with IA schemes illustrated in Fig. 2.5.

	D_0	D_1	D_2
(a)	1/12	1/12	1/12
(b)	1/12	1/3	13/12
(c)	1/12	1/3	2/3
(d)	4/27	4/9	4/9

2.3.5 Extensions on MD Index Assignment

As described above, the design of an MD quantizer can be divided into two stages: the IA and codebook training. The index assignment plays an important role in the MD quantizer. Some work has been done to obtain the optimal IA scheme of an MD quantizer. Because of its prohibitively high computational complexity, heuristic algorithms are widely applied.

Berger-Wolf and Reingold modeled IA over two and more channels as a combinational optimization problem of number arrangement in a matrix [19]. The authors proposed a method to derive lower bounds on the distortions at given channel rates, as well as an algorithm to give the upper bound for the arbitrary number of channels. Furthermore,

Balogh and Csirik studied the IA problem of the MDSQ with two identical channels for an unbounded discrete information source [82]. They used two measures of side distortions: the spread and the variance. For both measures, new lower and upper bounds were derived.

Görtz and Leelapornchai extended the binary switching algorithm [32] to obtain the optimal IA of an MD vector quantizer with an arbitrary number of descriptions [83]. The experimental results showed that this algorithm achieved higher robustness against packet losses than the ML IA scheme provided in [18]. Yahampath applied simulated annealing to obtain the optimal IA for an MD vector quantizer with an arbitrary number of descriptions at equal or unequal transmission rates [20].

2.3.6 The MDSQ over Noisy Channels

As a direct application, the MDSQ was applied in wireless communications by Yang and Vaishampayan [84]. Over slow fading Rayleigh channels, the MDSQ resulted in not only a smaller interleaving delay than the conventional channel code-based approaches, but also significant improvement over a system with a maximum ratio combiner.

The technique of the COQ was applied in the MD system as well. Yahampath developed an algorithm to allocate MD index pairs so as to minimize the overall distortion in the presence of packet loss [20]. Zhou and Chan applied an extended multiple-channel optimized quantizer design to compensate the distortion introduced by erased symbols [21]. However, if the actual channel conditions, such as packet loss rate, symbol error probability and loss probability, were quite different from those at the design stage, the performance of COQ-based MDC algorithms developed in [20] [21] degraded much, even worse than that of the original MDSQ.

2.4 FEC-based Multiple Description Coding

Forward error correction (FEC) is a well-known channel coding technique that provides protection against bit errors. FEC is also utilized to provide protection in MDC. Puri and Ramchandran exploited unequal error protection (UEP) to generate robust multiple descriptions from prioritized source bit streams [48]. Different from "source coding" based MDC techniques mentioned above, FEC-based MDC distributes non-overlapping information evenly in multiple descriptions according to its importance. More powerful FEC codes are exploited to protect more important information. As a result, more important infor-

mation is more likely to be recovered by the applied FEC codes. This technique can be widely and easily extended to various source, such as images [12], [13], [48], [76], [85], and videos [14], [53].

Puri and Ramchandran interleaved the source bit stream, and utilized Reed-Solomon (RS) codes to provide protection from lost descriptions [48]. The mechanism is illustrated in Fig. 2.9. The original progressive bit stream of the source is partitioned into m quality levels, which are protected by RS codes, and then grouped into n descriptions. Stronger RS codes are applied to protect more important quality levels. Every description is designed to be "equally" important. The ith level is protected by RS$(n, i, n-i+1)$[1] code. As the result, when the number of lost descriptions is not more than $n-i$, the lost description(s) can be recovered. This idea was further utilized to generate MD of videos so as to protect from lost packets [14], [53]. Moreover, based on FEC-based MDC, Sachs et al. added FEC codes within every packet so as to provide protection against both bit errors and packet losses for image transmissions over wireless packet networks [12].

2.5 Summary

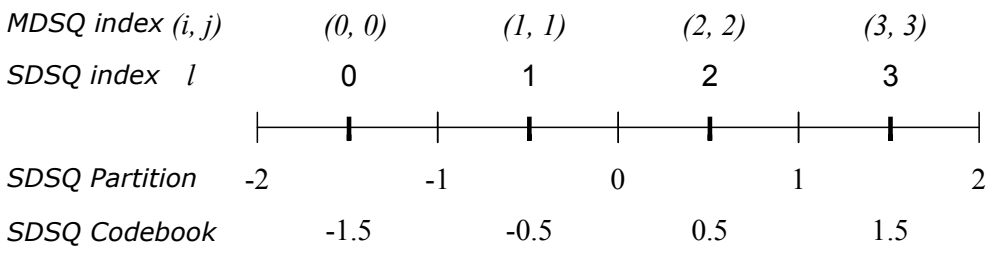
Due to limited bandwidth in transmissions, source signals are compressed by quantizers. Compressed data are vulnerable to bit errors incurred by noisy channels. Techniques for noisy channel quantization are developed to improve the fidelity of reconstructed signals at the receiver without ARQ. Over erasure channels, the MDC was introduced to achieve fine quality of the reconstructed signals with all descriptions, and acceptable quality with some descriptions. As an important step, approaches of IA are widely studied in the MDC system. In addition, the MDC system is applied in image and video communications to accomplish acceptable reconstruction quality in the presence of packet loss.

Existing work on IA in an MD system only used natural binary codes of the chosen index pairs, which are vulnerable to bit errors. In the MDC system, once bit errors are so many that applied FEC codes fail to decode, the contaminated descriptions are usually dropped. It results in significant performance degradation.

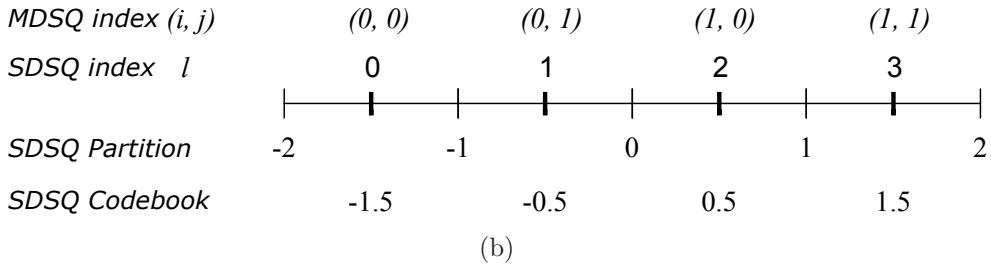
In our work presented in the following chapters, we, first, try to improve the robustness of MD index pairs against bit errors. Furthermore, we intend to utilize the residual

[1] An i-tuple message is mapped to an n-tuple RS codeword, where the minimum Hamming distance is $n-i+1$.

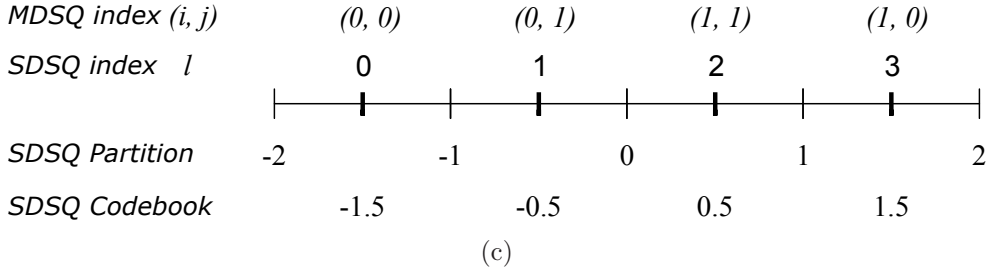
information in the contaminated descriptions to reduce adverse effects of bit errors. Therefore, we propose the error resilient multiple description coding (ERMDC). With deliberate design, the ERMDC exploits the same redundancy as the traditional MDSQ to achieve higher robustness against both bit errors and packet losses, which usually occur in hybrid wireline-wireless networks.



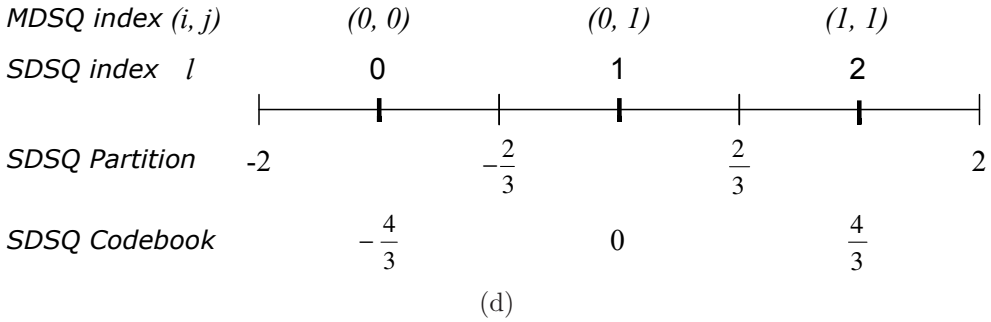
(a)



(b)



(c)



(d)

Fig. 2.6 SDSQ partitions, codebooks, indices and MDSQ index pairs associated with IA schemes illustrated in Fig. 2.5.

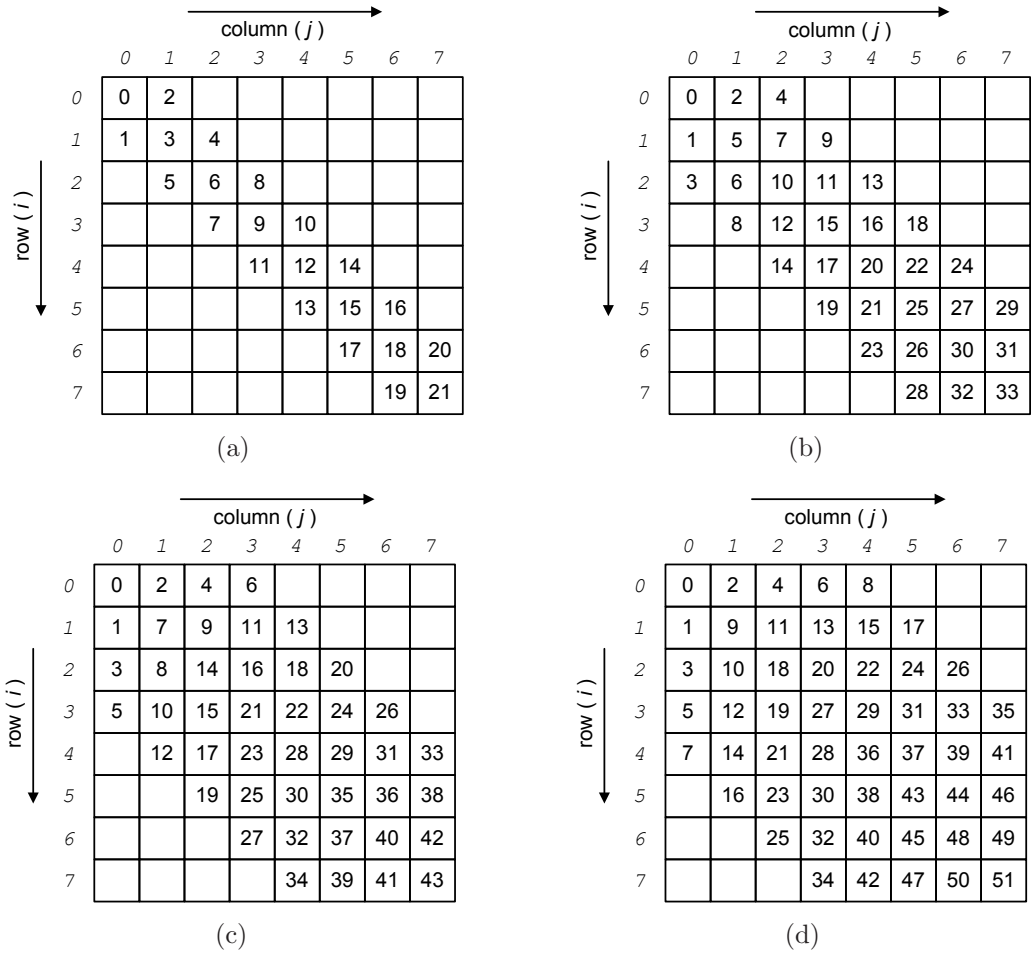


Fig. 2.7 Modified nested index assignment (MNIA) for $R_s = 3$, (a) $k = 1$, $L = 22$; (b) $k = 2$, $L = 34$; (c) $k = 3$, $L = 44$; and (d) $k = 4$, $L = 52$.

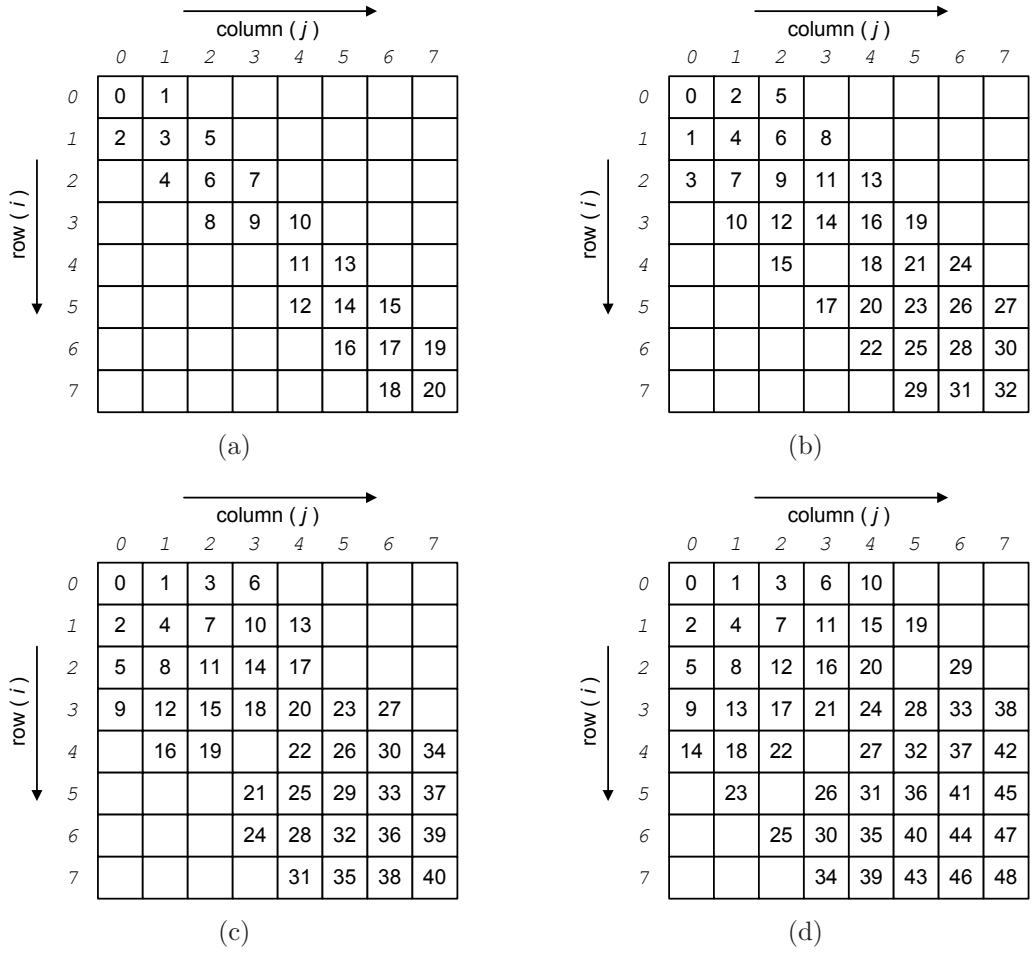


Fig. 2.8 Modified linear index assignment (MLIA) for $R_s = 3$, (a) $k = 1$, $L = 21$; (b) $k = 2$, $L = 33$; (c) $k = 3$, $L = 41$; and (d) $k = 4$, $L = 49$.

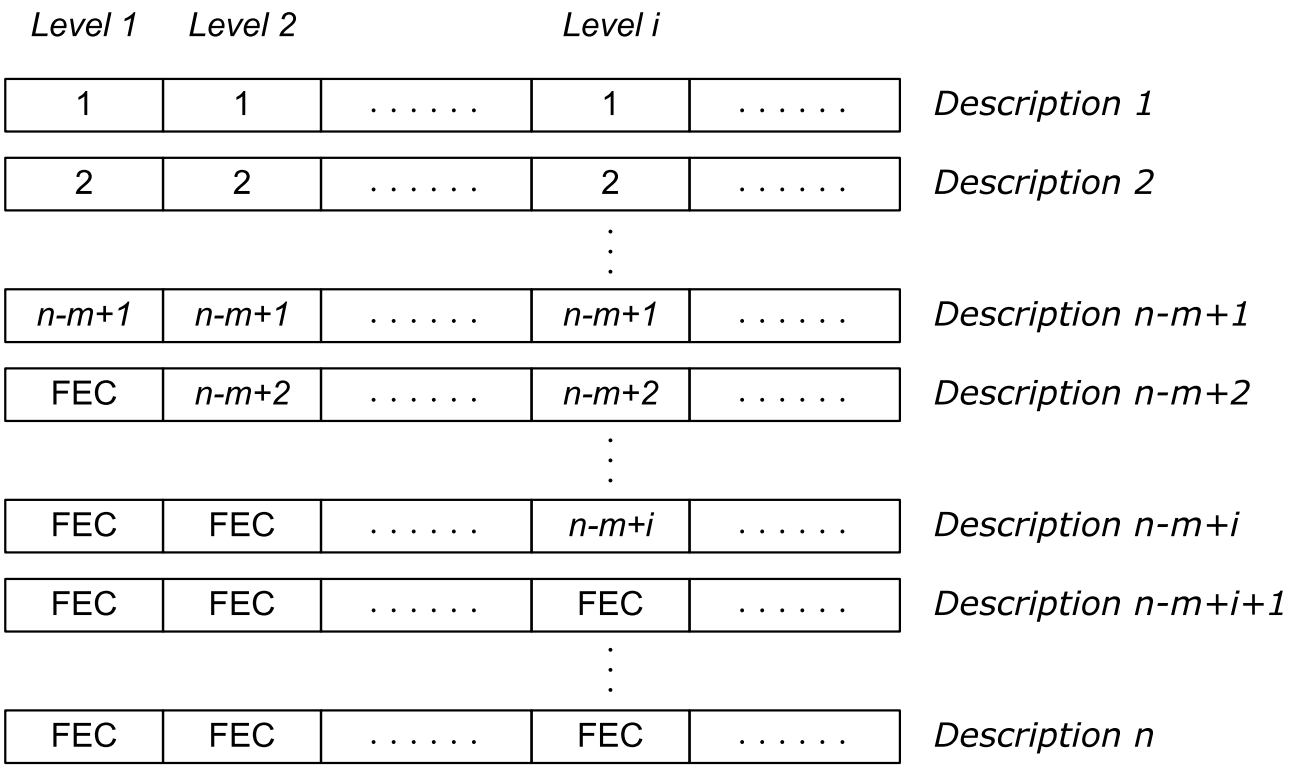


Fig. 2.9 FEC-based MDC: The progressive source bit streams are partitioned into m quality levels protected by RS codes, and grouped into n descriptions.

Chapter 3

Error Resilient Multiple Description Coding

As mentioned in Chapter 2, the MDC was originally proposed to provide protection against channel failures, and, further, packet losses over packet networks. We propose the error resilient multiple description coding (ERMDC) to combat both packet losses and bit errors. In this chapter, after a brief overview of the proposed ERMDC principle, the optimum ERMDC decoder is obtained. Experimental results indicate that, with similar redundancy, the ERMDC decoder outperforms the classical MDSQ decoder in the presence of bit errors[1].

3.1 Introduction

The encoder of the MDC decomposes source samples into two or more correlated descriptions, and transmits them independently over vulnerable channels. MDC decoders reconstruct the received signals with acceptable quality from each individual description, and finer quality from more descriptions. In this work, we only consider the situation with two descriptions. A classical MDC system with two channels and three decoders is illustrated in Fig. 1.1.

In a single description (SD) quantizer, the index assignment (IA) is a one-to-one mapping between quantizer outputs and a series of integers, called indices. In a quantizer-based MDC system, such as the MDSQ [18], the IA is a one-to-one mapping between SD indices

[1]This work has been presented at the 13th IEEE International Conference on Image Processing (ICIP'06), Oct. 2006 [22], and in IEEE Transactions on Signal Processing, Aug. 2008 [24].

and MD index pairs from multiple descriptions. It splits a single description into multiple descriptions with approximately equal and minimized side distortions. IA methods for MDSQ design was described by Vaishampayan [18]. An IA approach for more than two descriptions was developed by Berger-Wolf and Reingold [19]. Gortz and Leelapornchai presented a practically feasible IA algorithm for an MDVQ with an arbitrary number of descriptions [83]. A refined IA method was proposed by Balogh and Csirik to obtain two well-balanced descriptions [82].

In a conventional MDC system, all channels are assumed to have an on/off characteristic, i.e., data is either received correctly or lost completely [18]–[21]. However, due to noisy wireless channels, received descriptions usually contain bit errors. Channel codes or FEC codes are applied to accomplish average reliability in delay-constrained transmissions, such as image [48] and video communications [86].

Since the error correction capability of applied channel codes is usually limited and fixed, not all bit errors can be corrected. Therefore, after decoding FEC codes in the two descriptions, three scenarios usually exist at the receiver end:

- *Scenario I*: both descriptions are error-free;
- *Scenario II*: one description is error-free, but not both;
- *Scenario III*: both contain errors.

The conventional MDC system deals with *Scenario I* by using the central decoder. The achieved distortion is called the central distortion. In *Scenario II*, the erroneous description is thrown away. The correct description is decoded by a side decoder. The resulting side distortion is much higher than the central distortion. In Chapters 3 and 4, the error resilient multiple description coding (ERMDC) is developed to deal with *Scenario I* and *II* by utilizing the inherent redundancy and dependency among multiple descriptions to improve the capability of error detection and, furthermore, reduce the distortion of the reconstructed signals. This new technique is related to the noisy channel quantizer. The existing work on the IA in an MDC system only used natural binary labelling of the chosen index pairs. The minimum Hamming distance between two binary representations of selected index pairs is one bit. Therefore, as a result of one-bit transmission error, a selected index pair may be changed to either an unused index pairs, which is a detectable error, or another used index pair, which cannot be detected. In addition to achieve balanced descriptions,

an ERMDC encoder is proposed to improve the error detection capability of the ERMDC decoder. Since the ERMDC does not utilize such extra redundancy as channel codes and iterative decoding techniques, the computational complexity of the ERMDC is lower than those of [87], [88].

For *Scenario III*, the noisy channel quantizer, channel coding and the dependency between descriptions can be utilized to combat error-prone channels. The technique of channel-optimized quantizer for noisy channels was applied in the MDC system. An IA algorithm was developed by Yahampath for an MDVQ so as to minimize the overall distortion in the presence of packet loss [20]. An extended multiple-channel optimized quantizer design was applied by Zhou and Chan to compensate the distortion introduced by symbol erasures [21]. However, mismatching parameters resulted in serious performance degradation. On the other hand, with the help of extra redundancy, such as channel coding, the dependency between descriptions was utilized to combat bit errors in both descriptions. Combined with channel coding, multiple description block coding was studied by Barros et al. by using turbo cross decoding [87]. With the aid of index mapping, Barros et al. decoded the received bit flow over slow fading channels. Reconstruction levels of unused index pairs were replaced by the mean of the source data, which is a very coarse approximation. Similarly, characteristics of MD index mapping were also observed and exploited in [21], [88]. A Bayesian network was applied by Guionnet et al. to decode received data through noisy channels by using the dependency among MDSQ index pairs and variable length coding [88].

The main contribution of Chapters 3 and 4 is to develop a technique of designing an ERMDC system. The design procedure of an ERMDC consists of developing a robust MD encoder and an enhanced central decoder, called an *ERMDC encoder* and *decoder*, respectively. This ERMDC encoder-decoder pair is called an *ERMDC codec*. When used with the traditional MDC encoder, the ERMDC decoder alone exploits the inherent redundancy and dependency among multiple descriptions to achieve higher error tolerance and lower distortion than existing MDC encoder-decoder pairs. By enlarging the minimum Hamming distance, h_{min}, between binary representations of any two selected index pairs, the ERMDC encoder enhances the error detection capability of the ERMDC decoder. A genetic algorithm (GA) is used to design the IA scheme applied in the ERMDC encoder. Experimental results show that the ERMDC codec outperforms both the conventional MDC encoder-decoder pair and the pair of the conventional MDC encoder and the

ERMDC decoder.

This chapter is organized as follows: In Section 3.2, the ERMDC is briefly introduced as a big picture before describing details. The proposed ERMDC decoder is discussed in Section 3.3. Analytical and experimental results are shown in Section 3.4, followed by a discussion and conclusion in Section 3.5.

3.2 Error Resilient Multiple Description Coding

We develop here the concept of the ERMDC by using the MDSQ to generate two descriptions. We focus on resolution-constrained MDSQs, although the principles underpinning the ERMDC (decoding a corrupted description by using a correctly received one) can be exploited with other MD systems, including MD vector quantizers, and with more than two descriptions. In this section, after summarizing the MDSQ, we will introduce the basic idea of the ERMDC.

3.2.1 The Multiple Description Problem Revisited

In the MDSQ described in Section 2.3, the source samples are decomposed into index pairs at R_s bpss per channel. Each description is sent over a channel. Channel conditions are available at the decoder, but not at the encoder. If both channels work, the central decoder reconstructs the transmitted signals by using two received descriptions. If one channel fails, a side decoder reproduces the transmitted signals by using the received description. The objective of designing a pair of encoder and central decoder is to minimize the average central distortion when both channels work, subject to constraints on the maximum average side distortion and the equality of two side distortions when only one channel works.

If an MDSQ index pair (i, j) is transmitted over noisy channels, $\hat{i}$ and $\hat{j}$ are obtained at the receiver end after decoding FEC codes. In this chapter, we will discuss how to deal with transmission errors that incur received MD index pairs $(\hat{i}, \hat{j})$ cannot be mapped to SDSQ indices l, specifically, $l \neq a^{-1}(\hat{i}, \hat{j})$.

Let the central decoder be denoted as $g_0 : (\hat{i}, \hat{j}) \longmapsto \mathbb{R}$, and two side decoders be denoted as $g_1 : \hat{i} \longmapsto \mathbb{R}$ and $g_2 : \hat{j} \longmapsto \mathbb{R}$, respectively. So the corresponding reproduction levels are $g_0(\hat{i}, \hat{j})$, $g_1(\hat{i})$ and $g_2(\hat{j})$, accordingly. For *Scenario I*, where both descriptions are correct, i.e., $i = \hat{i}$ and $j = \hat{j}$, $l = a^{-1}(\hat{i}, \hat{j})$ and, then, $g_0(\hat{i}, \hat{j}) = g_0(i, j) = c_l$. For *Scenarios II* and *III*, where at least one description carries errors, i.e., $i \neq \hat{i}$ or $j \neq \hat{j}$, $g_0(\hat{i}, \hat{j}) \neq c_l$.

In this work, we use mean square error (MSE) to evaluate the distortion between the source and the reconstructed signals. The square error between two real values is denoted by $d(\cdot,\cdot)$, i.e., $d(x,y)=(x-y)^2$. Let $\hat{X}$ represent the output of the central decoder, $\hat{X}^{(1)}$ and $\hat{X}^{(2)}$ represent the outputs of side decoders, respectively. Accordingly, the average central distortion d_0 is given by

$$D_0 = E\Big\{d(X,\hat{X})\Big\} = \sum_{l\in\mathcal{L}}\int_{A_l} d\big(x, g_0(\hat{i},\hat{j})\big) f_X(x)dx, \tag{3.1}$$

and side distortions D_1 and D_2 are given by

$$D_1 = E\Big\{d\big(X,\hat{X}^{(1)}\big)\Big\} = \sum_{i\in\mathcal{I}}\int_{A_i^{(1)}} d\big(x, g_1(\hat{i})\big) f_X(x)dx, \tag{3.2}$$

$$D_2 = E\Big\{d\big(X,\hat{X}^{(2)}\big)\Big\} = \sum_{j\in\mathcal{J}}\int_{A_j^{(2)}} d\big(x, g_2(\hat{j})\big) f_X(x)dx. \tag{3.3}$$

And the average side distortion d_s is given by (2.22).

When one description is correctly received and the other one is lost, i.e., $\hat{i}=i$ or $\hat{j}=j$, $g_1(\hat{i})=c_i^{(1)}$ and $g_2(\hat{j})=c_j^{(2)}$. That is to say, (3.2) and (3.3) are equivalent to (2.20) and (2.21), respectively. However, in *Scenarios II* and *III*, $g_0(\hat{i},\hat{j})\neq g_0(i,j)$, when either $\hat{i}\neq i$ or $\hat{j}\neq j$. Therefore, the ERMDC decoder is designed to minimize $E\Big\{d\big(g_0(i,j),g_0(\hat{i},\hat{j})\big)\Big\}$.

3.2.2 ERMDC Decoder

For *Scenario I* and, especially, *Scenario II*, we propose an ERMDC decoder to utilize the residual information in the erroneous description. By referring to the correct description, the ERMDC decoder detects symbol errors in the erroneous description and, then, estimates the corresponding reconstruction levels so as to minimize the distortion caused by those errors. The structure of an ERMDC decoder with two channels and three receivers is illustrated in Fig. 3.1.

In *Scenario II*, if Description 1 is correct and Description 2 contains errors, symbol errors in Description 2 are detected by referring to Description 1, as demonstrated in Fig. 3.2. Here, we use classic MDSQ IA schemes [18] to explain how this decoder detects errors. In Fig. 2.8(b), the SDSQ index *0* is assigned to $(0,0)$. If $(0,0)$ is transmitted, and $(0,1)$ is obtained after channel decoding at the receiver end, we cannot tell if $(0,1)$ is right or wrong,

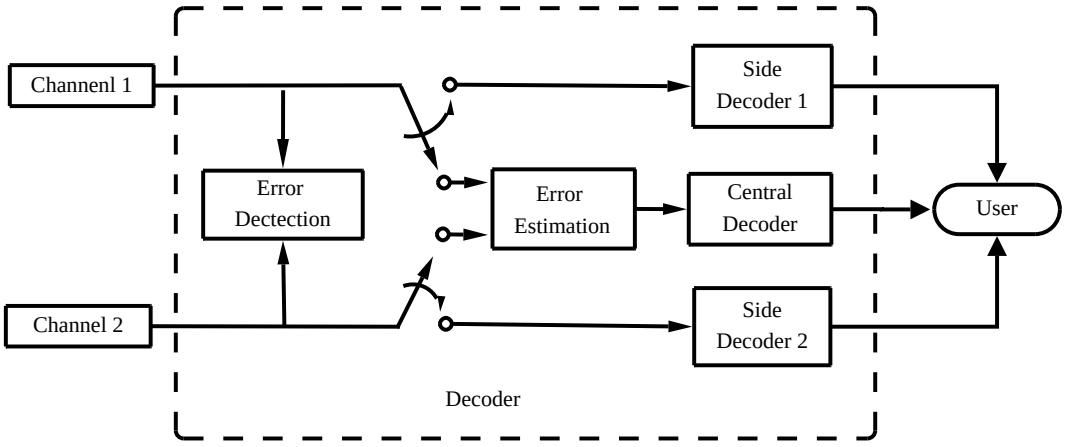


Fig. 3.1 General architecture of an ERMDC decoder.

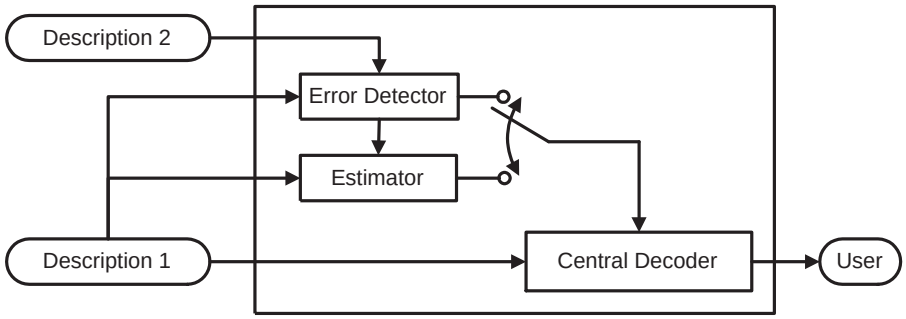


Fig. 3.2 The ERMDC decoder with corrupted Description 2 and correct Description 1.

since it is assigned to another SDSQ index *2*. This is an *undetectable error*. Nevertheless, if the obtained index pair is $(0, 5)$, we know that this is an error, because $(0, 5)$ is not mapped to an SDSQ index. This is a *detectable error*. Furthermore, if we know that Description 1 is right and Description 2 incurs some errors in decoding FEC code, *5* is determined as an error. Thereafter, we estimate the output value by using the methods described in Section 3.3.

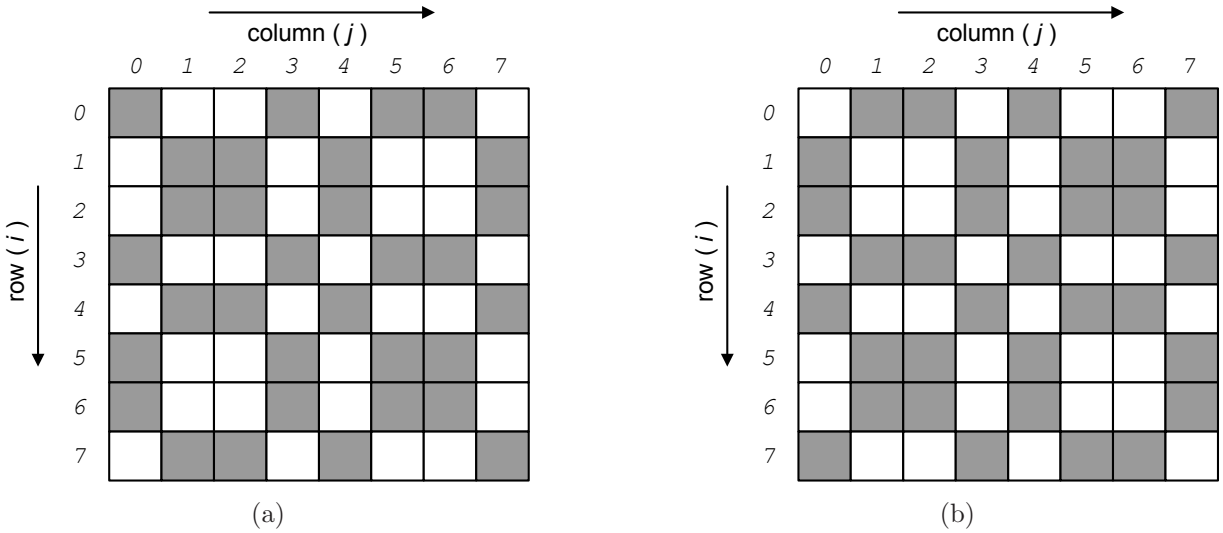


Fig. 3.3 Gray cells correspond to the valid index pairs in the selected IA schemes. (a) and (b) are two equivalent schemes of the robust IA with $R_s = 3$, $N = 32$, and $h_{min} = 2$.

3.2.3 ERMDC Encoder

Let $\mathbf{i}$ and $\mathbf{j}$ denote binary representations of $i \in \mathcal{I}$ and $j \in \mathcal{J}$, respectively, i.e.,

$$\mathbf{i} = [i_{(R_s-1)} \cdots i_1 i_0] \tag{3.4}$$

$$\mathbf{j} = [j_{(R_s-1)} \cdots j_1 j_0] \tag{3.5}$$

$$\tag{3.6}$$

are two binary R_s-tuples, where $i_k, j_k \in \{0, 1\}$, $k = 0,\ 1,\ \cdots,\ R_s - 1$. $i_{(R_s-1)}$ and $j_{(R_s-1)}$ are the most significant bits (MSBs), i_0 and j_0 are the least significant bits (LSBs) of $\mathbf{i}$ and $\mathbf{j}$. $\mathbf{i}$ and $\mathbf{j}$ are combined to form a $2R_s$-bit codeword $\mathbf{w}$:

$$\mathbf{w} \stackrel{\triangle}{=} [\mathbf{i}, \mathbf{j}] = [i_{(R_s-1)} \cdots i_1 i_0 j_{(R_s-1)} \cdots j_1 j_0]. \tag{3.7}$$

In the MDSQ, when $\mathbf{w}$'s are encoded in natural binary encoding as usual, any number of bit errors may result in undetectable errors. For example, in Fig. 2.8(b), a transmitted codeword [000001] or $(0, 1)$ may be changed to [001001] or $(1, 1)$ by a one-bit error. Since $(0, 1)$ and $(1, 1)$ are mapped to SDSQ index *2* and *4* respectively, *2* will be decoded as *4*

as the result of an undetectable error.

A robust IA algorithm is proposed for the ERMDC encoder to help the ERMDC decoder to achieve a higher ability of error detection. This IA approach assigns $\mathbf{w}$ to l with larger minimum Hamming distance, h_{min}, between any two $\mathbf{w}$'s. Thus, whenever the number of bit errors in a $\mathbf{w}$ is less than h_{min}, this $\mathbf{w}$ can be detected as an error. Figure 3.3 gives examples with $R_s = 3$, $N = 32$, $h_{min} = 2$. Index pairs corresponding to grayed cells are mapped to SDSQ indices. In this way, without using extra redundancy, the ERMDC decoder can detect all one-bit errors. Design details will be provided in the next chapter.

3.3 Design of ERMDC Decoder

Since the ERMDC decoder can work with the classical MDC encoder, we introduce the ERMDC decoder first in this section. The design goal is to find an optimal central decoder g_0 to minimize D_0 in *Scenarios I* and *II*. After deriving two optimal ERMDC decoding methods, a suboptimal decoder with lower computational complexity is also developed.

For a given $i \in \mathcal{I}$, the set of j that leads to a mapped index pair (i, j) is defined as

$$\mathcal{J}_i \stackrel{\triangle}{=} \{j \in \mathcal{J} : ij \in \mathcal{L}\}; \tag{3.8}$$

the set of j that does not lead to a mapped index pair (i, j) is defined as

$$\bar{\mathcal{J}}_i \stackrel{\triangle}{=} \{j \in \mathcal{J} : ij \notin \mathcal{L}\}. \tag{3.9}$$

Then, $\mathcal{J}_i \cup \bar{\mathcal{J}}_i = \mathcal{J}$, and $\mathcal{J}_i \cap \bar{\mathcal{J}}_i = \emptyset$. For example, in Fig. 2.8(a), at the second row, $\mathcal{J}_1 = \{0, 1, 2\}$ and $\bar{\mathcal{J}}_1 = \{3, 4, 5, 6, 7\}$.

Similarly, for a given $j \in \mathcal{J}$, two sets are

$$\mathcal{I}_j \stackrel{\triangle}{=} \{i \in \mathcal{I} : ij \in \mathcal{L}\} \tag{3.10}$$

and

$$\bar{\mathcal{I}}_j \stackrel{\triangle}{=} \{i \in \mathcal{I} : ij \notin \mathcal{L}\}, \tag{3.11}$$

where $\mathcal{I}_j \cup \bar{\mathcal{I}}_j = \mathcal{I}$ and $\mathcal{I}_j \cap \bar{\mathcal{I}}_j = \emptyset$.

According to whether the ERMDC decoder can detect them or not, transmission errors

are classified into two types: *detectable* and *undetectable* errors. D_d and D_u are expected distortions caused by detectable and undetectable errors, respectively. Therefore, the overall distortion caused by transmission errors, D_e, is given by

$$D_e = D_u + D_d, \tag{3.12}$$

where

$$D_e = \sum_{l \in \mathcal{L}} P_q(l) D_e(l), \tag{3.13}$$

$$D_u = \sum_{l \in \mathcal{L}} P_q(l) D_u(l), \tag{3.14}$$

$$D_d = \sum_{l \in \mathcal{L}} P_q(l) D_d(l). \tag{3.15}$$

$P_q(l) = P\{x \in A_l\}$ is the probability of the source falling in the partition A_l. $D_e(l)$, $D_u(l)$ and $D_d(l)$ are expected distortions of the source in the partition A_l due respectively to all, undetectable and detectable transmission errors. When an index pair (i, j) is transmitted, the corresponding received pair is $(\hat{i}, \hat{j})$. A transmission error can be detected as long as $\hat{j} \in \bar{\mathcal{J}}_{\hat{i}}$ or $\hat{i} \in \bar{\mathcal{I}}_{\hat{j}}$. Thus, for a given l, (i, j) and the corresponding received $(\hat{i}, \hat{j})$,

$$D_e(l) = D_e(i,j) = \begin{cases} D_d(l) = D_d(\hat{i}, \hat{j}), & \text{if } (\hat{i}, \hat{j}) \text{ is a detectable error;} \\ D_u(l) = D_u(\hat{i}, \hat{j}), & \text{otherwise.} \end{cases} \tag{3.16}$$

The overall expected distortion caused by quantization (i.e., without transmission errors), D_q, is given by

$$D_q = \sum_{l \in \mathcal{L}} P_q(l) D_q(l), \tag{3.17}$$

where

$$D_q(l) = \int_{A_l} d(x, c_l) f_X(x) dx \tag{3.18}$$

is the distortion contribution from the partition A_l.

3.3.1 Necessary Conditions for Optimal Decoder Design

Assuming high-rate quantization, the overall expected distortion contribution from the partition A_l is given by

$$\begin{aligned} D_0(l) &= \frac{\int_{A_l} f_X(x)\; d\big(x, g_0(\hat{i},\hat{j})\big)dx}{\int_{A_l} f_X(x)dx} \\ &\approx \frac{f_X(c_l)\int_{A_l} d\big(x, g_0(\hat{i},\hat{j})\big)dx}{f_X(c_l)\Delta_l} \\ &= \frac{1}{\Delta_l}\int_{A_l}\big(x - g_0(\hat{i},\hat{j})\big)^2 dx \\ &= \frac{1}{\Delta_l}\int_{A_l}\Big(x - c_l + c_l - g_0(\hat{i},\hat{j})\Big)^2 dx \\ &= \frac{\Delta_l^2}{12} + \Big(c_l - g_0(\hat{i},\hat{j})\Big)^2 \\ &= D_q(l) + D_e(l), \end{aligned} \tag{3.19}$$

where $\Delta_l = \int_{A_l} dx$, and $D_q(l) \approx \frac{\Delta_l^2}{12}$ in the high-rate system. Then, the overall average central distortion D_0 is given by

$$D_0 = \sum_{l\in\mathcal{L}} P_q(l) D_0(l) \tag{3.20}$$

$$= \sum_{l\in\mathcal{L}} P_q(l)\big[D_q(l) + D_e(l)\big]$$

$$= D_q + D_e \tag{3.21}$$

$$= D_q + D_u + D_d. \tag{3.22}$$

In an ideal system, all errors are found, so that the lower bound is given by

$$D_{low} = \lim_{D_u\to 0} D_0 = D_q + D_d \tag{3.23}$$

In (3.22), only D_d can be minimized to reduce D_0 by optimizing the reconstruction levels $g_0(\hat{i},\hat{j})$ of the detectable errors.

