SYMMETRIC MDC WITH REDUCED STORAGE SPACE DECODER

TWO TECHNIQUES FOR SYMMETRIC MULTIPLE DESCRIPTION CODING WITH REDUCED STORAGE SPACE DECODER

By

TING ZHENG, B.ENG.

A Thesis

Submitted to the School of Graduate Studies

In Partial Fulfilment of the Requirements

for the Degree

Master of Applied Science

MASTER OF APPLIED SCIENCE (2008) McMaster University
(Electrical and Computer Engineering) Hamilton, Canada

TITLE: Two Techniques for Symmetric Multiple Description Coding with Reduced Storage Space Decoder

AUTHOR: Ting Zheng, B.Eng. (Beijing Univ. of Tech.)

SUPERVISOR: Dr. Sorina Dumitrescu

NUMBER OF PAGES: xi, 99

Abstract

In this work we propose two techniques for symmetric multiple description coding with reduced storage space decoder.

The first technique is multiple description scalar quantizer with linear joint decoders. We propose an optimal design algorithm similar to Vaishampayan's algorithm, to which we add an index assignment optimization step. We also solve an additional challenge in the decoder optimization, by proving that the problem is a convex quadratic optimization problem with a closed form solution (under some mild conditions). Our tests show that the new method has very good performance when the probability of description loss is sufficiently low.

The other technique is an improvement to the traditional multiple description coding scheme based on uneven erasure protection. We evaluate the asymptotical performance of both schemes for a Gaussian memoryless source. The analysis reveals that the improvement reaches over 1 dB for up to ten descriptions and low probability of description loss. From our experiments we observe that the improved scheme is very competitive comparing to other multiple description techniques as well.

Acknowledgement

I would like to take this opportunity to thank the many individuals who have made the completion of this thesis possible.

First and foremost, I would like to express my gratitude to my supervisor Dr. Sorina Dumitrescu, for giving me the opportunity to pursue graduate studies, and for her guidance, encouragement and patience, which are highly appreciated and will always be remembered.

Special thanks to my readers, Dr. Wu and Dr. Zhang, for their valuable input. Sincere thanks to some special individuals I have met since I came to Canada. Heartfelt thank you's to Jie, Xiangjun, Mingkai, Xiaohan, Peipei, Jia, Yingbo and Bing, their kind help and friendship make these two years an experience filled with warmth and great pleasure.

Last but not least, I would like to express my grateful thanks to my parents. Thank you for your unconditional love and support, and thank you for telling me that you are deeply proud of me.

Contents

List of Figures

List of Tables

List of Abbreviations

MD	Multiple Description
MDC	Multiple Description Coding
MDQ	Multiple Description Quantizer
SQ	Scalar Quantizer
MDSQ	Multiple Description Scalar Quantizer
Opt-MDSQ	MDSQ with All Codebooks Stored
L-MDSQ	MDSQ with Linear Joint Decoders
IA	Index Assignment
MD-UEP	MDC Based on Uneven Erasure Protection
SRQ	Successively Refinable Quantizer
RS	Reed Solomon
pdf	Probability Density Function

Chapter 1

Introduction

1.1 Multiple Description Coding

The advent of modern communications raises new challenges for multimedia transmission. One of these challenges, encountered in both wired and wireless networks, is the necessity to alleviate the impact of packet loss. Multiple description coding (MDC) is an effective way to solve this problem. The basic idea in MDC is to create several descriptions of the signal such that it can be reconstructed to a certain fidelity from each individual description, and when more descriptions are available, they can refine each other, leading to a higher reconstruction fidelity. In transmission over packet-switched networks (Fig. 1.1) each packet can be regarded as an individual description. The use of MDC ensures that when some packets are lost due to network congestions and/or channel errors, the received packets can still be decoded leading to

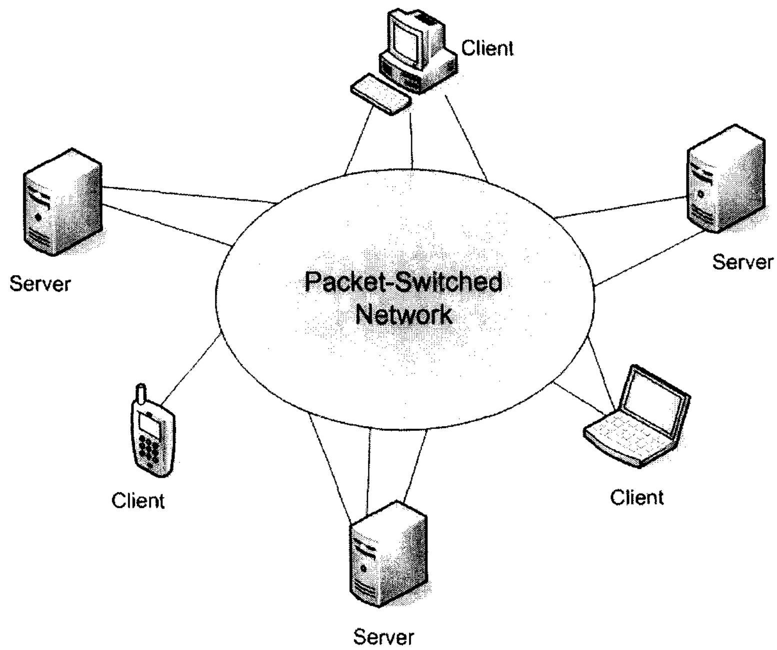


Figure 1.1: Modern packet-switched network.

a reconstruction quality which degrades gracefully as the number of losses increases.