For *Scenario II*, all bit errors are assumed independent and identically distributed (i.i.d.). For a given bit error rate (BER) p, the transfer probability from a b-bit symbol j

to $\hat{j}$ is

$$P_e(\hat{j}|j) = p^h(1-p)^{R_s-h}, \tag{3.24}$$

where h is the Hamming distance between j and $\hat{j}$.

Without losing generality, we assume that after FEC decoding, Description 1 is correct; however, bit errors in Description 2 are detected, but not corrected, i.e., $\hat{i} = i$, $\hat{j} \neq j$, and $\hat{j} \in \bar{\mathcal{J}}_i$. Therefore, the expected distortion $D_d^{(1)}(i,\hat{j})$ introduced by the detected error $(i,\hat{j})$ is given by

$$\begin{aligned} D_d^{(1)}(i,\hat{j}) &= E\left\{d\big(g_0(i,j), g_0(\hat{i},\hat{j})\big) \mid \hat{i} = i,\ \hat{j} \neq j,\ \ \hat{j} \in \bar{\mathcal{J}}_i\right\} \\ &= \sum_{j \in \mathcal{J}_i} P_q(ij) P_e(\hat{j}|j)\ d\big(g_0(i,j), g_0(i,\hat{j})\big). \end{aligned} \tag{3.25}$$

Thus,

$$D_d = \sum_{i \in \mathcal{I}} \sum_{\hat{j} \in \bar{\mathcal{J}}_i} D_d^{(1)}(i,\hat{j}). \tag{3.26}$$

In order to minimize D_d, the reconstruction level $g_0(i,\hat{j})$ is estimated by $e^{(1)}(i,\hat{j})$ in terms of all possible transmitted index pairs (i,j), $j \in \mathcal{J}_i$. By solving

$$\left.\frac{d\, D_d^{(1)}(i,\hat{j})}{d\, g_0(i,\hat{j})}\right|_{g_0(i,\hat{j}) = e_{opt}^{(1)}(i,\hat{j})} = 0, \tag{3.27}$$

the optimal value is given by

$$e_{opt}^{(1)}(i,\hat{j}) = \frac{\sum_{j \in \mathcal{J}_i} P_q(ij) P_e(\hat{j}|j)\ g_0(i,j)}{\sum_{j \in \mathcal{J}_i} P_q(ij) P_e(\hat{j}|j)}. \tag{3.28}$$

Another way to estimate $g_0(i,\hat{j})$ is to use the average reconstruction level $\bar{e}_i^{(1)}$ corresponding to the MDSQ index i. The expected distortion caused by detectable errors associated with the MDSQ index i is given by

$$\begin{aligned} D_d^{(1)}(i) =& E\left\{d\big(g_0(i,j), g_0(\hat{i},\hat{j})\big) \mid \hat{i} = i,\ \hat{j} \neq j,\ \ \hat{j} \in \bar{\mathcal{J}}_i\right\} \\ =& \sum_{j \in \mathcal{J}_i} P_q(ij) \sum_{\hat{j} \in \bar{\mathcal{J}}_i} P_e(\hat{j}|j)\ d(g_0(i,j), g_0(i,\hat{j})). \end{aligned} \tag{3.29}$$

Thus,

$$D_d = \sum_{i \in \mathcal{I}} D_d^{(1)}(i). \tag{3.30}$$

By solving

$$\left. \frac{dD_d^{(1)}(i)}{dg_0(i,\hat{j})} \right|_{g_0(i,\hat{j})=\bar{e}_{i_{opt}}^{(1)}} = 0, \tag{3.31}$$

the optimal $\bar{e}_i^{(1)}$ is given by

$$\bar{e}_{i_{opt}}^{(1)} = \frac{\sum_{j \in \mathcal{J}_i} \sum_{\hat{j} \in \bar{\mathcal{J}}_i} P_q(ij) P_e(\hat{j}|j)\, g_0(i,j)}{\sum_{j \in \mathcal{J}_i} \sum_{\hat{j} \in \bar{\mathcal{J}}_i} P_q(ij) P_e(\hat{j}|j)}. \tag{3.32}$$

According to (3.28) and (3.32), if probabilities of transmitting $j \in \mathcal{J}$ and receiving $\hat{j} \in \bar{\mathcal{J}}_i$ are the same, i.e., $P_e(j_1|j) \approx P_e(j_2|j)$, $j_1 \neq j_2$, and $j_1, j_2 \in \bar{\mathcal{J}}_i$, a suboptimal value is obtained by:

$$\begin{aligned} \bar{e}_{sub}^{(1)}(i,\hat{j}) &\approx \frac{\sum_{j \in \mathcal{J}_i} P_q(ij)\, g_0(i,j)}{\sum_{j \in \mathcal{J}_i} P_q(ij)} \\ &= g_1(i). \end{aligned} \tag{3.33}$$

3.3.2 Optimal and Suboptimal Decoders

In this work, we propose three methods to estimate reconstruction levels of erroneous MD index pairs to decrease D_d:

1. *Optimal*: Assuming that the a-priori source probabilities $P_q(l)$ or $P_q(ij)$ and the BER p are known at the receiver end, reconstruction levels of unused index pairs are calculated by using (3.32) or (3.28). We refer to the optimal method using (3.32) as *Optimal method 1*, which provides optimal solutions in the sense of average performance. *Optimal method 2* that uses (3.28) provides optimal solutions when the design parameters are very well matched to actual channel conditions.

2. *Suboptimal*: In the light of (3.33), reconstruction levels of unused index pairs are replaced by $g_1(i)$.

According to (3.25) (3.26) and (3.29) (3.30),

$$D_d^{(1)}(i) = \sum_{\hat{j} \in \bar{\mathcal{J}}_i} D_d^{(1)}(i, \hat{j}). \tag{3.34}$$

Hence, the performance achieved by Optimal method 1 is the average result of the performance achieved by Optimal method 2. That is to say, if the design parameters match the channel conditions very well, Optimal method 2 gives better performance; otherwise, Optimal method 1 is better.

The optimal ERMDC decoders require knowledge about the source and channels. However, the suboptimal decoder does not require such information. In Section 3.4, we will compare the performance of these three decoders.

3.4 Analytical and Experimental Results

At the transmitter end, entropy coding, such as Huffman coding and arithmetic coding, is usually applied to compress the quantized data further. At the receiver end, because of entropy coding, a single bit error may incur that the remaining bits are decoded incorrectly. Therefore, in the following experiments, we will test the performance of the proposed ERMDC in two situations: (i) no entropy coding is used, and (ii) entropy coding is applied. In the first situation, only transmission errors, which are simulated by separate bit errors, are considered. The statistics of errors are similar to the design parameters, in particular, the BER. In the second situation, incorrectly entropy decoding after receiving a bit error is simulated by burst bit errors. This situation is also used to test the sensitivity of proposed methods to parameter mismatch.

Analytical and experimental results are presented for i.i.d. Gaussian source samples with zero mean, unit variance and at the transmission rate of $R_s = 3$ bpss/channel. Since the ERMDC decoder can work with the traditional MDSQ encoder, the performance of the ERMDC decoder against separate and burst bit errors is presented in Sections 3.4.1 and 3.4.2. In the following experiments, 80000 samples are applied.

3.4.1 ERMDC Decoders Against Separate Bit Errors

Separate bit errors are independent, identically and uniformly distributed throughout the binary sequence of the description. Figure 3.4 shows analytical and experimental results of

MLIA$(3, 1, 21)$ and $(3, 2, 33)$, which are illustrated in Figs. 2.8(a) and (b).

In the existing work, if unused MD index pairs are found, the corresponding output levels are usually estimated by the mean of the source data [87]. The performance of this method is called *mean* in the figures. Curves plotted based on *Optimal methods 1* and *2* are obtained in terms of (3.32) and (3.28), respectively. The *suboptimal* curves of experiments are obtained by using (3.33). The central distortion D_0 and the side distortion D_s are shown as references. It shows that *Optimal method 2* achieves lower distortion than *Optimal method 1*. Even without knowledge about the source and channels, the *suboptimal* decoder achieves similar performance with *Optimal method 1*.

We define the *bit error tolerance* (BET) as the BER at which the distortion of a central decoder reaches D_s. Performance of ERMDC decoders degrades gracefully to D_s as the BER increases. Nevertheless, they achieve higher BET than *mean*. With higher redundancy, Fig. 3.4(a) shows that *MLIA*(3, 1, 21) gives higher BETs at 19.86% of *Optimal method 1*, and 21.63% of *Optimal method 2*, in contrast with 3.265% of *mean*. In Fig. 3.4(b), the BETs of *MLIA*(3, 2, 33), 16.72% of *Optimal method 1* and 17.63% of *Optimal method 2*, are still much higher than 8.565% of *mean*. In general, as the redundancy decreases, performance improvement achieved by the ERMDC decoder reduce. The reason is that less errors are detected with less redundancy.

3.4.2 ERMDC Decoders Against Burst Bit Errors

Experimental results associated with burst bit errors are shown in Fig. 3.5. Burst bit errors are simulated by inverting all bits from a certain position to the end of the binary sequence of the description. This position is determined by the ratio of the number of bit errors to the number of transmitted bits, i.e., the BER. Due to (3.23), ideally, the lower bound D_{low} can be achieved when $D_u = 0$.

As mentioned before, we are going to test the performance of the ERMDC on entropy encoded bitstreams or parameter mismatch. When the entropy decoder losses sync with the encoder, even though in some cases, some bits are still possibly correct, it is almost impossible to correctly decode the remaining part of the bitstream after losing sync. Thus, here inverting all bits after a certain position is simulating the worst case of losing sync in entropy decoding.

In Fig. 3.5, curves of D_{low}, called *lower bound (Method 1)*, *lower bound (Method 2)* and *lower bound (suboptimal)*, are obtained by replacing reconstruction levels of all er-

rors, including undetectable and detectable errors, with the optimal and suboptimal values accordingly. According to (3.22), the overall distortion is caused by quantization, undetectable and detectable errors. Therefore, the difference between the results and lower bounds is caused by undetectable errors. For $MLIA(3, 1, 21)$, because only two pairs of codewords, $(3, 4)$ and $(4, 4)$, $(3, 3)$ and $(3, 4)$, have three-bit Hamming distance, the difference between the experimental results and the lower bounds is marginal. It confirms that if there exist fewer undetectable errors, better performance can be achieved. Similarly, as the redundancy decreases, the performance improvements become smaller.

Because ERMDC decoders are not designed for burst bit errors, the design parameters do not match the bit error patterns in such circumstances. It is necessary to test whether the ERMDC decoders still keep their improvements under such circumstances. It indicates that they can highly tolerate burst bit errors. With the help of some redundancy, even at BER 50%, the achieved distortions of ERMDC decoders are still lower than D_s; whereas, curves of *mean* touch D_s at BERs lower than 20%. However, the performance of *Optimal method 2* is much worse than *Optimal method 1* and *suboptimal*. The reason is that *Optimal method 2* is more sensitive to parameter mismatch, even though it gives more accurate estimation with perfect knowledge of channels.

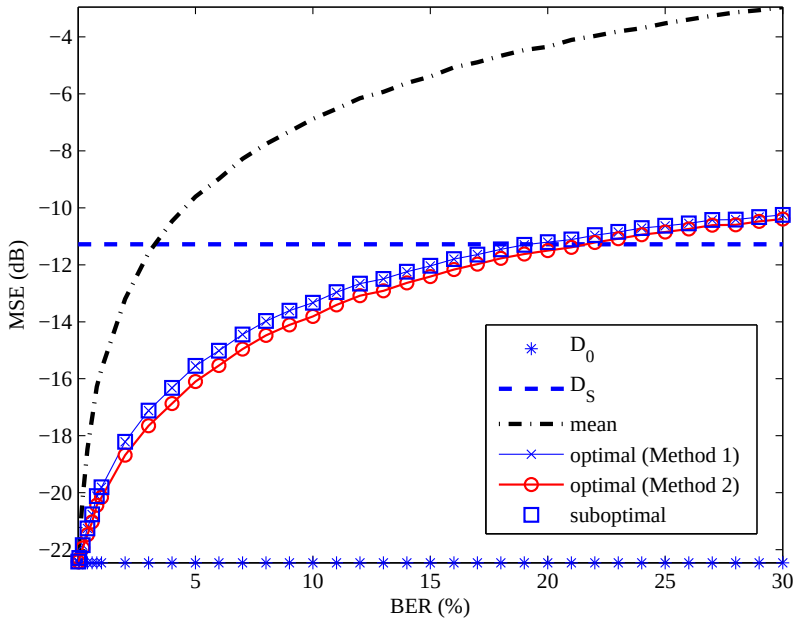
3.5 Discussion and Conclusion

In error-prone environment, instead of discarding a description that contains errors, we utilize the residual information in the corrupted description to achieve lower distortion than one single description does. After giving a short introduction of the ERMDC, we propose three ERMDC decoders to estimate the outputs of detectable errors.

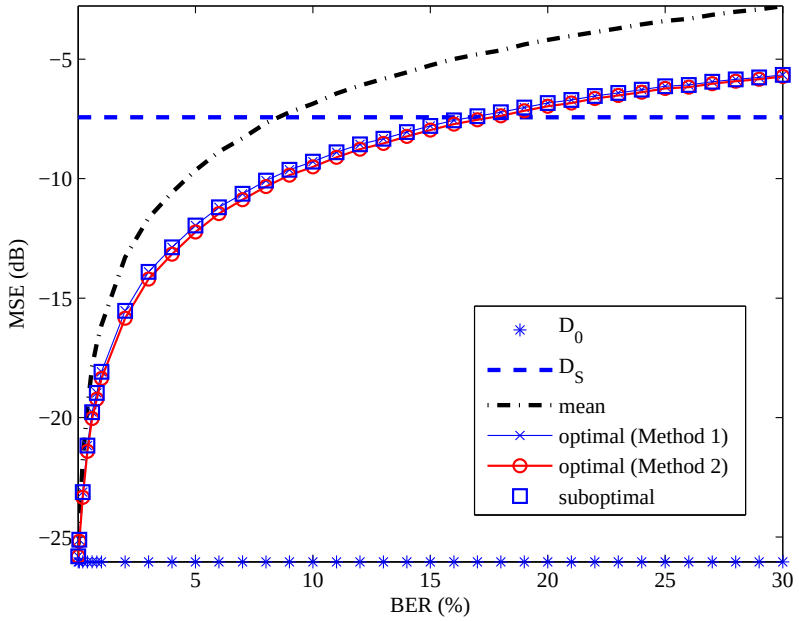
By estimating reconstruction levels of unused index pairs, ERMDC decoders outperform the conventional MDSQ decoder. ERMDC decoders achieve graceful performance degradation as the BER increases. However, the performance improvements reduce, as the redundancy decreases. Since the optimal ERMDC decoders are designed based on BERs, inaccurate BERs affect their performance, in particular, *Optimal method 2*. With lower computational complexity and without accurate knowledge of the source data and channel conditions, the suboptimal ERMDC decoder accomplishes very similar performance with the optimal. Therefore, the suboptimal decoder can be widely used in various scenarios.

From experimental results, we infer that a proper IA scheme may assist the ERMDC

decoder to provide more accurate estimation. Therefore, in the next chapter, we attempt to develop a procedure to find an IA scheme that is helpful in detecting errors.

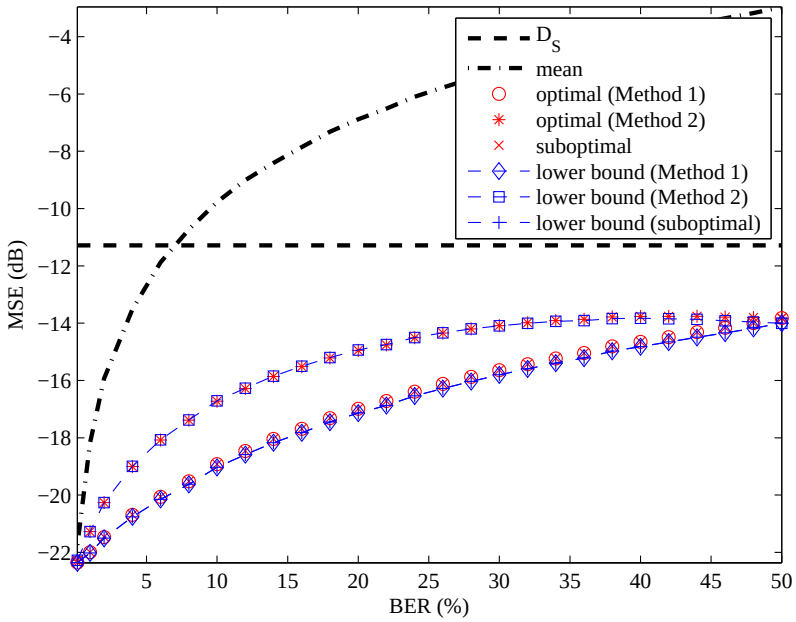


(a)

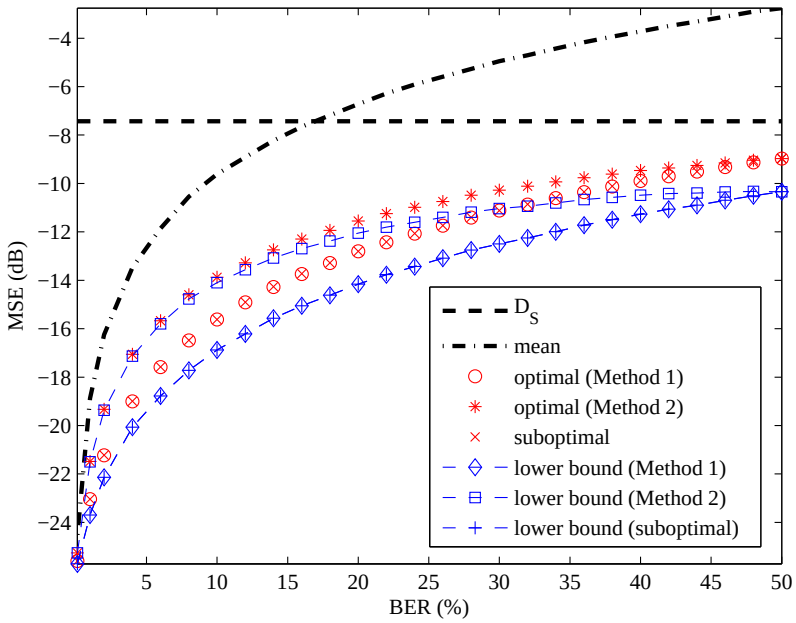


(b)

Fig. 3.4 Performance of ERMDC decoders associated with separate bit errors: (a) $MLIA(3, 1, 21)$, (b) $MLIA(3, 2, 33)$.



(a)



(b)

Fig. 3.5 Performance of the ERMDC decoder associated with burst bit errors: (a) $MLIA(3, 1, 21)$, (b) $MLIA(3, 2, 33)$.

Chapter 4

Design of ERMDC Encoder

After describing the decoder, in this chapter, we provide the design procedure of the ERMDC encoder. Experimental results show that, with the help of the ERMDC encoder, the performance of the ERMDC decoder is improved further[1].

4.1 Introduction

As mentioned in Section 2.1, the noisy channel quantizer was developed to achieve robustness against error-prone channels without the help of FEC coding. It is an approach of combining source and channel coding. Typical design methods of the noisy channel quantizer are the robust quantizer (RQ) and the channel-optimized quantizer (COQ). Compared with the RQ, the COQ needs more computations and provides lower performance, when channel conditions are different with its design parameters. Thus, we apply the RQ technique to realize the ERMDC encoder with low computational complexity and no extra redundancy.

Developing a robust IA scheme is the key step of designing a noisy channel quantizer. The best IA scheme provides the highest robustness against transmission errors. Although it is possible to find the optimal solution by using the exhaustive search, the cost is prohibitively high. Thus, a "close-to-optimal" IA scheme is usually found by using heuristic algorithms, such as the binary switching algorithm [32], the genetic algorithm (GA) [33], and the simulated annealing [21], [35]. In [33], experimental results indicated that the GA

[1]This work has been presented at the eighth International Workshop on Multimedia Signal Processing (MMSP'06), Oct. 2006 [23], and in IEEE Transactions on Signal Processing, Aug. 2008 [24].

found more accurate and more consistent solutions than the binary switching algorithm and the simulated annealing. Therefore, we use the GA in this chapter.

4.2 Design Procedure

Goals of designing an ERMDC encoder are (i) to find a robust IA scheme so as to improve the error detection capability of the ERMDC decoder; and (ii) to achieve two balanced descriptions with as low side distortions as possible. According to the ERMDC decoders described in Chapter 3, the higher error detection capability the decoder has, the more distortion is reduced. In order to improve the capability of detecting errors, we propose a novel method to find a robust IA scheme by increasing the minimum Hamming distance h_{min} between any two codewords. Then, based on the optimal partition and codebook of the SDSQ, the GA is applied to find an IA scheme to achieve two balanced descriptions with as low side distortions as possible. Thereafter, the algorithm developed in [18] is used to tune the partition and codebook of the selected IA scheme so as to reduce the side distortions further at the cost of slightly higher central distortion.

The design procedure of the ERMDC encoder is summarized in three steps:

1. Given an h_{min}, searching for qualified index pairs (i, j), the minimum Hamming distance between any two of which is h_{min}, within a $2^{R_s} \times 2^{R_s}$ IA matrix.

2. Assigning SDSQ indices l to selected index pairs (i, j) to achieve balanced descriptions with as low side distortions as possible.

3. Tuning the partition and codebook of the chosen IA scheme to fit the source data better by using the algorithm provided in [18].

In the following subsections, we will describe the algorithms for Step 1) and 2) in details, respectively. As for the algorithm for Step 3), it is developed by Vaishampayan [18] and briefly described in Section 2.3.2.

4.3 Searching for Qualified Index Pairs

An SDSQ index l is represented in m bits, i.e., the binary representation of l is $\mathbf{l} = [l_{m-1} \cdots l_0]$. By using an IA scheme, each SDSQ index $\mathbf{l}$ is mapped to a $2R_s$-bit codeword $\mathbf{w}$, where $2R_s \geq m$. Let $q = 2R_s - m$ denote the number of redundancy bits in each

Table 4.1 Algorithm to search qualified code words

$n \leftarrow 0$
FOR each $\mathbf{w}_0^{(n)} \in \mathcal{C}$
  $\mathbf{v} \leftarrow \mathbf{w}_0^{(n)}$
  $k \leftarrow 0$
  FOR each $\mathbf{v} \in \mathcal{C}$
    $\mathbf{v} \leftarrow \mathbf{w}_k^{(n)} + 1$
    FOR each $\mathbf{v} \in \mathcal{C}$ AND $\|\mathbf{v} - \mathbf{w}_t^{(n)}\| < h_{min}$, $\forall t, t = 0, \cdots, k$
      $\mathbf{v} \leftarrow \mathbf{v} + 1$
    END
    IF $\mathbf{v} \in \mathcal{C}$
      $k \leftarrow k + 1$
      $\mathbf{w}_k^{(n)} \leftarrow \mathbf{v}$
    END
  END
  $L^{(n)} \leftarrow k$
  $n \leftarrow n + 1$
END

codeword $\mathbf{w}$, which is defined as (3.7). If $q = 0$, no redundancy is provided. If $q > 0$, q-bit redundancy is provided in each $\mathbf{w}$.

Let $\mathcal{C} = \{0, 1\}^{2R_s}$ be the set of all possible codewords $\mathbf{w}$. In order to achieve the robustness against bit errors, we search for a subset $\mathcal{W} \subset \mathcal{C}$, so that

$$\mathcal{W} = \{\mathbf{w} \in \mathcal{C} : \forall \mathbf{v} \in \mathcal{W}, \mathbf{v} \neq \mathbf{w}, \|\mathbf{v} - \mathbf{w}\| \geq h_{min}\}, \tag{4.1}$$

where $\|\mathbf{v} - \mathbf{w}\|$ represents the Hamming distance between $\mathbf{v}$ and $\mathbf{w}$. The number of codewords $\mathbf{w} \in \mathcal{W}$ is L, and $\|\mathbf{w}\|$ represents the Hamming weight of $\mathbf{w}$.

Given R_s and h_{min}, we search $\mathcal{C}$ to find a satisfactory $\mathcal{W}$. The searching algorithm is described in Table 4.1. We use exhaustive search to find all possible n solutions. For example, if $R_s = 3$, at the first try, the first valid $\mathbf{w}$ is $\mathbf{w}_0 = [000000]$. Then every possible code word compares with $\mathbf{w}_0$, if the Hamming distance is equal or greater than h_{min}, this code word is kept as the second code word $\mathbf{w}_1$. Once again, the other combinations compare with $\mathbf{w}_0$ and $\mathbf{w}_1$, if both Hamming distances are equal or greater than h_{min}, this combination is kept as the third valid code word $\mathbf{w}_2$, and so on. At each time, the remaining

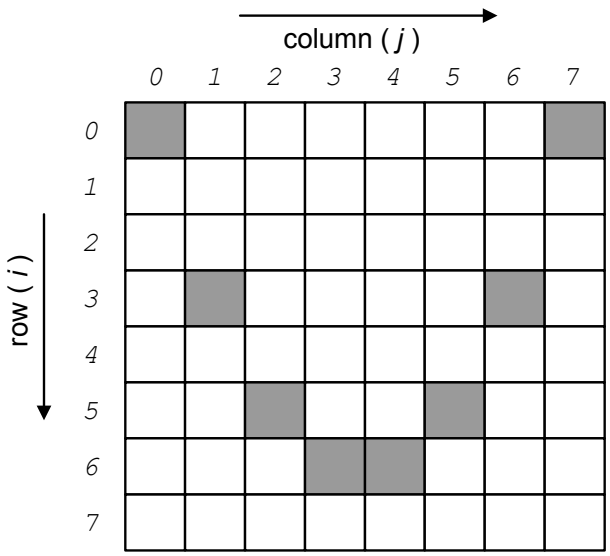


Fig. 4.1 A set of selected index pairs with $R_s = 3$, $N = 8$, and $h_{min} = 3$. Gray cells correspond to selected index pairs.

combinations compare with the valid code words, if all Hamming distances are equal or greater than h_{min}, this combination is kept as a valid code word. This kind of comparison terminates when no more combinations satisfy the requirement. In the nth solution, there are $L^{(n)}$ valid code words $\mathbf{w}$'s. Next iteration, $\mathbf{w}_0$ begins with [000001], until [1111111]. Of course, we can shorten this search by stopping iterations at the initial $\mathbf{w}_0 = [000111]$, because $[\mathbf{i}, \mathbf{j}]$ is actually equivalent to $[\mathbf{j}, \mathbf{i}]$.

After exhaustively searching $\mathcal{C}$, we choose $\mathcal{W}$ with the largest L. If there are more than one $\mathcal{W}$ with the largest L, the first one is selected. As examples, provided $R_s = 3$, solutions of $h_{min} = 2$ are shown in Fig. 3.3, one solution of $h_{min} = 3$ is shown in Fig. 4.1. We choose the first one for our experiments. Once $\mathbf{w}$'s are determined, they need to be assigned to the indices l's.

4.4 Index Assignment by Using Genetic Algorithm

The IA is a kind of permutation problem, in particular, an NP-complete problem. For $\mathcal{W}$ with L codewords, the total number of possible $\mathcal{W}$'s is $O(L!)$. Finding the best solution by using exhaustive search is impractical. A heuristic algorithms performs a deterministic search in a set of admissible configurations and often terminates in a local minimum. We apply the GA to find a "close-to-optimal" solution at a reasonable cost. The cost function

is used to evaluate solutions found in each iteration of the IA algorithm. The cost function is discussed first, followed by the description of the IA algorithm.

4.4.1 Cost Function

First, we describe the cost function for evaluating the candidate solutions in each generation (iteration). We consider the cost from two aspects, spreads and distortions. Distortions are defined in Chapter 3. Next, we give the definition of spreads.

Spreads

Unless otherwise specified, in this work, SDSQ indices l are naturally labeled. That is to say, a quantization partition associated with larger source values x is mapped to a larger index l.

Let

$$l_{max}^{(1)}(i) = \max_j a^{-1}(i,j), \tag{4.2}$$

$$l_{min}^{(1)}(i) = \min_j a^{-1}(i,j), \tag{4.3}$$

$$l_{max}^{(2)}(j) = \max_i a^{-1}(i,j), \tag{4.4}$$

$$l_{min}^{(2)}(j) = \min_i a^{-1}(i,j), \tag{4.5}$$

where $l_{max}^{(1)}(i)$ and $l_{min}^{(1)}(i)$ are the maximum and minimum SDSQ indices l associated with a given MD index i, respectively; $l_{max}^{(2)}(j)$ and $l_{min}^{(2)}(j)$ are the maximum and minimum SDSQ indices l associated with a given MD index j, respectively.

The spread $s^{(m)}(n)$ is defined as

$$s^{(m)}(n) = l_{max}^{(m)}(n) - l_{min}^{(m)}(n) + 1, \tag{4.6}$$

where $n = 0, 1, \cdots, M-1$, and $m = 1, 2$. For example, in Fig. 2.8(b), the spread associated with $j = 3$ is

$$\begin{aligned} s^{(2)}(3) &= \max_{i \in \mathcal{I}_3} a^{-1}(i,3) - \min_{i \in \mathcal{I}_3} a^{-1}(i,3) + 1 \\ &= 17 - 8 + 1 = 10. \end{aligned}$$

In terms of the definition of spreads, spreads are independent on characteristics of source signals and channel noise. If only the m-th description is received, the centroid of all SDSQ partitions corresponding to the MDSQ index n is used as the reconstruction level of n. Thus, the smaller a spread $s^{(m)}(n)$ is, the more accurate the reconstruction level is, in particular, in high rate systems.

$\mathcal{S}^{(m)}$ denotes the set of all spreads associated with the m-th description, where $m = 1, 2$. The average spread $\bar{s}^{(m)}$ associated with the m-th description is defined as

$$\bar{s}^{(m)} = E\Big\{\mathcal{S}^{(m)}\Big\}, \tag{4.7}$$

where $m = 1, 2$. In high rate systems,

$$\bar{s}^{(m)} = \frac{1}{M}\sum_{n=0}^{M-1} s^{(m)}(n). \tag{4.8}$$

The overall average spread $\bar{s}$ is given by

$$\bar{s} = \frac{\bar{s}^{(1)} + \bar{s}^{(2)}}{2}. \tag{4.9}$$

In high rate systems, $\bar{s}^{(m)}$ and $\bar{s}$ can be used as metrics of the corresponding side distortion and the average side distortion, respectively.

The maximum spread s_{max} is the maximum value of all row and column spreads in an IA scheme, i.e.,

$$s_{max} = \max_{m,n} s^{(m)}(n). \tag{4.10}$$

s_{max} is used to evaluate possible reconstruction distortions of an IA scheme. s_{max} of the IA scheme shown in Fig. 2.8(b) is 11. A smaller s_{max} suggests a lower side distortion. Similarly, the minimum spread

$$s_{min} = \min_{m,n} s^{(m)}(n). \tag{4.11}$$

The difference between s_{max} and s_{min} is

$$s_{dif} = s_{max} - s_{min}. \tag{4.12}$$

A smaller s_{dif} indicates less variance among spreads $s^{(m)}(n)$. It, furthermore, indicates

that the IA scheme is less sensitive and more robust to the statistics of source signals as well as bit error patterns.

The Cost

In order to achieve robustness as well as low and similar side distortions, we use three factors to evaluate each possible IA scheme: the maximum spread s_{max}, the average of the two side distortions, $D_s = \frac{D_1+D_2}{2}$, and the difference of the two side distortions, $|D_1 - D_2|$. By applying different weights α, β and γ on those factors accordingly, we obtain the cost function:

$$cost = \alpha \cdot s_{max} + \beta \cdot D_s + \gamma \cdot |D_1 - D_2|. \tag{4.13}$$

Spreads and s_{max} only depend on IA schemes. Once an IA scheme is determined, they are fixed and not affected by the statistics of actual source data and channel conditions. Therefore, the solution chosen based on s_{max} is insensitive to specific application environments.

The central decoder is independent on the specific IA scheme. The Lloyd-Max algorithm is used to obtain the optimal central partition and codebook of the source. Side distortions D_1 and D_2, then, D_s and $|D_1 - D_2|$ are calculated for each possible IA scheme. Smaller D_s and $|D_1 - D_2|$ are preferred so as to achieve lower and closer side distortions.

Because the central partition and codebook are optimized according to the training data, side distortions are highly related to the statistics of the training data. If the actual signals are not similar to the training data, the performance of the quantizer selected only based on distortions will degrade. Considering the influence of varying channel conditions, the performance degradation will be increased further. Therefore, by using different weights α, β and γ, the importance of three factors is adjusted so as to accommodate to the specific source data and application environment. If the applied source data are similar to the training set, and the transmission channels are almost error-free, lower and more balanced side distortions are preferred, i.e., β, $\gamma >> \alpha$ in (4.13). In the other hand, if the applied source data vary much or are transmitted over error-prone channels, s_{max} is used as the primary factor in the cost function, i.e., $\alpha >> \beta$, γ.

Table 4.2 The IA algorithm by using the GA.

1. At the first generation, each chromosome is initialized by a random permutation of integers 0 to $L-1$.
2. Evaluating and sorting chromosomes according to the increasing *cost*. The chromosome with the lower *cost* obtained by (4.13) is better.
3. The best 50% chromosomes are kept as parents. The other 50% are dropped.
4. Exchange the k-th and $(L-k-1)$-th integers in each parent to generate an offspring, where $0 \leq k \leq \lfloor \frac{L}{2} \rfloor$.
5. Except the best one, randomly exchange two integers within each chromosome at a certain mutation rate.
6. Go to Step 2 unless it converges.

4.4.2 Index Assignment Algorithm

The GA is an optimization and search technique based on the principles of genetics and natural evolution [89]. The GA provides attractive results when traditional optimization approaches fail. It uses a population with many chromosomes (individual realizations) to evolve according to certain selection rules so as to minimize the cost. There are L_{pop} chromosomes in each generation (iteration). Each chromosome is a permutation of all L SDSQ indices l over the selected $\mathcal{W}$.

The IA algorithm is described in Table 4.2. In each generation, all chromosomes are evaluated by using (4.13). Chromosomes with lower cost are better. In order to enlarge the search area, new chromosomes, called offsprings, are generated by mating parents that are the best 50% chromosomes. In order to avoid duplicated elements in each offspring, at Step 4, elements in each parent are switched in pairs. For example, if a parent is [1, 3, 2, 4, 0, 5], the offspring may be [5, 0, 4, 2, 3, 1]. Thereafter, at Step 5, a small portion of elements are switched randomly. This step is called mutation. For example, if in [5, 0, 4, 2, 3, 1], 0 is chosen to mute, it swaps with 3 to obtain a chromosome [5, 3, 4, 2, 0, 1]. The portion is determined by the mutation rate, which is the number of switched elements divided by the number of all elements. The best chromosome is kept

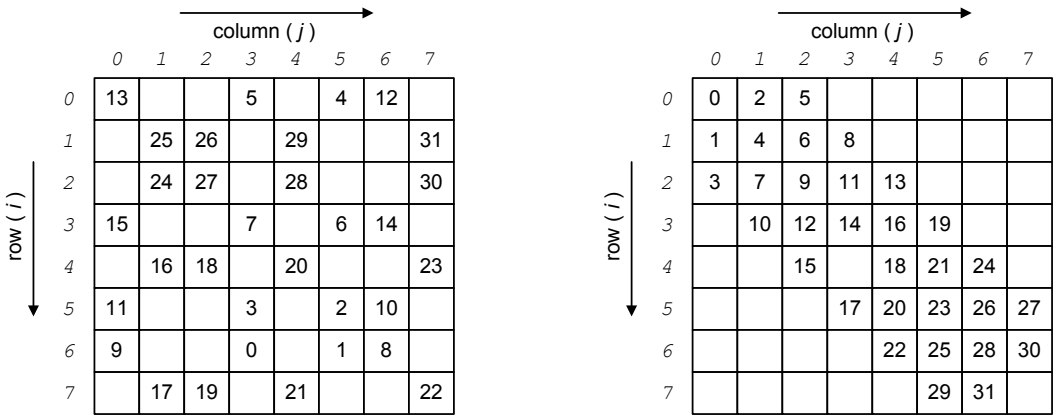


Fig. 4.2 (a) A robust IA with $R_s = 3$ and $N = 32$; (b) $MLIA(3, 2, 32)$.

intact during mutations. At Step 6, the algorithm terminates if it converges, that is to say, if no better chromosome is found in a pre-determined number of generations. After the algorithm terminates, the best chromosome is chosen as the solution.

4.5 Experimental Results

The ERMDC encoder combines with the ERMDC decoder to form the ERMDC codec. Analytical and experimental results are presented for i.i.d. Gaussian source samples with zero mean, unit variance and at the transmission rate of $R_s = 3$ bpss/channel. In Section 4.5.1, the ERMDC codec is tested under two circumstances where separate and burst bit errors are respectively dominant. In Section 4.5.2, the performance of the GA is compared in terms of different cost weights and design steps.

4.5.1 Performance of ERMDC Codecs

An ERMDC encoder with 3 bpss/channel is developed in terms of the procedure described in Section 4.2. The solution with $R_s = 3$, $h_{min} = 2$, as illustrated in Fig. 3.3(a), is used as the template for the IA. In the GA algorithm, $\alpha = \beta = \gamma = 1$, $L_{pop} = 1024$, and 2242 iterations are used. The mutation rate is 5%. A robust IA scheme as shown in Fig. 4.2(a) is found as the result. Then, the ERMDC encoder associated with the robust IA scheme is refined to adapt to the statistics of the source based on the algorithm provided in [18].

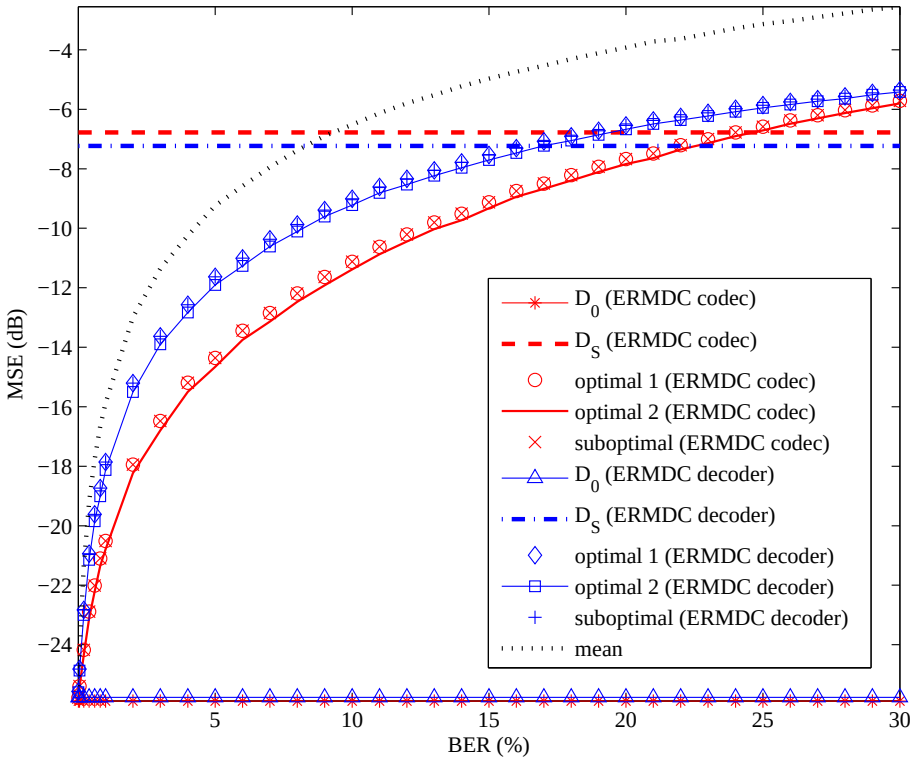


Fig. 4.3 Experimental results of the traditional method, ERMDC decoders and codecs associated with separate bit errors.

Separate Bit Errors

Experimental results associated with i.i.d. and uniformly distributed bit errors are plotted in Fig. 4.3. For the purpose of comparison, the performance of the *ERMDC decoder* and *mean* based on $MLIA(3, 2, 32)$, shown in Fig. 4.2(b), is also drawn. By exploiting the MDSQ encoder and the ERMDC decoders, reconstruction distortions are decreased up to 2.45 dB from *mean*. Reconstruction distortions of ERMDC codecs are further reduced up to 2.9 dB in the range of BERs $< 25\%$. Meanwhile, the BET increases from 8.39% of *mean*, to about 16.3% of ERMDC decoders, and then 24.1% of ERMDC codecs, accordingly. This gain is accomplished by detecting all one-bit errors and estimating the associated reconstruction levels. On the contrary, in conventional IA schemes, some one-bit errors cannot be detected. As the BER increases further, two- and three-bit errors appear more frequently, so advantages of the ERMDC encoder decrease.

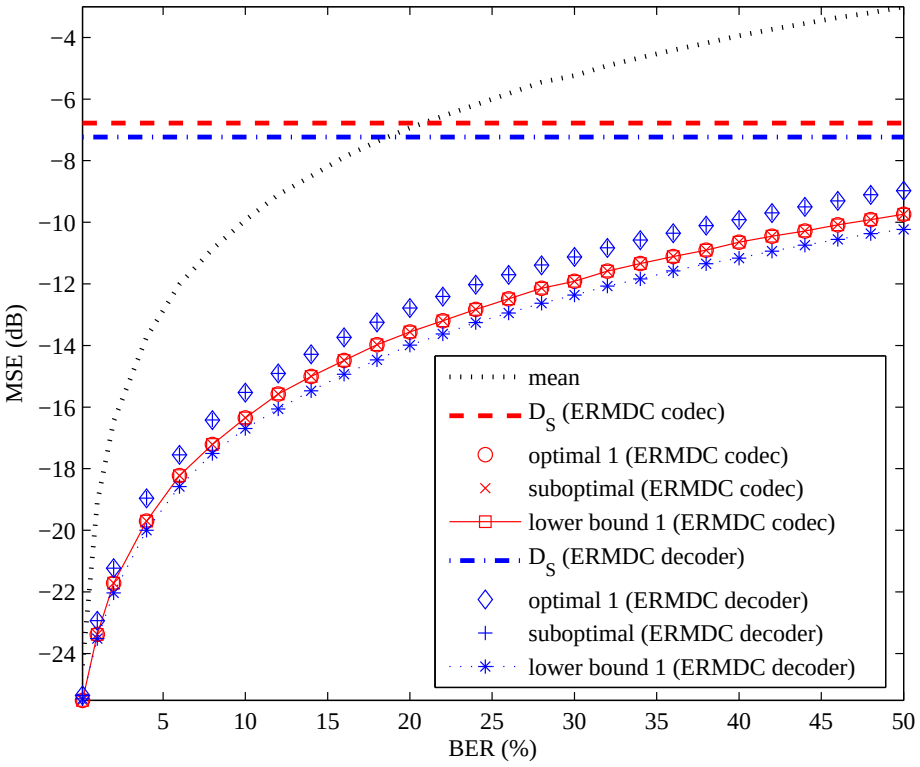


Fig. 4.4 Experimental results of the traditional method, ERMDC decoders and codecs associated with burst bit errors.