Another application of MDC is in diversity-based communications systems. In such systems there are several channels (or paths in a network) from a sender (or multiple senders) to the destination. Each description is transmitted over a separate channel. In this work we only consider the scenario where each channel either transmits without error or it breaks down. If the same data were transmitted over all the channels there would not be any advantage when all channels transmit successfully versus the case when only one is successful. With MDC, it is guaranteed that when only one channel works, enough data arrives at destination to ensure a good signal reconstruction, while a consistently higher quality reconstruction is achieved as more

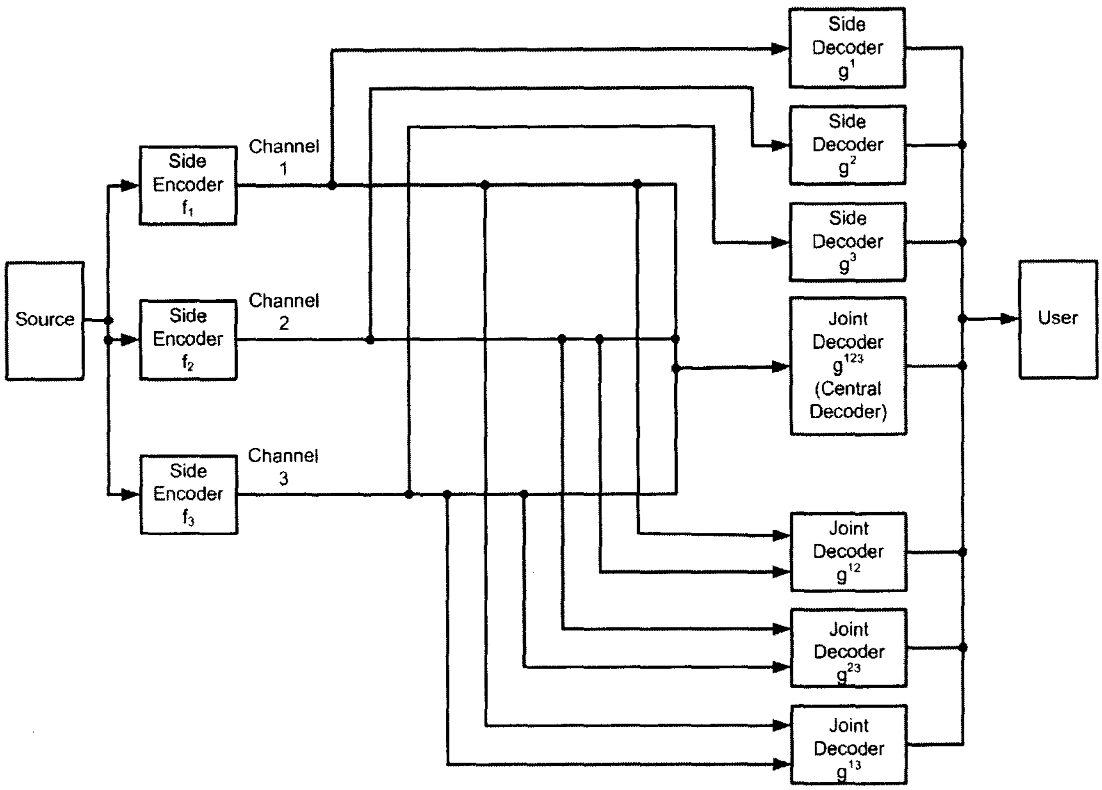


Figure 1.2: Multiple description scheme for three descriptions.

channels are successful.

In the most general setting, an MDC scheme generating K descriptions can be regarded as a system consisting of K encoders (also called side encoders), and $2^K - 1$ decoders, one for each subset of descriptions. Fig. 1.2 illustrates the block diagram of an MDC scheme for three descriptions. Each encoder generates a bitstream (description) of the same source and sends it over a separate channel. The sender does not know the channels failure pattern, but the receiver has this information. If only a subset of channels are successful, hence only some descriptions arrive at destination, the decoder corresponding to that subset of descriptions is used to jointly decode them. The K decoders corresponding to individual descriptions are called side decoders, while the others are termed joint decoders. Moreover, the joint decoder corresponding to the whole set of descriptions is known as the central decoder.

1.2 Information Theory Perspective of MDC

1.2.1 Introduction to Rate Distortion Theory

There is always a tradeoff between the length of the representation (bit rate) and the quality of the reconstruction. In order to achieve a higher quality reconstruction, a higher bit rate is needed. Rate distortion theory studies the achievable region of rates and distortions. The rate-distortion function is determined asymptotically, i.e., by assuming that the block length n tends to infinity, which is not possible in practical codes. However, it gives useful intuition on how quality varies with the length of the representation.

Assume that a source produces a sequence of independent, identically distributed random variables $X_1, X_2, \cdots, X_n, \cdots$, over some alphabet $\mathcal{X}$. An n-block source code consists of an encoder-decoder pair (f, g). The encoder f maps any source sequence $\mathbf{x}^{(n)}$ of length n to an index $f(\mathbf{x}^{(n)}) \in \{1, 2, \cdots, 2^{nR}\}$, where R is the bit rate per symbol. The decoder g maps any index from $\{1, 2, \cdots, 2^{nR}\}$ to a reproduction sequence $\mathbf{a}^{(n)} = g(f(x^{(n)})) \in \mathcal{X}^n$.

The distortion measure d gives a nonnegative numerical rating $d(x, a)$ to how well a source sample x is approximated by a reconstruction value a. The most commonly used distortion measure is the squared error measure $d(x, a) = (x-a)^2$. The distortion between sequences $\mathbf{x}^{(n)} = (x_1, x_2, \cdots, x_n)$ and $\mathbf{a}^{(n)} = (a_1, a_2, \cdots, a_n)$ is defined as the average distortion per symbol, which is

$$d(\mathbf{x}^{(n)}, \mathbf{a}^{(n)}) = \frac{1}{n}\sum_{i=1}^{n} d(x_i, a_i). \tag{1.1}$$

The distortion associated with the source code (f, g) is defined as the expected distortion between the source and its reproduction:

$$D = E[d(X^n, A^n)] = E[d(X^n, g(f(X^n)))]. \tag{1.2}$$

A rate distortion pair (R, D) is achievable if there exists a source code (f, g) with length n, rate R and distortion D for some positive integer n. The closure of the set of achievable rate distortion pairs (R, D) is called the rate distortion region. The rate-distortion function $R(D)$ is the minimum of all rates R such that (R, D) is in the rate distortion region for a given D. Conversely, the distortion-rate function $D(R)$ is the minimum of all distortions D such that (R, D) is in the rate distortion region for a given R.

Generally, the rate distortion region is very hard to determine. However, there are some results for a few sources and certain distortion measures. For a Gaussian memoryless source with variance σ^2, the distortion rate function is [38, 1, 8]:

$$D(R) = \sigma^2 2^{-2R}. \tag{1.3}$$

For a memoryless continuous-valued source with variance σ^2 and differential entropy $h(p)$, its distortion rate function with squared error measure is bounded by [38, 1, 8]:

$$\frac{1}{2\pi e}2^{2h(p)}2^{-2R} \leq D(R) \leq \sigma^2 2^{-2R}, \tag{1.4}$$

where the differential entropy $h(p)$ is determined by the source probability density function $p(x)$,

$$h(p) \triangleq -\int p(x)\log_2 p(x)dx. \tag{1.5}$$

For a memoryless Gaussian source with variance σ^2, we have:

$$h(p) = \frac{1}{2}\log(2\pi e\sigma^2). \tag{1.6}$$

Relation (1.4) implies that the Gaussian source is the most difficult source to compress. Among different sources with the same variance, the Gaussian source requires the largest number of bits to achieve the same distortion.

1.2.2 Multiple Description Rate Distortion Region

In MDC, each description has a rate and each combination of descriptions has a distortion (the distortion achieved by the joint decoder). The multiple description rate distortion region (MD region) for a particular source and distortion measure, in the case of K descriptions, is the closure of the set of simultaneously achievable K rates and $2^K - 1$ distortions in MDC.

The entire MD region has been known only for Gaussian memoryless sources with squared error measure for the case of two descriptions. The region was determined by Ozarow [31] and is referred to as the Ozarow MD region. Similarly to single description, the MD region for any continuous memoryless source with squared error measure can be bounded by the MD region for Gaussian source. In the case of two descriptions, the MD region is the closure of the sets of achievable rate distortion quintuples $(R_1, R_2, D_1, D_2, D_0)$, where R_1, R_2 denote the descriptions' rates, D_1, D_2 denote the individual distortions of the two descriptions (also called side distortions), and D_0 denotes the distortion achieved when the two descriptions are decoded jointly (also known as central distortion).

For a Gaussian memoryless source with unit variance and squared error measure, the Ozarow MD region is given by:

$$D_1 \geq 2^{-2R_1}, \tag{1.7}$$

$$D_2 \geq 2^{-2R_2}, \tag{1.8}$$

$$D_0 \geq 2^{-2(R_1+R_2)}\gamma, \tag{1.9}$$

where γ is defined as:

$$\gamma \triangleq \begin{cases} 1, & \text{if} \quad \Pi \leq \Delta \\ \frac{1}{1-(\sqrt{\Pi}-\sqrt{\Delta})^2}, & \text{if} \quad \Pi > \Delta \end{cases}$$