Burst Bit Errors

In Section 3.4.2, it indicates that *Optimal method 2* degrades significantly when parameters are mismatching. Here, we only test *Optimal method 1* and *suboptimal* that are more robust against parameter mismatch. Performance of the *ERMDC codec* associated with burst bit errors is shown in Fig. 4.4. For comparison, experimental results of *Optimal method 1* and *suboptimal* ERMDC decoders, and *mean* based on MDSQ IA$(3, 2, 32)$ are plotted. The performance of *Optimal method 1* and *suboptimal* ERMDC codecs is similar, and it is about 3 dB lower than their side distortions D_s at BER 50%. Because ERMDC codecs find all three-bit errors, their performance reaches the lower bound of the ERMDC codec. Since the side distortion D_s of the *ERMDC codec* is a little higher, the lower bound of the ERMDC codec is slightly higher than that of the ERMDC decoder.

Table 4.3 GA performance associated with various weights of the cost function before tuning. The performance is evaluated by using spreads, side distortions and generations. The first row shows the results obtained by using the algorithm provided in Section 2.3. $D_0 = 0.002525$.

No.	α	β (γ)	s_{max}	s_{min}	s_{dif}	D_1	D_2	D_s	σ (%)	generations
1	–	–	11	3	8	0.226995	0.221651	0.224323	1.19	–
2	1	0	10	7	3	0.227438	0.259546	0.243492	6.59	2168
3			11	7	4	0.222710	0.299554	0.261132	14.71	223
4			10	6	4	0.220707	0.240115	0.230411	4.21	1828
5			11	7	4	0.238648	0.305612	0.272130	12.30	162
6			11	7	4	0.247601	0.238988	0.243295	1.77	268
7	1	1	10	7	3	0.239491	0.231814	0.235653	1.63	3208
8			10	7	3	0.231615	0.230856	0.231236	0.16	1677
9			10	7	3	0.232924	0.240670	0.236797	1.64	1249
10			10	6	4	0.234178	0.232148	0.233163	0.44	3716
11			10	7	3	0.229554	0.236221	0.232888	1.43	1449
12	0	1	16	5	11	0.166404	0.170360	0.168382	1.17	1994
13			16	4	12	0.162541	0.165867	0.164204	1.01	1393
14			16	4	12	0.170261	0.166736	0.168498	1.05	1316
15			16	4	12	0.164386	0.162549	0.163468	0.56	1009
16			16	4	12	0.160618	0.168664	0.164641	2.44	1975

4.5.2 Performance of Genetic Algorithm

In order to demonstrate the performance of the proposed GA as well as the influence of three factors s_{max}, D_s and $|D_1 - D_2|$ in (4.13) on choosing IA scheme, we test the GA with various weights α, β, γ on i.i.d. Gaussian source samples with zero-mean and unit variance. Let $\beta = \gamma$ so as to simplify the discussion. In experiments, we investigate the influence of various (α, β) combinations. The GA is applied on the candidate index pairs shown in Fig. 3.3(a).

Experimental results before using the tuning algorithm, which is the third step of designing an ERMDC encoder, are illustrated in Table 4.3. For each (α, β), the GA is run

Table 4.4 Distortions after tuning partitions and codebooks by using the algorithm described in Table 2.2. $\lambda_1 = \lambda_2 = 0.005$. The stopping threshold is 5×10^{-5}.

No.	No. in Table 4.3	D_0	D_1	D_2	D_s	σ (%)
1	1	0.002626	0.192798	0.191860	0.192329	0.24
2	4	0.002610	0.205491	0.223683	0.214587	4.24
3	8	0.002644	0.209272	0.213795	0.211534	1.07
4	15	0.002650	0.149784	0.152058	0.150921	0.75

five times with 1024 chromosomes and the mutation rate 5%. In Table 4.3, the maximum spread s_{max}, the minimum spread s_{min}, two side distortions D_1, D_2, and the average side distortion D_s are provided. For evaluating IA schemes according to the spread, a smaller s_{max} and an s_{min} closer to s_{max} are preferred.

$$\sigma = \frac{|D_1 - D_s| + |D_2 - D_s|}{2D_s} \tag{4.14}$$

is used to evaluate the balance of two descriptions. A smaller σ means that the balance is better. The number of generations to converge is also provided. For the purpose of comparison, the performance achieved by the traditional IA scheme illustrated in Fig. 4.2(b) without using the tuning algorithm is shown in the first row.

When $\beta = 0$, the GA searches for a best solution only taking into account s_{max}. $\alpha = 0$ means that the GA searches for minimum side distortions without considering spreads. As the result, the achieved side distortions D_s at $\alpha = 0$ are significantly lower than those at $\beta = 0$ and the traditional IA scheme. At $\alpha = \beta = 1$, the GA achieves good and consistent s_{max}, s_{min}, D_s and σ. A good trade-off among the spread, the side distortions and the balance is achieved at the cost of more generations.

According to the design procedure of the ERMDC encoder described in Section 4.2, the algorithm described in Table 2.2 is used at the third step to tune the partition and codebook of the selected IA scheme. In Table 4.4, the performance of ERMDC encoders after tuning is compared. For the purpose of comparison, distortions achieved by the traditional IA scheme illustrated in Fig. 4.2(b) are shown in the first row. The other three IA schemes are No. 4, 8, and 15 in Table 4.3. They are the IA schemes with the lowest D_s with respect

to ($\alpha = 1$, $\beta = 0$), ($\alpha = 1$, $\beta = 1$), and ($\alpha = 0$, $\beta = 1$) accordingly. It shows that after tuning, the side distortions are further reduced at the cost of slightly higher central distortions. Similar to the results before tuning, at ($\alpha = 0$, $\beta = 1$), the IA scheme selected by the GA achieves much lower side distortions than the traditional IA scheme described in Section 2.3.4.

It indicates that the spread and the side distortions are, to some extend, conflicting. When the test signal is similar to the training signal, the side distortions are preferable, that is to say, ($\alpha = 0$, $\beta = 1$); otherwise, the spread is preferable, specifically, $\alpha = 1$. Furthermore, it suggests that the GA provides excellent flexibility for the encoder design.

4.6 Discussion and Conclusion

In this chapter, we propose an ERMDC encoder so as to enhance the error detection capability of the ERMDC decoder. Instead of natural binary codes, we enlarge the minimum Hamming distance among codewords so that the ERMDC decoder can detect more errors. Once the ERMDC decoder finds an error, it applies the proposed optimal or suboptimal methods to estimate the reconstruction value. Consequently, the reconstruction distortion is reduced. In addition, the GA is utilized to search for a good IA scheme. The GA is generalized to find a good trade-off by taking into account both the spread and side distortions. Experimental results showed that, compared with the conventional MDC system, the ERMDC codecs achieve significant improvements against both separate and burst bit errors.

However, since exhaustive search and the GA are used in finding a good IA scheme, the computational complexity is prohibitively high, especially, at large bit rates. Therefore, in the next chapter, we are going to simplify this procedure.

The ERMDC System

When applying the ERMDC in the MDC system to generate an ERMDC system, the ERMDC encoder and decoder substitute the classical MDC encoder and central decoder, respectively. If we assume that the BET and BER are known at the receiver end, the ERMDC decoder works when the BER is lower than the BET; once the BER becomes higher than the BET, the side decoder takes the place. The ERMDC system can achieve the best performance at the receiver end in terms of current channel conditions.

Compared with the conventional MD system, such as the MDSQ, the ERMDC system achieves almost the same central distortion in the error-free situation, a slight degradation in the packet-loss environment, and much improvement in the presence of noisy channels. Also, since the designs of ERMDC encoder and suboptimal decoder have nothing to do with channel conditions, the achieved performance is not affected by parameter mismatching between the design and application stages. Therefore, the ERMDC system can be applied in a wide range of communication environments.

Performance of the ERMDC system will be also tested in various environments in Chapter 6, such as slow Rayleigh fading channels in Section 6.1 and hybird wireline-wireless networks in Section 6.2.

Chapter 5

Generalized Fast Index Assignment for ERMDC

5.1 Introduction

In Chapter 4, a robust IA algorithm is developed for the ERMDC encoder to increase the minimum Hamming distance among MD index pairs, so that the ERMDC decoder can detect more transmission errors, and, furthermore, reduce the reconstruction distortion. However, because this IA algorithm is realized by exhaustive search and the GA, the training time of the ERMDC encoder becomes significantly longer, as the number of quantization levels increased. This time-consuming process is improper in applications where short training time is preferred. Moreover, the solution obtained by the GA changes with different initial conditions and mutations. Hence, it is necessary for the encoder to transmit the obtained IA scheme to the decoder. This transmission increases the overhead and the complexity of the ERMDC system. Therefore, an IA technique with low computational complexity and less transmission overhead is required so as to make the ERMDC applicable.

In this chapter, we propose a novel generalized robust IA algorithm with low computational complexity for the ERMDC encoder[1]. Any arbitrary number of parity bits can be easily added to combat both packet losses and bit errors. This algorithm can be implemented "on-the-fly" so as to produce two descriptions as soon as possible. Experimental

[1]This work has been partially presented at the IEEE International Conference on Image Processing, Oct. 2008 [26], and the Asilomar Conference on Signals, Systems and Computers [27].

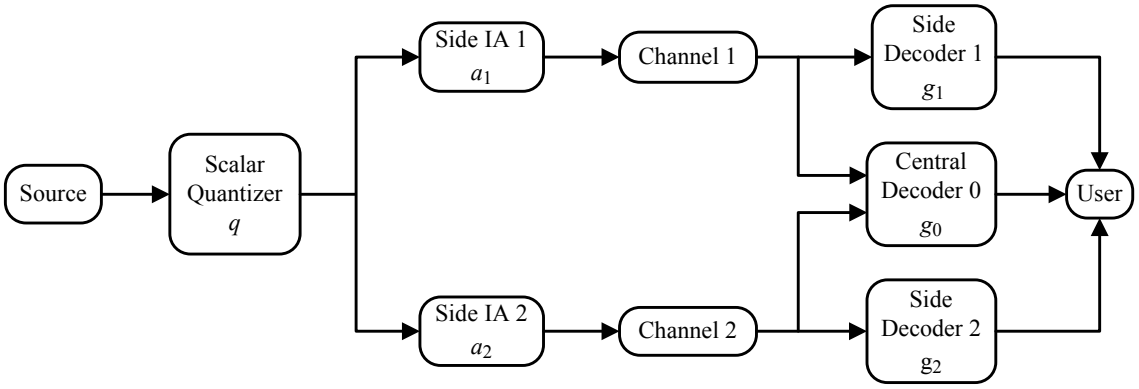


Fig. 5.1 The ERMDC system with two channels and three receivers.

results indicate that the proposed algorithm outperforms existing algorithms, such as IA algorithms applied in the MDSQ [18] and the polyphase transform selective quantization (PTSQ) [47], in the sense of computational complexity or rate distortion. Moreover, by using this algorithm, the IA scheme used at the encoder does not need to be transmitted.

The proposed IA algorithm is described in Section 5.3, followed by the optimal bit allocation scheme derived in Section 5.4. In Section 5.5, experimental results show that the proposed algorithm outperforms existing algorithms. In Section 5.6, a numerical method for arbitrary source and IA scheme is provided. Moreover, closed form formulas are derived for uniformly distributed signals and the proposed IA scheme. In order to accommodate progressive transmissions, two embedded IA schemes are provided and compared in Section 5.7. Further discussion and conclusion is given in Section 5.8.

5.2 Problem Formulation and Notations

As discussed in the previous chapter, searching for the optimum IA scheme of the ERMDC encoder-decoder pair is an NP-complete problem. Even though we provide a feasible method to obtain a "close-to-optimum" solution, the computational complexity of the proposed method grows significantly, as the bit rate increases. In order to simplify this optimization, in this chapter, the ERMDC is designed for a high rate system without taking into account the source properties. Thus, instead of the average side distortion D_s, the average spread $\bar{s}$ is minimized as a measurement of D_s subject to the constraint of having approximately balanced descriptions, i.e., $\bar{s}^{(1)} \approx \bar{s}^{(2)}$. As a result, in addition to

lower computational complexity, the obtained ERMDC system can be applicable to a wide range of source signals.

We define $\mathbb{F}_2 = \{0, 1\}$ as a set, on which modulo-2 addition $\oplus$ is defined. Let $\mathbb{F}_2^n$ consist of binary vectors $\mathbf{w} = [w_{n-1} \cdots w_1 w_0]$ of length n, where $w_k \in \mathbb{F}_2$, $k = 0, 1, \cdots, n-1$. In addition, each element of the binary vector $\mathbf{w}$ corresponds to a bit in the binary representation of the integer w with n bits. w_{n-1} corresponds to the most significant bit (MSB), w_0 corresponds to the least significant bit (LSB).

Let $\mathbf{l} \in \mathbb{F}_2^{R_0}$ be binary representations of l:

$$\mathbf{l} = \left[l_{(R_0-1)} \cdots l_1 \, l_0 \right]. \tag{5.1}$$

Bits of $\mathbf{l}$ are information bits. As defined in (3.4) and (3.5), $\mathbf{i} \in \mathbb{F}_2^{R_s}$ and $\mathbf{j} \in \mathbb{F}_2^{R_s}$ are binary representations of i and j respectively. $[\mathbf{i}, \mathbf{j}]$ denotes the binary representation of the index pair (i, j). The total code rate R of each source sample is $R = 2R_s$ bpss.

We also call the binary vectors $\mathbf{w} = [\mathbf{i}, \mathbf{j}]$ *codewords*, i.e.,

$$\begin{aligned} \mathbf{w} &= \left[w_{(R-1)} \cdots w_1 \, w_0 \right] \\ &= [i_{(R_s-1)} \cdots i_0 \, j_{(R_s-1)} \cdots j_0]. \end{aligned}$$

Let $\mathcal{W} \subseteq \mathbb{F}_2^R$ be the set of codewords $\mathbf{w}$ with minimum Hamming distance h_{min}. $\|\mathbf{w} - \mathbf{v}\|$ denotes the Hamming distance between $\mathbf{w}$ and $\mathbf{v}$, where $\mathbf{w}, \mathbf{v} \in \mathcal{W}$ and $\mathbf{w} \neq \mathbf{v}$. So, $\|\mathbf{w} - \mathbf{v}\| \geq h_{min}$.

$\rho = R - R_0$ denotes the number of bits of redundancy. $\mathcal{B}(R_0, \rho)$ represents an ERMDC encoder that adds ρ redundancy bits to each R_s information bits. If $R_0 = \rho$, the redundancy is generated by duplicating information bits. In the following, we only consider the situation where $R_0 > \rho$.

5.3 Generalized Fast Index Assignment Algorithm

In this section, the proposed generalized fast IA algorithm consists of three steps according to the number of redundancy bits ρ:

1. $\mathcal{B}(R_0, 0)$: No redundancy is added. Information bits are split into two descriptions with similar side spreads.

2. $\mathcal{B}(R_0, 1)$: One-bit redundancy is added. Two descriptions are generated with $R = R_0 + 1$ bpss and the same side spreads.

3. $\mathcal{B}(R_0, \rho)$, $\rho \geq 2$: ρ-bit redundancy is added. Two descriptions are generated with $R = R_0 + \rho$ bpss and the same side spreads.

In addition, the optimal bit allocation algorithm is provided.

5.3.1 Index Assignment without Redundancy

When $\rho = 0$, since no redundancy is added in converting $\mathbf{l}$ to $[\mathbf{i}, \mathbf{j}]$, information bits are simply split into $\mathbf{i}$ and $\mathbf{j}$. In order to minimize $\bar{s}$ subject to similar $\bar{s}^{(1)}$ and $\bar{s}^{(2)}$, the MSB $l_{(R_0-1)}$ should be arranged in a different description than the next $\frac{R_0}{2}$ significant bits, i.e., $l_{(R_0-2)}\ l_{(R_0-3)}\ \cdots\ l_{\left(\frac{R_0}{2}-1\right)}$. In order to keep the same numbers of bits in both descriptions, the remaining bits in the description where $l_{(R_0-1)}$ is located have to be filled with $\frac{R_0}{2} - 1$ LSBs, i.e., $l_{\left(\frac{R_0}{2}-2\right)}\ \cdots\ l_1\ l_0$. Thus, $\mathcal{B}(R_0, 0)$ is produced by

$$\mathbf{i} = \left[\ l_{(R_0-2)}\ l_{(R_0-3)}\ \cdots\ l_{\left(\frac{R_0}{2}-1\right)}\ \right], \qquad \text{and} \tag{5.2}$$

$$\mathbf{j} = \left[\ l_{(R_0-1)}\ l_{\left(\frac{R_0}{2}-2\right)}\ \cdots\ l_1\ l_0\ \right]. \tag{5.3}$$

Side spreads are given by

$$\bar{s}^{(1)} = 2^{(R_0-1)} + 2^{\left(\frac{R_0}{2}-1\right)}, \tag{5.4}$$

$$\bar{s}^{(2)} = 2^{(R_0-1)} - 2^{\left(\frac{R_0}{2}-1\right)} + 1. \tag{5.5}$$

Thus,

$$\bar{s} = 2^{(R_0-1)} + 0.5. \tag{5.6}$$

(5.2) and (5.3) indicate that generating two descriptions is actually to split information bits into two descriptions. Therefore, the computational complexity of encoding each R_0 information bits, which correspond to a source sample, is $O(R_0)$.

As an example, generation of $\mathcal{B}(4, 0)$ is illustrated in Fig. 5.2. Four information bits are split into $\mathbf{i}$ and $\mathbf{j}$ in Fig. 5.2(a), specifically, $i_1 \leftarrow l_2$, $i_0 \leftarrow l_1$, $j_1 \leftarrow l_3$, and $j_0 \leftarrow l_0$. The resulting map between indices l and index pairs (i, j) is shown in Fig. 5.2(b). As a result, $\bar{s}_1 = 10$, $\bar{s}_2 = 7$ and $\bar{s} = 8.5$.

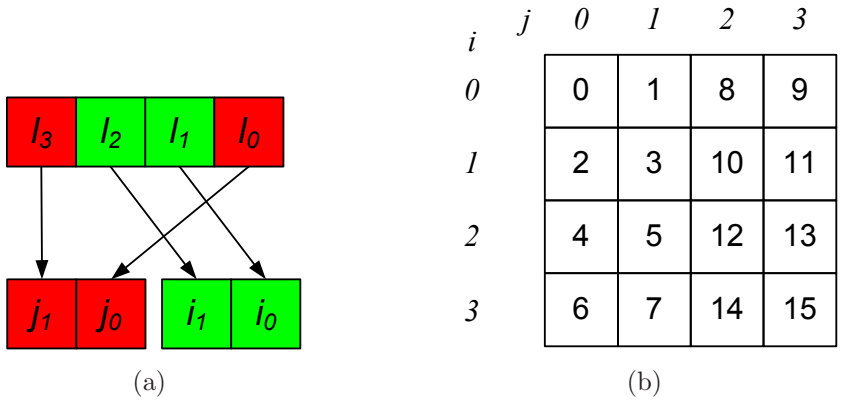


Fig. 5.2 Generation of $\mathcal{B}(4,0)$: (a) four bits of $\mathbf{l}$ are split to $\mathbf{i}$ and $\mathbf{j}$. (b) the resulting map between SDSQ indices l and ERMDC index pairs (i,j).

5.3.2 Index Assignment with One-Bit Redundancy

When $\rho = 1$, one-bit redundancy is added. First, we prove Theorem 5.3.1, based on which the bit allocation algorithm is provided.

Theorem 5.3.1. *Let a codeword $\boldsymbol{w}$ consist of $(R-1)$-bit binary vector $\boldsymbol{l} = [l_{R-2} \ \cdots \ l_0]$ without redundancy and a parity bit e. For a binary constant n, $n \in \{0,1\}$, let a set $\mathcal{W}_n$ be defined by*

$$\mathcal{W}_n = \Big\{ \boldsymbol{w} : \bigoplus_{k=0}^{R-1} \boldsymbol{w}_k = n \Big\}. \tag{5.7}$$

$\mathcal{W}_n$ has minimum Hamming distance $h_{min} = 2$.

Proof. Let $\mathbf{l}$ and $\mathbf{l}'$ be two different $(R-1)$-bit binary vectors without redundancy. Thus,

$$||\mathbf{l} - \mathbf{l}'|| \geq 1, \tag{5.8}$$

that is to say,

$$\sum_{k=0}^{R-2} (l_k \oplus l'_k) \geq 1. \tag{5.9}$$

Let $\mathbf{w} = [\mathbf{l}\ e]$, $\mathbf{v} = [\mathbf{l}'\ e']$, and $\bigoplus_{k=0}^{R-1} w_k = \bigoplus_{k=0}^{R-1} v_k$. Therefore,

$$\bigoplus_{k=0}^{R-2} l_k \oplus e = \bigoplus_{k=0}^{R-2} l'_k \oplus e', \tag{5.10}$$

that is to say,

$$e' \oplus e = \bigoplus_{k=0}^{R-2} (l_k \oplus l'_k). \tag{5.11}$$

Then,

$$\begin{aligned} ||\mathbf{w} - \mathbf{v}|| &= ||\mathbf{l} - \mathbf{l}'|| + (e \oplus e') \\ &= \sum_{k=0}^{R-2} (l_k \oplus l'_k) + \bigoplus_{k=0}^{R-2} (l_k \oplus l'_k). \end{aligned} \tag{5.12}$$

If $||\mathbf{l} - \mathbf{l}'||$ is a positive even number,

$$\bigoplus_{k=0}^{R-2} (l_k \oplus l'_k) = 0. \tag{5.13}$$

Thus, $||\mathbf{w} - \mathbf{v}|| \geq 2$. Otherwise, $||\mathbf{l} - \mathbf{l}'||$ is a positive odd number,

$$\bigoplus_{k=0}^{R-2} (l_k \oplus l'_k) = 1. \tag{5.14}$$

Thus, $||\mathbf{w} - \mathbf{v}|| \geq 2$, too.

Consequently, $||\mathbf{w} - \mathbf{v}|| \geq 2$. Therefore, $\mathcal{W}_n$ has minimum Hamming distance $h_{min} = 2$. □

There are two sets satisfying (5.7):

$$\mathcal{W}_0 = \left\{ \mathbf{w} : \bigoplus_{k=0}^{R-1} w_k = 0 \right\}, \qquad \text{and} \tag{5.15}$$

$$\mathcal{W}_1 = \left\{ \mathbf{w} : \bigoplus_{k=0}^{R-1} w_k = 1 \right\}. \tag{5.16}$$

In the following, without loss of generality, we only consider $\mathcal{W} = \mathcal{W}_0$.

In terms of Theorem 5.3.1, $\mathcal{B}(R_0, 1)$ is generated by one-to-one mapping indices $\mathbf{l} \in \mathcal{L}$ to codewords $[\mathbf{i}, \mathbf{j}] \in \mathcal{W}$ so as to minimize $\bar{s}$, subject to

$$\bigoplus_{k=0}^{R_s-1} (i_k \oplus j_k) = 0, \tag{5.17}$$

$$\bar{s}^{(1)} = \bar{s}^{(2)}. \tag{5.18}$$

The minimum Hamming distance between any two $[\mathbf{i}, \mathbf{j}]$ is 2. Therefore, one-bit errors are detectable.

For the purpose of minimizing $\bar{s}$, except $l_{(R_0-1)}$, the MSB of l, all other information bits $l_{(R_0-2)} \cdots l_1 l_0$ are split to $i_{(R_s-1)} \cdots i_2 i_1$ and $j_{(R_s-1)} \cdots j_2 j_1$, respectively. Due to satisfying (5.17), the LSBs of $\mathbf{i}$ and $\mathbf{j}$, i.e., i_0 and j_0, are used as parity bits and generated by using modulo-2 addition of corresponding information bits. Specifically, $\mathcal{B}(R_0, 1)$ is produced by

$$\begin{aligned} \mathbf{i} &= \left[l_{(R_0-3)} \cdots l_{\left(\frac{R_0-3}{2}\right)} \, i_0 \right], \qquad \text{and} \\ \mathbf{j} &= \left[l_{(R_0-2)} \, l_{\left(\frac{R_0-5}{2}\right)} \cdots l_0 \, j_0 \right], \end{aligned} \tag{5.19}$$

where

$$\begin{aligned} i_0 &= l_{(R_0-1)} \oplus \bigoplus_{k=\frac{R_0-3}{2}}^{R_0-3} l_k, \qquad \text{and} \\ j_0 &= l_{(R_0-1)} \oplus l_{(R_0-2)} \oplus \bigoplus_{k=0}^{\frac{R_0-5}{2}} l_k. \end{aligned} \tag{5.20}$$

Hence, side spreads are given by

$$\bar{s}^{(1)} = 2^{(R_0-2)} - 2^{(\frac{R_0-3}{2})} + 1 \tag{5.21}$$

$$\bar{s}^{(2)} = 2^{(R_0-2)} + 2^{(\frac{R_0-3}{2})}. \tag{5.22}$$

In order to achieve (5.18), when $l_{(R_0-1)} = 1$, $\mathbf{i}$ and $\mathbf{j}$ are exchanged. Therefore, when $l_{(R_0-1)} = 0$, the output codewords are $\mathbf{w} = [\mathbf{i}, \mathbf{j}]$; when $l_{(R_0-1)} = 1$, the output codewords

Table 5.1 Fast index assignment algorithm for the ERMDC.

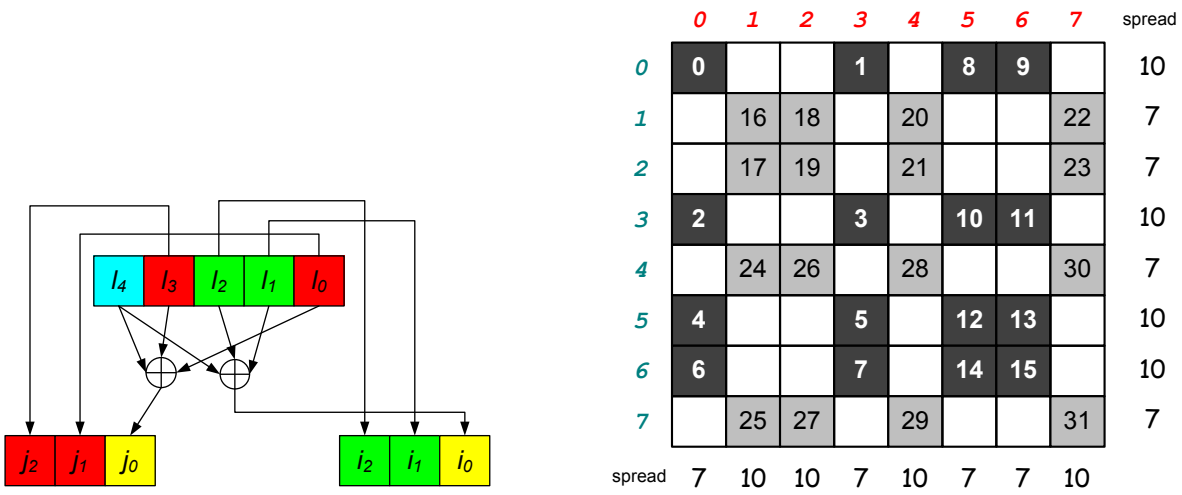
SET R_0;
$\mathbf{l} \leftarrow \left[\ l_{(R_0-1)}\ l_{(R_0-2)}\ \cdots\ l_0\ \right]$;
$n \leftarrow R_0 - 2$;
$k \leftarrow \frac{R_0-1}{2}$; $\quad h \leftarrow \frac{R_0-1}{2}$;
$i_0 \leftarrow l_{(R_0-1)}$; $\quad j_0 \leftarrow l_{(R_0-1)}$;
WHILE $n \geq 0$
  IF $n < (R_0 - 2)$ AND $n > \frac{R_0-5}{2}$
    $i_k \leftarrow l_n$;
    $i_0 \leftarrow i_0 \oplus l_n$;
    $k \leftarrow k - 1$;
  ELSE
    $j_h \leftarrow l_n$;
    $j_0 \leftarrow j_0 \oplus l_n$;
    $h \leftarrow h - 1$;
  END
  $n \leftarrow n - 1$;
END
IF $l_{(R_0-1)} = 1$
  swap i and j;
END
output i and j.

are $\mathbf{w} = [\mathbf{j}, \mathbf{i}]$.

Consequently, $\bar{s}$, $\bar{s}^{(1)}$ and $\bar{s}^{(2)}$ are given by

$$\bar{s} = \bar{s}^{(1)} = \bar{s}^{(2)} = 2^{(R_0-2)} + 0.5. \tag{5.23}$$

The bit allocation algorithm is elaborated in Table 5.1. Except the MSB $l_{(R_0-1)}$, the other (R_0-1) information bits are directly sent to two descriptions. Two parity bits i_0 and j_0 are generated by modulo-2 summation of corresponding information bits. Hence, the computational complexity of dealing with each R_0 information bits is $O(2R_0)$. Generation of $\mathcal{B}(5,1)$ is shown in Fig. 5.3 as an example, where $\bar{s} = \bar{s}^{(1)} = \bar{s}^{(2)} = 8.5$.



(a) (b)

Fig. 5.3 Generation of $\mathcal{B}(5,1)$: (a) generation of **i** and **j** when $l_4 = 0$; (b) the resulting map between indices l and index pairs (i, j).

5.3.3 Geometric IA Algorithm for One-Bit Redundancy

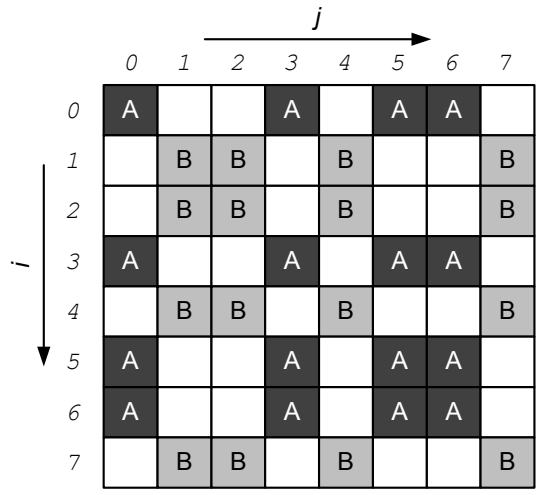


Fig. 5.4 The ERMDC index pairs used in the IA scheme illustrated in Fig. 5.3(b) can be divided into two non-overlapping groups A and B.

In this subsection, a geometric IA algorithm for one-bit redundancy is proposed. One-bit redundancy is added by allocating specific slots, instead of modulo-2 operations used in the previous subsection.

Figure 5.3(b) suggests that the ERMDC index pairs (i, j) can be divided into two groups

A and B, as illustrated in Fig. 5.4. Since the rows and columns that are occupied by the slots located in Group A are not occupied by the slots of Group B, these two groups are non-overlapping.

We use Theorem 5.3.2 to explicitly explain this observation. Theorem 5.3.2 claims that these two groups are determined by the bit l_{R_0-1}. That is to say, Group A is associated with $l_{R_0-1}=0$; Group B is associated with $l_{R_0-1}=1$.

Theorem 5.3.2. *For $\mathcal{B}(R_0,1)$, SDSQ indices $\boldsymbol{l}$ and $\boldsymbol{l}'$ are respectively decomposed into ERMDC codewords $[\boldsymbol{i},\boldsymbol{j}]$ and $[\boldsymbol{i}',\boldsymbol{j}']$ by using* (5.19) *and* (5.20)*. Provided that $l_{R_0-1}\neq l'_{R_0-1}$, where l_{R_0-1} and l'_{R_0-1} are MSBs of $\boldsymbol{l}$ and $\boldsymbol{l}'$ respectively, $\boldsymbol{i}\neq\boldsymbol{i}'$ and $\boldsymbol{j}\neq\boldsymbol{j}'$.*

Proof. Let $\mathbf{l}=\left[l_{R_0-1}\,l_{R_0-2}\cdots l_1\,l_0\right]$, $\mathbf{i}=[i_{R_s-1}\cdots i_1\,i_0]$, $\mathbf{j}=[j_{R_s-1}\cdots j_1\,j_0]$; and $\mathbf{l}'=\left[l'_{R_0-1}\,l'_{R_0-2}\cdots l'_1\,l'_0\right]$, $\mathbf{i}'=[i'_{R_s-1}\cdots i'_1\,i'_0]$, $\mathbf{j}'=[j'_{R_s-1}\cdots j'_1\,j'_0]$, where l_{R_0-1} and l'_{R_0-1} are MSBs of l and l' respectively, $R_0=2R_s-1$.

In terms of (5.17), (5.19) and (5.20),

$$\begin{aligned}\bigoplus_{k=0}^{R_s-1}(i_k\oplus j_k)=0\\ \bigoplus_{k=0}^{R_s-1}(i'_k\oplus j'_k)=0\end{aligned}\tag{5.24}$$

where

$$\begin{aligned}i_0&=l_{R_0-1}\oplus\bigoplus_{k=1}^{R_s-1}i_k,\\ j_0&=l_{R_0-1}\oplus\bigoplus_{k=1}^{R_s-1}j_k,\\ i'_0&=l'_{R_0-1}\oplus\bigoplus_{k=1}^{R_s-1}i'_k,\\ j'_0&=l'_{R_0-1}\oplus\bigoplus_{k=1}^{R_s-1}j'_k.\end{aligned}\tag{5.25}$$

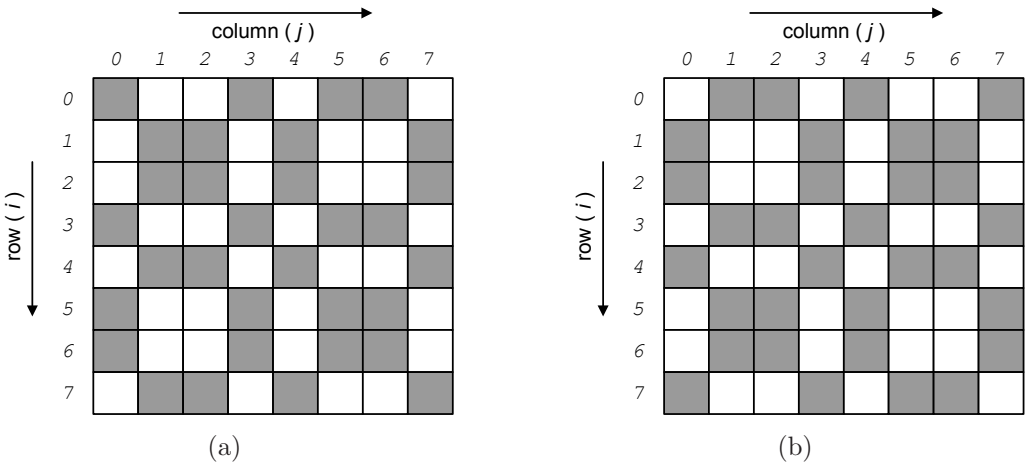


Fig. 5.5 Two IA patterns associated with one-bit redundancy: (a) the IA pattern corresponds to (5.15); (b) the IA pattern corresponds to (5.16). Greyed cells are selected ERMDC index pairs.

According to (5.25),

$$\begin{aligned} l_{R_0-1} &= \bigoplus_{k=0}^{R_s-1} i_k = \bigoplus_{k=0}^{R_s-1} j_k, \\ l'_{R_0-1} &= \bigoplus_{k=0}^{R_s-1} i'_k = \bigoplus_{k=0}^{R_s-1} j'_k. \end{aligned} \tag{5.26}$$

Suppose that $l_{R_0-1} \neq l'_{R_0-1}$,

$$\begin{aligned} \bigoplus_{k=0}^{R_s-1} i_k &\neq \bigoplus_{k=0}^{R_s-1} i'_k, \\ \bigoplus_{k=0}^{R_s-1} j_k &\neq \bigoplus_{k=0}^{R_s-1} j'_k. \end{aligned} \tag{5.27}$$

Therefore, $\mathbf{i} \neq \mathbf{i}'$ and $\mathbf{j} \neq \mathbf{j}'$. □

Because i and j are used as the row and column numbers in the IA matrix, Theorem 5.3.2 indicates that the rows and columns are exclusively occupied by Group A or B.

According to Theorem 5.3.2, if (5.15) is satisfied, we get the IA pattern shown in Fig. 5.5(a); if (5.16) is satisfied, we get the IA pattern shown in Fig. 5.5(b). In Fig. 5.5,

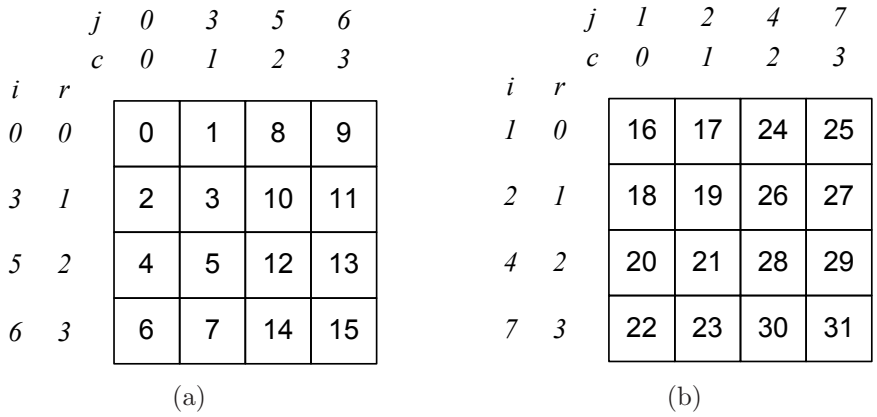


Fig. 5.6 Using the geometric IA algorithm to obtain the IA scheme shown in Fig. 5.3(b): (a) $\mathcal{L}_A$ is assigned to Group A illustrated in Fig. 5.4; (b) $\mathcal{L}_B$ is assigned to Group B illustrated in Fig. 5.4.

greyed cells are selected ERMDC index pairs. Two patterns shown in Fig. 5.5 are the same as those illustrated in Fig. 3.3, which are found by using exhaustive search in Chapter 4. Therefore, we obtain the same IA pattern with much lower computational complexity.

Based on Theorem 5.3.2, we develop a *geometric IA* method to map SDSQ indices to ERMDC index pairs. The resulting IA scheme is equivalent to that obtained by using the method described in (5.19) and (5.20).

In Fig. 5.6, we use an example to illustrate how to use the geometric IA method. In this example, without losing generality, we use the IA pattern given in Fig. 5.5(a). $R_0 = 5$, $R_s = 3$, and $\rho = 1$. The target IA scheme is shown in Fig. 5.3(b).

First, as shown in Fig. 5.4, selected cells are divided into two non-overlapping groups A and B. All cells in Group A are taken out to reshape a $2^{R_0-1} \times 2^{R_0-1}$ matrix, as illustrated in Fig. 5.6(a). i and j are row and column numbers of the original $2^{R_0+1} \times 2^{R_0+1}$ matrix; r and c are row and column numbers of the new $2^{R_0-1} \times 2^{R_0-1}$ matrix. SDSQ indices $l \in \mathcal{L} = \{0, 1, \cdots, 2^{R_0} - 1\}$ are divided into two sets $\mathcal{L}_A$ and $\mathcal{L}_B$ according to $l_{R_0-1} = 0$ and $l_{R_0-1} = 1$ respectively. Hence, $\mathcal{L}_A = \{0, 1, \cdots, 2^{R_0-1} - 1\}$ and $\mathcal{L}_B = \{2^{R_0-1}, 2^{R_0-1} + 1, \cdots, 2^{R_0} - 1\}$. SDSQ indices in $\mathcal{L}_A$ are assigned to Group A by using the IA method provided in Section 5.3.1. The resulting IA scheme is shown in Fig. 5.6(a). Thereafter, referring to i and j, Group A is mapped back to the original matrix. Similarly, SDSQ indices in $\mathcal{L}_B$ are assigned to Group B, as illustrated in Fig. 5.6(b). However, in order to keep balanced spreads, before mapping back to the original matrix, the matrix is transposed.

At last, the IA scheme shown in Fig. 5.3(b) is obtained.

Therefore, according to this algorithm, it can be inferred that the average spread of $\mathcal{B}(R_0, 1)$ equals to that of $\mathcal{B}(R_0 - 1, 0)$.

5.3.4 Index Assignment for More Than One Bit Redundancy

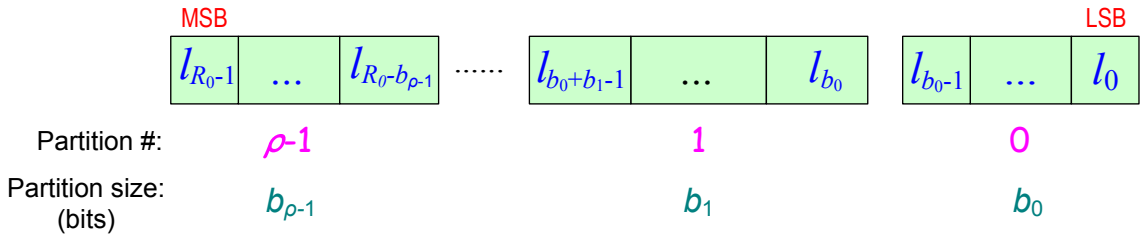


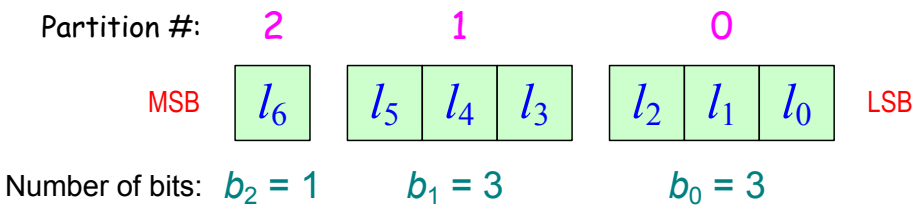
Fig. 5.7 R_0 bits of $\mathbf{l}$ are divided into ρ partitions in sequence in $\mathcal{B}(R_0, \rho)$, $\rho \geq 2$.