for

$$\Pi \triangleq (1 - D_1)(1 - D_2) \tag{1.10}$$

$$\Delta \triangleq D_1 D_2 - 2^{-2(R_1+R_2)}. \tag{1.11}$$

The MD region under the assumptions that $R_1 = R_2 = 1, D_1 = D_2$ is plotted

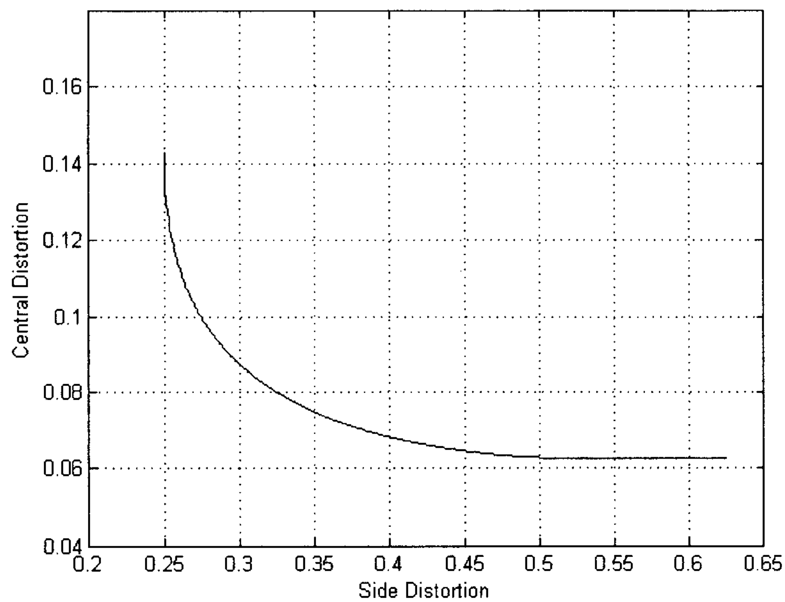


Figure 1.3: The Ozarow MD region for two balanced descriptions.

in Fig. 1.3. (1.9) implies that the central distortion must exceed the distortion rate minimum by the factor γ. When the side distortions are large, $\gamma = 1$ so the central distortion can reach the minimum. Otherwise, we have to compromise the central reconstruction for the good side reconstructions.

For more than two descriptions, an achievable region was provided in [33] for the *symmetric* case, where each description has the same rate, and the distortion constraint only depends on the number of descriptions available. Wang and Viswanath [51, 50] established a tight sum rate lower bound for certain cases with only two levels of distortion constraints. Tian, Mohajer and Diggavi [42] provided a novel lower bound for symmetric scalar Gaussian MDC with K levels of distortion constraints.

1.3 MDC Techniques

The main MD methods investigated in the literature are MD based on uneven erasure protection, MD quantization, MD correlating transforms, and MD coding with frames. This section briefly describes these MD techniques. More detailed information can be found in [19].

MD Based on Uneven Erasure Protection (MD-UEP). A method to achieve symmetric multiple descriptions, which has been used in image processing applications, is by combining a successively refinable (i. e., progressively refinable or scalable, or embedded) source code with uneven erasure protection [29, 34, 14, 40]. The scalable code stream is split into consecutive segments of decreasing importance. Then the source segments with decreasing importance are encoded with progressively weaker erasure protection channel codes. Further, the descriptions are formed across the channel codewords.

Very recently, [42] showed that regardless of the number of descriptions, the gap between the MD-UEP coding scheme and the optimal MD coding is less than 1.5 bits in individual description rate. This result suggests that MD-UEP is a very competitive approach, due to its simplicity and to the fact that the gap can be bounded by a small constant value.

MD Quantization. In MD quantization, the descriptions are generated using quantization. Vaishampayan proposed a design algorithm for fixed-rate MD scalar quantizer (MDSQ) for two descriptions [45], for the special case called *balanced* descriptions. Two descriptions are said to be balanced if they have the same rate, and achieve the same distortion when used individually. This algorithm is extended to entropy-constrained scalar quantizers in [47]. The design of MD lattice vector quantization was also addressed in [37, 48].

MD Correlating Transforms. In this framework the descriptions are generated by a linear transform with the purpose of correlating signals. Redundancy is added through the way transform coefficients are generated [49]. The statistical dependency between transform coefficients can be used to improve the estimation of the lost coefficients. The general approach is to multiply a transform matrix T to the vector of random variables. Let us consider an example as is shown in [49]. Assume X_1, X_2 are two independent, zero-mean Gaussian random variables with variances σ_1^2, σ_2^2, respectively, $\sigma_1^2 \neq \sigma_2^2$. We can form the descriptions (Y_1, Y_2) of (X_1, X_2) by

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \tag{1.12}$$

This way, $Y_1 = (X_1 + X_2)/\sqrt{2}$, $Y_2 = (X_1 - X_2)/\sqrt{2}$, and Y_1, Y_2 are correlated with correlation coefficient

$$E[Y_1 Y_2] = \frac{\sigma_1^2 - \sigma_2^2}{\sigma_1^2 + \sigma_2^2}. \tag{1.13}$$

If one description is lost, we can estimate the lost description to better quality than if X_1 and X_2 are directly used as descriptions. [20, 21] extend this simple technique to more general transforms and longer vectors.

MD Coding with Frames. The idea of quantized frame expansion [22, 23] is to left-multiply a source vector $x^{(N)}$ with length N by a matrix $F \in \mathbb{R}^{M \times N}, M > N$ to produce a vector of M transform coefficients. These coefficients are scalar quantized and partitioned into K sets, $K \leq M$, to form K descriptions. The quantized expansion coefficients are represented by $y = Q(Fx)$. The source vector can be reconstructed by solving a least-squares problem: $\arg\min_x \|y - Fx\|^2$. Improvement can be made by introducing more complicated reconstruction methods [7, 24, 35].