Next, we describe the algorithm producing $\mathcal{B}(R_0, \rho)$, $\rho \geq 2$. In order to add ρ-bit redundancy, the methods to add one-bit redundancy are recursively applied. As illustrated in Fig. 5.7, R_0 information bits of each $\mathbf{l}$ are divided into ρ partitions in sequence. Each partition is protected by one-bit redundancy. The granularity of the k-th partition b_k denotes the number of information bits involved in this partition, $k = 0, 1, \cdots, \rho - 1$. Thus,

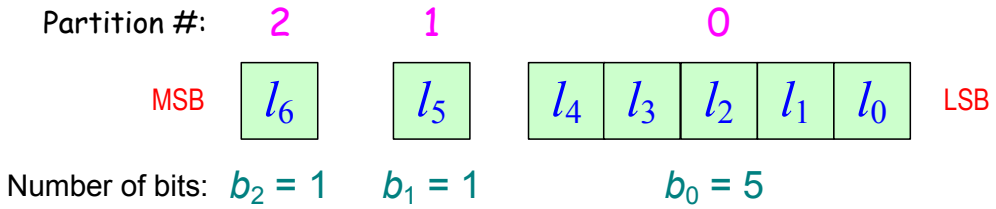
$$R_0 = \sum_{k=0}^{\rho-1} b_k. \tag{5.28}$$

We use a partition sequence $\langle b_{\rho-1}, \cdots, b_0 \rangle$ to represent a partition scheme for $\mathcal{B}(R_0, \rho)$. There may exist more than one partition sequence for a $\mathcal{B}(R_0, \rho)$. For instance, two partition sequences for $\mathcal{B}(7, 3)$ are $\langle 1, 3, 3 \rangle$ and $\langle 1, 1, 5 \rangle$, as illustrated in Fig. 5.8. Specifically, in Fig. 5.8(a), $\langle 1, 3, 3 \rangle$ means to divide $\mathbf{l} = [l_6 l_5 l_4 l_3 l_2 l_1 l_0]$ to three partitions: $[l_6]$, $[l_5 l_4 l_3]$ and $[l_2 l_1 l_0]$, each of which is protected by one-bit redundancy.

For a given partition sequence of $\mathcal{B}(R_0, \rho)$, the IA scheme is accomplished partition by partition. For partitions with $b_k = 1$, two descriptions are generated by duplicating the information bit; then, the modulo-2 summation of these two bits is 0, i.e., (5.17) is satisfied. For partitions with $b_k \geq 3$, two descriptions are produced in the same way as $\mathcal{B}(R_0, 1)$. Therefore, in each partition, two descriptions are generated with the minimized average side spread and balanced side spreads, and (5.17) is satisfied. Consequently, with



(a)



(b)

Fig. 5.8 Two realizations of of $\mathcal{B}(7,3)$: (a) $\langle 1,3,3\rangle$; (b) $\langle 1,1,5\rangle$.

minimized $\bar{s}$ and $\bar{s}_1 = \bar{s}_2$, the IA scheme is uniquely determined by this partition sequence; furthermore, each $[\mathbf{i},\mathbf{j}]$ satisfies (5.17). Similar to the algorithm described in Section 5.3.2, the computational complexity corresponding to each source sample is $O(2R_0)$.

5.3.5 Error Detection, Correction and Estimation

Redundancy added in the ERMDC system provides protection from packet losses and bit errors. At the receiver, the correctness of each packet can be determined by checking applied FEC codes, such as cyclic redundancy check (CRC). If both descriptions are lost or carry bit errors, this transmission fails. When only one description is received and correct, the corresponding side decoder reconstructs signals with the side distortion. When both descriptions are received correctly, the central decoder recovers received signals with the central distortion. In the case that one of two received descriptions carries bit errors, each code word $[\mathbf{i},\mathbf{j}]$ is checked partition by partition in terms of (5.17). If (5.17) is not satisfied, the reconstruction value is estimated by using the algorithms provided in Section 3.3.2. In particular, at partitions with unit granularities, taking the bits from the correct descriptions means not to introduce bit errors, because the duplicated information bits are decomposed into two descriptions.

5.4 Optimal Bit Allocation Schemes

In the previous section, we present how to assign indices with low complexity. In particular, in Section 5.3.4, more than one bit of redundancy is added, and each bit of redundancy is allocated to a partition. As shown in Fig. 5.8, for a given $\mathcal{B}(R_0,\ \rho)$, more than one bit allocation schemes may exist. So, we discuss how to measure the quality of different bit allocation schemes, and the optimal bit allocation scheme based on the measurement.

5.4.1 Performance Measurement of Bit Allocation Schemes

As mentioned in Section 4.4.1, spreads are used in evaluating IA schemes, specifically, S_{max} and S_d. In a high rate system, we derive the overall average spread $\bar{s}$ of the IA method proposed in Section 5.3.4.

For a given $\mathcal{B}(R_0,\ \rho)$, $\rho \geq 2$, we use $\bar{s}$ as the metric of the average side distortion D_s. Provided a partition sequence $\langle b_{\rho-1}, \cdots, b_0 \rangle$, the average side spread of the k-th partition $\bar{s}_{p_k}$ is given by

$$\bar{s}_{p_k} = \begin{cases} 1, & b_k = 1; \\ 2^{b_k-2} + 0.5, & b_k \geq 3. \end{cases} \tag{5.29}$$

b_k is the granularity of the k-th partition defined in Section 5.3.4.

Therefore, the overall average side spread $\bar{s}$ is given by

$$\bar{s} = \bar{s}_{p_0} + \sum_{k=1}^{\rho-1} (\bar{s}_{p_k} - 1) \cdot 2^{\sum_{t=0}^{k-1} b_t}. \tag{5.30}$$

$\bar{s}$ is a good measurement of D_s associated with IA schemes with the same SDSQ levels. However, since the influence of different SDSQ levels on D_s is not considered in calculating $\bar{s}$, it is not enough to evaluate D_s of IA schemes with different SDSQ levels. Thus, taking into account the total number of SDSQ levels $L = 2^{R_0}$, the normalized average spread $\bar{s}_n$ is introduced as

$$\bar{s}_n = \frac{\bar{s}}{2^{R_0}}. \tag{5.31}$$

5.4.2 Optimal Bit Allocation

In terms of (5.29)–(5.30), a principle for designing $\mathcal{B}(R_0, \rho)$ can be inferred: (i) in terms of (5.29), smaller granularity b_k produces lower $\bar{s}_{p_k}$; (ii) in addition to (5.30), it can be in-

ferred that in order to get smaller $\bar{s}$, larger granularities should be located in less significant partitions, namely, $b_k \leq b_{k-1}$, $k = 1, 2, \cdots, \rho - 1$. For example, for $\langle 1,\ 1,\ 5 \rangle$, $\bar{s} = 8.5$; for $\langle 1,\ 3,\ 3 \rangle$, $\bar{s} = 14.5$. Thus, $\langle 1,\ 1,\ 5 \rangle$ is better than $\langle 1,\ 3,\ 3 \rangle$.

We derive the optimal bit allocation scheme for $\mathcal{B}(R_0, \rho)$. In terms of (5.29), for the k-th partition, the minimum average side distortion

$$\min_{b_k} \bar{s}_{p_k} = 1, \tag{5.32}$$

is achieved, when $b_k = 1$.

Let

$$\bar{s}_{p_H} = \sum_{k=1}^{\rho-1} (\bar{s}_{p_k} - 1) \cdot 2^{\sum_{t=0}^{k-1} b_t}. \tag{5.33}$$

When $b_k = 1$, $k = 1, \cdots, \rho - 1$,

$$\min \bar{s}_{p_H} = 0 \tag{5.34}$$

is achieved.

Based on (5.30), $\bar{s} = \bar{s}_{p_0} + \bar{s}_{p_H}$. Therefore, given $\mathcal{B}(R_0,\ \rho)$, when $b_k = 1$, $k = 1, \cdots, \rho - 1$, and $b_0 = R_0 - \rho + 1$,

$$\min \bar{s} = \bar{s}_{p_0} = \begin{cases} 1, & b_0 = 1; \\ 2^{b_0 - 2} + 0.5, & b_0 \geq 3 \end{cases} \tag{5.35}$$

is achieved.

Hence, the optimal bit allocation scheme is

$$\langle \underbrace{1,\ 1,\ \cdots,\ 1}_{\rho - 1},\ R_0 - \rho + 1 \rangle\ . \tag{5.36}$$

5.5 Experimental Results and Performance Comparison

In this section, we exploit experimental results to demonstrate performance of the fast IA algorithm proposed above. In Section 5.5.1, computational complexity of the fast IA algorithm, the GA algorithm and the MDSQ is compared. In Section 5.5.2, we demon-

Table 5.2 Comparison among the fast IA, the GA and the MDSQ. For the fast IA and the GA, $\rho = 1$.

	Fast IA			GA				MDSQ			
R	s_{max}	s_{dif}	D_s (dB)	s_{max}	s_{dif}	D_s (dB)	t (mins)	ρ	s_{max}	s_{dif}	D_s (dB)
6	10	3	-19.8	10	1	-19.3	0.37	0.96	11	7	-20.7
8	36	7	-19.8	51	2	-17.6	90	1.08	37	26	-21.6
10	136	15	-19.8	439	51	-11.5	503	1.01	172	126	-21.7

strate which metric, the average spread $\bar{s}$ or the normalized average spread $\bar{s}_n$, is better to represent the average side distortion D_s. Then, performance of the fast IA algorithm is compared with that of existing algorithms over erasure and noisy channels in Section 5.5.3.

5.5.1 Computational Complexity

The fast IA is designed to cut down the computational complexity of designing the ERMDC encoder. In this section, we compare computational complexity of the proposed fast IA algorithm for $\mathcal{B}(R_0, 1)$ with the IA algorithm proposed in Chapter 4, in which the GA is used to find a "close-to-optimum" solution, and the MDSQ [18]. In the following, we mention the algorithm provided in Chapter 4 as *the GA*, which is the key step and consumes most of the computational complexity.

Experimental results are shown in Table 5.2. In experiments, uniformly distributed source samples in $[-0.5, 0.5]$ are quantized by uniform scalar quantizers. Average side distortions D_s in dB are shown as references. In the fast IA and the GA, $\rho = 1$, and, thus, $R = R_0 + 1$.

As mentioned in Section 4.4.1, spreads are independent on characteristics of source signals and channel noise. A smaller spread suggests that the reconstruction value is more accurate. A smaller maximum spread s_{max} means a lower side distortion. Smaller spread difference s_{dif} represents smaller variance of spreads and, hence, a more balanced system. Therefore, we use s_{max} and s_{dif} to evaluate IA schemes in high rate systems.

Simulations written in C++ run on a PC with Intel Pentium IV 3 GHz CPU, 1 G bytes memory, which means there is no memory limitation, especially, on the GA. The computational complexity of the GA is influenced by many factors, such as the number of genes, chromosomes, generations, and sorting algorithms. The elapsed time t in minutes

is utilized to evaluate the computational complexity of the GA. It indicates that if the time is long enough, GA can find a better IA scheme, i.e., smaller s_{dif}. As the size of the search space that is represented by R increases, the speed of finding a good solution decreases significantly. In a small search space, such as $R = 6$ bpss, the GA finds a better solution than the fast IA in less than one minute. However, when it searches a space with $R = 10$ bpss, a good solution cannot be obtained even in more than eight hours. In contrast, with substantial lower computational complexity of $O(2R_0)$, the fast IA algorithm always produces a "close-to-optimum" IA scheme in a constant time that is not more than one millisecond.

The performance achieved by using the MLIA of the MDSQ, illustrated in Section 2.3.4, is also shown in Table 5.2. In the MDSQ, ρ is usually not an integer. For purposes of comparison, we chose ρ of the MDSQ close to 1. It shows that IA schemes obtained by the fast IA algorithm have both smaller s_{max} and s_{dif}, even though side distortions D_s are a little higher. However, the MDSQ is not robust to bit errors. It suggests that IA schemes obtained by the fast IA algorithm are more robust against bit errors at the cost of higher side distortions, which is similar to the results shown in Chapter 4.

Index assignment of an MDSQ is realized in two steps: (i) all $L \approx 2^{R_0}$ MDSQ index pairs are scanned and, then, stored in an IA table; (ii) each SDSQ index l is mapped to an MDSQ index pair (i, j) by checking the IA table with 2^{R_0} entries. Consequently, for encoding K indices l, the total computational complexity is $(2^{R_0} + KR_0)$. It is a little higher than that of the fast IA algorithm, which is $2KR_0$.

As for $\mathcal{B}(R_0, \rho)$, $\rho \geq 2$, the computational complexity of the fast IA remains $O(2R_0)$. Moreover, because the fast IA algorithm decomposes each SDSQ index l to an ERMDC index pair (i, j) bit by bit, the ERMDC encoder can output two descriptions as information bits are being fed. That is to say, the implementation of the decomposition is so-called "on-the-fly". In addition, when applying MDSQ in bitplane-based image compression algorithms, since ρ is not an integer, a conversion between the number of bitplanes and the number of MDSQ quantization levels is required. In contrast, the fast IA algorithm can be easily adapted without this kind of conversion.

5.5.2 Spreads vs. Side Distortions

In Section 5.4.1, we claim that in high rate systems, the average spread $\bar{s}$ and the normalized average spread $\bar{s}_n$ can be used as metrics of the average side distortion so as to evaluate

Table 5.3 Bit allocation schemes of $\mathcal{B}(R_0, \rho)$, where $R_0 + \rho = 10$ and $\rho \geq 1$.

Partition sequences	R_0	$\bar{s}$	$\bar{s}_n$	D_s ($\times 10^{-4}$)
$\langle 9 \rangle$	9	128.5	0.2510	104
$\langle 1, 7 \rangle$	8	32.5	0.1270	26.1
$\langle 3, 5 \rangle$	8	56.5	0.2207	99.3
$\langle 1, 1, 5 \rangle$	7	8.5	0.0664	6.54
$\langle 1, 3, 3 \rangle$	7	14.5	0.1133	24.8
$\langle 1, 1, 1, 3 \rangle$	6	2.5	0.0391	1.73

different bit allocation schemes. Here, we use experiments to verify the validity of this statement.

In experiments, bit allocation schemes for $\mathcal{B}(R_0, \rho)$, where $R_0 + \rho = 10$ and $\rho \geq 1$, are compared. A uniform scalar quantizer is applied with i.i.d. source samples uniformly distributed in $[-0.5, 0.5]$. Six bit allocation schemes are obtained: $\mathcal{B}(9, 1) = \langle 9 \rangle$, $\mathcal{B}_1(8, 2) = \langle 1,\ 7 \rangle$, $\mathcal{B}_2(8, 2) = \langle 3,\ 5 \rangle$, $\mathcal{B}_1(7, 3) = \langle 1,\ 1,\ 5 \rangle$, $\mathcal{B}_2(7, 3) = \langle 1,\ 3,\ 3 \rangle$, $\mathcal{B}(6, 4) = \langle 1,\ 1,\ 1,\ 3 \rangle$. Associated $\bar{s}$, $\bar{s}_n$ and D_s are shown in Table 5.3. Behaviours of D_s and $\bar{s}_n$ are further compared in Fig. 5.9. It indicates that compared with $\bar{s}$, $\bar{s}_n$ provides more accurate description of the behaviour of D_s with respect to central distortions. Therefore, we use $\bar{s}_n$ as the metric in following experiments. According to experimental results, the optimal bit allocation schemes associated with $\rho = 1, 2, 3, 4$ are $\mathcal{B}(9, 1) = \langle 9 \rangle$, $\mathcal{B}_1(8, 2) = \langle 1,\ 7 \rangle$, $\mathcal{B}_1(7, 3) = \langle 1,\ 1,\ 5 \rangle$, $\mathcal{B}(6, 4) = \langle 1,\ 1,\ 1,\ 3 \rangle$ respectively. This experimental result is consistent with the optimal bit allocation scheme (5.36) obtained in Section 5.4.2.

5.5.3 Performance Against Packet Losses and Bit Errors

In order to evaluate performance over erasure and noisy channels, we compare the proposed method with the MDSQ (Section 2.3), and a variant of the PTSQ [47]. In this variant of the PTSQ, in order to protect from bit errors, the redundancy of $\mathcal{B}(R_0, \rho)$ is generated by repeating ρ most significant information bits. For example, for $\mathcal{B}(7, 3)$, $\mathbf{l} = [l_6 \cdots l_0]$, $\mathbf{i} = [l_6 l_5 l_4 l_3 l_2 l_0]$ and $\mathbf{j} = [l_6 l_5 l_4]$. Therefore, the side distortion of this variant is higher than that of the original PTSQ [47], in which the redundancy is optimized according to the side bitrate.

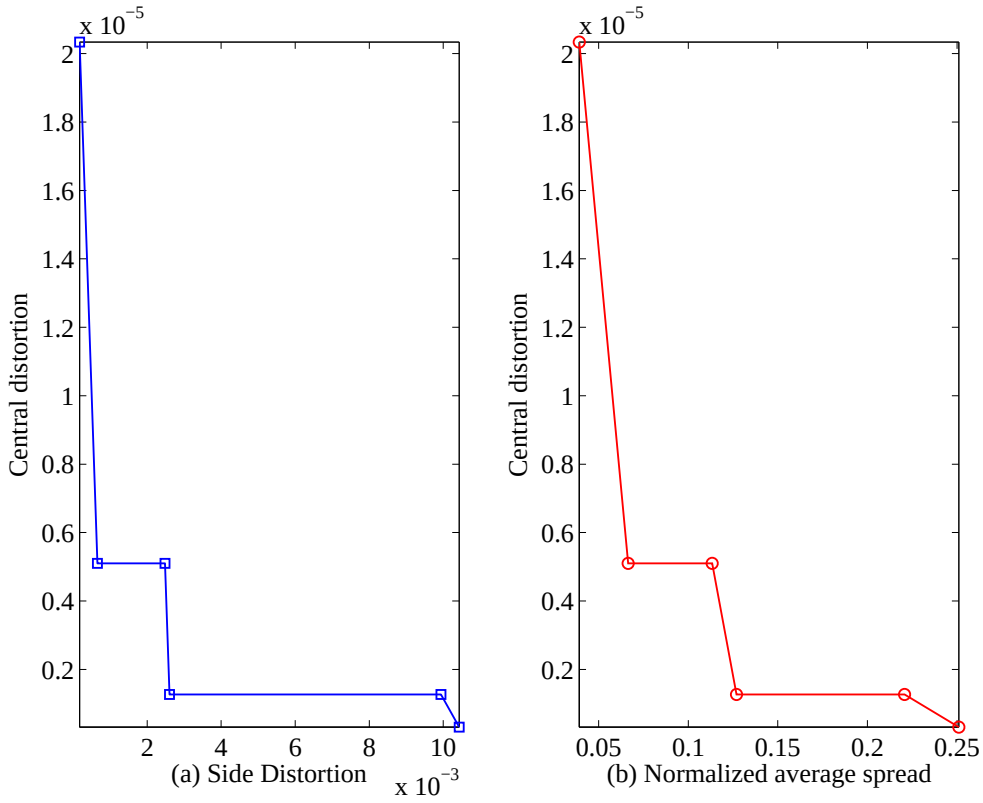


Fig. 5.9 Comparison between behaviours of side distortions and normalized average spread with respect to the central distortion at the total coding rate 10 bpss. Data points in each figure from the left to the right correspond to partition sequences $\langle 1,1,1,3\rangle$, $\langle 1,1,5\rangle$, $\langle 1,3,3\rangle$, $\langle 1,7\rangle$, $\langle 3,5\rangle$, $\langle 9\rangle$.

Over erasure channels, we test these three algorithms at the total coding rate 10 bpss. Experimental results are shown in Figs. 5.10 and 5.11. In Fig. 5.10, the source signal is i.i.d. uniformly distributed and quantized by a uniform scalar quantizer. In contrast, in Fig. 5.11, the source signal is i.i.d. normally distributed and quantized by a Lloyd-Max scalar quantizer. In both figures, only performance of the best schemes of the fast IA algorithm at given ρ, i.e., $\mathcal{B}(9,1) = \langle 9\rangle$, $\mathcal{B}(8,2) = \langle 1,\ 7\rangle$, $\mathcal{B}(7,3) = \langle 1,\ 1,\ 5\rangle$, and $\mathcal{B}(6,4) = \langle 1,\ 1,\ 1,\ 3\rangle$, is plotted. Performance of $\mathcal{B}(9,1)$, $\mathcal{B}(8,2)$, $\mathcal{B}(7,3)$ and $\mathcal{B}(6,4)$ of the PTSQ is also plotted. The redundancy of the MDSQ is generated by selecting different numbers of diagonals within the IA matrix (referred to Section 2.3).

Figures 5.10 and 5.11 show that side distortions of the fast IA are much lower than those of the PTSQ, and close to those of the MDSQ, no matter which source signal is

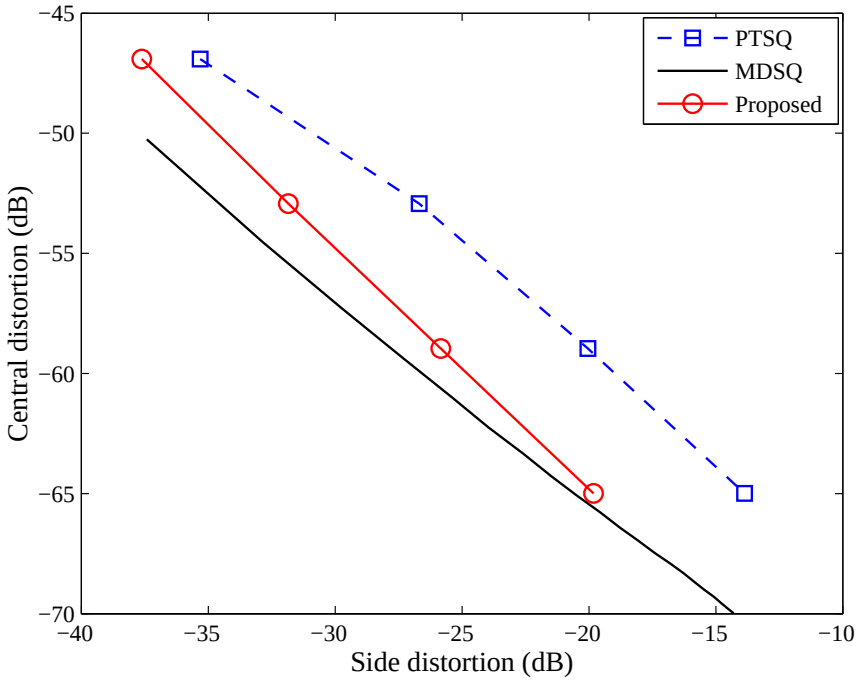


Fig. 5.10 Side distortions of various bit allocations with respect to central distortions at the total code rate 10 bpss. The source signal is i.i.d. uniformly distributed and quantized by a uniform scalar quantizer.

used. Especially, at low redundancy, the PTSQ provides poor performance, since its side distortions are greater or equal to the variance of the source. In contrast, side distortions of the proposed algorithm are still lower than the variance, and even lower than those of the MDSQ. Therefore, the proposed IA algorithm achieves higher robustness against packet losses than the PTSQ, and similar robustness with the MDSQ.

Figures 5.10 and 5.11 also indicate that except with one-bit redundancy, the relative performance of all three IA algorithms associated with different source signals is similar. Therefore, in following experiments, i.i.d. Gaussian source samples with zero mean and unit variance are used as the source signal and quantized by Lloyd-Max quantizers.

Thereafter, we compare the robustness of various IA schemes against both packet losses and bit errors. Bit errors are i.i.d. and uniformly distributed.

In Fig. 5.12, the performance of the proposed algorithm is plotted in the sense of distortions associated with BERs. For $R_0 = 7$, we compare performance of three fast IA schemes: $\mathcal{B}(7,1) = \langle 7 \rangle$, $\mathcal{B}_1(7,3) = \langle 1,\ 1,\ 5 \rangle$, and $\mathcal{B}_2(7,3) = \langle 1,\ 3,\ 3 \rangle$. The performance

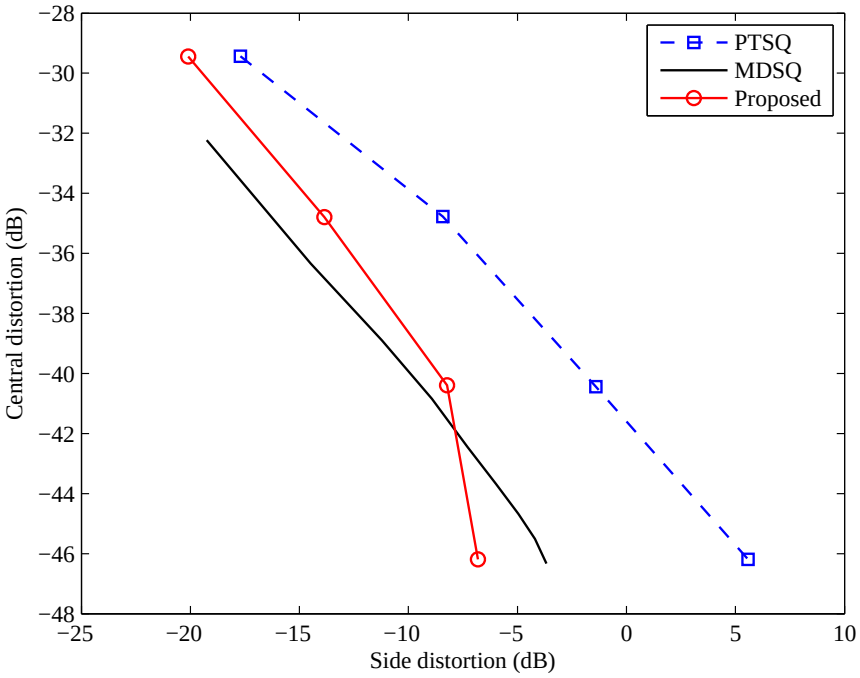


Fig. 5.11 Side distortions of various bit allocations with respect to central distortions at the total code rate 10 bpss. The source signal is i.i.d. normally distributed and quantized by a Lloyd-Max scalar quantizer.

of the MDSQ is also provided for the purpose of comparison. Since the redundancy of the MDSQ is determined by the number of diagonals chosen in the IA matrix, as described in Section 2.3.4, the number of SDSQ levels usually cannot be represented in integer number of bits. Hence, we choose an MDSQ with 4 bpss at each description and 124 SDSQ levels, which is close to $128 = 2^7$, the number of quantization levels of the SDSQ for the fast IA algorithm.

Figure 5.12 shows that with similar redundancy, the fast IA scheme $\mathcal{B}(7, 1)$ is more robust than the MDSQ against both packet losses and bit errors. As the redundancy increases, the proposed algorithm provides higher robustness against packet losses and bit errors. In addition, $\langle 1,\ 1,\ 5\rangle$ outperforms $\langle 1,\ 3,\ 3\rangle$ even with the same redundancy.

Figure 5.13 shows performance achieved by the proposed algorithm and the PTSQ against bit errors. $\mathcal{B}(7, 3)$ is realized respectively by the proposed algorithm, i.e., $\langle 1,\ 1,\ 5\rangle$, and the PTSQ. In the proposed algorithm, the optimal and suboptimal estimations described respectively in (3.28) and (3.33) are applied. The average side distortion D_s of

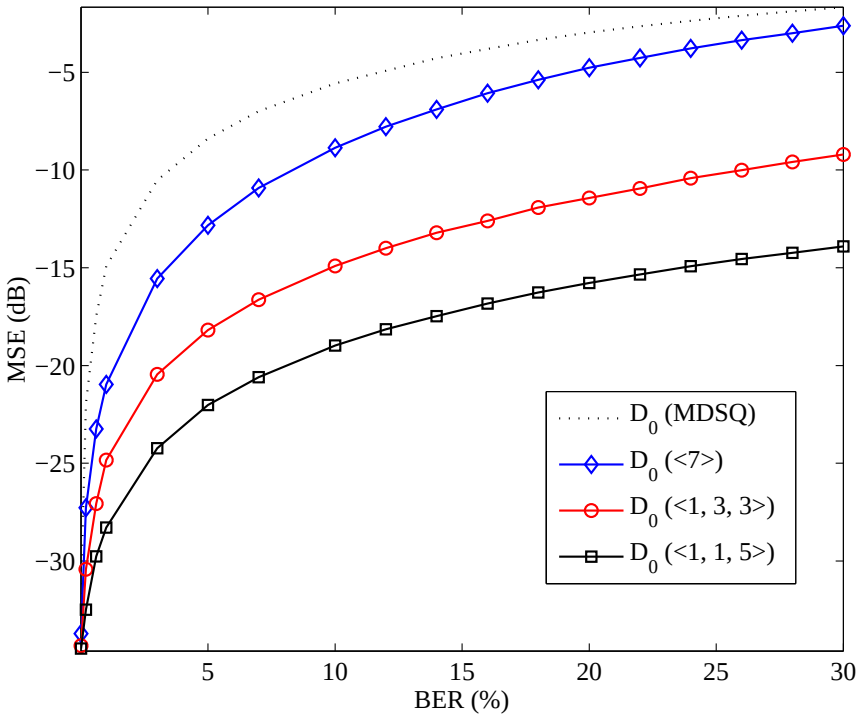


Fig. 5.12 Central distortion achieved by different ERMDC schemes and the MDSQ associated with BERs. The average side distortions associated with the MDSQ, $\langle 7 \rangle$, $\langle 1, 3, 3 \rangle$ and $\langle 1, 1, 5 \rangle$ are -4.26 dB, -6.78 dB, -8.65 dB and -13.86 dB, respectively.

$\langle 1,\ 1,\ 5 \rangle$ is also plotted as the reference. It shows that with knowledge of statistics of source samples and BERs, the optimal ERMDC apparently outperforms the PTSQ. With little knowledge of channel conditions and low computational complexity, the suboptimal ERMDC slightly outperforms the PTSQ. However, because D_s of the PTSQ scheme is much higher than that of $\langle 1,\ 1,\ 5 \rangle$, as shown in Fig. 5.11, the proposed algorithm achieves much higher robustness against packet losses than the PTSQ. Therefore, the PTSQ cannot provide as consistent robustness against both bit errors and packet losses as the proposed algorithm.

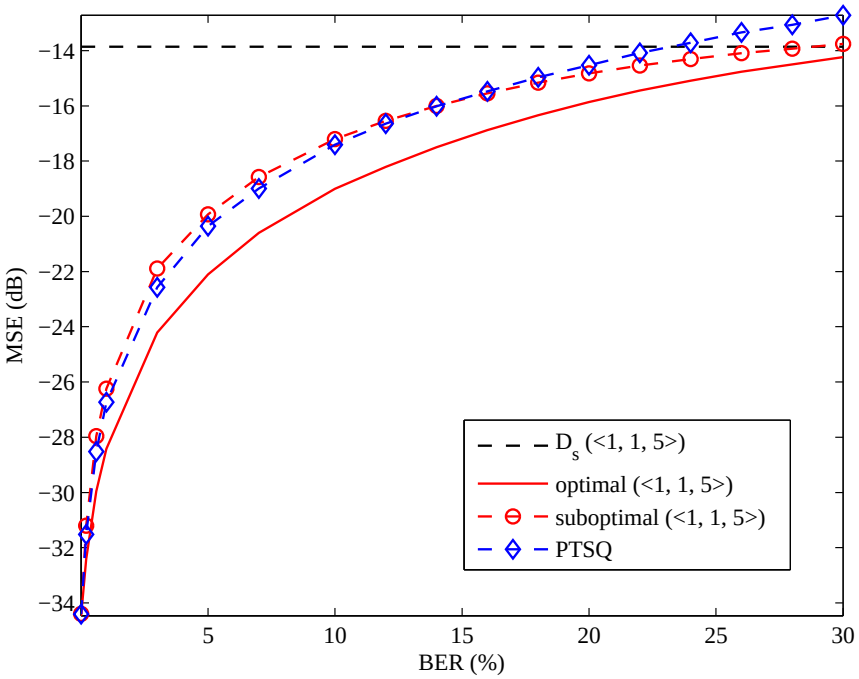


Fig. 5.13 Performance comparison between the ERMDC and the PTSQ associated with BERs.

5.6 Performance Analysis

In the previous sections of this chapter, we provide generalized fast IA algorithms for the ERMDC system, and their performance based on experiments. In order to provide a fast and simple means of evaluating the performance of the proposed algorithms, in this section, we give numerical solutions and/or closed-form formulas under such circumstances as high rate systems, uniform random signals and Gaussian random signals.

In the following, we summarize the derivation results. Please see Appendix A for the details. The obtained solutions are later utilized in Chapter 6 to derive the optimal redundancy associated with a given packet loss rate and BER.

5.6.1 High rate systems

Assume a high rate system, the source $x \in [x_{min}, x_{max})$ with the pdf $f_X(x)$ as the source signals. Let the range of x be $\delta = x_{max} - x_{min}$. Source samples x are uniformly quantized into $L = 2^{R_0}$ levels. The quantization partition associated with l or (i, j) is given by

$v_{ij} : \left[x_{ij}^L,\ x_{ij}^H\right)$. Thus, each interval

$$\Delta = x_{ij}^H - x_{ij}^L = \frac{\delta}{L} = 2^{-R_0}\delta.$$

The side bit rate is R_s bpss. Therefore, SDSQ indices $l = 0, 1, \cdots, L-1$, and RMDSQ indices $i, j = 0, 1, \cdots, M-1$, where $M = 2^{R_s}$. The output associated with l or (i, j) is approximately given by

$$c_{ij} \approx \frac{x_{ij}^H + x_{ij}^L}{2}\ . \tag{5.37}$$

Side codebooks associated with i and j are respectively given by

$$c_i^{(1)} \approx \frac{\sum\limits_{j\in\mathcal{J}} c_{ij}\cdot f_X(c_{ij})}{\sum\limits_{j\in\mathcal{J}} f_X(c_{ij})}\ , \quad \text{and} \tag{5.38}$$

$$c_j^{(2)} \approx \frac{\sum\limits_{i\in\mathcal{I}} c_{ij}\cdot f_X(c_{ij})}{\sum\limits_{i\in\mathcal{I}} f_X(c_{ij})}\ , \tag{5.39}$$

where $\mathcal{I}$ and $\mathcal{J}$ are defined in Section 3.2.1. See the details of the derivation in Appendix A.1.

Therefore, side distortions associated with i and j are respectively given by

$$D_i^{(1)} = \frac{\sum\limits_{j\in\mathcal{J}} \left(c_{ij} - c_i^{(1)}\right)^2 f_X(c_{ij})}{\sum\limits_{j\in\mathcal{J}} f_X(c_{ij})} + \frac{\Delta^2}{12}\ , \tag{5.40}$$

$$D_j^{(2)} = \frac{\sum\limits_{i\in\mathcal{I}} \left(c_{ij} - c_j^{(2)}\right)^2 f_X(c_{ij})}{\sum\limits_{i\in\mathcal{I}} f_X(c_{ij})} + \frac{\Delta^2}{12}\ . \tag{5.41}$$

Average side distortions corresponding to two descriptions are given by

$$D_1 = \frac{\sum_{i\in\mathcal{I}} D_i^{(1)} f_X(i)}{\sum_{i\in\mathcal{I}} f_X(i)}\,, \tag{5.42}$$

$$D_2 = \frac{\sum_{j\in\mathcal{J}} D_j^{(2)} f_X(j)}{\sum_{j\in\mathcal{J}} f_X(j)}\,, \tag{5.43}$$

where $f_X(i) = \sum_{j\in\mathcal{J}} f_X(c_{ij})$, and $f_X(j) = \sum_{i\in\mathcal{I}} f_X(c_{ij})$.

Therefore, the overall average side distortion D_s is obtained by (2.22).

Assume that x are i.i.d. Gaussian random variables with zero mean and unit variance. Then,

$$f_X(x) = \frac{1}{\sqrt{2\pi}}\, e^{-\frac{x^2}{2}}.$$

Therefore,

$$c_i^{(1)} = \frac{2\cdot\sum_{j\in\mathcal{J}}\left[\exp\left(-\frac{(x_{ij}^L)^2}{2}\right) - \exp\left(-\frac{(x_{ij}^H)^2}{2}\right)\right]}{\sum_{j\in\mathcal{J}}\left[\operatorname{erf}\left(\frac{x_{ij}^H}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{x_{ij}^L}{\sqrt{2}}\right)\right]}\,, \tag{5.44}$$

$$c_j^{(2)} = \frac{2\cdot\sum_{i\in\mathcal{I}}\left[\exp\left(-\frac{(x_{ij}^L)^2}{2}\right) - \exp\left(-\frac{(x_{ij}^H)^2}{2}\right)\right]}{\sum_{i\in\mathcal{I}}\left[\operatorname{erf}\left(\frac{x_{ij}^H}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{x_{ij}^L}{\sqrt{2}}\right)\right]}, \tag{5.45}$$

where the error function is defined as

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\, dt\ .$$

Thereafter, D_s can be obtained by using (5.40)(5.41) and (2.22).

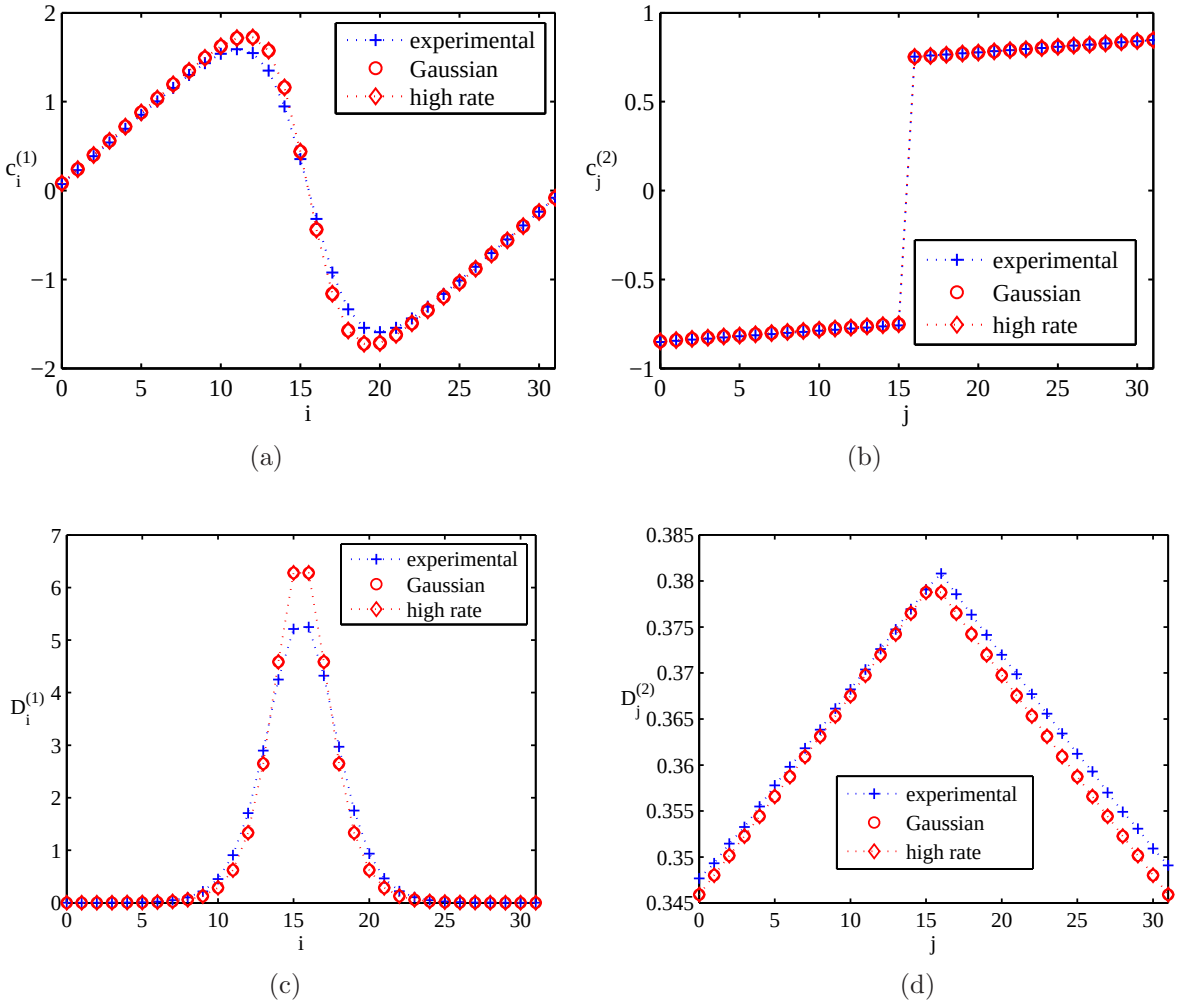


Fig. 5.14 Comparison between experimental and analytical results associated with $\mathcal{B}(10,0)$: (a) side codebook $c_i^{(1)}$, (b) side codebook $c_j^{(2)}$, (c) side distortion $D_i^{(1)}$, and (d) side distortion $D_j^{(2)}$.

Experimental and Analytical Results

We use Gaussian random variables with zero mean and unit variance to test the numerical solutions we obtained above. Side codebooks $c_i^{(1)}$ and $c_j^{(2)}$, side distortions $D_i^{(1)}$ and $D_j^{(2)}$ for $R_0 = 10$ bpss, $R_s = 5$ bpss and $\rho = 0$ bpss in high rate systems are calculated according to (5.38)–(5.41), respectively. $c_i^{(1)}$ and $c_j^{(2)}$ can also be calculated specifically for Gaussian random variables in high rate systems according to (5.44) and (5.45), respectively. Experimental and analytical results are compared in Fig. 5.14. It shows that analytical

Table 5.4 Comparison of side distortions between experimental and analytical results associated with $\mathcal{B}(10, 0)$.

Results	D_1	D_2	D_s
Experiment	0.2430	0.3691	0.3061
High rate	0.2110	0.3625	0.2867
Gaussian	0.2110	0.3625	0.2867

and experimental results are close. Furthermore, average side distortions are illustrated in Table 5.4. It indicates that numerical methods provide very similar side distortions to the actual ones.

Therefore, numerical methods, especially, the numerical method for arbitrary random signals in high rate systems, give us a good way to estimate actual side distortions based on a-priori statistics of the source signals. Based on this, in the next chapter, we try to determine the optimal redundancy associated with a given BER and PLR in order to achieve the highest robustness in the sense of rate distortion.

5.6.2 Uniform input

Assume that the source $x \in [0, \sigma)$ is an i.i.d. and uniformly distributed random signal. The overall average side distortion D_s of $\mathcal{B}(R_0, \rho)$ is defined as a function of ρ. When $\rho \geq 2$, we only consider the optimal bit allocation provided in (5.36). That is to say, the bit allocation scheme is

$$\langle \underbrace{1,\ 1,\ \cdots,\ 1}_{\rho - 1},\ R_0 - \rho + 1 \rangle\ .$$

Therefore, based on the derivation provided in Appendix A.2,

$$D_s(\rho) = \begin{cases} \dfrac{\sigma^2}{24}\ , & \rho = 0; \\ \\ 2^{-2\rho} \cdot \dfrac{\sigma^2}{24}\ , & \text{otherwise}, \end{cases} \tag{5.46}$$

i.e., $D_s(\rho) = D_s(0) - 6.02\rho$ dB. When $\sigma = 1$, $D_s(\rho) = -13.8 - 6.02\rho$ dB.

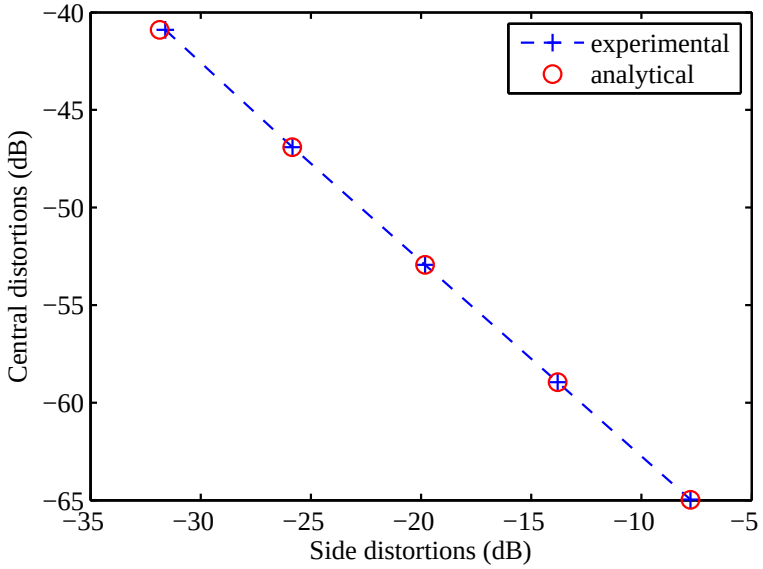


Fig. 5.15 Comparison of experimental and analytical results associated with IA schemes: $\mathcal{B}(10,0)$, $\mathcal{B}(9,1) = \langle 9 \rangle$, $\mathcal{B}(8,2) = \langle 1,7 \rangle$, $\mathcal{B}(7,3) = \langle 1,1,5 \rangle$, and $\mathcal{B}(6,4) = \langle 1,1,1,3 \rangle$.

Experimental and analytical results

Uniformly distributed random signals $x \in [-1,1]$ are used as the source. IA schemes with the total coding rate $R = 10$ and $\rho \geq 0$ are obtained according to the method provided in Section 5.3. We only consider the optimal IA schemes in terms of (5.36). Therefore, there are five IA schemes: $\mathcal{B}(10,0)$, $\mathcal{B}(9,1) = \langle 9 \rangle$, $\mathcal{B}(8,2) = \langle 1,7 \rangle$, $\mathcal{B}(7,3) = \langle 1,1,5 \rangle$, and $\mathcal{B}(6,4) = \langle 1,1,1,3 \rangle$. It shows that analytical results according to (5.46) are very close to experimental results. There is a little difference at $\mathcal{B}(6,4)$, because we ignore $\frac{\Delta^2}{24}$ when making an approximation in (A.26).

5.6.3 Gaussian random signals

Assume a high rate system, i.i.d. Gaussian random variables $x \in [-\frac{\delta}{2}, \frac{\delta}{2}]$ with mean $\mu = 0$ and variance $\sigma = 1$ are uniformly quantized and represented by SDSQ indices

$$l = -\frac{L}{2}, \cdots, -1, 0, 1, \cdots, \frac{L}{2} - 1 \, .$$

Then, the partition interval $\Delta = \frac{\delta}{L}$. Assume $\delta >> \sigma$, that is to say, very few variables fall out of this range $[-\frac{\delta}{2}, \frac{\delta}{2}]$, so that the effect of clipping can be ignored.

SDSQ indices l are one-to-one mapped to MDSQ index pairs (i, j) by using the IA method for $\mathcal{B}(R_0, 0)$ provided in Section 5.3.1, where

$$i = -\frac{M}{2}, \cdots, -1, 0, 1, \cdots, \frac{M}{2} - 1,$$
$$j = -\frac{M}{2}, \cdots, -1, 0, 1, \cdots, \frac{M}{2} - 1.$$

Note that for convenience of derivation, SDSQ indices l, ERMDC indices i and j are here defined as signed integers, instead of unsigned integer as usually defined in the rest of this dissertation.

Let l^L and l^H denote SDSQ indices $l < 0$ and $l \geq 0$, respectively. When $l < 0$ or $j < 0$,

$$l^L = \frac{M}{2} i + j - \frac{M}{2};$$

when $l \geq 0$ or $j \geq 0$,

$$l^H = \frac{M}{2} i + j + \frac{L}{4}.$$

We distinguish l^L and l^H by replacing two-sided j with corresponding $\tilde{j} = |j|$, that is to say, $\tilde{j} = 0, 1, \cdots, \frac{M}{2} - 1$. Then,

$$l^L = \frac{M}{2}\, i + \tilde{j} - \frac{L}{4}; \tag{5.47}$$
$$l^H = \frac{M}{2}\, i + \tilde{j} + \frac{L}{4}. \tag{5.48}$$

So l^L and l^H are symmetric.

In a high rate system, according to (5.37),

$$c_{ij} = \frac{x_{ij}^H + x_{ij}^L}{2} = \left(l + \frac{1}{2}\right) \cdot \Delta. \tag{5.49}$$

Probability density functions f_i^L and f_i^H associated with $x < 0$ and $x \geq 0$ respectively are given by

$$\begin{aligned} f_i^L &\approx f_X(x_{ij}^L) \\ &= \frac{1}{\sqrt{2\pi}} \cdot \exp\left\{ -\left(Mi - \frac{L}{2}\right)^2 \cdot \frac{\Delta^2}{8} \right\}, \end{aligned} \tag{5.50}$$

and

$$\begin{aligned} f_i^H &\approx f_X(x_{ij}^H) \\ &= \frac{1}{\sqrt{2\pi}} \cdot \exp\left\{ -\left(Mi + \frac{L}{2} \right)^2 \cdot \frac{\Delta^2}{8} \right\} \end{aligned} \tag{5.51}$$

Define Φ_i as

$$\Phi_i = \frac{f_i^H}{f_i^L + f_i^H} = \frac{1}{1 + \exp\left(\frac{M^3}{4}\, i \cdot \Delta^2 \right)} \ . \tag{5.52}$$

Hence, side codebooks $c_i^{(1)}$ and $c_j^{(2)}$ are given by

$$c_i^{(1)} = \frac{M}{2} \cdot \left(i + M\Phi_i - \frac{M}{2} \right) \cdot \Delta \ ; \tag{5.53}$$

$$c_j^{(2)} \approx \begin{cases} -\sqrt{\dfrac{2}{\pi}} \ , & j < 0 \ , \\ \sqrt{\dfrac{2}{\pi}} \ , & j \geq 0 \ . \end{cases} \tag{5.54}$$

Average side distortions D_1 and D_2 are given by

$$D_1 = \frac{M^5}{8}\, \Delta^3 \cdot \sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} \frac{f_i^L\, f_i^H}{f_i^L + f_i^H} + \frac{\Delta^2}{12} \ ; \tag{5.55}$$

$$D_2 \approx 1 - \frac{2}{\pi} + \frac{\Delta^2}{12}. \tag{5.56}$$

It shows that (5.55) is still not very simple. In addition, (5.55) and (5.56) are only for a given IA scheme and Gaussian distribution. Therefore, the numerical method provided in Section 5.6.1 is preferred in further analysis.

5.7 Embedded Fast Index Assignment

Embedded coding is usually utilized in progressive image transmissions. The embedded MDSQ was proposed in [54] for progressive image transmissions over unreliable channels.

The IA algorithm proposed in Section 5.3 is not originally designed for progressive transmission. Especially, in the case of $\mathcal{B}(R_0, 1)$, since the parity bits, which are the LSBs of i and j, are generated by using the MSB l_{R_0} of SDSQ index l, decoding and error detection has to proceed after receiving all bits of i and j.

However, the proposed IA algorithm can be easily modified to generate embedded coding. Here, we provide two ways to extend the proposed IA algorithm to accommodate robust progressive transmissions.

In order to develop these embedded codes, we consider a more general case of allocating information bits and redundancy bits. Specifically, for $\mathcal{B}(R_0, \rho)$, R_0 bits of information are divided into γ partitions, so that in the k-th partition there are b_k bits of information protected by e_k bits of redundancy, subject to

$$R_0 = \sum_{k=0}^{\gamma-1} b_k \ , \tag{5.57}$$

$$\rho = \sum_{k=0}^{\gamma-1} e_k \ . \tag{5.58}$$

The corresponding bit allocation scheme is $\big\langle (b_{\gamma-1}, e_{\gamma-1}), \cdots, (b_1, e_1), (b_0, e_0) \big\rangle$, where the $(\gamma-1)$-th partition is the most significant partition, and the zeroth partition is the least significant partition.

Different from previous discussion, γ may not equal to ρ, and e_k may not be one. As a particular case, when $\gamma = \rho$ and $e_k = 1$, the bit allocation scheme is $\big\langle (b_{\rho-1}, 1), \cdots, (b_1, 1), (b_0, 1) \big\rangle$. By omitting $e_k = 1$, we get $\big\langle b_{\rho-1}, \cdots, b_1, b_0 \big\rangle$, which is the bit allocation scheme discussed before.

Unequal error protection

In the first method, for $\mathcal{B}(R_0, \rho)$, R_0 bits of information are divided into $(\rho+1)$ partitions, so that in each of ρ most significant partitions, one-bit information is protected by one-bit redundancy. Specifically, the bit allocation scheme is given by

$$\big\langle \underbrace{(1,1),\ \cdots,\ (1,1)}_{\rho}\ ,\ (R_0-\rho, 0) \big\rangle \ . \tag{5.59}$$

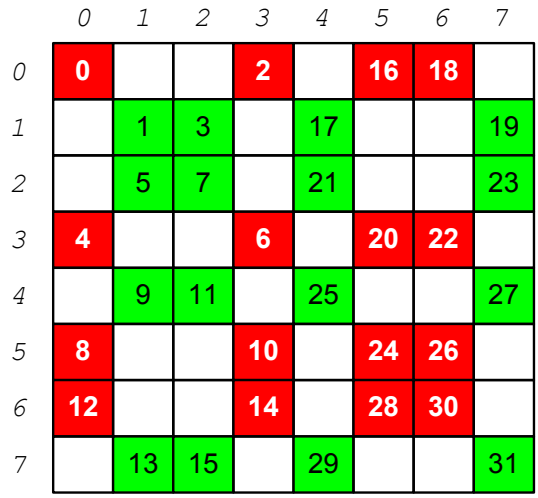


Fig. 5.16 Embedded IA scheme obtained by the proposed algorithm.

This is actually an *unequal error protection* (UEP) method. Redundancy bits are generated by duplicating information bits in the same partition. The IA of the least significant partition is obtained by using the method for $\mathcal{B}(R_0-\rho, 0)$, which is provided in Section 5.3.1. For a given $\mathcal{B}(R_0, \rho)$, the average spread of this method equals to the IA method provided in Section 5.3.

Therefore, in this method, once an information bit is received, the decoder starts to set the corresponding bit in the output symbol until all bits are received. In addition, (5.17) is still satisfied and allows for error detection.

Equal error protection

The second method exploits *equal error protection* (EEP) to generate ERMDC index pairs (i, j). This method is almost the same as the method provided in Section 5.3, except that for $\mathcal{B}(R_0, 1)$, parity bits i_0 and j_0 are produced by using the LSB l_0 instead of the MSB l_{R_0-1}. Specifically,

$$\begin{aligned} \mathbf{i} &= \left[l_{(R_0-2)} \cdots l_{\left(\frac{R_0-1}{2}\right)} \, i_0 \right], \qquad \text{and} \\ \mathbf{j} &= \left[l_{(R_0-1)} \, l_{\left(\frac{R_0-3}{2}\right)} \cdots l_1 \, j_0 \right], \end{aligned} \tag{5.60}$$

where

$$\begin{aligned} i_0 &= l_0 \oplus \bigoplus_{k=\frac{R_0-1}{2}}^{R_0-2} l_k, \qquad \text{and} \\ j_0 &= l_0 \oplus l_{(R_0-1)} \oplus \bigoplus_{k=1}^{\frac{R_0-3}{2}} l_k. \end{aligned} \tag{5.61}$$

In cases of $\rho = 0$ and $\rho \geq 2$, the IA methods given in Sections 5.3.1 and 5.3.4 are utilized respectively.

The embedded EEP IA scheme for $\mathcal{B}(5,1)$ is illustrated in Fig. 5.16. The average spread $\bar{s}$ of this IA scheme is 16. It is larger than 8.5, which is the average spread of the IA scheme shown in Fig. 5.3(b). The IA scheme shown in Fig. 5.3(b) is designed according to the method provided in Section 5.3.2, i.e., using the MSB l_{R_0-1} to produce parities.

Similar to the UEP method, in this method, once an information bit is received, decoding can proceed. In addition, (5.17) is still satisfied and allows for error detection.

Performance comparison

Here, we compare performance of three fast IA methods, which are the ERMDC IA method provided in Section 5.3, embedded UEP and EEP IA algorithms proposed in this section. Performance of the MDSQ and PTSQ is also plotted as the reference. In the following experiments, Gaussian signal with zero mean and unit variance is used as the source and uniformly quantized.

Side distortions with respect to central distortions of various IA methods are compared in Fig. 5.17. The total coding rate $R = 10$, and $\rho = 1, 2, 3, 4$. It shows that performance of the ERMDC and embedded UEP IA methods is very close. Their performance is much better than that of the PTSQ, a little worse than that of the MDSQ at high redundancy, and better at low redundancy. The embedded EEP IA method is worse than the other two fast IA methods, but outperforms the PTSQ, especially at low redundancy.

In Fig. 5.18, central distortions achieved by these IA methods at $R = 10$ and $\rho = 1$ are plotted. It shows that the ERMDC accomplishes the highest robustness. The UEP achieves similar robustness to the MDSQ and PTSQ; however, the EEP outperforms UEP marginally.

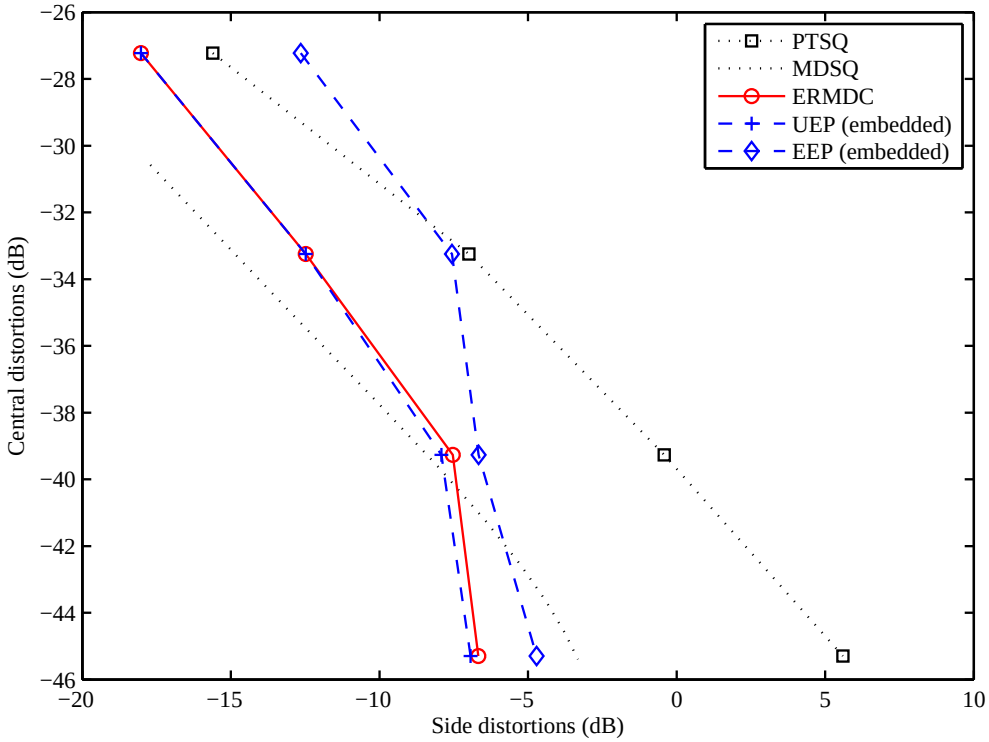


Fig. 5.17 Central distortions achieved by various IA methods associated with side distortions.

In summary, the EEP and UEP outperform the MDSQ and PTSQ against both packet losses and bit errors, even though they are worse than the ERMDC. Taking into account their low computational complexity and high robustness, they are very attractive in progressive transmissions.

5.8 Discussion and Conclusion

In this chapter, a novel generalized fast IA algorithm for the ERMDC is proposed. The proposed algorithm exploits the redundancy equivalent to any arbitrary number of parity bits to enhance the robustness of the ERMDC system against both bit errors and packet losses. Without checking a predetermined IA table, the proposed algorithm is able to be implemented "on-the-fly". Experimental results show that the proposed algorithm achieves consistent robustness against both bit errors and packet losses. When only taking into account packet losses, the proposed algorithm accomplishes performance close to the MDSQ, and much better than the PTSQ. When only considering bit errors, the proposed algorithm

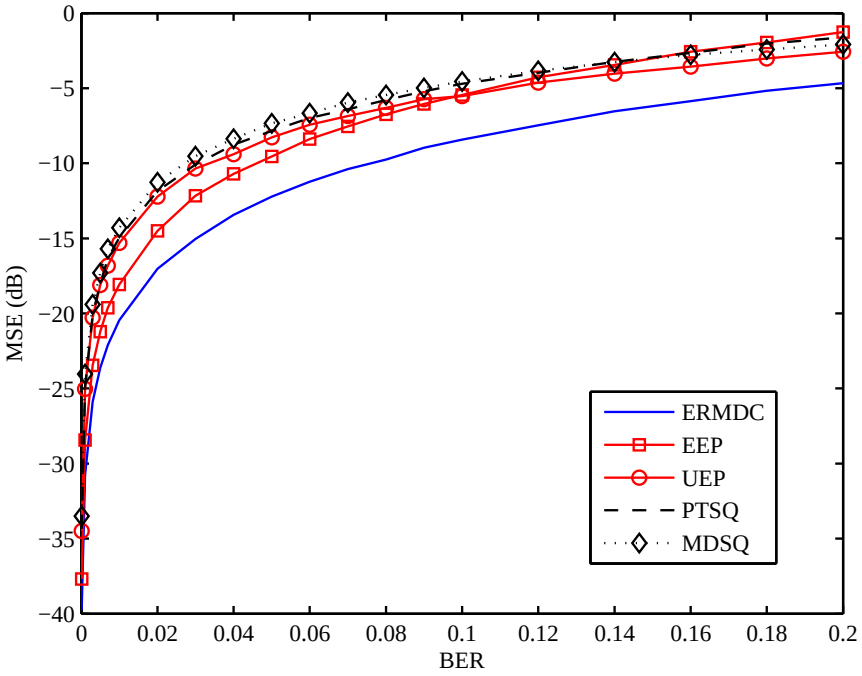


Fig. 5.18 Distortion achieved by various IA methods associated with BERs.

outperforms both the MDSQ and the PTSQ.

In the MDSQ, since IA schemes must be determined before MDSQ encoding, it is difficult to change the redundancy thereafter. In contrast, the proposed algorithm can flexibly adjust granularities of partitions at any time during ERMDC encoding in terms of the current requirement of robustness and channel conditions.

So as to conduct further analysis of the proposed ERMDC system, a numerical method for arbitrary IA scheme and source is provided in high rate systems. In addition, closed form formulas of the overall average side distortion are derived for the proposed IA algorithm with uniformly distributed signals. Experimental results show that the numerical method and closed form formulas provide very good description of the associated ERMDC codec.

In order to accommodate to progressive transmissions, based on the IA algorithm proposed in Section 5.3, two embedded fast IA algorithms are presented in Section 5.7. Both of them are capable of protecting from packet losses and bit errors. It suggests that the IA algorithms proposed in this chapter can be flexibly and easily adapted to various application requirements.

In the next chapter, we will try to apply the ERMDC codec over slow Rayleigh fading channels. Because FEC codes are well-known for their high efficiency on protecting from bit errors, we are going to combine the ERMDC with FEC codes to achieve higher efficiency and robustness. In addition, we will investigate how much redundancy is appropriate for a certain channel condition.

Chapter 6

Performance of ERMDC

In Chapters 3 and 4, we propose the decoder and encoder of the ERMDC. Experimental results show that the ERMDC codec outperforms the traditional MDSQ. In this chapter, we will test the proposed ERMDC codec over typical wireless channels, such as Rayleigh fading channels, in Section 6.1[1]. In Chapter 5, a generalized fast IA algorithm is developed to simplify the design of the ERMDC encoder. In Section 6.2, the ERMDC is combined with FEC codes to form an ERMDC system, which provides protection against packet losses and bit errors[2]. This ERMDC system is tested on a novel test bed that is a three-state Markov chain. The experimental results show that the ERMDC system outperforms both packet loss recovery techniques respectively based on MDC and FEC. Furthermore, the optimal redundancy in the ERMDC system is derived in Section 6.3.

6.1 Soft Input ERMDC for Rayleigh Fading Channels

Instead of binary symmetric channels (BSC) used in previous chapters, in this section, we attempt to apply the ERMDC codec over wireless channels. The MDSQ was applied by Yan and Vaishampayan to achieve low-delay communications over Rayleigh fading channels [84]. In order to match the Rayleigh fading model to the classic on/off channel model of MDC systems, a fading parameter was predetermined as a threshold. Even though an upper bound of the channel erasure probability was derived, it was too loose to be used for

[1]This work has been presented at the IEEE International Conference on Multimedia and Expo, July 2007.

[2]This work has been presented at the 69th IEEE Vehicular Technology Conference, April 2009.

the MDSQ design at low channel signal-to-noise ratio (SNR). Zhou et al. extended the multiple-channel optimized quantizer to Rayleigh fading channels by applying soft decision at the receiver [90]. In order to determine reconstruction symbols "0", "1" and "erasure", thresholds were set up in terms of derived channel erasure rates and BERs. Based on the dependency between descriptions, symbol "erasure" was estimated so as to reduce the resulting distortion. Since "erasures" were assumed in both descriptions, the achieved improvement was very limited.

Because FEC codes can only provide limited protection from bit errors, it is possible that the number of bit errors in only one description exceeds the error correction capability of the applied FEC codes. We considered this situation and the situation with two correct descriptions, and developed optimal and suboptimal ERMDC decoders to reduce the distortion caused by detectable transmission errors in Chapter 3. A robust IA scheme is provided in Chapter 4 to improve the performance of the ERMDC decoder. This pair of ERMDC encoder and decoder can be seen as a kind of joint source-channel coding. In Chapters 3 and 4, received modulated signals are demodulated to a bitstream. Once transmission errors in this bitstream are detected, the reproduction values are estimated so as to reduce the reconstruction distortion. Instead, in this section, by directly exploiting received modulated signals, we estimate more accurately the reproduction levels of detected transmission errors so as to achieve higher robustness against noisy and fading channels. The resulting distortions of reconstructed signals over additive white Gaussian noise (AWGN) and slow Rayleigh fading channels are lower than those obtained by the conventional soft detector. This performance improvement is achieved without introducing any extra redundancy or much computational complexity.

We will present the details of the novel soft decoder based on AWGN in Section 6.1.1. This soft decoder is further applied for slow Rayleigh fading channels. In Section 6.1.2, experimental results show that the proposed algorithm achieves graceful performance degradation, and outperforms existing works. The conclusion is given in the last subsection.

6.1.1 ERMDC over Rayleigh Fading Channels

We consider that information is transmitted over two independent wireless channels without memory. These two channels might be two realizations of the same channel at different time instances. In this subsection, two scenarios are discussed: first, the transition probability is derived for AWGN channels without fading; thereafter, this probability is extended to

channels with slow Rayleigh fading and AWGN.

AWGN channels

Each source sample $x \in X$ is encoded into an R_0-bit SDSQ index l, then decomposed into an ERMDC index pair (i, j), and, thereafter, transmitted via a BPSK modulator over two independent wireless channels. In particular, each ERMDC index i and j is coded in R_s bits as $[i_{R_s-1} \ \cdots \ i_1 \ i_0]^T$ and $[j_{R_s-1} \ \cdots \ j_1 \ j_0]^T$. Each bit of i and j is modulated as $s_m^{(1)}$ and $s_m^{(2)}$, respectively, where $s_m^{(1)}$ or $s_m^{(2)} \in \{+\sqrt{E_s}, -\sqrt{E_s}\}$, $m \in \{0, \ \cdots, \ R_s - 1\}$. Thus, each symbol of i and j is modulated via BPSK as $\mathbf{S}^{(n)} = \left[s_{R_s-1}^{(n)}, \ \cdots, \ s_0^{(n)}\right]^T$, respectively, where $n = 1, 2$. When only considering wireless channels with AWGN $\mathbf{W}^{(n)}$, the received signal $\mathbf{Y}^{(n)}$ corresponding to the transmitted signal $\mathbf{S}^{(n)}$ of n-th channel is given by

$$\mathbf{Y}^{(n)} = \mathbf{S}^{(n)} + \mathbf{W}^{(n)}, \quad n = 1, 2, \tag{6.1}$$

where $\mathbf{W}^{(n)}$ is AWGN with zero mean and variance σ^2, i.e., its distribution is

$$p_{_W}(w) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{w^2}{2\sigma^2}\right). \tag{6.2}$$

The received indices $\hat{i}$ and $\hat{j}$ can be determined by using soft detector

$$\hat{i} = \arg\min_{i \in \mathcal{I}} \sum_{m=0}^{R_s-1} \left\| y_m^{(1)} - s_m^{(1)} \right\|^2, \tag{6.3}$$

$$\hat{j} = \arg\min_{j \in \mathcal{J}} \sum_{m=0}^{R_s-1} \left\| y_m^{(2)} - s_m^{(2)} \right\|^2, \tag{6.4}$$

where $y_m^{(1)}$ and $y_m^{(2)}$ are received signals corresponding to m-th bits in binary representations of received indices $\hat{i} \in \mathcal{I}$ and $\hat{j} \in \mathcal{J}$, respectively. Because $\hat{i}$ and $\hat{j}$ are separately determined, if $a^{-1}(\hat{i}, \hat{j}) \notin \mathcal{L}$, transmission errors occur.

The FEC codes added in each description at the transmitter is utilized to correct bit errors. In order to illustrate the soft decoder's operation, we assume that all bit errors in Description 1 are corrected after FEC decoding, i.e., $i = \hat{i}$; however, bit errors in Description 2 cannot be corrected due to exceeding the error correction capability of applied FEC codes, i.e., $j \neq \hat{j}$. In this case, soft decision is used by Yang and Vaishampayan [84].

Referring to the corrected index $\hat{i}$, $\hat{j}$ can be determined by

$$\hat{j} = \arg\min_{j\in\mathcal{J}_i} \sum_{m=0}^{R_s-1} \left\| y_m^{(2)} - s_m^{(2)} \right\|^2 . \tag{6.5}$$

Instead, we utilize (3.28) to develop a novel method to estimate the reconstruction level $e_{opt}^{(1)}(i,\hat{j})$ corresponding to the unused index pair $(i,\hat{j})$, where $\hat{j} \in \bar{\mathcal{J}}_i$. The transition probability $P_e(\hat{j}|j)$ in (3.28) is given by

$$P_e(\hat{j}|j) = \prod_{m=0}^{R_s-1} P_e(\hat{j}_m|j_m), \tag{6.6}$$

where j_m is the m-th bit of transmitted index j, $\hat{j}_m$ is the received bit corresponding to j_m. In light of (6.1) and (6.2), (6.6) can be written as

$$P_e(\hat{j}|j) = \left(\frac{1}{\sigma\sqrt{2\pi}}\right)^{R_s} \exp\left(-\frac{\sum_{m=0}^{R_s-1} \left\| y_m^{(2)} - s_m^{(2)} \right\|^2}{2\sigma^2} \right). \tag{6.7}$$

Rayleigh fading channels

Next, we consider the scenario with slow Rayleigh fading channels. Let $\mathbf{Z}^{(n)}$, $\mathbf{A}^{(n)}$ and $\mathbf{W}^{(n)}$, $n = 1, 2$, represent respectively the received signal, the Rayleigh parameter, and AWGN corresponding to the transmitted signal $\mathbf{S}^{(n)}$, where

$$\mathbf{Z}^{(n)} = \mathbf{A}^{(n)}\mathbf{S}^{(n)} + \mathbf{W}^{(n)}, \quad n = 1, 2. \tag{6.8}$$

Under the assumption of slow fading, the Rayleigh parameter $\mathbf{A}^{(n)}$ is assumed to be constant over the interval of transmitting a symbol. $\mathbf{A}^{(n)} = diag\{a^{(n)}\}$, $n = 1, 2$, where $a^{(n)}$ are Rayleigh parameters corresponding to indices i and j, respectively. The distribution of $\mathbf{A}^{(n)}$ is

$$p_{_A}(a) = \frac{2a}{\Omega}\exp(-\frac{a^2}{\Omega}), \quad a \geq 0,$$

where $\Omega = E\{A^2\}$.

With the assumption that perfect estimates of the Rayleigh parameters $\mathbf{A}^{(n)}$ are avail-

able at the receiver, the soft decision described in (6.3), (6.4) and (6.5) are changed correspondingly to

$$\hat{i} = \arg\min_{i\in\mathcal{I}} \sum_{m=0}^{R_s-1} \left\| z_m^{(1)} - a^{(1)} s_m^{(1)} \right\|^2, \tag{6.9}$$

$$\hat{j} = \arg\min_{j\in\mathcal{J}} \sum_{m=0}^{R_s-1} \left\| z_m^{(2)} - a^{(2)} s_m^{(2)} \right\|^2, \tag{6.10}$$

and

$$\hat{j} = \arg\min_{j\in\mathcal{J}_i} \sum_{m=0}^{R_s-1} \left\| z_m^{(2)} - a^{(2)} s_m^{(2)} \right\|^2. \tag{6.11}$$

The transition probability $P_e(\hat{j}|j)$ for slow Rayleigh fading channels can be calculated by

$$P_e(\hat{j}|j) = \left(\frac{1}{\sigma\sqrt{2\pi}}\right)^{R_s} \exp\left(-\frac{\sum_{m=0}^{R_s-1} \left\| z_m^{(2)} - a_j^{(2)} s_m^{(2)} \right\|^2}{2\sigma^2} \right). \tag{6.12}$$

For either AGWN or slow Rayleigh fading channels, once $P_e(\hat{j}|j)$ is obtained according to (6.7) or (6.12), $e_{opt}^{(1)}(i,\hat{j})$ can be calculated by using (3.28).

6.1.2 Experimental Results

Experimental results are expressed as signal-to-distortion ratio (SDR) in dB versus channel SNR measured at the receiver end. Specifically, SDR $= \frac{E_s}{D_0}$, SNR $= \frac{E_s}{W_0}$ for AWGN channels and SNR $= \Omega\frac{E_s}{W_0}$ for Rayleigh fading channels, where W_0 is the power of AWGN.

In the following experiments, we test various combinations of three decoders and two encoders:

- Encoders:
  - *MDSQ encoder*: The traditional IA scheme such as Fig. 6.1(a) is used.
  - *ERMDC encoder*: The robust IA scheme such as Fig. 6.1(b), which is developed in Chapter 4, is applied.
- Decoders:

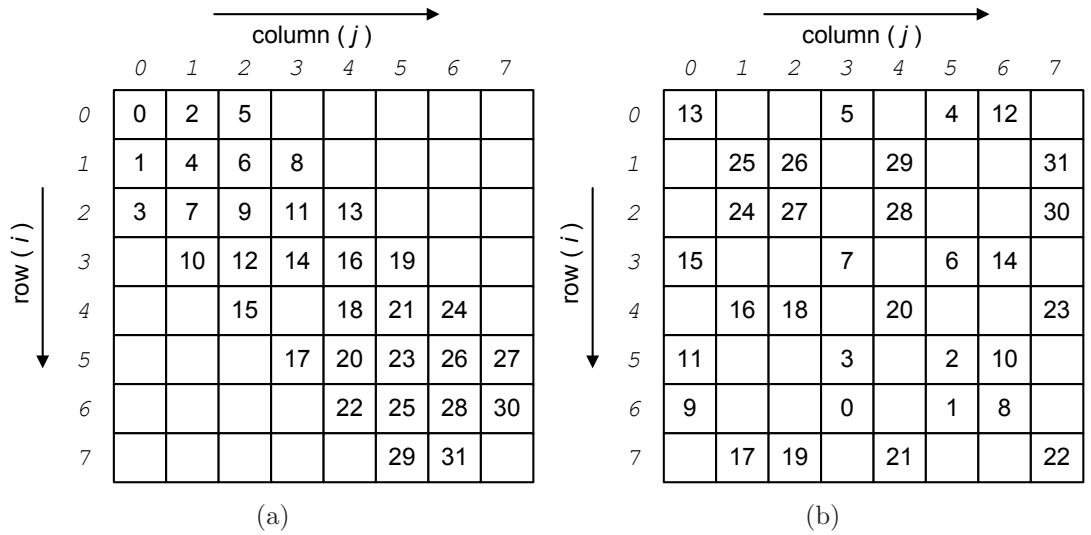


Fig. 6.1 IA schemes at 3 bpss/channel: (a) MDSQ IA, (b) robust IA.

- *Soft detector*: The conventional soft detector developed by Yang and Vaishampayan [84] uses the algorithm described by (6.5).
- *ERMDC decoder*: The soft input ERMDC decoder proposed in this section is utilized.
- *ERMDC BER decoder*: The ERMDC decoder realized by calculating $P_e(\hat{j}|j)$ via BERs. The BER p is calculated approximately by $p = Q\left(\sqrt{\frac{2E_s}{W_0}}\right)$.

Experimental results over AWGN channels are shown in Fig. 6.2. The conventional soft detector and ERMDC decoder proposed here are applied with MDSQ and ERMDC encoders, respectively. The ERMDC encoder-decoder pair outperforms other encoder-decoder pairs up to 7 dB. As a result of cooperating with the ERMDC encoder, the soft input ERMDC decoder achieves improvement of up to 2 dB compared with the soft detector, and up to 5 dB compared with the ERMDC BER decoder. The reason of the improvement brought by the ERMDC encoder is that any one and three-bit errors are detected. With the help of the ERMDC encoder, soft decoders reduce the distortion caused by detectable errors. Furthermore, the soft input ERMDC decoder decreases more effectively the distortion caused by detectable errors. It should be emphasized that the improvement is achieved without extra redundancy. In addition, SDRs at SNR $\geq$ 0 dB are higher than side distortions, because bit errors in one description are assumed to be corrected by applied FEC codes.

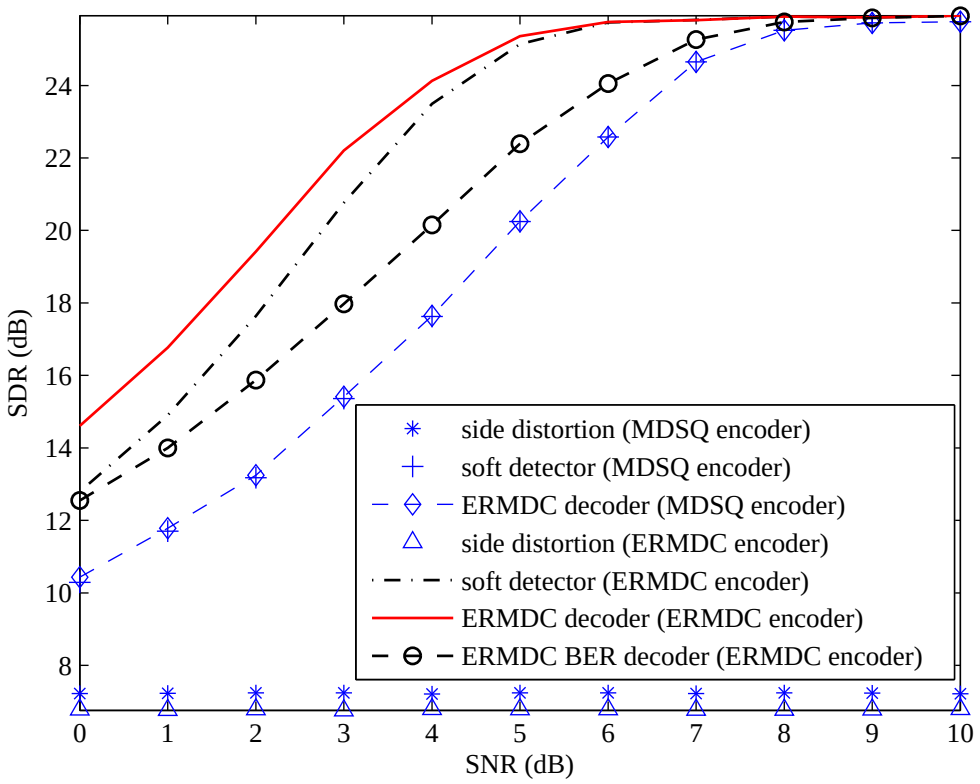


Fig. 6.2 Performance comparison of various MD encoder-decoder pairs over AWGN channels.

Figure 6.3 shows results over slow Rayleigh fading channels. Similar to the performance shown in Fig. 6.2, with the aide of FEC codes, the achieved performance is higher than side distortions. Even though due to channel fading, the performance improvement is not as significant as that with AGWN channels, the soft input ERMDC encoder-decoder pair still outperforms other encoder-decoder pairs up to 2 dB. It indicates that the soft input ERMDC encoder-decoder pair provides not only graceful performance degradation, but also more robustness against noisy wireless channels. As we expect, based on soft receiver inputs, the estimates can be made more precisely than that based on BER. Thus, the soft input ERMDC decoder always provides better performance than the ERMDC BER decoder.

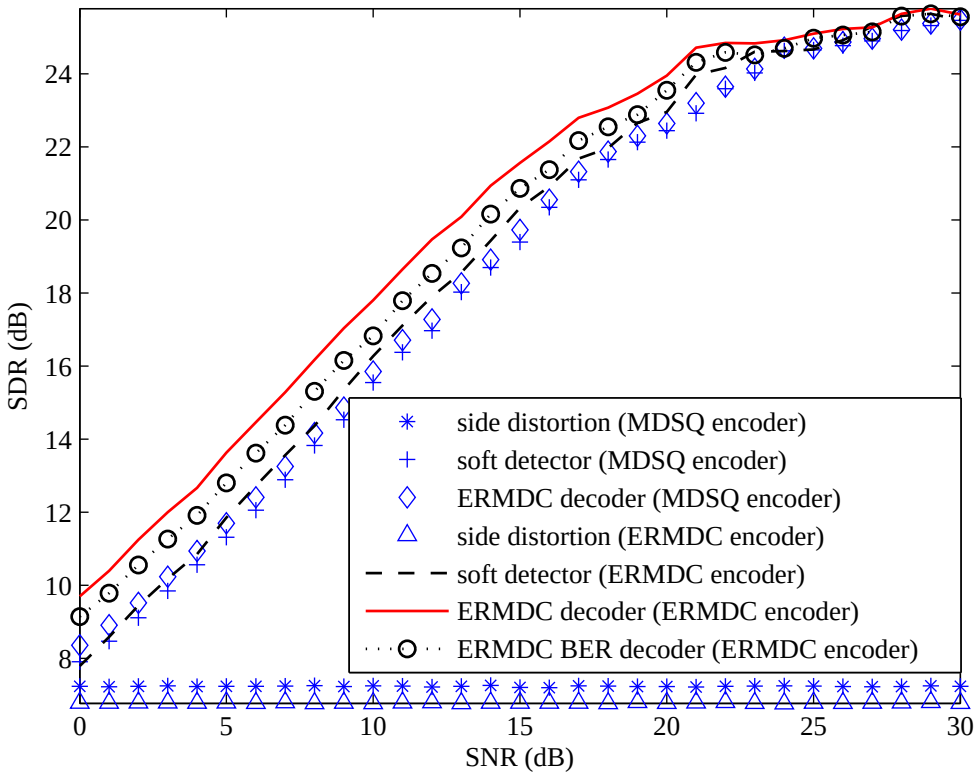


Fig. 6.3 Performance comparison of various MD encoder-decoder pairs over slow Rayleigh fading channels.

6.1.3 Conclusion

The ERMDC is extended for AWGN and slow Rayleigh fading channels. With soft receiver inputs, reproduction levels are estimated by using the proposed algorithms to decrease the distortion introduced by detectable transmission errors. Experimental results show that this novel ERMDC encoder-decoder pair achieves more robustness against noisy and slow fading channels, and significantly outperforms the existing algorithms.

6.2 Joint MD-FEC Coding

In this section, we attempt to combine MDC and FEC techniques to provide protection from packet losses and bit errors. Hybrid wireline and wireless networks are utilized as a typical environment in the presence of packet losses and bit errors to test the performance

of the proposed algorithm.

After a brief introduction of existing packet recovery techniques in Section 6.2.1, an ERMDC system with joint MD-FEC coding is proposed in Section 6.2.2. A three-state Markov chain is proposed to model the environment in the presence of packet losses and bit errors in Section 6.2.3, followed by experimental results in Section 6.2.4. A short conclusion is provided in Section 6.2.5.

6.2.1 Introduction

Nowadays, wireline and wireless networks are converging to form hybrid networks. It has become an important trend in the telecommunications industry [1]. Packet losses and bit errors are two important reasons that cause high reconstruction distortions, or even transmission failure at the receiver end. When data is transmitted from a high-speed wireline network to a low-speed wireless network, packets are usually dropped in order to alleviate traffic congestion. Due to the vulnerability of wireless channels, wireless networks usually suffer from bit errors and channel failure. As a result, end-to-end transmissions through hybrid networks are often affected adversely by not only packet losses but also bit errors. Meanwhile, as important sources of revenues for service providers, delay-sensitive services, such as voice, audio and video communications, play very important roles in current and future networks. Therefore, reliable and low-delay transmission techniques are required by these services to protect information from bit errors and packet losses.