1.4 Contribution and Organization of this Thesis

In the most general setting, the MDC scheme generating K descriptions has $2^K - 1$ decoders. If each decoder needs to store some information to enable decoding, then the storage space requirement increases significantly with the number of descriptions. High storage space needs could be an issue, especially in applications where the memory resources are scarce (for example, transmission to mobile devices). Such applications motivate the study of MDC with reduced storage space decoder.

In this thesis we propose two techniques for symmetric MDC with reduced storage space decoder. One is MDSQ with linear joint decoders. The other is an improved MD-UEP scheme.

For MDSQ, in order to reduce the storage space at the decoder we need to compromise the decoding optimality. The traditional MDSQ stores a codebook for each decoder. The solution we propose is to store a codebook for each side decoder and a few joint decoders, and generate the other codebooks as linear combinations of the side codebooks during decoding. We address the problem of optimal design of such systems. The algorithm we propose is a generalized Lloyd algorithm, similar to the one introduced by Vaishampayan [45], to which we add an index assignment optimization step at each iteration. The challenge is in the step of decoder optimization, which is more complex than in the traditional design [45], since the codebooks can no longer be optimized separately. Fortunately, as we will show later, the problem turns out to be a convex quadratic optimization problem with a closed form solution (under some mild conditions).

MD-UEP is also an MD strategy of reduced storage space at the decoder. The second technique we propose is an improvement to the traditional MD-UEP. The main idea is to partition the set of samples into equal-sized subsets, and encode each subset separately using a successively refinable quantizer. Further, interleaved erasure protection codes of decreasing strength are applied across the sub-streams. We show that the proposed MD-UEP framework strictly outperforms the previous one. We evaluate the asymptotical performance using the expected distortion at the decoder as the performance measure, and compare it with the traditional MD-UEP. For a Gaussian memoryless source, the asymptotic improvement in performance can attain as much as 1.68 dB (for 3 descriptions and very low probability of description loss), with a tendency to decrease as the number of descriptions and the probability of description loss increase. This new MD-UEP technique was first presented in [15].

We also perform an experimental study of the proposed MDC techniques and compare them with traditional MDSQ and MD-UEP. Our tests show that the new MD-UEP technique is competitive to MDSQ.

The thesis is structured as follows. Chapter 2 presents the notations and definitions for scalar quantization and MDSQ, and introduces the generalized Lloyd algorithm for optimal design of MDSQ with more than two descriptions. Chapter 3 introduces the MDSQ with linear joint decoders and proposes an optimal design algorithm. Chapter 4 presents the improved MD-UEP scheme and its asymptotical performance analysis. We show the experimental results and the comparison between the two techniques (also with the traditional MDC schemes) in Chapter 5. Chapter 6 concludes the thesis.

Chapter 2

Definition and Optimal Design of MDSQ

In this chapter, we present the notations and definitions for scalar quantization and MDSQ, as well as the generalized Lloyd algorithm for optimal design of MDSQ with more than two descriptions. Since this algorithm can only guarantee a locally optimal solution, the choice of the initial encoder, which includes an index assignment, influences its performance. Therefore, we reserve the last section to the discussion of the index assignment.

2.1 Scalar Quantization

A scalar quantizer (SQ) consists of an encoder-decoder pair (f, g). Given a continuous random variable X, with the probability density function (pdf) $f_X(x)$, the encoder f

maps any sample x to an index i in the set $\{1, 2, \cdots, N\}$, and the decoder g maps any index $i \in \{1, 2, \cdots, N\}$, to a reconstruction value (or reproduction value) $a_i \in \mathbb{R}$. The set of all reconstruction values is called the codebook. The quantizer cells are the sets $A_1, A_2, \cdots, A_N$, where $A_i = \{x : f(x) = i\}$, $i = 1, 2, \cdots, N$. They form a partition of $\mathbb{R}$, in other words, they satisfy

$$\bigcup_{i=1}^{N} A_i = \mathbb{R}, \tag{2.1}$$

$$A_i \bigcap A_j = \emptyset \quad \text{for} \quad i \neq j. \tag{2.2}$$

Such a quantizer is called an N-level SQ. The rate R of a quantizer is defined as the average number of bits used to represent the indices $i \in \{1, 2, \cdots, N\}$. SQ's can be divided into two classes: fixed-rate SQ and entropy-constrained SQ. In fixed-rate SQ, the same number of bits is used to represent all indices, hence the rate is $r = \lceil \log_2 N \rceil$. On the other hand, in entropy-constrained SQ the indices are entropy-coded, therefore the number of bits used to represent each index depends on the entropy of the corresponding cell. The rate of entropy-constrained SQ is the sum of entropies of all cells, i.e., $r = \sum_{i=1}^{N} -P_i \log_2 P_i$, where $P_i = \int_{A_i} f_X(x)dx$. In this work, only fixed-rate SQ is considered, therefore, the specification of fixed-rate will be omitted.

The performance of an SQ is measured by the distortion between the input signal and its reconstruction. In this work, we use the squared error distortion measure, hence, the expected distortion of the source reconstruction is

$$E(d) = \sum_{i=1}^{N} \int_{A_i} (x - a_i)^2 f_X(x) dx. \tag{2.3}$$

The objective of optimal SQ design is to find an encoder-decoder pair (f, g) which minimizes the expected distortion $E(d)$.

A popular method for optimal SQ design is the Lloyd algorithm [27, 28]. This is an iterative algorithm which optimizes the encoder and the decoder one at a time. It begins with an initial encoder, and proceeds iteratively, each iteration consisting of the following two steps: (1) fix the encoder and optimize the decoder, and (2) fix the decoder and optimize the encoder. The value of $E(d)$ decreases at each iteration, and since it is bounded below by zero, the sequence of expected distortions is guaranteed to converge to a local minimum.

Decoder Optimization. At this step the encoder is fixed and the decoder is optimized. The optimal codebook must satisfy the *centroid condition* [18]. In other words, the optimal reconstruction value a_i, for the ith cell, $1 \leq i \leq N$, has to be the center of mass (centroid) of the region A_i, i.e.,

$$a_i = E[X|X \in A_i] = \int_{A_i} x f_{X|A_i}(x) dx = \frac{\int_{A_i} x f_X(x) dx}{\int_{A_i} f_X(x) dx}. \tag{2.4}$$

Encoder Optimization. In this step the decoder is fixed and the encoder is optimized. The encoder which minimizes (2.3) can be obtained by mapping any input sample x to the reproduction value which incurs the minimum distortion, in other words, to the closest reproduction value. This condition is known as the *nearest* *neighbor condition* [18]. Formally, it requires that A_i consists of all input samples which are closer to a_i than to any other $a_j, j \neq i$, i.e.,

$$f(x) = i \quad \text{only if} \quad (x - a_i)^2 \leq (x - a_j)^2 \quad \text{for all} \quad j \neq i. \tag{2.5}$$

The above condition implies that all quantizer cells are intervals, hence the encoder partition is specified by the boundaries (thresholds) between consecutive intervals. Let $\mathbf{t} = (t_0, t_1, \cdots, t_N)$, denote the vector of thresholds, such that $A_i = \{x | t_{i-1} \leq x < t_i\}$, for $2 \leq i \leq N$, and $A_1 = \{x | t_0 < x < t_1\}$, with $t_0 = -\infty, t_N = \infty$. Then the nearest neighbor condition implies that

$$t_i = \frac{1}{2}(a_i + a_{i+1}), \tag{2.6}$$

for all $1 \leq i \leq N - 1$.

Lloyd's algorithm for optimal SQ design converges to a locally optimal solution, in general. The globally optimal solution is guaranteed only for some classes of pdf's, for example, log-concave pdf's [17, 43, 26, 44]. Globally optimal SQ design algorithms for discrete distributions, which run in polynomial time in the size of the input alphabet, have also been proposed [5, 39, 53, 52].

2.2 MDSQ

2.2.1 Definition and Notations

A K-description MDSQ consists of K side encoders $f_1, \cdots, f_K$, and $2^K - 1$ decoders, each decoder $g_{\mathcal{L}}$ corresponding to a non-empty subset $\mathcal{L}$ of descriptions, $\mathcal{L} \subseteq \mathcal{K} =$

$\{1, \cdots, K\}$. Recall that we use the term central decoder to describe the decoder corresponding to the whole set of descriptions $\mathcal{K}$, and use the term side decoders to describe the decoders corresponding to the sets which have only one description. The decoders other than the side decoders are referred to as joint decoders. Therefore, among the $2^K - 1$ component decoders of an MDSQ, there are K side decoders, one central decoder, and $2^K - 1 - K$ joint decoders. Note that the central decoder is a special case of a joint decoder.

Given a source sample x, first the side encoders map x into some K-tuple $(i_1, \cdots, i_K) \in \mathcal{I}_\mathcal{K}$, where i_k is generated by the kth encoder, $1 \leq k \leq K$, and $\mathcal{I}_\mathcal{K}$ denotes the set of all K-tuples generated by all side encoders. Formally, the encoder f_k is a function $f_k : \mathbb{R} \to \{1, \cdots, M_k\}$, where M_k is some positive integer, and defines the number of cells in the kth side encoder. The K side encoders generate a partition $\mathcal{A}$ of $\mathbb{R}, \mathcal{A} = \{A_{i_1,\cdots,i_K} | (i_1, \cdots, i_K) \in \mathcal{I}_K\}$, where $A_{i_1,\cdots,i_K} = \{x | f_1(x) = i_1, \cdots, f_K(x) = i_K\}$. The partition $\mathcal{A}$ is called the central partition.

Each index i_k is transmitted over the kth channel. Each channel has some probability of breaking down, so at the receiver, there are two kinds of situations with respect to each description: either the channel works properly and the received index is correct, or the channel breaks down and nothing is received.

Let $\mathcal{L} = \{l_1, \cdots, l_k, \cdots, l_s\} \subseteq \mathcal{K}$ denote a subset of descriptions, where $1 \leq k \leq s$, and $1 \leq s \leq K$. Assume only those descriptions in the subset $\mathcal{L}$ are received at the decoder. Then the decoder $g_\mathcal{L}$ corresponding to the arrived descriptions is used to reconstruct the source sample. The decoder $g_{\mathcal{L}}$ maps each s-tuple $(i_{l_1}, \cdots, i_{l_s})$ to some reconstruction level $a^{\mathcal{L}}_{i_{l_1}, \cdots, i_{l_s}} \in \mathbb{R}$, where $1 \leq i_{l_k} \leq M_k$, for all $1 \leq k \leq s$.