Reliable delivery mechanisms for delay-sensitive services have been widely studied to combat packet losses and bit errors. Forward error correction (FEC) has been widely applied to protect information from bit errors. Packet-loss recovery techniques are usually classified into two categories: (i) sender-based and (ii) receiver-based [4]. Typical sender-based techniques are: automatic repeat request (ARQ), FEC, multiple description coding (MDC) and interleaving. Because the delay is of critical importance in delay-sensitive services, FEC and MDC are applied at the cost of some redundancy and short delays. When sender-based mechanisms fail, receiver-based error concealment schemes, such as insertion, interpolation, and regeneration, produce replacements of lost packets.

As a means of joint source-channel coding, MDC was originally proposed to protect from channel failure [16], [41]. The performance of the MDSQ was studied by Yang and Vaishampayan in low-delay communications over a Rayleigh fading channel, and compared with a maximum ratio combining system and FEC-based systems [84]. It showed that

with the same interleaving delay, the MDSQ-based system outperformed the other two systems. Furthermore, Kim and Kleijn compared rate-distortion performance over one and two channels between FEC and MDC-based packet-loss recovery on Gilbert channel model [91]. Experimental results indicated that FEC-based techniques were robust against single-packet loss; on the other hand, MDC-based techniques were more robust against burst-packet loss.

However, in the existing work, the performance of MDC-based techniques is only investigated in the presence of either packet losses or noisy channels, but not both. Since the ERMDC achieves high robustness against both packet losses and bit errors with the help of FEC, in this section, we propose an enhanced ERMDC system, in which MDC and FEC are utilized jointly to protect from both packet losses and bit errors. FEC provides protection against bit errors, and MDC provides protection against packet losses. In addition, the central decoder is refined to deal with the situation, in which both received descriptions carry bit errors after decoding FEC. We investigate the end-to-end performance of the proposed ERMDC system over hybrid networks. Experimental results show that the joint application of MDC and FEC-based techniques outperforms each individual.

Furthermore, due to characteristics of hybrid networks, namely, the combination of packet losses and bit errors, the traditional testbed, such as the two-state Gilbert channel model [92], is not suitable. Therefore, we propose a novel three-state Markov chain channel model. In the proposed channel model, there are three states:

(i) packets are received correctly, or

(ii) packets are received with bit errors, or

(iii) packets are lost.

In experiments, given a packet error rate (PER) and a packet loss rate (PLR), we investigate distortions of the enhanced ERMDC system associated with different BERs at the bit-error state. For purposes of comparison, performance of traditional MDC and FEC-based methods is also provided. To compare fairly, source samples are quantized with the same number of levels, and protected by similar redundancy; nevertheless, processing delays are different.

6.2.2 Enhanced ERMDC System with Joint MD-FEC Coding

Due to high efficiency in protection from bit errors, FEC codes are utilized to achieve an enhanced ERMDC system with joint MD-FEC coding. At the receiver end, received descriptions are checked, and bit errors are possibly corrected based on the applied FEC codes. Based on results of FEC decoding, the ERMDC system reduces the reconstruction distortion by estimating outputs of received index pairs with errors.

Generally speaking, at the transmitter end, different types of FEC codes are added according to characteristics of the network. If congestion is the dominant reason of packet losses, and channel conditions are not severe, cyclic redundancy check (CRC) is added to check the correctness of received packets. Otherwise, FEC codes with low redundancy should be added to provide basic protection from bit errors.

At the receiver end, there exist six possible situations when two descriptions are transmitted. When both descriptions are received, the refined central decoder, which will be described in details later, is applied. When only one description is received and correct, side decoders are used. Otherwise, the transmission fails. All possible situations and associated solutions are listed in Table 6.1.

The Refined Central Decoder

As mentioned in previous chapters, at the receiver end, a central decoder is used when both descriptions are received and the applied FEC codes are decoded. The dependency between i and j, i.e., (5.17), is checked to determine if every index pair $(\hat{i}, \hat{j})$ is received correctly.

In the case that both descriptions are received correctly, the output level $\hat{x}_0 = g_0(\hat{i}, \hat{j})$. If only one description is correct, but the other one carries bit errors, each index pair (i, j) is checked in terms of (5.17). If (5.17) is satisfied, the output level $\hat{x}_0 = g_0(\hat{i}, \hat{j})$. Otherwise, the output level $\hat{x}_0$ is estimated according to the suboptimal method provided in Chapter 3, specifically, side codebooks.

However, as for the case that both received descriptions still carry bit errors after decoding applied FEC codes, in the previous chapters, both descriptions were dropped, and the transmission is considered as a failure. Here, we propose a simple method that does not require knowledge of channel conditions. By checking (5.17), index pairs are discriminated whether they carry bit errors or not. Outputs of correct index pairs are

Table 6.1 Possible receiving cases and their solutions in an ERMDC system with joint MD-FEC coding.

No.	**Receiving cases**	**Solutions**
1	Two descriptions are received correctly.	**Central decoder**: *The output level* $\hat{x}_0 = g_0(\hat{i}, \hat{j})$.
2	One description is received correctly, but the other one carries bit errors.	**Central decoder**: *If* (5.17) *is satisfied,* $\hat{x}_0 = g_0(\hat{i}, \hat{j})$*; otherwise,* $\hat{x}_0$ *is estimated according to the associated side codebook.*
3	Both descriptions carry bit errors.	**Central decoder**: *If* (5.17) *is satisfied,* $\hat{x}_0 = g_0(\hat{i}, \hat{j})$*; otherwise, estimated by the mean of the source* μ.
4	One description is received correctly, the other one is lost.	*Side decoder is applied.*
5	One description carries bit errors, the other one is lost.	*The transmission fails.*
6	Both descriptions are lost.	*The transmission fails.*

obtained by $\hat{x}_0 = g_0(\hat{i}, \hat{j})$, and those of index pairs with bit errors are estimated by the mean of the source μ. Therefore, if the BER is not too high, an improvement may be achieved.

6.2.3 Channel Model in the presence of Packet Losses and Bit Errors

For channels with memory, the discrete-time, finite-state Markov models are commonly used. As a two-state Markov model, the Gilbert channel model has been widely exploited to simulate scenarios with either bit errors or packet losses. In the scenario with bit errors, the two states are a good error-free state, and a bad state with a BER. In the scenario with lost packets, the two states are a state where a packet is received, and a state where a packet is lost.

It is obvious that the classical Gilbert model is not suitable to simulate the situation where both packet losses and bit errors should be taken into account simultaneously, such as the hybrid wireline and wireless network. Therefore, we propose a novel three-state Markov model as illustrated in Fig. 6.4. In the proposed model, there are three states at

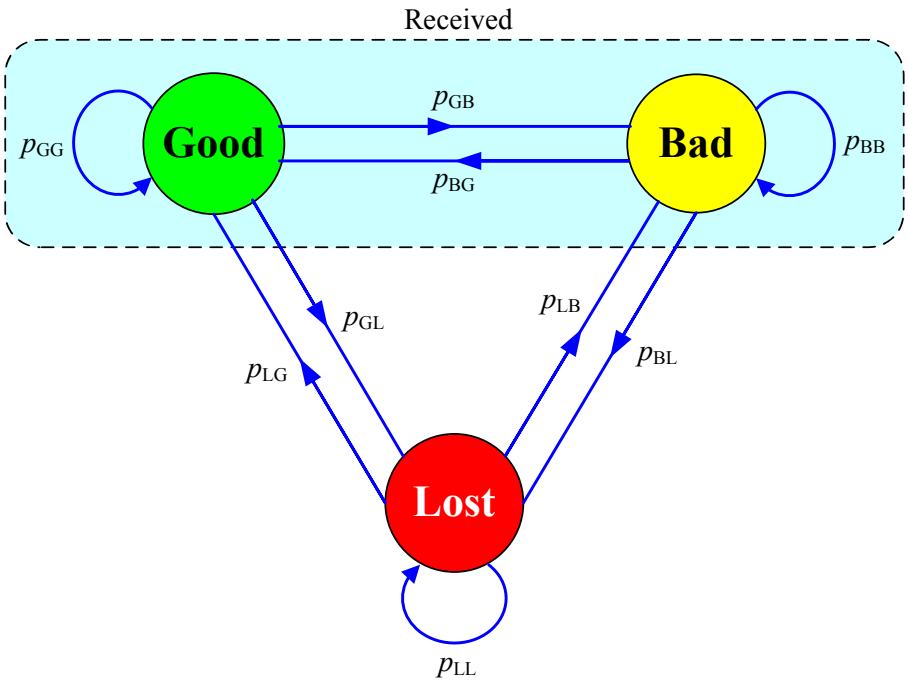


Fig. 6.4 Channel model based on a three-state Markov chain.

the receiver end:

(i) "***Good***": The transmitted packet is received without error.

(ii) "***Bad***": The transmitted packet is received with bit errors at a certain BER, and

(iii) "***Lost***": The transmitted packet is lost.

Transition probabilities between any two states are given in a transition probability matrix T. The transition probability matrix $\mathbf{T}$ is given by

$$\mathbf{T} = \begin{array}{c} \\ Good \\ Bad \\ Lost \end{array} \begin{array}{c} \begin{array}{ccc} Good & Bad & Lost \end{array} \\ \begin{pmatrix} p_{GG} & p_{GB} & p_{GL} \\ p_{BG} & p_{BB} & p_{BL} \\ p_{LG} & p_{LB} & p_{LL} \end{pmatrix} \end{array} \tag{6.13}$$

Assume communications start only when channel conditions are good, that is to say, the initial state distribution is $\Pi_0 = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}$. And the steady-state distribution is

$$\Pi_{ss} = \begin{pmatrix} \boldsymbol{p}_G & \boldsymbol{p}_B & \boldsymbol{p}_L \end{pmatrix},$$

where $\boldsymbol{p}_G$, $\boldsymbol{p}_B$ and $\boldsymbol{p}_L$ are the probabilities of the three states "good", "bad", and "lost". $\boldsymbol{p}_L$ is actually the PLR.

At the transmitter, as a typical FEC code, assume that Reed-Solomon codes RS(N, K) are applied to each packet to protect from bit errors. RS(N, K) codes are defined over Galois field $GF(2^{R_f})$, thus, $N = 2^{R_f} - 1$. Let p_b be the BER at the "Bad" state. Therefore, for each R_f-bit symbol, the symbol error probability is

$$p_s = 1 - (1 - p_b)^{R_f} \ . \tag{6.14}$$

We assume that one packet contains only one RS codeword. Consequently, the RS codeword error probability p_e after RS decoding at the receiver end is given by

$$p_e = \sum_{t=N-K+1}^{N} \binom{N}{t} p_s^t \ (1 - p_s)^{N-t}. \tag{6.15}$$

Taking into account the distribution of the "Bad" state, the actual PER p_E after decoding the RS codes is

$$p_E = p_e \cdot p_B \ . \tag{6.16}$$

In the following experiments, given a transition probability matrix $\mathbf{T}$, we are going to investigate performance of various techniques in the rate-distortion sense, associated with various BERs at the "Bad" state.

6.2.4 Experimental Results

Redundancy is exploited to protect information from packet losses and bit errors. According to the targets of its protection, redundancy or protection can be basically classified into two categories:

(i) packet level, which provides the capability of packet-loss recovery; and

(ii) bit level, which provides the capability of detecting and recovering bit errors.

Let $R = \frac{k}{n}$ be the code rate, where k input bits yield n output bits at a certain encoding stage. Let R_p and R_b be code rates at packet level and bit level, respectively. Thus, the total code rate is $R_t = R_p \cdot R_b$.

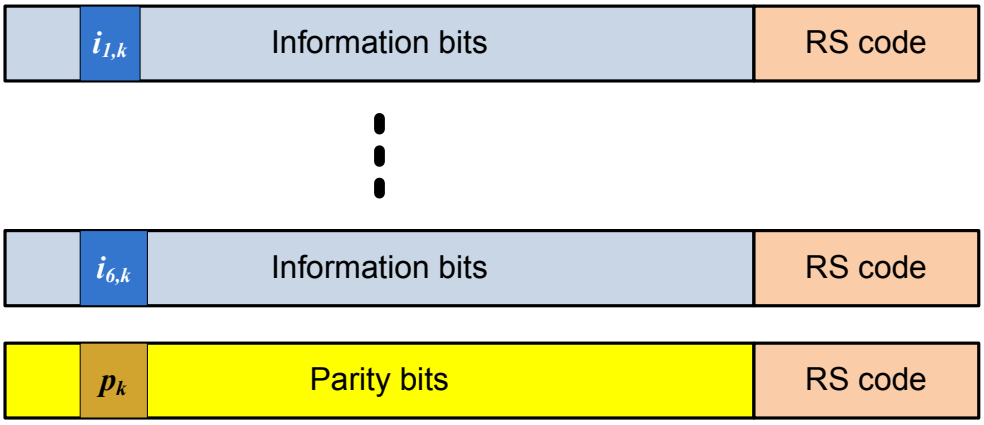


Fig. 6.5 Generation of parity packets: The k-th bit p_k of the parity packet is generated by $p_k \oplus \sum_{h=1}^{6} i_{h,k} = 0$, where $i_{h,k}$ is the k-th bit in the h-th information packet.

An example is illustrated in Fig. 6.5. Each six information packets are protected by a parity packet. The parity packet is generated by exclusive OR (XOR) of the six information packets. These six information packets and the parity packet constitute a packet group. In each packet, RS(N, K) codes are applied to protect from bit errors. If one packet in a packet group is lost or cannot be corrected by the applied RS code, this packet can be recovered. Therefore, the bit-level code rate is $R_b = \frac{K}{N}$, the packet-level code rate is $R_p = \frac{1}{6}$, and the total code rate is $R_t = \frac{K}{6N}$.

In experiments, we investigate various combinations of bit-error and packet-loss recovery techniques:

- *SD-RS*: Single description protected by RS codes.
- *SD-FEC*: Single description protected by RS codes and parity packets.
- *MDSQ-1*: Two descriptions are encoded by using the classical MDSQ encoder-decoder pair (referred to Section 2.3) and protected by RS codes.
- *MDSQ-2*: Two descriptions are generated by the ERMDC encoder provided in Chapter 5 and decoded by the MDSQ decoder. These descriptions are protected by RS codes. That is to say, after decoding RS codes, any received descriptions with bit errors are treated as being lost, and estimated by μ.

- *ERMDC-1*: Two descriptions are generated by ERMDC encoder provided in Chapter 5 and decoded by using the ERMDC decoder proposed in Chapter 3. Descriptions are protected by RS codes. Specifically, when both received descriptions carry bit errors, the outputs are estimated by the mean of the source μ.

- *ERMDC-2*: This is the enhanced ERMDC system proposed in this section. That is to say, two descriptions are generated by ERMDC encoder provided in Chapter 5 and decoded by using the ERMDC decoder refined in Section 6.2.2. Descriptions are protected by RS codes.

In order to compare them fairly in the sense of rate distortion, similar redundancy is added, that is to say, total code rates of these techniques are similar.

Experimental setting

Independent Gaussian random variables with zero mean and unit variance are used as the source. They are quantized by a uniform scalar quantizer. Then, packet-level redundancy is added by using FEC or MDC, and followed by bit-level protection provided by RS codes. Thereafter, each description is encapsulated into a series of packets, and transmitted over one channel, which is simulated by the proposed channel model with a given $\mathbf{T}$ and a BER. At the receiver, bit errors are detected and possibly recovered by the applied RS codes. Packet-level redundancy is utilized to recover lost packets. If bit errors or lost packets are unable to be recovered, $\mu = 0$ is used as the output.

The detailed settings of various techniques are shown in Table 6.2. Each source sample is uniformly quantized as widely applied in image and video compression, and, then, represented in seven bits. The quantization distortion is -32.83 dB.

In the SD-RS and SD-FEC, packets are only transmitted over one channel. Each packet consists of two RS codes as indicated in Table 6.2. That is to say, each packet consists of $\lceil 7 \times 127 \times 2/8 \rceil = 223$ bytes. As illustrated in Fig. 6.5, in the SD-FEC, each six information packets are protected by a parity packet, and RS(N, K) codes are applied in every packet.

For MDC-based techniques, each source sample is decomposed into two descriptions with 4 bpss each. Each description is split into 255-byte packets. Two descriptions are transmitted over two independent channels. In the MDSQ-1, since it is hard to find a proper index assignment scheme with 128 levels, we choose the scheme with 124 levels. In the ERMDC, one parity bit is added in each seven information bits.

Table 6.2 Experimental settings of various techniques.

Schemes	SDSQ levels	Channels	R_p (%)	RS code	R_b (%)	R_t (%)
SD-RS	128	1	1	(255, 215)	84.31	84.31
SD-FEC	128	1	85.71	(255, 251)	98.43	84.37
MDSQ-1	124	2	84.77	(255, 247)	96.86	82.11
MDSQ-2, ERMDC-1, 2	128	2	87.5	(255, 247)	96.86	84.75

Various techniques have different processing delays. The SD-RS processes each packet as soon as a packet is received. MDC-based techniques need to wait for receiving two packets from two channels before decoding. Hence, the SD-RS and MDC-based techniques have similar delays. Nevertheless, the SD-FEC requires the longest processing delay, namely, seven packets here.

Performance Comparison

First, we try the situation where bit errors and packet losses are equally important reasons in performance degradation. Experimental results associated with various BERs at the "Bad" state are shown in Figs. 6.6 and 6.7. The transition probability matrix $\mathbf{T}_1$ is given by

$$\mathbf{T}_1 = \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.8 & 0.1 & 0.1 \\ 0.8 & 0.1 & 0.1 \end{pmatrix}.$$

Therefore, $\Pi_{ss} = \begin{pmatrix} 0.8 & 0.1 & 0.1 \end{pmatrix}$. Specifically, p_B and p_L are 10%, respectively.

In Fig. 6.6, MDC-based techniques are compared with respect to various BERs. It shows that ERMDC-2 achieves lower distortion than ERMDC-1 until a high BER of 6%. At higher BERs, distortions achieved by ERMDC-2 are higher than those of ERMDC-1, because (5.17) cannot detect high proportion of bit errors, specifically, even number of bit errors in an MD index pair. In addition, it shows that MDSQ-1 achieves much higher distortion than ERMDC-2. The reason is that the average side distortion d_s of MDSQ-1, which is -3.4 dB, is higher than that of ERMDC-2, which is -7.05 dB. Thus, in order to compare them fairly, we use the performance of the ERMDC encoder provided in Chapter 5 combined with the MDSQ decoder, called *MDSQ-2* here, to represent the performance of

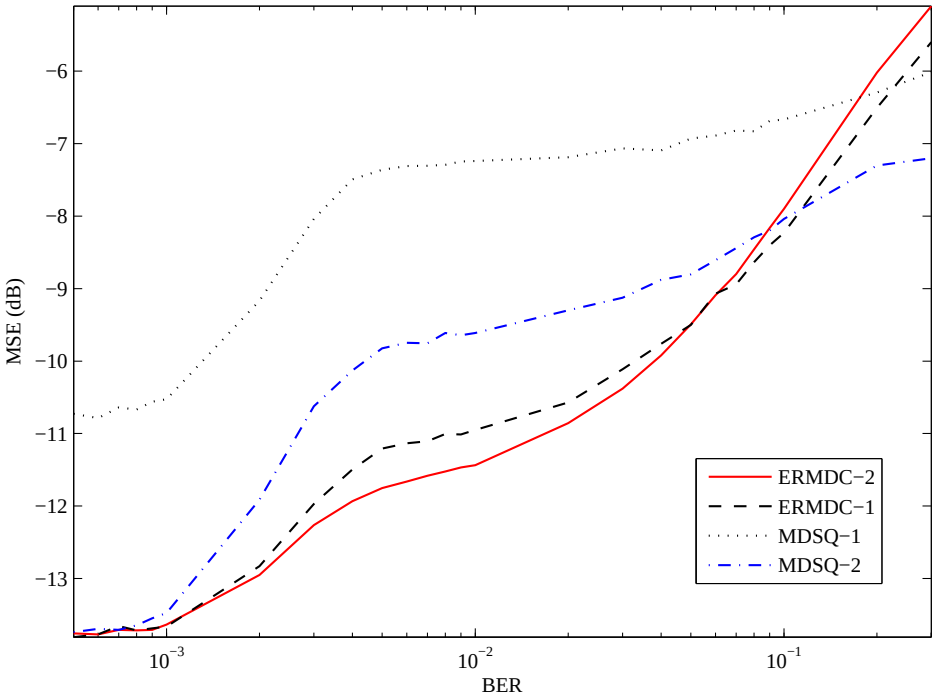


Fig. 6.6 Performance comparison of MDC-based techniques with respect to various BERs. Each channel is constrained by the transition probability matrix $\mathbf{T}_1$.

the traditional MDSQ. That is to say, after decoding RS codes, any received packets with bit errors are treated as being lost, and estimated by μ. For clarity, ERMDC-1 and MDSQ-1 will not be used in the following comparisons with FEC-based techniques.

In Fig. 6.7, MDC and FEC-based techniques are compared with respect to various BERs. At low BERs, distortions are mainly caused by packet losses. Therefore, techniques with packet-loss recovery, such as the SD-FEC, MDSQ-2 and ERMDC-2, are more robust against packet losses than the SD-RS, which is unable to recover any lost packet. The distortion obtained by the SD-FEC is about 1 dB higher than the ERMDC-2, because ERMDC-2 can still obtain partial information, even though only one description is received. MDSQ-2 provides similar robustness with ERMDC-2.

As the BER increases, the distortion of the SD-RS stays constant until the applied RS codes begin to fail. Starting from this point, the performance of the SD-RS degrades rapidly. Similarly, performance of MDSQ-2 degrades fast, because the MDSQ decoder does

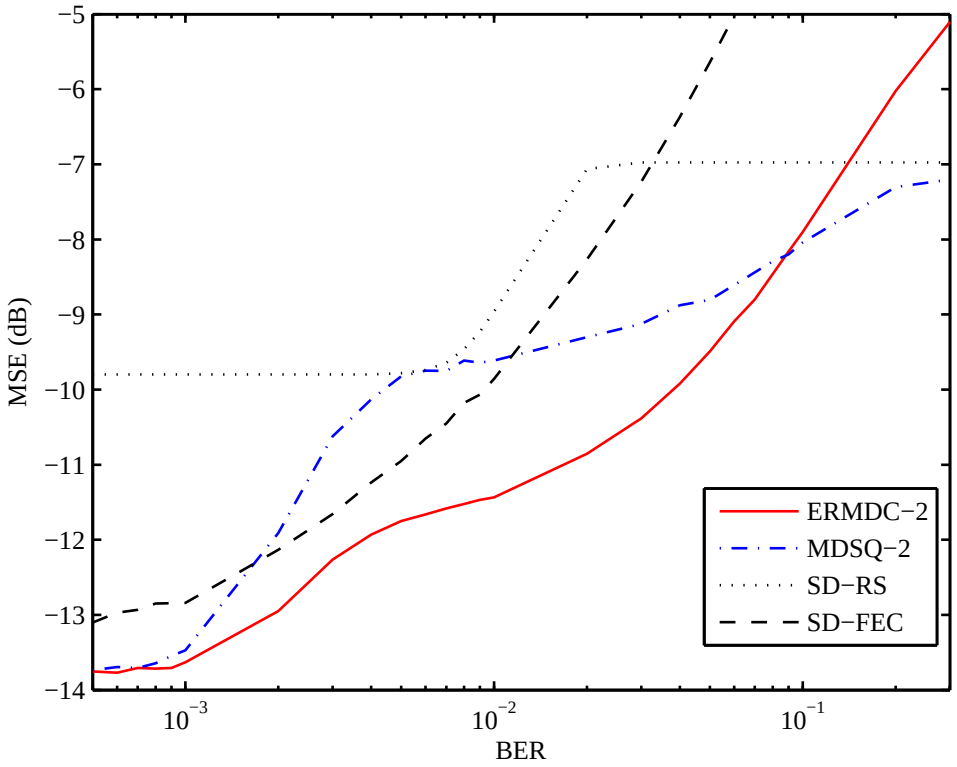


Fig. 6.7 Performance comparison of MDC and FEC-based techniques with respect to various BERs. Each channel is constrained by the transition probability matrix $\mathbf{T}_1$.

not provide the capability of recovering bit errors. In contrast, the SD-FEC and ERMDC-2 achieve graceful performance degradation. Especially, due to the capability of detecting error index pairs and estimating outputs, ERMDC-2 achieves the highest robustness against both packet losses and bit errors. At the BER 2%, the distortion of ERMDC-2 is 4 dB lower than those of the SD-RS and the SD-FEC. At the BER 4%, ERMDC-2 reaches the lowest distortion obtained by the SD-RS.

Second, we try the situation where bit errors are the dominant reason of performance degradation. Experimental results associated with various BERs is shown in Fig. 6.8. The

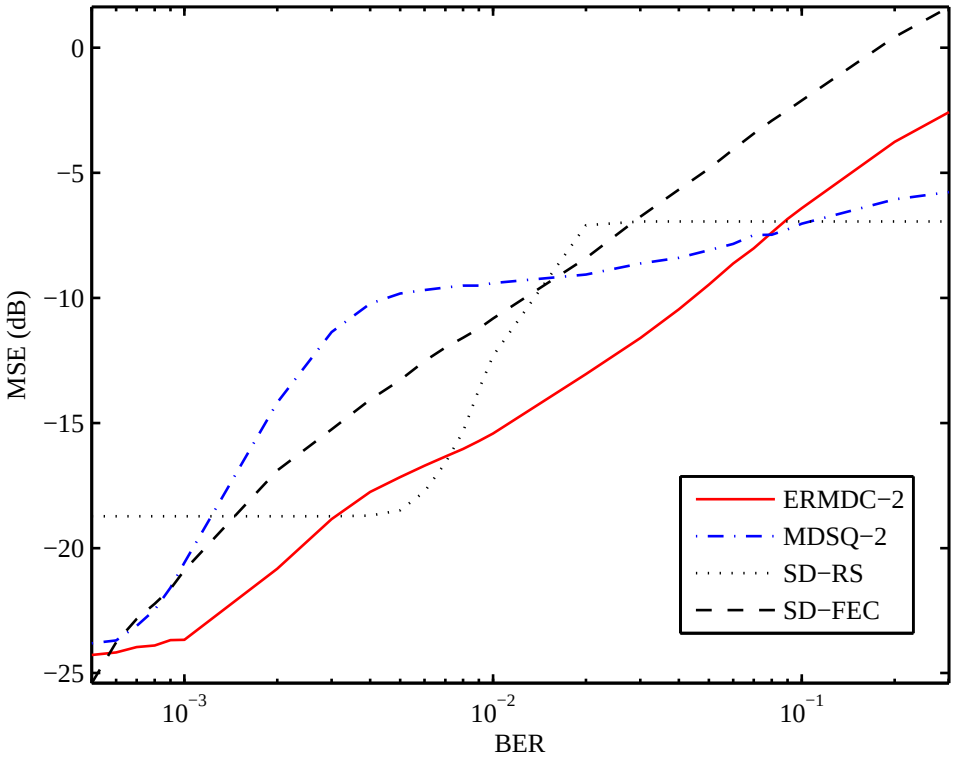


Fig. 6.8 Performance comparison with respect to various BERs. Each channel is constrained by the transition probability matrix $\mathbf{T}_2$.

transition probability matrix $\mathbf{T}_2$ is given by

$$\mathbf{T}_2 = \begin{pmatrix} 0.8 & 0.19 & 0.01 \\ 0.8 & 0.19 & 0.01 \\ 0.8 & 0.19 & 0.01 \end{pmatrix}.$$

$\Pi_{ss} = \begin{pmatrix} 0.8 & 0.19 & 0.01 \end{pmatrix}$. Specifically, $p_B = 19\%$, and $p_L = 1\%$.

Figure 6.8 indicates that when BER $< 6\times10^4$, the SD-FEC obtains the lowest distortion, because each parity packet can recover all information from one lost packet in a packet group, whereas ERMDC-2 can only recover partial information of lost packets. As the BER increases, the distortion of the SD-FEC rises fast. However, thanks to its higher bit-error resilience, the performance of ERMDC-2 deteriorates more slowly.

Third, we try the situation where packet losses are the dominant reason of performance

degradation. Experimental results associated with various BERs is shown in Fig. 6.9. The transition probability matrix $\mathbf{T}_3$ is given by

$$\mathbf{T}_3 = \begin{pmatrix} 0.8 & 0.01 & 0.19 \\ 0.8 & 0.01 & 0.19 \\ 0.8 & 0.01 & 0.19 \end{pmatrix}.$$

$\Pi_{ss} = \begin{pmatrix} 0.8 & 0.01 & 0.19 \end{pmatrix}$. Specifically, $p_B = 1\%$, and $p_L = 19\%$.

Figure 6.9 indicates that MDC-based methods significantly outperform FEC-based methods when packet losses are dominant. ERMDC-2 still achieves the highest robustness until the BER reaches 15%. When the BER is very high, error detection in ERMDC-2 may make mistakes. Specifically, because (5.17) can only detect one-bit or odd number bits of errors, even number bits of errors are unable to be detected. Therefore, higher estimation distortion is introduced at high BERs. As the result, MDSQ-2 achieves lower distortion at high BERs ($> 15\%$).

In the fourth experiment, we consider the situation where lost and erroneous packets occur in bursts. Experimental results associated with various BERs is shown in Fig. 6.10. The transition probability matrix $\mathbf{T}_4$ is given by

$$\mathbf{T}_4 = \begin{pmatrix} 0.7 & 0.25 & 0.05 \\ 0.25 & 0.7 & 0.05 \\ 0.25 & 0.05 & 0.7 \end{pmatrix}.$$

$\Pi_{ss} \approx \begin{pmatrix} 0.46 & 0.40 & 0.14 \end{pmatrix}$. Specifically, $p_B \approx 40\%$, and $p_L \approx 14\%$. Consequently, in this experiment, the average length of burst packet losses is about 3.33 packets.

Figure 6.10 shows that at low BERs, MDC-based techniques substantially outperform FEC-based techniques. SD-FEC can only recover one lost packet out of each seven consecutive packets. Hence, when packet losses occur in bursts, i.e., on average about 3.3 packets in this experiment, the performance of SD-FEC is poor. Its performance is similar to that of SD-RS at low BERs, and becomes poorer as the BER increases. The traditional MDC technique, i.e., MDSQ-2, achieves similar performance to that of the ERMDC at low BERs. However, once the number of bit errors exceeds the error correction capability of applied RS codes, its performance degrades rapidly as the BER increases, and becomes worse than those of FEC-based techniques. Similarly, the performance of ERMDC-2 deteriorates as

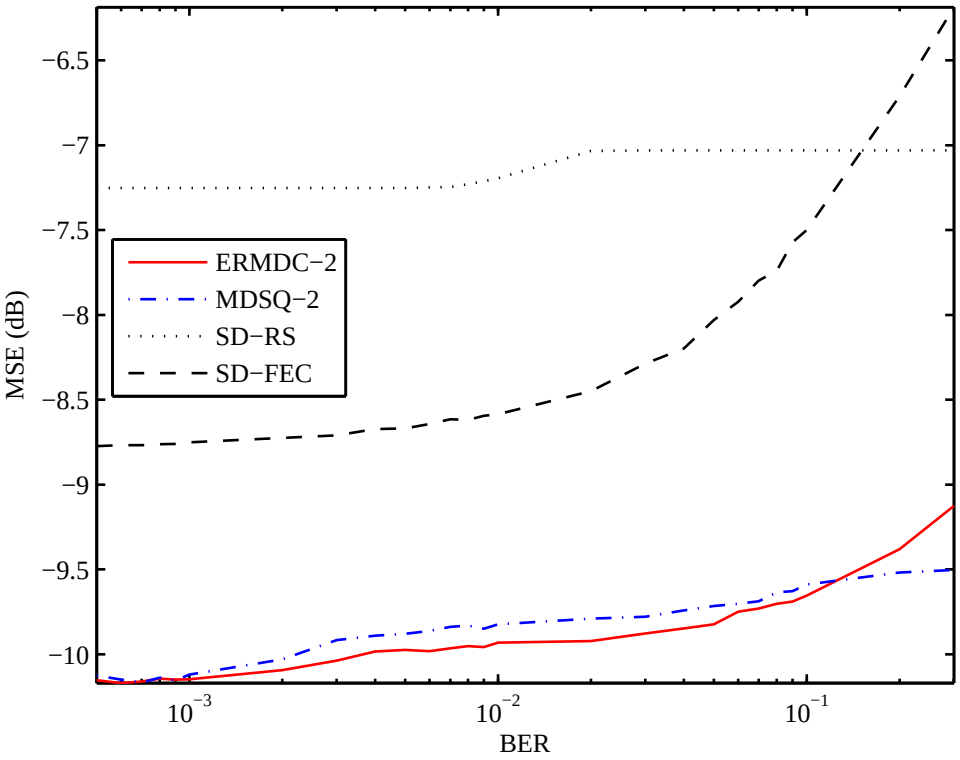


Fig. 6.9 Performance comparison with respect to various BERs. Each channel is constrained by the transition probability matrix $\mathbf{T}_3$.

the BER grows, when the number of bit errors begins to exceed the error correction capability of applied RS codes. Nevertheless, the error resilient capability of the ERMDC offsets, to some extent, the performance degradation. Thus, the performance degradation becomes slower, but continues, as the BER increases. Furthermore, since a higher density of bit errors exceeds the error resilient capability of ERMDC-2, more false error detection makes its performance worse than those of SD-RS and MDSQ-2. Consequently, ERMDC-2 achieves the lowest end-to-end distortion in a wide range of BERs, and its performance gracefully degrades as the BER increases.

6.2.5 Discussion and Conclusion

In this section, the end-to-end performance of the ERMDC system with joint MD-FEC coding against both packet losses and bit errors is investigated in the sense of rate distor-

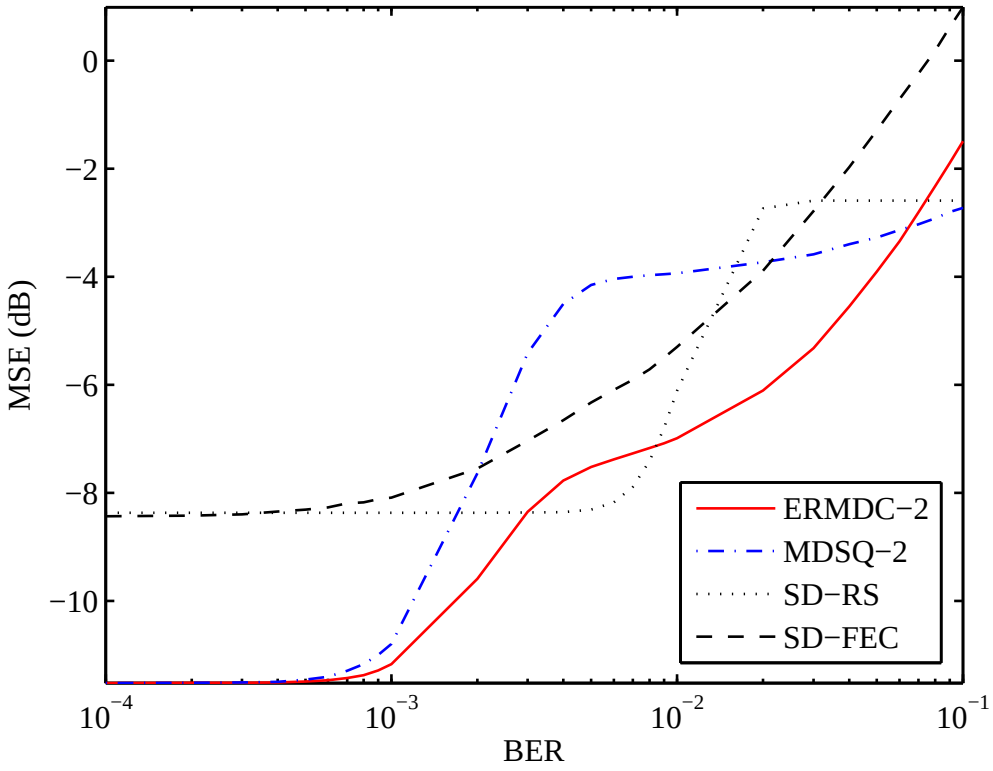


Fig. 6.10 Performance comparison of MDC-based techniques with respect to various BERs. Each channel is constrained by the transition probability matrix $\mathbf{T}_4$.

tion. Redundancy is added by means of FEC and/or MDC so as to protect information against bit errors and/or packet losses. In order to simulate the environment where packet losses and bit errors are two important reasons of performance degradation, a novel three-state Markov model is proposed. In the proposed model, three states are an error-free state, a state with bit errors, and a packet-loss state. Various techniques with capabilities of recovering bit errors and/or packet losses are compared based on the proposed channel model. Experimental results show that the ERMDC system with joint MD-FEC coding achieves the highest robustness against both packet losses and bit errors. The method of FEC-based packet-loss recovery achieves the closest behaviour; however, its processing delay is much longer. Although the single description protected by FEC codes achieves higher robustness against bit errors, it cannot alleviate adverse effects of packet loss. Therefore, it is appropriate to apply the ERMDC system on delay-sensitive communications in the

presence of both packet losses and bit errors.

6.3 Optimal MDC Redundancy

In the last section, it was showed that the joint application of MD and FEC coding outperforms either technique applied individually. Since redundancy is added in both MDC and FEC, the problem of optimally allocating redundancy between MDC and FEC arises for a given total code rate. In this section, we will only consider a simple method, specifically, received packets with remaining errors after decoding FEC codes are dropped. That is to say, for a given FEC code and a given total code rate, we are exploring how to optimize MDC redundancy with respect to the PER p_E and PLR p_L by only applying the traditional MDC decoding method.

In Section 6.3.1, after briefly describing the optimization problem, the problem is solved for such source signals as uniformly distributed inputs and arbitrarily distributed inputs in the high rate regime. Analytical and experimental results are provided in Section 6.3.2.

6.3.1 The Redundancy Optimization Problem

In the ERMDC system protected by FEC coding, redundancy in MDC and FEC is added to provide protection from packet losses and bit errors. For a given total code rate R bpss and FEC coding scheme, we are going to minimize the overall average distortion at the receiver end as a function of MDC redundancy ρ bpss associated with channel conditions, such as PLRs p_E and PERs p_L. Therefore, the source is represented in $R_0 = R - \rho$ bpss. In the following discussion, assume that RS(N, K) codes are applied, and the source is uniformly quantized at R_0 bpss.

As described in Table 6.1, six possible receiving cases exist at the receiver end. Here, we only consider a simple situation where received packets with remaining errors after decoding FEC codes are dropped in Receiving cases 2 and 3 described in Table 6.1. That is to say, we only apply the traditional MDC decoding method.

Let D_0 be the central distortion, D_s be the average side distortion, and D_l be the distortion when transmission fails. Thus, D_l equals to the variance of the source. We list probabilities and distortions of all possible receiving cases in Table 6.3.

Therefore, as a function of MDC redundancy ρ, the overall average distortion D of the

Table 6.3 Probabilities and distortions of all possible receiving cases.

Receiving case	Probability	Distortion
1. Both descriptions are received	$(1-p_L)^2$	
a) both are correct	$\times(1-p_E)^2$	D_0
b) only one is correct	$\times 2p_E(1-p_E)$	D_s
c) both are wrong	$\times p_E^2$	D_l
2. Only one description is received	$2p_L(1-p_E)$	
a) the received is correct	$\times(1-p_E)$	D_s
b) the received is wrong	$\times p_E$	D_l
3. Both descriptions are lost	p_L^2	D_l

ERMDC system at the receiver is given by

$$D(\rho) = A \cdot D_0 + B \cdot D_s + C \cdot D_l, \tag{6.17}$$

where

$$A = (1-p_L)^2 \cdot (1-p_E)^2; \tag{6.18}$$

$$B = 2(1-p_L)(1-p_E)\left[(1-p_L)p_E + p_L\right]; \tag{6.19}$$

$$C = (1-p_L)^2 p_E^2 + 2p_L(1-p_L)p_E + p_L^2. \tag{6.20}$$

Uniformly distributed inputs

When source samples $x \in [0,\ \sigma]$ are uniformly distributed. The central distortion D_0 is given by [93]

$$D_0 = \frac{\sigma^2}{12} \cdot 2^{-2R_0} = \frac{\sigma^2}{12} \cdot 2^{-2(R-\rho)}. \tag{6.21}$$

In terms of (5.46), the average side distortion D_s is given by

$$D_s = \frac{\sigma^2}{24} \cdot 2^{-2\rho}. \tag{6.22}$$

Let ρ_u^* be the minimum redundancy, i.e., $\frac{\partial D}{\partial \rho}\Big|_{\rho=\rho_u^*} = 0$. Hence, by applying (6.21) and (6.22) in (6.17),

$$\rho_u^* = \frac{R}{2} + \frac{1}{4}\log_2 \frac{(1-p_L)p_E + p_L}{(1-p_L)(1-p_E)}. \tag{6.23}$$

Since all ρ should be a non-negative integer, we round the calculation result to get the actual ρ_u^*. Various methods to get the integer value of ρ_u^* will be discussed in Section 6.3.2. In case that the result obtained by using (6.23) is negative, set $\rho_u^* = 0$. When $p_L, p_E << 1$, (6.23) is simplified to

$$\rho_u^* = \frac{R}{2} + \frac{1}{4}\log_2(p_E + p_L). \tag{6.24}$$

That is to say, the optimum redundancy ρ_u^* is a function of the sum of the PER and PLR.

Arbitrary distribution in high rate systems

For arbitrary source signals, we assume high rate systems. Based on Section 5.6.1, $D_0 = \frac{\Delta^2}{12}$, D_s is obtained by using (5.42), (5.43) and (2.22). According to (6.17), the minimum redundancy ρ^* at a given R, p_L and p_E can be obtained by exhaustively searching all possible cases, specifically,

$$\rho^* = \arg\min_{\rho} D(\rho), \tag{6.25}$$

where $\rho = 1, 2, \cdots, \frac{R}{2} - 1$.

6.3.2 Analytical and Experimental Results of the MDC System

In the following, we investigate the relationship among the redundancy ρ, the total coding rate R of the MDC system, the packet loss rate p_L, and the packet error rate p_E after decoding applied RS codes. In addition, we compare analytical and experimental results.

Uniformly distributed inputs

In this subsection, we use uniformly distributed random variables $x \in [0, 1]$ as the source. As in (6.23), since the redundancy ρ_u^* should be a non-negative integer, three methods, namely, *rounding*, *flooring* and *ceiling*, can be used to get the integer value of ρ_u^*. In Fig. 6.11, analytical results obtained by three methods are compared. Specifically, according to the optimum redundancy ρ_u^* obtained by different methods, the overall average distortions $D(\rho_u^*)$ associated with various p_L and p_E are plotted in Fig. 6.11. It shows that rounding always provides the best solution. The simplified solution provided in (6.24), called *estimated*, is also plotted in this figure. Integer values of the simplified solution are obtained by rounding. It indicates that when $p_L, p_E << 1$, the simplified solutions are the same as (6.23). Therefore, in the following, we use the rounded simplified solution as the analytical results.

Thereafter, we investigate the relation among ρ_u^* and p_L, p_E by using analytical results. $D(\rho_u^*)$, $D(\rho)$ at $R = 10$ bpss, $\rho = 1, 2, 3, 4$, associated with various p_L and p_E are plotted in Fig. 6.12. It shows that $\rho = 3$ usually provides the lowest distortion except at very low p_L and p_E, where $\rho = 2$ gives the best solution.

The relation among ρ_u^*, R and p_L ($p_L = p_E$) is shown in Fig. 6.13. Figure 6.13(a) shows that basically higher redundancy is needed for higher p_L and p_E. Figure 6.13(b) indicates that given p_L and p_E, for every two more bits in R, one bit goes to the redundancy, another one goes to the information.

Finally, we test the analytical solution in experiments. The experimental results at $R = 8$ bpss, $\rho = 1, 2, 3$ bpss are shown in Fig. 6.14(a). The optimal solutions obtained by (6.24) are marked by circles. In experiments, RS(255, 247) are applied. Every 255 bytes are encapsulated into an individual packet. The PLR $p_L = 0.01$. Since we are investigating the effects of both packet losses and RS decoding failures, we set a high BER $p_b = 0.01$ so as to achieve a high probability of RS decoding failure. Thus, after decoding the received packet, the probability that RS decoding fails is $p_e = 0.9981$. As the result, the PER $p_E = 0.9981 \times p_B \approx p_B$. In these experiments, only traditional MDC decoding method is used, that is to say, erroneous packets after decoding applied RS codes are dropped. It shows that the analytical method accurately gives the best solutions in experiments.

High rate system

In order to investigate performance under the assumption of high rate systems, we use i.i.d. Gaussian random variables with zero mean and unit variance as the source.

Figure 6.14(b) gives experimental results associated with $R = 8$ bpss, $\rho = 1, 2, 3$ bpss at $p_L = 1\%$ and $p_b = 1\%$. Optimal solutions obtained respectively according to (6.24) and (6.25) are also plotted. It shows that (6.25) can accurately find the best solution. On the contrary, (6.24) cannot always provide optimal solutions, because of the different statistics of source signals. However, as a fast and coarse estimate, (6.24) is still usable.

Next, we investigate the optimal redundancy ρ^* associated with the total coding rate R of the MDC system based on analytical results at $p_b = 1\%$. Figure 6.15(a) shows the relation at various p_L and p_E. Figure 6.15(b) shows the relation when $p_L = p_E$. It shows that at low PLRs and PERs, one-bit redundancy can provide the best protection, even though R increases up to 10 bpss; as R increases further, one of every two more bits is approximately assigned to redundancy.

6.3.3 Experimental results of the ERMDC system

In this subsection, we attempt to utilize the analytical methods developed for the MDC system in the ERMDC system. We investigate the difference between the optimal redundancy of the MDC and the ERMDC systems by experiments. Encoder-decoder pairs of MDSQ-2 and ERMDC-1, which were explained in Section 6.2.4, are used in experiments as typical systems of MDC and ERMDC respectively.

In Figs. 6.16(a) and 6.16(b), curves associated with MDSQ-2 and ERMDC-1 systems are plotted when $\rho = 1, 2, 3$ bpss, $R = 8$ bpss, $p_L = 1\%$, and $p_b = 1\%$. Figure 6.16(a) shows experimental results when using uniformly distributed signals as the source. Figure 6.16(b) shows experimental results when using Gaussian random signals with zero mean and unit variance as the source. Both figures show that the ERMDC system significantly improves the robustness against erroneous packets. For example, in Fig. 6.16(a), the crossover between the best two curves associated with $\rho = 2$ and $\rho = 3$ is moved from 0.6% to 9%; in Fig. 6.16(b), the crossover between the best two curves associated with $\rho = 1$ and $\rho = 2$ is moved from 0.6% to 2.5%. In addition, the lowest overall distortions of ERMDC systems at $p_E = 9\%$ are about 4 dB lower for both types of source signals. Furthermore, from the point view of the optimal redundancy, the ERMDC systems save up to two bits.

It is obvious that analytical methods for the MDC system cannot provide accurate optimal solutions for the ERMDC system. However, the analytical solutions for the MDC system are still able to be used as a good estimate for the ERMDC system. Specifically, at low and very high PERs, the optimal solutions of the ERMDC system are the same as those of the MDC system. Since the ERMDC system provides high robustness, in general case, a good guess is the optimal analytical solution at low PERs.

6.4 Conclusion

In this chapter, we tested the proposed ERMDC system in various application environments, such as slow Rayleigh fading channels, hybrid wireline and wireless networks. Experimental results show that the ERMDC system is easily adapted to these environments, and its performance exceeds that of existing methods. In addition, under the same circumstances, the ERMDC system uses lower bit rate to provide lower reconstruction distortions than the traditional MDC systems.

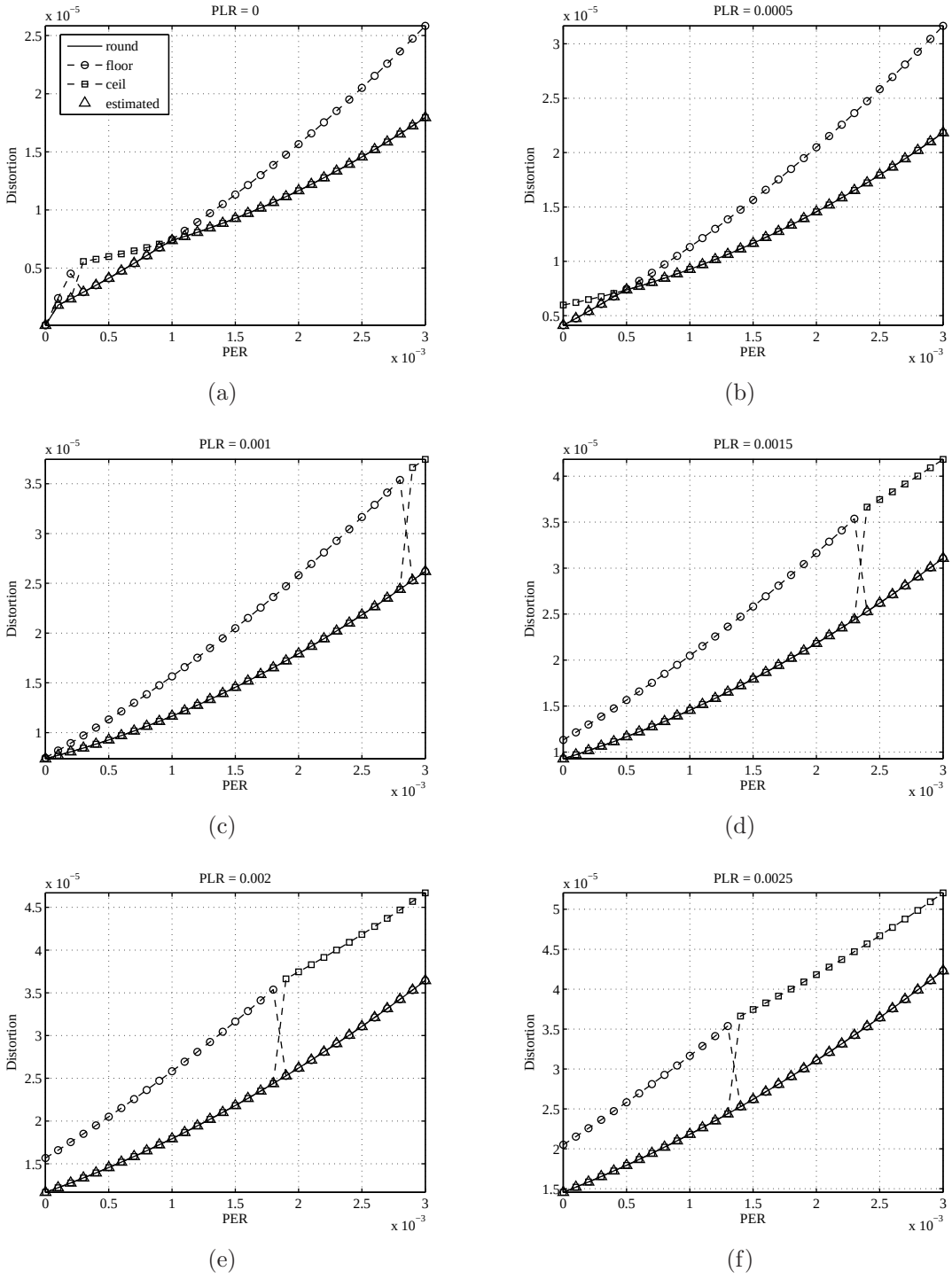


Fig. 6.11 Optimal distortion $D(\rho_u^*)$ with respect to different methods that convert the value obtained by (6.23) to an integer.

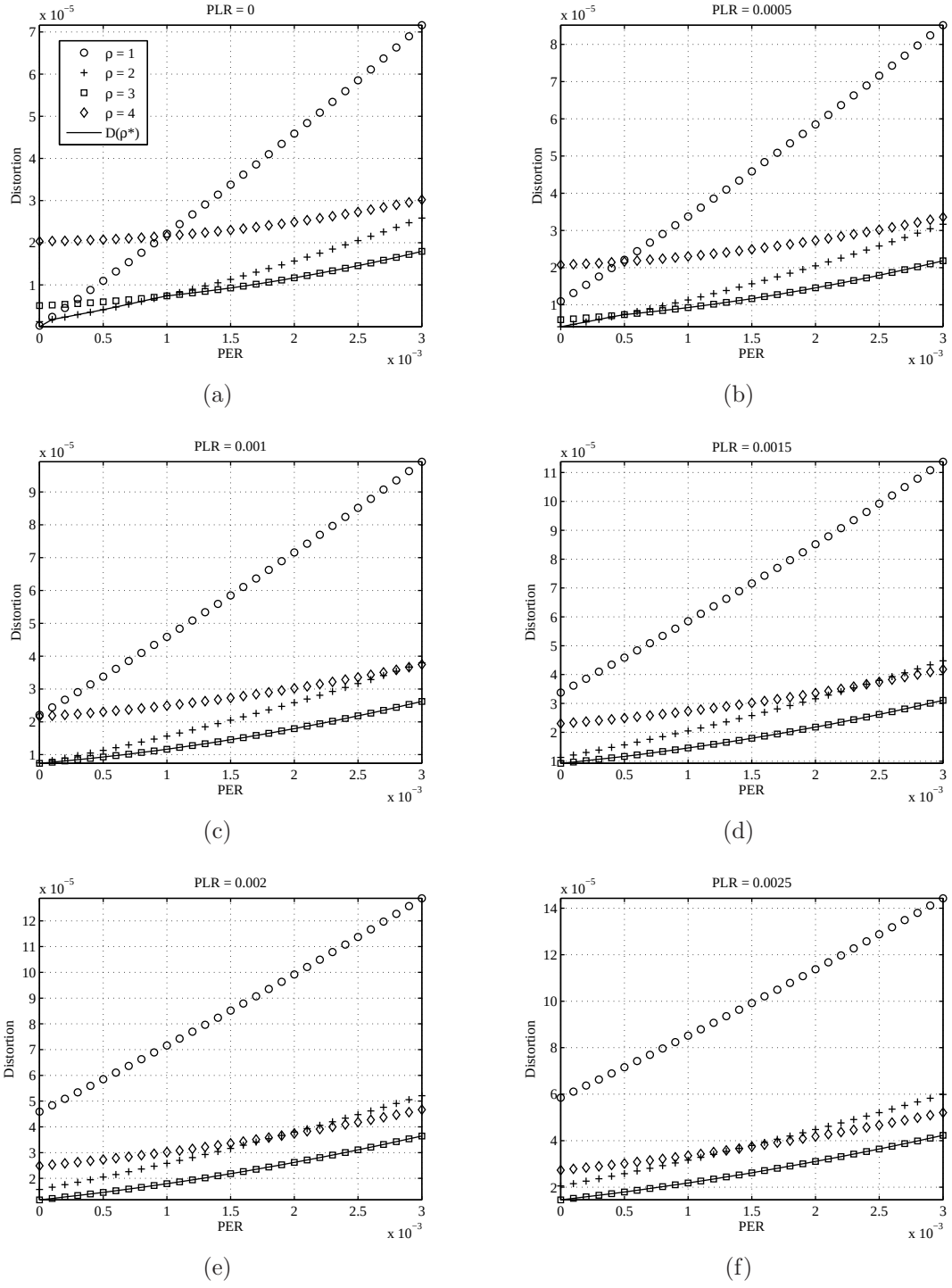
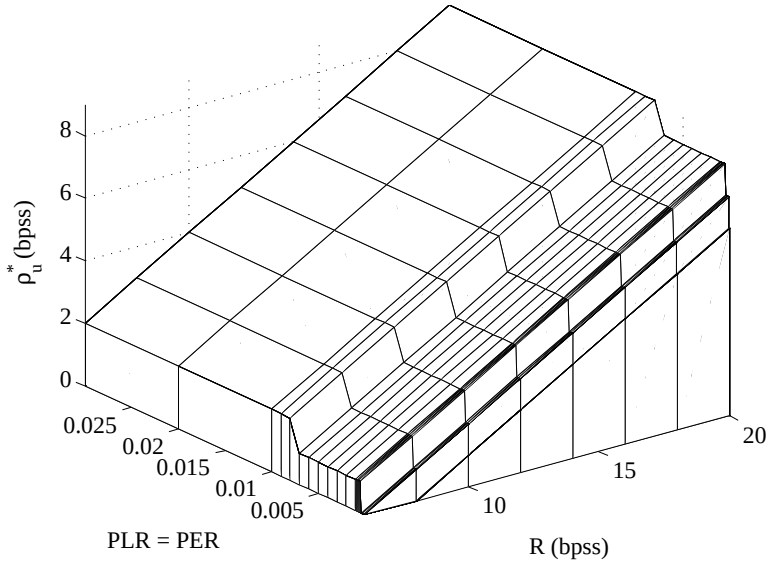
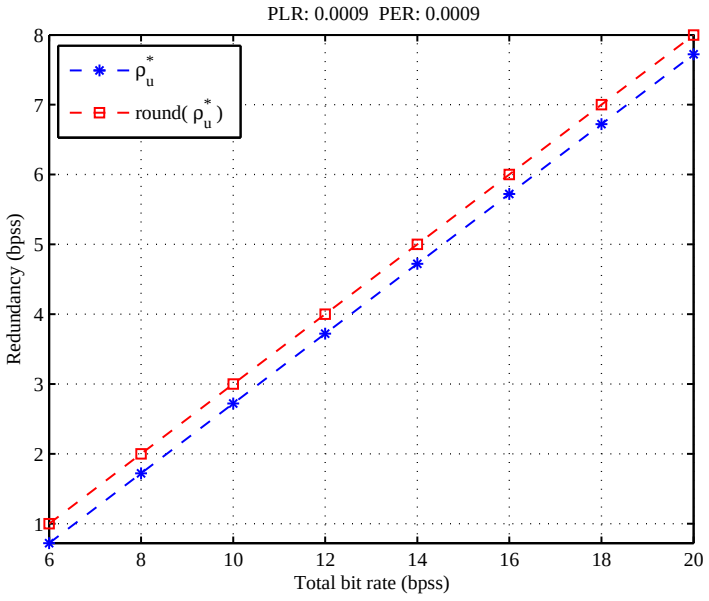


Fig. 6.12 $D(\rho)$ with respect to various ρ, p_L and p_E at $R = 10$ bpss.

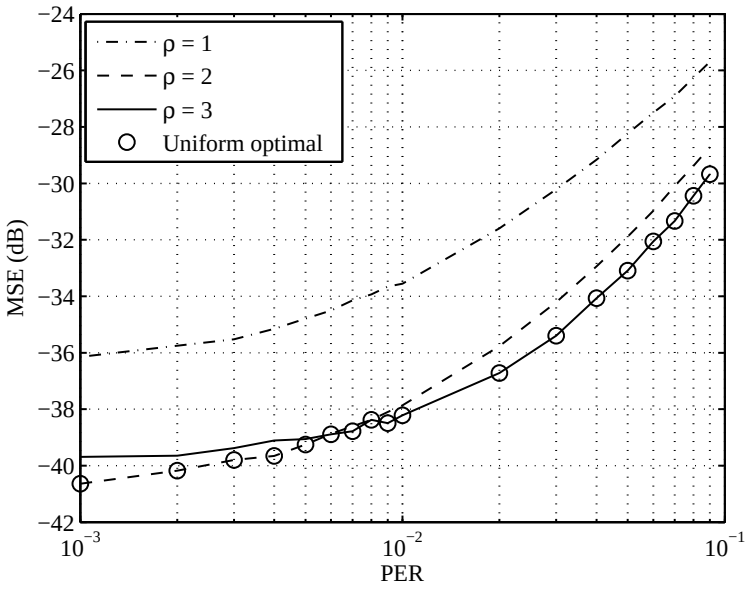


(a)

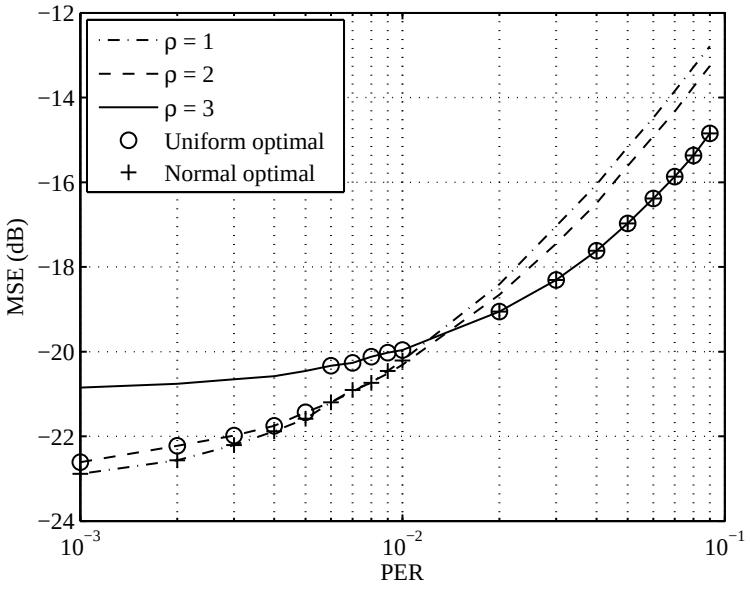


(b)

Fig. 6.13 Optimal bit rate of redundancy ρ_u^* with respect to the total bit rate R and PLR (PLR = PER): (a) at various PLR; (b) at a certain PLR.

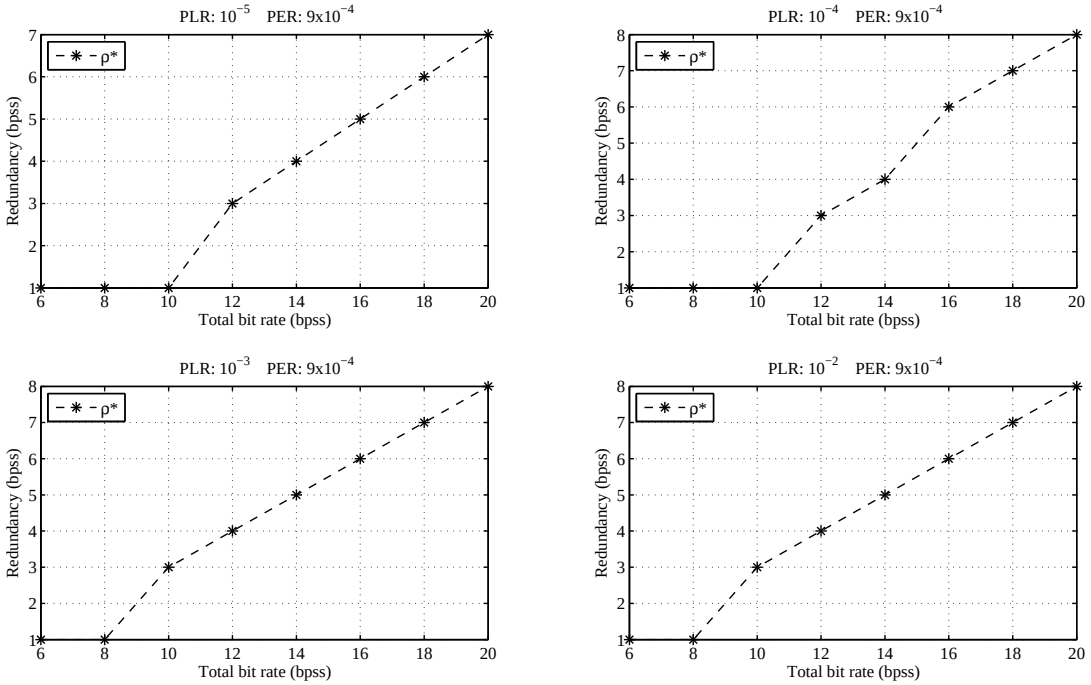


(a)

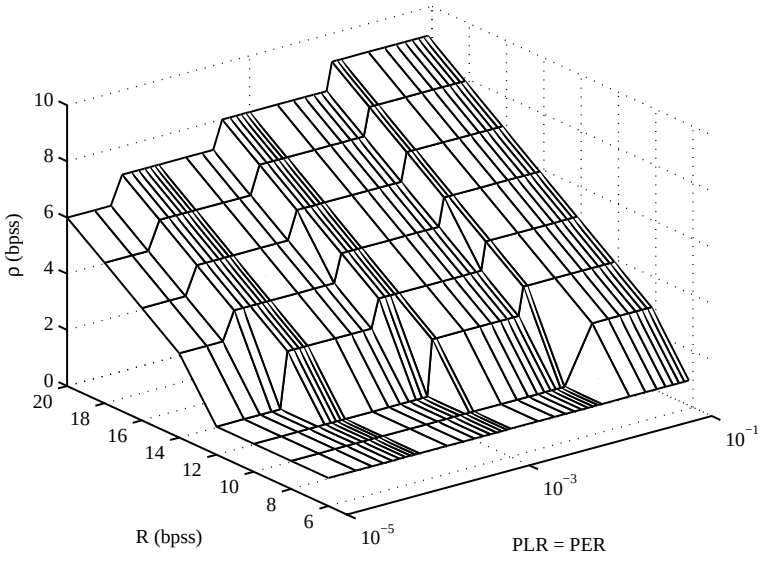


(b)

Fig. 6.14 $D(\rho)$ with respect to various ρ and p_E at $R = 8$ bpss: (a) Source samples are uniformly distributed; (b) Source samples are Gaussian random variables.

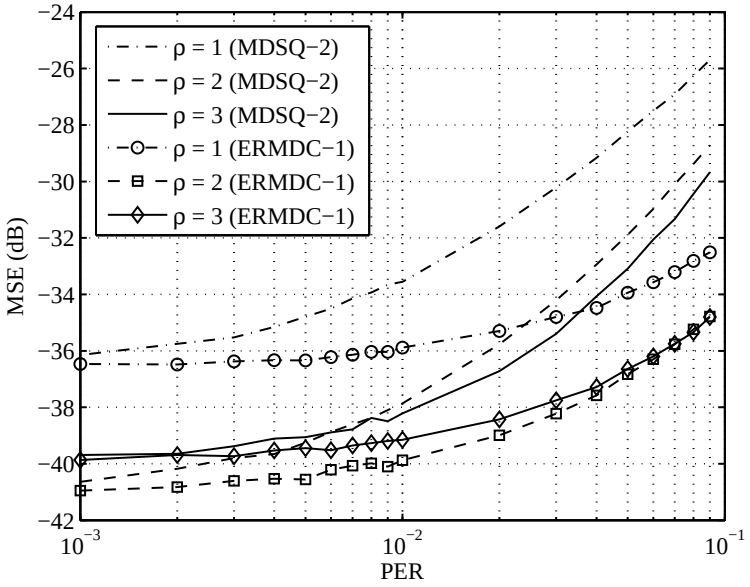


(a)

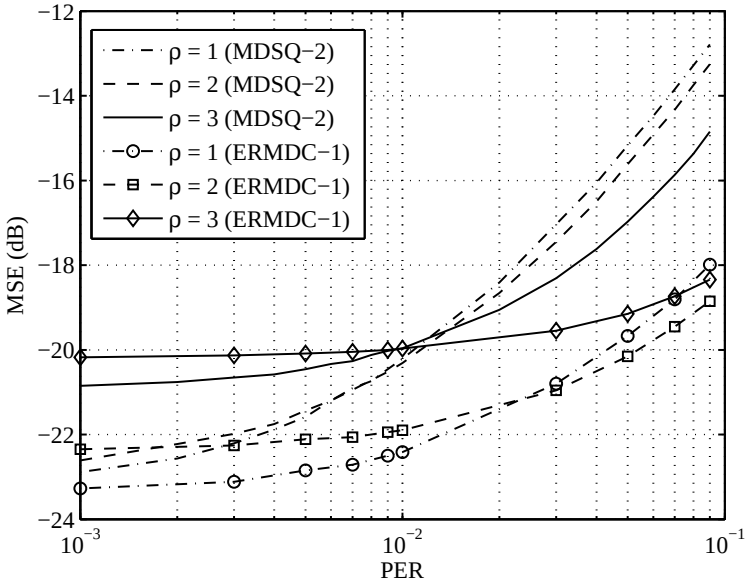


(b)

Fig. 6.15 ρ^* with respect to various p_L and p_E at $R = 8$ bpss: (a) $p_L \neq p_E$; (b) $p_L = p_E$.



(a)



(b)

Fig. 6.16 $D(\rho)$ of MDSQ-2 and ERMDC-1 systems with respect to various ρ and p_E at $R = 8$ bpss: (a) Source samples are uniformly distributed; (b) Source samples are Gaussian random variables.

Chapter 7

Conclusion and Future Work

In this work, based on techniques of noisy channel quantizers and classical MDC systems, the ERMDC system is proposed to provide protection from packet losses and bit errors. In this chapter, the main contributions of the dissertation are summarized, and future work is discussed.

7.1 Contribution

The classic MDC system was originally developed to combat channel failure. It has been extended to protect from erasure packets over packet networks, where packet loss is equivalent to channel failure. In wireless communications, even though FEC codes are regularly applied to protect from bit errors, descriptions after decoding FEC codes may still contain bit errors. In a typical MDC system, erroneous descriptions are usually dropped, although this method results in significant degradation in the sense of rate-distortion performance, because of the large performance gap between the central distortion and the side distortion.

Therefore, in order to provide high robustness against both packet losses and bit errors, we propose the ERMDC that comprises a robust MDC encoder and an enhanced MDC decoder. Three main objectives of designing the ERMDC includes:

(i) *Exploiting the inherent redundancy and dependency among multiple descriptions;*

(ii) *Minimizing the distortion introduced by bit errors;*

(iii) *Minimizing effects of losing any arbitrary description(s).*

Corresponding to the design objectives of the ERMDC, in Chapter 4, methods applied at the ERMDC encoder are as follows:

(i) Compared with FEC coding, the technique of robust quantizers that is applied does not add extra redundancy and encoding/decoding delays.

(ii) Instead of natural labeling in classic MDC, the Hamming distance between any two MD index pairs is expanded as far as possible, so that the ERMDC decoder is able to detect transmission errors more easily. Exhaustive search is applied to achieve this task.

(iii) It is preferred to achieve balanced or equally important multiple descriptions with as high fidelity as possible. Since obtaining an optimal ERMDC encoder is an NP-complete problem, the GA is exploited to find a "close-to-optimum" solution in an acceptable time.

The ERMDC decoder fulfills the design targets by following means:

(i) Unused MD index pairs are treated as transmission errors.

(ii) Output values associated with unused MD index pairs are optimally estimated based on channel conditions.

(iii) Instead of discarding erroneous descriptions, residual information in them is utilized by the means mentioned above.

Consequently, the ERMDC achieves graceful performance degradation in the presence of bit errors. With the same redundancy, the ERMDC successfully accomplishes higher robustness against bit errors than existing MDC techniques as well as similar robustness against packet losses. Furthermore, by utilizing analog received signals as soft input, the ERMDC outperforms other MDC methods over typical wireless channels, such as AWGN and slow Rayleigh fading channels.

As mentioned above, exhaustive search and the GA is applied in the ERMDC encoder to achieve the design targets. However, the computational complexity of these methods increases exponentially, as the search space grows. In Chapter 5, a generalized fast robust IA scheme is proposed for the ERMDC encoder to add arbitrary bit numbers of redundancy in a systematic way. The computational complexity of this algorithm is significantly

lower than the ERMDC encoder developed before. Compared with existing methods, the ERMDC with this algorithm accomplishes similar robustness against packet losses, and higher robustness against bit errors. Moreover, because of its high flexibility and low computational complexity, it is possible to implement "on-the-fly", and the added redundancy is easily adapted to channel conditions. Furthermore, this algorithm is modified to meet the requirement of embedded coding with low complexity at the cost of lower robustness.

In order to examine its overall performance, the ERMDC system is compared with other error resilient techniques, such as the classical MDC and FEC-based techniques, in the presence of both packet losses and bit errors. Since there is no proper model to simulate channels with both packet losses and bit errors, a three-state Markov chain is proposed. With similar redundancy, the ERMDC system with simple FEC codes provides the highest and most consistent robustness against both packet losses and bit errors.

In addition, the performance of the ERMDC is analyzed in a theoretical way. Based on analytical and numerical solutions, the redundancy is optimized for cases of discarding all erroneous packets. Experimental results show that it provides a good reference for designing an optimal ERMDC system.

7.2 Future work

The work presented in this dissertation is just the beginning of the ERMDC. There are still many possibilities to explore.

Even though in this dissertation we always use scalar quantizers, the ERMDC should be extended to vector quantizers without difficulty. Of course, it is necessary to adapt the ERMDC somehow so as to accommodate to vector quantizers. In addition, here, we only consider two descriptions. It is possible to apply the ERMDC in more than two descriptions.

When MDC and FEC are jointly applied, the ERMDC system achieves better performance against both packet losses and bit errors. We discuss a simple situation where the average distortion achieved by a traditional MDC system is minimized at a given PLR and PER. Experimental results suggest that the analytical methods for the MDC system are not suitable for the ERMDC systems. More accurate and faster methods are needed to obtain an optimal ERMDC system.

Since redundancy is added in both MDC and FEC, it is necessary to investigate how to

optimally allocate redundancy between MDC and FEC depending on channel conditions. Moreover, it is possible for FEC decoding to utilize information provided by the ERMDC so as to reduce the reconstruction distortion further. How to achieve this task is very interesting.

The MDC system is actually a diversity system. Typical diversity systems in wireless communications are multiple input/multiple output (MIMO) and orthogonal frequency division multiplexing (OFDM). An example of typical diversity systems in wireless networks is adhoc networks. It is possible to apply the ERMDC over these diversity systems so as to improve the robustness of these systems.

It is important to apply the ERMDC in transmitting multimedia signals, such as speech, images and videos. Since characteristics of various multimedia signals are quite different, the ERMDC has to be modified so as to accommodate to the specific signals.

Appendix A

Derivation of Side codebooks and side distortions

Here, we provide the derivation of side codebooks and side distortions for the RMDSQ system respectively based on source signals with uniform distribution and Gaussian distribution.

A.1 High rate systems

In this section, assuming high rate systems, we derive side codebooks and side distortions for arbitrary random signals, and Gaussian random signals. The resulting numerical solutions can be used with arbitrary IA schemes.

A.1.1 Arbitrary Random Signals

Assume that in a high rate system, the source $x \in [x_{min}, x_{max})$ with the pdf $f_X(x)$ as the source signals. Let the range of x be $\delta = x_{max} - x_{min}$. Source samples x are uniformly quantized into $L = 2^{R_0}$ levels. The quantization partition associated with l or (i, j) is given by $A_{ij} : \left[x_{ij}^L,\ x_{ij}^H\right)$. Thus, each partition interval has the length

$$\Delta = x_{ij}^H - x_{ij}^L = \frac{\delta}{L} = 2^{-R_0}\delta.$$

The side bit rate is R_s bpss. Therefore, SDSQ indices $l = 0, 1, \cdots, L-1$, and RMDSQ indices $i, j = 0, 1, \cdots, M-1$, where $M = 2^{R_s}$. The output associated with l or (i, j) is

approximately given by

$$\begin{aligned} c_{ij} &= \frac{\int_{A_{ij}} x \cdot f_X(x) dx}{\int_{A_{ij}} f_X(x) dx} \\ &\approx \frac{f_X\left(x_{ij}^L\right) \cdot \int_{A_{ij}} x dx}{f_X\left(x_{ij}^L\right) \cdot \Delta} \\ &= \frac{1}{2\Delta}\left(x_{ij}^H - x_{ij}^L\right)\left(x_{ij}^H + x_{ij}^L\right) \\ &= \frac{x_{ij}^H + x_{ij}^L}{2} \,. \end{aligned} \tag{A.1}$$

The side output associated with a certain i is given by

$$\begin{aligned} c_i^{(1)} &= \frac{\sum\limits_{j\in\mathcal{J}} \int_{A_{ij}} x \; f_X(x) \; dx}{\sum\limits_{j\in\mathcal{J}} \int_{A_{ij}} f_X(x) \; dx} \\ &\approx \frac{\sum\limits_{j\in\mathcal{J}} f_X(x) \cdot \int_{A_{ij}} x dx}{\sum\limits_{j\in\mathcal{J}} f_X(x) \cdot \int_{A_{ij}} dx} \\ &= \frac{\sum\limits_{j\in\mathcal{J}} f_X(x) \cdot c_{ij} \cdot \Delta}{\sum\limits_{j\in\mathcal{J}} f_X(x) \cdot \Delta} \\ &= \frac{\sum\limits_{j\in\mathcal{J}} c_{ij} \cdot f_X(c_{ij})}{\sum\limits_{j\in\mathcal{J}} f_X(c_{ij})} \,. \end{aligned} \tag{A.2}$$

Similarly, the side output associated with a certain j is given by

$$\begin{aligned} c_j^{(2)} &= \frac{\sum_{i\in\mathcal{I}} \int_{A_{ij}} x\ f_X(x)\ dx}{\sum_{i\in\mathcal{I}} \int_{A_{ij}} f_X(x)\ dx} \\ &\approx \frac{\sum_{i\in\mathcal{I}} c_{ij}\cdot f_X(c_{ij})}{\sum_{i\in\mathcal{I}} f_X(c_{ij})}\ . \end{aligned} \tag{A.3}$$

Therefore, the side distortion associated with a certain i is given by

$$\begin{aligned} D_i^{(1)} &= \frac{\sum_{j\in\mathcal{J}} \int_{A_{ij}} (x-c_i^{(1)})^2 f_X(x)dx}{\sum_{j\in\mathcal{J}} \int_{A_{ij}} f_X(x)dx} \\ &= \frac{\sum_{j\in\mathcal{J}} f_X(x)\cdot \int_{A_{ij}} (x-c_i^{(1)})^2 dx}{\sum_{j\in\mathcal{J}} f_X(x) \int_{A_{ij}} dx} \\ &= \frac{\sum_{j\in\mathcal{J}} f_X(x)\cdot \int_{A_{ij}} (x-c_i^{(1)})^2 dx}{\Delta\cdot \sum_{j\in\mathcal{J}} f_X(x)}\ . \end{aligned} \tag{A.4}$$

Let

$$\begin{aligned}
A &= \frac{1}{\Delta}\cdot\int_{A_{ij}}\left(x-c_i^{(1)}\right)^2 dx \\
&= \frac{1}{\Delta}\cdot\int_{A_{ij}}\left(x-c_{ij}+c_{ij}-c_i^{(1)}\right)^2 dx \\
&= \frac{1}{\Delta}\cdot\left\{\int_{A_{ij}}(x-c_{ij})^2 dx + 2\left(c_{ij}-c_i^{(1)}\right)\cdot\int_{A_{ij}}(x-c_{ij})dx + \left(c_{ij}-c_i^{(1)}\right)^2\int_{A_{ij}}dx\right\} \quad \text{(A.5)} \\
&= \frac{1}{\Delta}\cdot\left\{\int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} u^2\ du + 2\left(c_{ij}-c_i^{(1)}\right)\cdot\int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} u\ du + \left(c_{ij}-c_i^{(1)}\right)^2\cdot\Delta\right\} \\
&= \frac{\Delta^2}{12} + \left(c_{ij}-c_i^{(1)}\right)^2,
\end{aligned}$$

then

$$\begin{aligned}
D_i^{(1)} &= \frac{\sum_{j\in\mathcal{J}} f_X(x)\cdot A}{\sum_{j\in\mathcal{J}} f_X(x)} \\
&= \frac{\sum_{j\in\mathcal{J}}\left(c_{ij}-c_i^{(1)}\right)^2 f_X(c_{ij})}{\sum_{j\in\mathcal{J}} f_X(c_{ij})} + \frac{\Delta^2}{12}\ .
\end{aligned} \quad \text{(A.6)}$$

The average side distortion corresponding to all possible i is given by

$$D_1 = \frac{\sum_{i\in\mathcal{I}} D_i^{(1)} f_i}{\sum_{i\in\mathcal{I}} f_i}, \quad \text{(A.7)}$$

where $f_i = \sum_{j\in\mathcal{J}} f_X(c_{ij})$.

Similarly, the side distortion associated with a certain j is given by

$$D_j^{(2)} = \frac{\sum_{i\in\mathcal{I}} \left(c_{ij} - c_j^{(2)}\right)^2 f_X(c_{ij})}{\sum_{i\in\mathcal{I}} f_X(c_{ij})} + \frac{\Delta^2}{12} \tag{A.8}$$

The average side distortion corresponding to all possible j is given by

$$D_2 = \frac{\sum_{j\in\mathcal{J}} D_j^{(2)} f_j}{\sum_{j\in\mathcal{J}} f_j}, \tag{A.9}$$

where $f_j = \sum_{i\in\mathcal{I}} f_X(c_{ij})$.

Therefore, the overall average side distortion is obtained by using (2.22).

A.1.2 Gaussian Random Signals

Assume that x are i.i.d. Gaussian random variables with zero mean and unit variance. Then,

$$f_X(x) = \frac{1}{\sqrt{2\pi}}\, e^{-\frac{x^2}{2}}.$$

Therefore,

$$
\begin{aligned}
c_i^{(1)} &= \frac{\sum\limits_{j\in\mathcal{J}} \int_{x_{ij}^L}^{x_{ij}^H} x\ f_X(x)\ dx}{\sum\limits_{j\in\mathcal{J}} \int_{x_{ij}^L}^{x_{ij}^H} f_X(x)\ dx} \\
&= \frac{\sum\limits_{j\in\mathcal{J}} \int_{x_{ij}^L}^{x_{ij}^H} x\ \frac{1}{\sqrt{2\pi}}\ e^{-\frac{x^2}{2}}\ dx}{\sum\limits_{j\in\mathcal{J}} \int_{x_{ij}^L}^{x_{ij}^H} \frac{1}{\sqrt{2\pi}}\ e^{-\frac{x^2}{2}}\ dx} \\
&= \frac{2\cdot\sum\limits_{j\in\mathcal{J}} \left[\exp\left(-\frac{(x_{ij}^L)^2}{2}\right) - \exp\left(-\frac{(x_{ij}^H)^2}{2}\right)\right]}{\sum\limits_{j\in\mathcal{J}} \left[\operatorname{erf}\left(\frac{x_{ij}^H}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{x_{ij}^L}{\sqrt{2}}\right)\right]};
\end{aligned}
\tag{A.10}
$$

$$
c_j^{(2)} = \frac{2\cdot\sum\limits_{i\in\mathcal{I}} \left[\exp\left(-\frac{(x_{ij}^L)^2}{2}\right) - \exp\left(-\frac{(x_{ij}^H)^2}{2}\right)\right]}{\sum\limits_{i\in\mathcal{I}} \left[\operatorname{erf}\left(\frac{x_{ij}^H}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{x_{ij}^L}{\sqrt{2}}\right)\right]},
\tag{A.11}
$$

where the error function

$$
\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt\ .
$$

Thereafter, D_s can be obtained by using (A.6)–(A.9) and (2.22).

A.2 Uniform random signals

To derive the closed form for uniformly distributed random signals, we begin with the IA without redundancy; thereafter, closed forms associated with index assignment with redundancy are derived.

A.2.1 No redundancy is added ($\rho = 0$)

First, we consider $\mathcal{B}(R_0, 0)$, i.e., the total coding rate $R = R_0$, and no redundancy is added.

We use uniformly distributed $x \in [0, 1]$ as the source signals. Source samples x are uniformly quantized into $L = 2^{R_0}$ levels. Thus, the pdf is $f_X = \frac{1}{L}$; each interval $\Delta = \frac{1}{L} = 2^{-R_0}$. The side bit rate is $R_s = \dfrac{R_0}{2}$ bpss. SDSQ indices $l = 0, 1, \cdots, L-1$ are mapped to RMDSQ index pairs (i, j) by using the IA algorithm provided in Section 5.3.1, where $i, j = 0, 1, \cdots, M-1$, where $M = 2^{R_s}$. An example of this IA scheme is shown in Fig. 5.2. The partition associated with l is given by $v_l = [l \cdot \Delta,\ (l+1) \cdot \Delta)$. The output associated with l or (i, j) is $c_l = c_{ij} = (l + \frac{1}{2}) \cdot \Delta$.