The set of K side encoders $(f_1, \cdots, f_K)$ is denoted by $\mathbf{f}$ and is called the encoder of the MDSQ. The set of decoders $(g_{\mathcal{L}})_{\mathcal{L} \subseteq \mathcal{K}}$ is denoted by $\mathbf{g}$ and is called the decoder of the MDSQ. An MDSQ is completely determined by the encoder $\mathbf{f}$ and the decoder $\mathbf{g}$. Note that $\mathbf{f}$ is completely specified by the central partition $\mathcal{A}$ and the assignment of K-tuples to the set of cells in the central partition, which is called the index assignment (IA). In this work, we are only interested in the case of fixed-rate MDSQ, where the rate of each side encoder, $r_k, 1 \leq k \leq K$, is the number of bits used to represent any index generated by the kth side encoder. Hence $r_k = \lceil \log_2 M_k \rceil$.

The expected distortion of the source reconstruction when only a subset $\mathcal{L}$ of descriptions are received at the decoder is

$$d_{\mathcal{L}} = \sum_{(i_{l_1}, \cdots, i_{l_s}) \in \mathcal{I}_{\mathcal{L}}} \int_{\bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})} (x - a^{\mathcal{L}}_{i_{l_1}, \cdots, i_{l_s}})^2 f_X(x) dx, \tag{2.7}$$

where $f_{l_k}^{-1}(i_{l_k})$ denotes the set of values which are mapped by f_{l_k} to the index i_{l_k}. Here $\mathcal{I}_{\mathcal{L}}$ denotes the set of all possible s-tuples of indices corresponding to $\mathcal{L}$. The problem of optimal MDSQ design could be formulated as the problem of minimizing the central distortion $d_{\mathcal{K}}$ with constraints imposed on the rates and the other component distortions:

$$\begin{aligned} \text{minimize} \quad & d_{\mathcal{K}} \\ \text{subject to} \quad & d_{\mathcal{L}} \leq D_{\mathcal{L}}, \text{for all} \quad \mathcal{L} \subset \mathcal{K}, \\ & r_k = R_k, k = 1, 2, \cdots, K. \end{aligned} \tag{2.8}$$

where $D_{\mathcal{L}}, \mathcal{L} \subseteq \mathcal{K}$, and $R_k, 1 \leq k \leq K$, are some fixed values. For fixed-rate MDSQ, the constraints on the rates can be satisfied by fixing the values of $M_1, \cdots, M_K$, such that $M_k = 2^{R_k}, 1 \leq k \leq K$.

The constraints on distortions can be eliminated by using the Lagrangian relaxation method. Precisely, the Lagrangian functional is defined as

$$\mathcal{L}(\mathbf{f}, \mathbf{g}, \{\lambda_{\mathcal{L}}\}_{\mathcal{L}}) = \sum_{\mathcal{L} \subseteq \mathcal{K}} \lambda_{\mathcal{L}} d_{\mathcal{L}}, \tag{2.9}$$

where $\lambda_{\mathcal{L}} \geq 0, \mathcal{L} \subseteq \mathcal{K}$, are the Lagrangian multipliers, and $\lambda_{\mathcal{K}} = 1$. Then the problem is converted to the unconstrained optimization problem:

$$\text{minimize}_{\mathbf{f},\mathbf{g}} \quad \mathcal{L}(\mathbf{f}, \mathbf{g}, \{\lambda_{\mathcal{L}}\}_{\mathcal{L}}). \tag{2.10}$$

If there are Lagrangian multipliers such that the solution $(\mathbf{f}^*(\lambda_{\mathcal{L}}), \mathbf{g}^*(\lambda_{\mathcal{L}}))$ to the problem (2.10) satisfies the constraints in (2.8) with equality, then $(\mathbf{f}^*(\lambda_{\mathcal{L}}), \mathbf{g}^*(\lambda_{\mathcal{L}}))$ is a solution to (2.8). It is guaranteed that such Lagrangian multipliers exist only if $(\{R_k\}_k, \{D_{\mathcal{L}}\}_{\mathcal{L}})$ is on the lower convex hull of the operational MD region. Therefore this conversion is able to solve optimally only some instances of the problem, while for the others it provides a good approximation.

Another variant for formulating the problem of optimal MDSQ design is as the minimization of the overall expected distortion at the receiver with constraints imposed on rates:

$$\begin{aligned} \text{minimize} \quad & \sum_{\mathcal{L} \subseteq \mathcal{K}} p_{\mathcal{L}} d_{\mathcal{L}} \\ \text{subject to} \quad & r_k = R_k, k = 1, 2, \cdots, K, \end{aligned} \tag{2.11}$$

where $p_{\mathcal{L}}$ is defined as the probability that only those descriptions in subset $\mathcal{L}$ are received. If we set the Lagrangian multipliers in (2.9) to the values of $p_{\mathcal{L}}$'s in (2.11), then problem (2.11) becomes equivalent to (2.10).