Due to

$$\mathbf{l} = [l_{(R_0-1)} \ \cdots \ l_1 \ l_0], \tag{A.12}$$

$$\mathbf{i} = [i_{(R_s-1)} i_{(R_s-2)} \cdots i_1 i_0] = [l_{(R_0-2)} l_{(R_0-3)} \cdots l_{(\frac{R_0}{2}-1)}] \qquad \text{and} \tag{A.13}$$

$$\mathbf{j} = [j_{(R_s-1)} j_{(R_s-2)} \cdots j_1 j_0] = [l_{(R_0-1)} l_{(\frac{R_0}{2}-2)} \cdots l_1 l_0], \tag{A.14}$$

$$j = \sum_{k=0}^{R_s-1} j_k \cdot 2^k, \qquad \text{and} \tag{A.15}$$

$$i = \sum_{k=0}^{R_s-1} i_k \cdot 2^k; \tag{A.16}$$

therefore, $l = a^{-1}(i, j)$ is given by

$$\begin{aligned} l &= l_{R_0-1} \cdot 2^{(R_0-1)} + \sum_{k=0}^{(R_s-2)} l_k \cdot 2^k + \sum_{k=R_s-1}^{2R_s-2} l_k \cdot 2^k \\ &= l_{(R_0-1)} \cdot 2^{(R_0-1)} + \sum_{k=0}^{R_s-1} j_k \cdot 2^k - j_{(R_s-1)} \cdot 2^{(R_s-1)} + 2^{(R_s-1)} \cdot \sum_{k=0}^{R_s-1} i_k \cdot 2^k \\ &= \frac{M}{2} \cdot i + j + \frac{1}{2}(L - M) \cdot j_{(R_s-1)} \end{aligned} \tag{A.17}$$

Therefore, c_{ij} is given by

$$c_{ij} = \left[j + \frac{1}{2}(L-M) \cdot j_{(R_s-1)} + \frac{M}{2} \cdot i + \frac{1}{2} \right] \Delta. \tag{A.18}$$

Side codebooks $c_i^{(1)}$ and $c_j^{(2)}$

The side output $c_i^{(1)}$ associated with a certain i is given by

$$\begin{aligned} c_i^{(1)} &= \frac{\sum_{j=0}^{M-1} \left[j + \frac{1}{2}(L-M) \cdot j_{(R_s-1)} + \frac{1}{2} + \frac{M}{2} \cdot i \right] \cdot f_X \cdot \Delta}{\sum_{j=0}^{M-1} f_X} \\ &= \frac{f_X \cdot \Delta \cdot \sum_{j=0}^{M-1} \left[j + \frac{1}{2}(L-M) \cdot j_{(R_s-1)} + \frac{1}{2} + \frac{M}{2} \cdot i \right]}{L \cdot f_X} \\ &= \frac{\Delta}{M} \cdot \left[\sum_{j=0}^{M-1} j + \frac{1}{2}(L-M) \cdot \frac{M}{2} + M \cdot \left(\frac{1}{2} + \frac{M}{2} \cdot i \right) \right] \\ &= (2Mi + L + M) \cdot \frac{\Delta}{4}. \end{aligned} \tag{A.19}$$

Similarly, the side output $c_j^{(2)}$ associated with a certain j is given by

$$\begin{aligned} c_j^{(2)} &= \frac{\Delta}{M} \cdot \sum_{i=0}^{M-1} \left[j + \frac{1}{2}(L-M) \cdot j_{(R_s-1)} + \frac{1}{2} + \frac{M}{2} \cdot i \right] \\ &= \left[j + \frac{1}{2} + \frac{1}{2}(L-M) \cdot \left(j_{(R_s-1)} + \frac{1}{2} \right) \right] \cdot \Delta. \end{aligned} \tag{A.20}$$

Side distortions D_1 and D_2

According to (A.6),

$$\begin{aligned} D_i^{(1)} &= \frac{\sum_{i=0}^{M-1} \left(c_{ij} - c_i^{(1)}\right)^2 f_X}{\sum_{i=0}^{M-1} f_X} + \frac{\Delta^2}{12} \\ &= \frac{1}{M} \sum_{i=0}^{M-1} \left(c_{ij} - c_i^{(1)}\right)^2 + \frac{\Delta^2}{12} \end{aligned} \tag{A.21}$$

Due to (A.18) and (A.19), (A.21) is simplified to

$$D_i^{(1)} = (L-1)\,(3L+4)\,\frac{\Delta^2}{48} + \frac{\Delta^2}{12}\,. \tag{A.22}$$

Since x is uniformly distributed, $D_1 = D_i^{(1)}$.

Similarly, due to (A.8), (A.18) and (A.20), the side distortion associated with a certain j is given by

$$\begin{aligned} D_j^{(2)} &= \frac{1}{M} \sum_{i=0}^{M-1} \left(c_{ij} - c_j^{(2)}\right)^2 + \frac{\Delta^2}{12} \\ &= (L-1)\,L\,\frac{\Delta^2}{48} + \frac{\Delta^2}{12}\,. \end{aligned} \tag{A.23}$$

Therefore, $D_2 = D_j^{(2)}$.

Overall average side distortion D_s

Here, we define the overall average side distortion $D_s(\rho)$ as a function of the bit number ρ of redundancy per source sample. Therefore, according to (A.21) (A.23) and (2.22), the

overall average side distortion at $\rho = 0$ is given by

$$D_s(0) = (L^2 - 1)\,\frac{\Delta^2}{24} + \frac{\Delta^2}{12} \tag{A.24}$$

$$= (L^2 + 1)\,\frac{\Delta^2}{24} \tag{A.25}$$

$$\approx \frac{L^2}{24L^2} \tag{A.26}$$

$$= \frac{1}{24}\,, \tag{A.27}$$

i.e., $D_s(0) \approx -13.8$ dB.

If $X = [0, \sigma)$, $\Delta = \dfrac{\sigma}{L}$,

$$D_s(0) = \frac{\sigma^2}{24}\,. \tag{A.28}$$

A.2.2 One-bit redundancy is added ($\rho = 1$)

According to the conclusion obtained in Section 5.3.3, the IA scheme of $\mathcal{B}(R_0, 1)$ is comprised of two non-overlapping IA schemes of $\mathcal{B}'(R_0 - 1, 0)$, which are associated with SDSQ index sets $\mathcal{L}_A = \{0,\, 1,\, \cdots,\, 2^{R_0-1} - 1\}$ and $\mathcal{L}_B = \{2^{R_0-1},\, 2^{R_0-1} + 1,\, \cdots,\, 2^{R_0} - 1\}$ respectively. Therefore, the average side distortion of $\mathcal{B}(R_0, 1)$ equals to that of each $\mathcal{B}'(R_0 - 1, 0)$. That is to say, for the source $X = [0, \sigma)$, the source associated with $\mathcal{B}'(R_0 - 1, 0)$ is $X' = \left[0,\, \dfrac{\sigma}{2}\right)$ or $\left[\dfrac{\sigma}{2},\, \sigma\right)$. Hence, in terms of (A.28), the overall average side distortion is

$$D_s(1) = \frac{\sigma^2}{96}\,. \tag{A.29}$$

If $\sigma = 1$, $D_s = \dfrac{1}{96}$, i.e., -19.8 dB.

A.2.3 Arbitrary number of redundancy bits are added

For $\mathcal{B}(R_0,\, \rho)$, $R_0 \geq 3$ and $\rho \geq 2$, we only consider the optimal bit allocation provided in (5.36). That is to say, the bit allocation scheme is

$$\langle \underbrace{1,\ 1,\ \cdots,\ 1}_{\rho - 1},\ R_0 - \rho + 1 \rangle\,.$$

Therefore, the side distortion is determined by the least significant bit partition. Let Δ_0 be the quantization partition interval associated with the least significant bit partition. Thus, $\Delta_0 = 2^{-(\rho-1)}\sigma$. In terms of (A.29), the overall average side distortion $\mathcal{B}(R_0,\, \rho)$ is given by

$$\begin{aligned} D_s(\rho) &= \frac{\Delta_0^2}{96} \\ &= \frac{[2^{-(\rho-1)}\sigma]^2}{96} \\ &= 2^{-2(\rho-1)} \cdot \frac{\sigma^2}{96} \\ &= 2^{-2\rho} \cdot \frac{\sigma^2}{24} \\ &= 2^{-2\rho} \cdot D_s(0)\ . \end{aligned} \tag{A.30}$$

In addition, when $\rho = 1$, according to (A.29), $D_s(1) = 2^{-2} \cdot D_s(0)$. Therefore, for arbitrary $\mathcal{B}(R_0, \rho)$, $\rho \geq 0$,

$$\begin{aligned} D_s(\rho) &= 2^{-2\rho} \cdot \frac{\sigma^2}{24} \\ &= 2^{-2\rho} \cdot D_s(0)\ , \end{aligned} \tag{A.31}$$

i.e., $D_s(\rho) = D_s(0) - 6.02\rho$ dB.

A.3 Gaussian random signals

In this section, assume that in a high rate system, i.i.d. Gaussian random variables x with mean $\mu = 0$ and variance $\sigma = 1$ are uniformly quantized and represented by SDSQ indices l. Let $x_{min} = -\frac{\delta}{2}$ and $x_{max} = \frac{\delta}{2}$. Then $x \in [-\frac{\delta}{2}, \frac{\delta}{2}]$, partition interval $\Delta = \frac{\delta}{L}$, and the partition v_l or A_{ij} associated with l is given by

$$A_{ij} = \left[x_{ij}^L, x_{ij}^H\right) = [l \cdot \Delta, (l+1) \cdot \Delta).$$

SDSQ indices

$$l = -\frac{L}{2}, \cdots, -1, 0, 1, \cdots, \frac{L}{2} - 1$$

i \ j	0	1	2	3
0	-8	-7	0	1
1	-6	-5	2	3
2	-4	-3	4	5
3	-2	-1	6	7

Fig. A.1 Index assignment without redundancy.

are one-to-one mapped to MDSQ index pairs (i, j), where

$$i = -\frac{M}{2}, \cdots, -1, 0, 1, \cdots, \frac{M}{2} - 1,$$
$$j = -\frac{M}{2}, \cdots, -1, 0, 1, \cdots, \frac{M}{2} - 1.$$

Note that for convenience of derivation, SDSQ indices l, ERMDC indices i and j are here defined as signed integers, instead of unsigned integer as usually defined in the rest of this dissertation. An example is illustrated in Fig. A.1.

l^L and l^H denote SDSQ indices $l < 0$ and $l \geq 0$, respectively. When $l < 0$ or $j < 0$,

$$l^L = \frac{M}{2} i + j - \frac{M}{2};$$

when $l \geq 0$ or $j \geq 0$,

$$l^H = \frac{M}{2} i + j + \frac{L}{4}.$$

We distinguish l^L and l^H by replacing two-sided j with corresponding $\widetilde{j} = |j|$, that is to say, $\widetilde{j} = 0, 1, \cdots, \frac{M}{2} - 1$. Then,

$$l^L = \frac{M}{2} i + \widetilde{j} - \frac{L}{4}; \tag{A.32}$$
$$l^H = \frac{M}{2} i + \widetilde{j} + \frac{L}{4}. \tag{A.33}$$

So l^L and l^H are symmetric.

In a high rate system, according to (A.1),

$$c_{ij} = \frac{x_{ij}^H + x_{ij}^L}{2} = \left(l + \frac{1}{2}\right) \cdot \Delta. \tag{A.34}$$

A.3.1 Side codebook $c_i^{(1)}$

Let $c_i^{(1)}$ be the output corresponding to i, when i is given.

$$c_i^{(1)} = \frac{\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} c_{ij} \cdot f_X(c_{ij})}{\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_X(c_{ij})} \tag{A.35}$$

Since $M << L$, we assume that when $l < 0$, namely, $j < 0$, $f_X(c_{ij}) \approx f_i^L$; and when $l \geq 0$, namely, $j \geq 0$, $f_X(c_{ij}) \approx f_i^H$. Then, the denominator of (A.35) can be simplified by

$$\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_X(c_{ij}) \approx \frac{M}{2} \cdot \left(f_i^L + f_i^H\right) \tag{A.36}$$

The numerator of (A.35) can be simplified by

$$\begin{aligned}
\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} c_{ij}\cdot f_X\left(c_{ij}\right) &= \sum_{j=-\frac{M}{2}}^{-1} c_{ij}\cdot f_i^L + \sum_{j=0}^{\frac{M}{2}-1} c_{ij}\cdot f_i^H \\
&= \Delta\cdot f_i^L\cdot \sum_{j=-\frac{M}{2}}^{-1}\left(l^L+\frac{1}{2}\right) + \Delta\cdot f_i^H\cdot \sum_{j=0}^{\frac{M}{2}-1}\left(l^H+\frac{1}{2}\right) \\
&= \Delta\cdot f_i^L\cdot \sum_{j=-\frac{M}{2}}^{-1} l^L + \Delta\cdot f_i^H\cdot \sum_{j=0}^{\frac{M}{2}-1} l^H + \Delta\cdot\left(f_i^L+f_i^H\right)\cdot\frac{M}{2} \\
&= \Delta\cdot f_i^L\cdot \sum_{\widetilde{j}=0}^{\frac{M}{2}-1}\left[\frac{M}{2}i+\widetilde{j}-\frac{L}{4}\right] + \Delta\cdot f_i^H\cdot \sum_{\widetilde{j}=0}^{\frac{M}{2}-1}\left[\frac{M}{2}i+\widetilde{j}+\frac{L}{4}\right] \\
&\quad + \Delta\cdot\left(f_i^L+f_i^H\right)\cdot\frac{M}{2} \\
&= \Delta\cdot\left(f_i^L+f_i^H\right)\cdot\sum_{\widetilde{j}=0}^{\frac{M}{2}-1}\widetilde{j} + \Delta\cdot\left(f_i^L+f_i^H\right)\cdot\frac{L}{4}\,i \\
&\quad + \Delta\cdot\left(f_i^H-f_i^L\right)\cdot\frac{M^3}{8} + \Delta\cdot\left(f_i^L+f_i^H\right)\cdot\frac{M}{2} \\
&= \Delta\cdot\left(f_i^L+f_i^H\right)\cdot\frac{M}{4}\left(\frac{M}{2}-1\right) + \Delta\cdot\left(f_i^L+f_i^H\right)\cdot\frac{L}{4}\,i \\
&\quad + \Delta\cdot\left(f_i^H-f_i^L\right)\cdot\frac{M^3}{8} + \Delta\cdot\left(f_i^L+f_i^H\right)\cdot\frac{M}{2}\ .
\end{aligned} \tag{A.37}$$

Therefore, according to (A.36) and (A.37), (A.35) can be simplified as

$$\begin{aligned}
c_i^{(1)} &= \left\{\frac{1}{2}\left(\frac{M}{2}-1\right) + \frac{M}{2}\,i + \frac{f_i^H-f_i^L}{f_i^L+f_i^H}\cdot\frac{L}{4}+1\right\}\cdot\Delta \\
&= \left\{\frac{M}{2}\,i + \frac{L}{2}\cdot\frac{f_i^H}{f_i^L+p_i^H} + \frac{M}{4} - \frac{L}{4} + \frac{1}{2}\right\}\cdot\Delta\ .
\end{aligned} \tag{A.38}$$

Taking into account the pdf of Gaussian distribution, f_i^L and f_i^H are respectively given by

$$\begin{aligned} f_i^L &\approx f_X(x_{ij}^L) = f_X(l \cdot \Delta) \\ &= f_X\left\{\left(\frac{M}{2}\, i + \widetilde{j} - \frac{L}{4}\right) \cdot \Delta\right\} \\ &\approx f_X\left\{\left(Mi - \frac{L}{2}\right) \cdot \frac{\Delta}{2}\right\} \\ &= \frac{1}{\sqrt{2\pi}} \cdot \exp\left\{-\left(Mi - \frac{L}{2}\right)^2 \cdot \frac{\Delta^2}{8}\right\}\ , \end{aligned} \tag{A.39}$$

and

$$\begin{aligned} f_i^H &\approx f_X(x_{ij}^H) = f_X(l \cdot \Delta) \\ &= f_X\left\{\left(\frac{M}{2} i + \widetilde{j} + \frac{L}{4}\right) \cdot \Delta\right\} \\ &\approx f_X\left\{\left(Mi + \frac{L}{2}\right) \cdot \frac{\Delta}{2}\right\} \\ &= \frac{1}{\sqrt{2\pi}} \cdot \exp\left\{-\left(Mi + \frac{L}{2}\right)^2 \cdot \frac{\Delta^2}{8}\right\} \end{aligned} \tag{A.40}$$

Therefore,

$$\begin{aligned} \frac{f_i^L}{f_i^H} &= \exp\left\{\left(Mi + \frac{L}{2}\right)^2 \cdot \frac{\Delta^2}{8} - \left(Mi - \frac{L}{2}\right)^2 \cdot \frac{\Delta^2}{8}\right\} \\ &= \exp\left(\frac{M^3}{4}\, i \cdot \Delta^2\right)\ . \end{aligned} \tag{A.41}$$

Define Φ_i as

$$\Phi_i = \frac{f_i^H}{f_i^L + f_i^H} = \frac{1}{1 + \dfrac{f_i^L}{f_i^H}} = \frac{1}{1 + \exp\left(\frac{M^3}{4}\, i \cdot \Delta^2\right)}\ . \tag{A.42}$$

Hence, $c_i^{(1)}$ is finally given by

$$\begin{aligned} c_i^{(1)} &= \left\{ \frac{M}{2}\, i + \frac{L}{2} \cdot \Phi_i + \frac{M}{4} - \frac{L}{4} + \frac{1}{2} \right\} \cdot \Delta \\ &\approx \left\{ \frac{M}{2} i + \frac{L}{2} \cdot \Phi_i - \frac{L}{4} \right\} \cdot \Delta \\ &= \frac{M}{2} \cdot \left(i + M\Phi_i - \frac{M}{2} \right) \cdot \Delta \ . \end{aligned} \tag{A.43}$$

A.3.2 Average side distortion D_1

Let D_l or D_{ij} be the side distortion with respect to each l. When $l < 0$, $l^L = \frac{M}{2}\, i + \widetilde{j} - \frac{L}{4}$,

$$c_{ij}^L = \left(l + \frac{1}{2} \right) \cdot \Delta \approx \left(\frac{M}{2}\, i - \frac{L}{4} \right) \cdot \Delta \ . \tag{A.44}$$

Thus,

$$\begin{aligned} D_{ij}^L &= \left(c_{ij}^L - c_i^{(1)} \right)^2 \\ &\approx \left\{ \left(\frac{M}{2}\, i - \frac{L}{4} \right) \cdot \Delta - \left(\frac{M}{2}\, i + \frac{L}{2} \cdot \Phi_i - \frac{L}{4} \right) \cdot \Delta \right\}^2 \\ &= \frac{L^2}{4} \cdot \Phi_i^2 \cdot \Delta^2 \ . \end{aligned} \tag{A.45}$$

When $l \geq 0$, $l^H = \frac{M}{2}\, i + \widetilde{j} + \frac{L}{4}$,

$$c_{ij}^H = \left(l + \frac{1}{2} \right) \cdot \Delta \approx \left(\frac{M}{2}\, i + \frac{L}{4} \right) \cdot \Delta \ . \tag{A.46}$$

Thus,

$$\begin{aligned} D_{ij}^H &= \left(c_{ij}^H - c_i^{(1)} \right)^2 \\ &\approx \left\{ \left(\frac{M}{2} i + \frac{L}{4} \right) \cdot \Delta - \left(\frac{M}{2}\, i + \frac{L}{2} \cdot \Phi_i - \frac{L}{4} \right) \cdot \Delta \right\}^2 \\ &= \frac{L^2}{4} \cdot (1 - \Phi_i)^2 \cdot \Delta^2 \end{aligned} \tag{A.47}$$

Let D_i be the side distortion associated with each i, when i is given. According to (A.6),

$$\begin{aligned}
D_i^{(1)} &\approx \frac{\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} \left(c_{ij} - c_i^{(1)}\right)^2 \cdot f_X(c_{ij})}{\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_X(c_{ij})} + \frac{\Delta^2}{12} \\
&\approx \frac{\sum_{\widetilde{j}=0}^{\frac{M}{2}-1} \left(D_{ij}^L \cdot f_i^L + D_{ij}^H \cdot f_i^H\right)}{\frac{M}{2} \cdot (f_i^L + f_i^H)} + \frac{\Delta^2}{12} \\
&\approx \frac{\frac{M}{2} \cdot \left(D_{ij}^L \cdot f_i^L + D_{ij}^H \cdot f_i^H\right)}{\frac{M}{2} \cdot (f_i^L + f_i^H)} + \frac{\Delta^2}{12} \\
&= D_{ij}^L \cdot (1 - \Phi_i) + D_{ij}^H \cdot \Phi_i + \frac{\Delta^2}{12} + \frac{\Delta^2}{12} \\
&= \left\{\Phi_i^2 \cdot (1 - \Phi_i) + (1 - \Phi_i)^2 \cdot \Phi_i\right\} \cdot \frac{L^2}{4} \cdot \Delta^2 + \frac{\Delta^2}{12} \\
&= \Phi_i(1 - \Phi_i) \cdot \frac{L^2}{4} \cdot \Delta^2 + \frac{\Delta^2}{12}
\end{aligned} \tag{A.48}$$

The distribution of each i is given by

$$\begin{aligned} f_i &= \frac{\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_X(c_{ij})}{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1}\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_X(c_{ij})} \\ &= \frac{\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_X(c_{ij})\cdot\Delta}{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1}\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_X(c_{ij})\cdot\Delta} \\ &\approx \frac{\frac{M}{2}\left(f_i^L + f_i^H\right)\cdot\Delta}{\int_{-\infty}^{\infty} f_X(x)\,dx} \\ &= \frac{M}{2}\left(f_i^L + f_i^H\right)\Delta \end{aligned} \tag{A.49}$$

Therefore, the average side distortion of this description is given by

$$\begin{aligned} D_1 &= \frac{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} D_i\, f_i}{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} f_i} + \frac{\Delta^2}{12} = \sum_{i=-\frac{M_1}{2}}^{\frac{M_1}{2}-1} D_i\, f_i + \frac{\Delta^2}{12} \\ &= \frac{M}{2}\frac{L^2}{4}\Delta^3 \cdot \sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} \frac{f_i^L\, f_i^H}{\left(f_i^L + f_i^H\right)^2}\cdot\left(f_i^L + f_i^H\right) + \frac{\Delta^2}{12} \\ &= \frac{M^5}{8}\Delta^3 \cdot \sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} \frac{f_i^L\, f_i^H}{f_i^L + f_i^H} + \frac{\Delta^2}{12}\,. \end{aligned} \tag{A.50}$$

A.3.3 Side codebook $c_j^{(2)}$

Let $c_j^{(2)}$ be the output with respect to j, when j is given. According to (A.2),

$$c_j^{(2)} = \frac{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} c_{ij} \cdot f_X\left(c_{ij}\right)}{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} f_X\left(c_{ij}\right)}. \tag{A.51}$$

Therefore, when $j < 0$,

$$c_j^{(2)} \approx \frac{\int_{-\infty}^{0} x \cdot f_X(x)\ dx}{\int_{-\infty}^{0} f_X(x)\ dx} = -\sqrt{\frac{2}{\pi}}\ ; \tag{A.52}$$

when $j \geq 0$,

$$c_j^{(2)} \approx \frac{\int_{0}^{\infty} x \cdot f_X(x)\ dx}{\int_{0}^{\infty} f_X(x)\ dx} = \sqrt{\frac{2}{\pi}}\ . \tag{A.53}$$

A.3.4 Average side distortion D_2

Let $D_j^{(2)}$ be the distortion corresponding to each j, when j is given. According to (A.8),

$$\begin{aligned} D_j^{(2)} &= \frac{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} \left(c_{ij} - c_j^{(2)}\right)^2 f_X(c_{ij})}{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} f_X(c_{ij})} + \frac{\Delta^2}{12} \\ &= \widetilde{D}_j^{(2)} + \frac{\Delta^2}{12}\ , \end{aligned} \tag{A.54}$$

where

$$\widetilde{D}_j^{(2)} = \frac{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} \left(c_{ij} - c_j^{(2)}\right)^2 f_X(c_{ij})}{\sum_{i=-\frac{M}{2}}^{\frac{M}{2}-1} f_X(c_{ij})} \ . \tag{A.55}$$

When $l < 0$,

$$\begin{aligned}
\widetilde{D}_j^{(2)} &\approx \frac{\int_{-\infty}^{0}(x - c_j^{(2)})^2 \cdot f_X(x)\ dx}{\int_{-\infty}^{0} f_X(x)\ dx} \\
&= \frac{\int_{-\infty}^{0} x^2 \cdot f_X(x)\ dx}{\int_{-\infty}^{0} f_X(x)\ dx} - 2\, c_j^{(2)} \cdot \frac{\int_{-\infty}^{0} x \cdot f_X(x)\ dx}{\int_{-\infty}^{0} f_X(x)\ dx} + \left(c_j^{(2)}\right)^2 \cdot \frac{\int_{-\infty}^{0} \cdot f_X(x)\ dx}{\int_{-\infty}^{0} f_X(x)\ dx} \\
&= 1 - 2\, c_j^{(2)} \cdot c_j^{(2)} + \left(c_j^{(2)}\right)^2 \\
&= 1 - \left(c_j^{(2)}\right)^2 \\
&= 1 - \frac{2}{\pi}
\end{aligned} \tag{A.56}$$

Similarly, when $l \geq 0$, $\widetilde{D}_j^{(2)} \approx 1 - \frac{2}{\pi}$. Thus, for all l, $\widetilde{D}_j^{(2)} \approx 1 - \frac{2}{\pi}$.

Consequently, according to (A.54),

$$D_j^{(2)} \approx 1 - \frac{2}{\pi} + \frac{\Delta^2}{12} \ . \tag{A.57}$$

Therefore, according to (A.9), the average side distortion D_2 of this description is given by

$$\begin{aligned}
D_2 &= \frac{\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} D_j^{(2)} \cdot f_X\left(D_j^{(2)}\right)}{\sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_X\left(D_j^{(2)}\right)} \\
&\approx 1 - \frac{2}{\pi} + \frac{\Delta^2}{12} \ .
\end{aligned} \tag{A.58}$$

Consequently, the overall average side distortion D_s can be obtained in terms of (A.50)(A.58) and (2.22).

References

[1] J.-P. Joseph, P. Justl, F. R. J. Magee, A. Mukhopadhyay, and D. Sun, "Converged wireline-wireless network evolution: Opportunities and challenges," *Bell Labs Tech. J.*, vol. 10, no. 2, pp. 57–80, Aug. 2005.

[2] F. Dickson, "IPTV content & services 2007: Telecom companies turn up the heat," iSuppli Corporation, Tech. Rep., June 2007. [Online]. Available: http://www.isuppli.com/Abstract/P8989_20070711143536.pdf [Accessed: July 2, 2009]

[3] "VoIP service revenue up 52% in 2007; boom expected to continue," Infonetics Research, Tech. Rep., Aug. 13, 2008. [Online]. Available: http://www.infonetics.com/pr/2008/ms08.vip.nr.asp [Accessed: July 2, 2009]

[4] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE Network*, vol. 12, no. 5, pp. 40–48, Sept./Oct. 1998.

[5] D. Pan, "A tutorial on MPEG/audio compression," *IEEE Trans. Multimedia*, vol. 2, no. 2, pp. 60–74, Summer 1995.

[6] J. Chakareski, S. Han, and B. Girod, "Layered coding vs. multiple descriptions for video streaming over multiple paths," *Multimedia Systems*, vol. 10, no. 4, pp. 275–285, Apr. 2005.

[7] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, no. 12, pp. 3445–3462, December 1993.

[8] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 243–250, June 1996.

[9] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Trans. Consumer Electron.*, vol. 46, no. 4, pp. 1103–1127, Nov. 2000.

[10] D. S. Taubman and M. W. Marcellin, "JPEG2000: Standard for interactive imaging," *Proc. IEEE*, vol. 90, no. 8, pp. 1336–1357, August 2002.

[11] J. Hagenauer and T. Stockhammer, "Channel coding and transmission aspects for wireless multimedia," *Proc. IEEE*, vol. 87, no. 10, pp. 1764–1777, Oct. 1999.

[12] D. G. Sachs, R. Anand, and K. Ramchandran, "Wireless image transmission using multiple-description based concatenated codes," in *Proc. SPIE Visual Commun. Image Processing*, vol. 3974, Jan. 2000, pp. 300–311.

[13] A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Unequal loss protection: Graceful degradation of image quality over packet erasure channels through forward error correction," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 819–828, June 2000.

[14] J. Goshi, A. E. Mohr, R. E. Ladner, E. A. Riskin, and A. Lippman, "Unequal loss protection for H.263 compressed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 3, pp. 412–419, Mar. 2005.

[15] V. K. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Processing Mag.*, vol. 18, no. 5, pp. 74–93, Sept. 2001.

[16] L. Ozarow, "On a source-coding problem with two channels and three receivers," *Bell Syst. Tech. J.*, vol. 59, no. 10, pp. 1909–1921, Dec. 1980.

[17] A. El Gamal and T. Cover, "Achievable rates for multiple descriptions," *IEEE Trans. Inform. Theory*, vol. 28, no. 6, pp. 851–857, Nov. 1982.

[18] V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Trans. Inform. Theory*, vol. 39, no. 3, pp. 821–834, May 1993.

[19] T. Y. Berger-Wolf and E. M. Reingold, "Index assignment for multichannel communication under failure," *IEEE Trans. Inform. Theory*, vol. 48, no. 10, pp. 2656–2668, Oct. 2002.

[20] P. Yahampath, "On index assignment and the design of multiple description quantizers," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 4, May 2004, pp. iv–597–iv–600.

[21] Y. Zhou and W.-Y. Chan, "Multiple description quantizer design using a channel optimized quantizer approach," in *Proc. Conf. Inform. Sci. Syst.*, Mar. 2004.

[22] R. Ma and F. Labeau, "Enhanced multiple description decoder for error-prone channels," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2006, pp. 805–808.

[23] ——, "Robust index assignment for MDSQ encoder over noisy channels," in *Proc. IEEE Int. Workshop Multimedia Signal Processing*, Oct. 2006, pp. 286–290.

[24] ——, "Error-resilient multiple description coding," *IEEE Trans. Signal Processing*, vol. 56, no. 8, pp. 3996–4007, Aug. 2008.

[25] ——, "Soft input error resilient multiple description coding for Rayleigh fading channels," in *Proc. IEEE Int. Conf. Multimedia Expo*, July 2007, pp. 1147–1150.

[26] ——, "Fast index assignment for robust multiple description coding," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2008, pp. 2052–2055.

[27] ——, "Generalized fast index assignment for robust multiple description scalar quantizers," in *Proc. Asilomar Conf. Signals, Syst., Computers*, Oct. 2008, pp. 1287–1291.

[28] ——, "End-to-end performance of robust multiple description scalar quantizer," in *Proc. IEEE Veh. Technol. Conf.*, Apr. 2009.

[29] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, July 1948.

[30] S. Vembu, S. Verdu, and Y. Steinberg, "The source-channel separation theorem revisited," *IEEE Trans. Inform. Theory*, vol. 41, no. 1, pp. 44–54, Jan. 1995.

[31] K. Zeger and A. Gersho, "Zero redundancy channel coding in vector quantisation," *IET Electron. Lett.*, vol. 23, no. 12, pp. 654–656, June 1987.

[32] ——, "Pseudo-Gray coding," *IEEE Trans. Commun.*, vol. 38, no. 12, pp. 2147–2158, Dec. 1990.

[33] W.-W. Chang, T.-H. Tan, and D.-Y. Wang, "Robust vector quantization for wireless channels," *IEEE J. Select. Areas Commun.*, vol. 19, no. 7, pp. 1365–1373, July 2001.

[34] N. Farvardin and V. Vaishampayan, "Optimal quantizer design for noisy channels: An approach to combined source-channel coding," *IEEE Trans. Inform. Theory*, vol. 33, no. 6, pp. 827–837, Nov. 1987.

[35] N. Farvardin, "A study of vector quantization for noisy channels," *IEEE Trans. Inform. Theory*, vol. 36, no. 4, pp. 799–809, July 1990.

[36] N. Farvardin and V. Vaishampayan, "On the performance and complexity of channel-optimized vector quantizers," *IEEE Trans. Inform. Theory*, vol. 37, no. 1, pp. 155–160, Jan. 1991.

[37] M. R. Garey and D. S. Johnson, *Computers and intractability: a guide to the theory of NP-completeness*. W. H. Freeman & Co., 1990.

[38] K. Zeger and A. Gersho, "Vector quantizer design for memoryless noisy channels," in *Proc. IEEE Int. Conf. Commun.*, vol. 3, June 1988, pp. 1593–1597.

[39] J.-H. Chen, G. Davidson, A. Gersho, and K. Zeger, "Speech coding for the mobile satellite experiment," in *Proc. IEEE Int. Conf. Commun.*, June 1987, pp. 756–763.

[40] C. J. Kuo, C. H. Lin, and C. H. Yeh, "Noise reduction of VQ encoded images through anti-Gray coding," *IEEE Trans. Image Processing*, vol. 8, no. 1, pp. 33–40, Jan. 1999.

[41] J. K. Wolf, A. D. Wyner, and J. Ziv, "Source coding for multiple descriptions," *Bell Syst. Tech. J.*, vol. 59, no. 8, pp. 1417–1426, Oct. 1980.

[42] A. Ingle and V. Vaishampayan, "DPCM system design for diversity systems with applications to packetized speech," *IEEE Trans. Speech Audio Processing*, vol. 3, pp. 48–57, Jan. 1995.

[43] Y. Wang, M. T. Orchard, and A. R. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," in *Proc. IEEE Int. Workshop Multimedia Signal Processing*, June 1997, pp. 419–424.

[44] M. Srinivasan and R. Chellappa, "Multiple description subband coding," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, 4-7 Oct. 1998, pp. 684–688.

[45] V. K. Goyal, J. Kova**v**cević, R. Arean, and M. Vetterli, "Multiple description transform coding of images," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, Oct. 1998, pp. 674–678.

[46] V. K. Goyal, J. Kova**v**cević, and M. Vetterli, "Multiple description transform coding: Robustness to erasures using tight frame expansions," in *Proc. IEEE Int. Symposium Inform. Theory*, 16 - 21 August 1998, p. 408.

[47] W. Jiang and A. Ortega, "Multiple description coding via polyphase transform and selective quantization," in *Proc. SPIE Visual Commun. Image Processing*, vol. 3653, Jan. 1999, pp. 998–1008.

[48] R. Puri and K. Ramchandran, "Multiple description source coding using forward error correction codes," in *Proc. Asilomar Conf. Signals, Syst., Computers*, vol. 1, Oct. 1999, pp. 342–346.

[49] P. Sagetong and A. Ortega, "Optimal bit allocation for channel-adaptive multiple description coding," in *Proc. SPIE Visual Commun. Image Processing*, vol. 3974, Jan. 2000, pp. 53–63.

[50] P. Sherwood, X. Tian, and K. Zeger, "Efficient image and channel coding for wireless packet networks," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, 2000, pp. 132–135.

[51] S. D. Servetto, K. Ramchandran, V. A. Vaishampayan, and K. Nahrstedt, "Multiple description wavelet based image coding," *IEEE Trans. Image Processing*, vol. 9, no. 5, pp. 813–826, May 2000.

[52] N. Varnica, M. Fleming, and M. Effros, "Multi-resolution adaptation of the SPIHT algorithm for multiple description," in *Proc. Data Compression Conf.*, Mar. 2000, pp. 303–312.

[53] R. Puri, K. Ramchandran, K. W. Lee, and V. Bharghavan, "Forward error correction (FEC) codes based multiple description coding for Internet video streaming and multicast," *Signal Processing: Image Commun.*, vol. 16, no. 8, pp. 745–762, May 2001.

[54] T. Guionnet, C. Guillemot, and S. Pateux, "Embedded multiple description coding for progressive image transmission over unreliable channels," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, 7-10 Oct. 2001, pp. 94–97.

[55] A. C. Ashwin, K. R. Ramakrishman, and S. H. Srinivasan, "A multiple description method for wavelet based image coding," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, 22-25Sept. 2002, pp. II–709 – II–712.

[56] M. Pereira, M. Antonini, and M. Barlaud, "Channel adapted multiple description coding scheme using wavelet transform," in *Proc. ICIP*, 2002, pp. II–197–200.

[57] K. P. Subbalakshmi and S. Somasundaram, "Multiple description image coding framework for EBCOT," in *Proc. IEEE Int. Conf. Image Processing*, 2002, pp. III–541–544.

[58] I. V. Bajic and J. W. Woods, "Domain-based multiple description coding of images and video," *IEEE Trans. Image Processing*, vol. 12, no. 10, pp. 1211 – 1225, October 2003.

[59] H. Chen and Y. Yang, "Multiple descriptions based wavelet image coding," in *Proc. Int. Conf. Parallel Distrib. Comput., Applicat., Technol.*, Aug. 2003, pp. 805–808.

[60] I.-K. Eom and Y.-S. Kim, "Robust EZW coding with shared threshold," *IET Electron. Lett.*, vol. 39, no. 21, pp. 1514 – 1515, 16 Oct. 2003.

[61] T. Gan and K.-K. Ma, "Sliding-window packetization for forward error correction based multiple description transcoding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 5, Apr. 2003, pp. V–756–759.

[62] T. Tillo and G. Olmo, "A novel multiple description coding scheme compatible with the JPEG2000 decoder," *IEEE Signal Processing Lett.*, vol. 11, no. 11, pp. 908–911, Nov. 2004.

[63] Y. Wang, A. R. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proc. IEEE*, vol. 93, no. 1, pp. 57–70, Jan. 2005.

[64] C. Cai, J. Chen, K.-K. Ma, and S. K. Mitra, "Multiple description wavelet coding with dual decomposition and cross packetization," *Signal, Image and Video Processing*, vol. 1, no. 1, pp. 53–61, Apr. 2007.

[65] C. Cai, J. Chen, and S. K. Mitra, "Structure unanimity multiple description coding," *Signal Processing: Image Commun.*, vol. 22, no. 1, pp. 59–68, Jan. 2007.

[66] S.-H. Yang and P.-F. Cheng, "Robust transmission of SPIHT-coded images over packet networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 5, pp. 558–567, May 2007.

[67] V. A. Vaishampayan and J. Domaszewicz, "Design of entropy-constrained multiple-description scalar quantizers," *IEEE Trans. Inform. Theory*, vol. 40, no. 1, pp. 245–250, Jan. 1994.

[68] M. Fleming and M. Effros, "Generalized multiple description vector quantization," in *Proc. Data Compression Conf.*, Mar. 1999, pp. 3–12.

[69] H. Jafarkhani and V. Tarokh, "Multiple description trellis-coded quantization," *IEEE Trans. Commun.*, vol. 47, pp. 799–803, June 1999.

[70] Y. Frank-Dayan and R. Zamir, "Universal lattice-based quantizers for multiple descriptions," in *Proc. Data Compression Conf.*, Mar. 2000, pp. 500–509.

[71] V. A. Vaishampayan, N. J. A. Sloane, and S. D. Servetto, "Multiple-description vector quantization with lattice codebooks: Design and analysis," *IEEE Trans. Inform. Theory*, vol. 47, no. 5, pp. 1718–1734, July 2001.

[72] J.-C. Batllo and V. A. Vaishampayan, "Asymptotic performance of multiple description transform codes," *IEEE Trans. Inform. Theory*, vol. 43, no. 2, pp. 703–707, Mar. 1997.

[73] V. K. Goyal and J. Kovacevic, "Optimal multiple description transform coding of Gaussian vectors," in *Proc. Data Compression Conf.*, Mar. 1998, pp. 388–397.

[74] ——, "Generalized multiple description coding with correlating transforms," *IEEE Trans. Inform. Theory*, vol. 47, no. 6, pp. 2199–2224, Sept. 2001.

[75] Y. Wang, M. Orchard, V. Vaishampayan, and A. Reibman, "Multiple description coding using pairwise correlating transforms," *IEEE Trans. Image Processing*, vol. 10, pp. 351–366, Mar. 2001.

[76] M. Grangetto, E. Magli, and G. Olmo, "Ensuring quality of service for image transmission: Hybrid loss protection," *IEEE Trans. Inform. Theory*, vol. 13, no. 6, pp. 751–757, June 2004.

[77] D. S. Taubman and M. W. Marcellin, *JPEG2000: Image Compression, Fundamentals, Standards and Practice*. Kluwer Academic Publishers, 2002.

[78] A. I. Gavrilescu, A. Munteanu, P. Schelkens, and J. Cornelis, "Embedded multiple description scalar quantisers," *IET Electron. Lett.*, vol. 39, no. 13, pp. 979–980, 26 June 2003.

[79] A. I. Gavrilescu, A. Munteanu, J. Cornelis, and P. Schelkens, "Generalisation of embedded multiple description scalar quantisers," *IET Electron. Lett.*, vol. 41, no. 2, pp. 63 – 65, 20 Jan. 2005.

[80] A. Munteanu, J. Cornelis, G. V. der Auwera, and P. Cristea, "Wavelet-based lossless compression scheme with progressive transmission capability," *International Journal of Imaging Systems and Technology*, vol. 10, no. 1, pp. 76–85, Jan. 1999.

[81] A. C. Miguel, A. E. Mohr, and E. A. Riskin, "SPIHT for generalized multiple description coding," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 24-28 Oct. 1999, pp. 842–846.

[82] J. Balogh and J. A. Csirik, "Index assignment for two-channel quantization," *IEEE Trans. Inform. Theory*, vol. 50, no. 11, pp. 2737–2751, Nov. 2004.

[83] N. Gortz and P. Leelapornchai, "Optimization of the index assignments for multiple description vector quantizers," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 336–340, Mar. 2003.

[84] S.-M. Yang and V. A. Vaishampayan, "Low-delay communication for Rayleigh fading channels: An application of the multiple description quantizer," *IEEE Trans. Commun.*, vol. 43, no. 11, pp. 2771–2783, Nov. 1995.

[85] A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Generalized multiple description coding through unequal loss protection," in *Proc. ICIP*, vol. 1, Oct. 1999, pp. 411–415.

[86] K. Lee, R. Puri, T. Kim, K. Ramchandran, and V. Bharghavan, "An integrated source coding and congestion control framework for video streaming in the internet," in *Proc. IEEE Conf. Computer Commun.*, Mar. 2000, pp. 747–756.

[87] J. Barros, J. Hagenauer, and N. Görtz, "Turbo cross decoding of multiple descriptions," in *Proc. IEEE Int. Conf. Commun.*, Apr. 2002, pp. 1398–1402.

[88] T. Guionnet, C. Guillemot, and E. Fabre, "Soft decoding of multiple descriptions," in *Proc. IEEE Int. Conf. Multimedia Expo*, vol. 2, Aug. 2002, pp. 601–604.

[89] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*, 2nd ed. John Wiley & Sons, Inc., 2004.

[90] Y. Zhou, W.-Y. Chan, and T. H. Falk, "Multiple-channel optimized quantizers for Rayleigh fading channels," in *Proc. Asilomar Conf. Signals, Syst., Computers*, vol. 1, Nov. 2004, pp. 687–691.

[91] M. Y. Kim and W. B. Kleijn, "Rate-distortion comparisons between FEC and MDC based on Gilbert channel model," in *Proc. IEEE Int. Conf. Networks*, Sept. 28-Oct. 1 2003, pp. 495–500.

[92] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J.*, vol. 39, pp. 1253–1266, Sept. 1960.

[93] K. Sayood, *Introduction to Data Compression*, 2nd ed. Morgan Kaufmann Publishers, 2000.