In the special case of symmetric descriptions, the rates of all descriptions are the same, i.e., $r_1 = \cdots = r_k = r$, and $p_{\mathcal{L}}$ only varies with the number of received descriptions. In other words, $p_{\mathcal{L}_i} = p_{\mathcal{L}_j} = p(s)$ if $|\mathcal{L}_i| = |\mathcal{L}_j| = s$.

The first algorithm to solve problem (2.10) was proposed by Vaishampayan [45]. He considered the case of two descriptions, and his approach is a generalization of Lloyd algorithm for SQ design. This algorithm begins with an initial encoder, and proceeds iteratively, each iteration consisting of the following two steps: (1) fix the encoder and optimize the decoder, and (2) fix the decoder and optimize the encoder. The value of the Lagrangian decreases at each iteration, and since it is bounded below by zero, the sequence of Lagrangians is guaranteed to converge to a local minimum. In general, the globally optimal solution is not guaranteed but for some special cases (e.g., convex cells and log-concave pdfs [12]). In [30, 11, 13] globally optimal algorithms for MDSQ design are proposed for discrete distributions, under the assumption that all cells are convex.

2.2.2 Generalized Lloyd Algorithm for Optimal MDSQ Design

Vaishampayan's algorithm [45] can be extended in a straightforward manner to the general K-description case, $K \geq 2$ [9]. In this section we describe this extension. Here we use the second approach (2.11) for formulating the problem of optimal MDSQ design. By substituting (2.7) in (2.11) the optimization problem becomes

$$\text{minimize}_{\mathbf{f},\mathbf{g}} \sum_{\mathcal{L} \subseteq \mathcal{K}} p_{\mathcal{L}} \sum_{(i_{l_1},\cdots,i_{l_s}) \in \mathcal{I}_{\mathcal{L}}} \int_{\bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})} (x - a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}})^2 f_X(x) dx \tag{2.12}$$

Decoder optimization. When the encoder is fixed, the decoder can be optimized by separately minimizing each term in the summation in (2.12). This turns into a simpler problem of separately minimizing each integral, leading to the solution:

$$a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}} = E[X | X \in \bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})] = \frac{\int_{\bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})} x f_X(x) dx}{\int_{\bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})} f_X(x) dx} \tag{2.13}$$

Encoder Optimization. At this step the decoder is fixed and the encoder is optimized. Here it is useful to rewrite the expression in (2.12) into the equivalent form:

$$\sum_{(i_1,\cdots,i_K) \in \mathcal{I}_{\mathcal{K}}} \int_{A_{i_1,\cdots,i_K}} [\sum_{\mathcal{L} \subseteq \mathcal{K}} p_{\mathcal{L}} (x - a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}})^2 f_X(x)] dx, \tag{2.14}$$

where $l_1, \cdots, l_s$, denote the elements of $\mathcal{L}$. The purpose of this step is to design the sets $A_{i_1,\cdots,i_K}$ for all $(i_1, \cdots, i_K) \in \mathcal{I}_{\mathcal{K}}$, such that they form a partition of $\mathbb{R}$ and (2.14) is minimized. A sufficient condition for minimizing (2.14) is

$$A_{i_1,\cdots,i_K} = \{x \in \mathbb{R} | \sum_{\mathcal{L} \subseteq \mathcal{K}} p_{\mathcal{L}}(x - a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}})^2 \leq \sum_{\mathcal{L} \subseteq \mathcal{K}} p_{\mathcal{L}}(x - a^{\mathcal{L}}_{i'_{l_1},\cdots,i'_{l_s}})^2, \quad \text{for all} \quad (i'_1, \cdots, i'_K) \in \mathcal{I}_{\mathcal{K}} - \{(i_1, \cdots, i_K)\}\}. \tag{2.15}$$

Further, by denoting

$$\alpha_{i_1,\cdots,i_K} = \sum_{\mathcal{L} \subseteq \mathcal{K}} p_{\mathcal{L}} a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}}$$
$$\beta_{i_1,\cdots,i_K} = \sum_{\mathcal{L} \subseteq \mathcal{K}} p_{\mathcal{L}} (a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}})^2, \tag{2.16}$$

we can write (2.15) into another form which is similar to the one given in [45]:

$$A_{i_1,\cdots,i_K} = \{x \in \mathbb{R} | 2\alpha_{i_1,\cdots,i_K} x - \beta_{i_1,\cdots,i_K} \geq 2\alpha_{i'_1,\cdots,i'_K} x - \beta_{i'_1,\cdots,i'_K}, \quad \text{for all} \quad (i'_1, \cdots, i'_K) \in \mathcal{I}_{\mathcal{K}} - \{(i_1, \cdots, i_K)\}\}. \tag{2.17}$$

It is clear that $A_{i_1,\cdots,i_K}$ is either an interval or an empty set. Let us order the elements of $\mathcal{I}_{\mathcal{K}}$ in increasing order of $\alpha_{i_1,\cdots,i_K}$. If for two distinct K-tuples $(i_1, \cdots, i_K)$ and $(i'_1, \cdots, i'_K)$, we have $\alpha_{i_1,\cdots,i_K} = \alpha_{i'_1,\cdots,i'_K}$, then either 1)$\beta_{i_1,\cdots,i_K} \neq \beta_{i'_1,\cdots,i'_K}$ or 2)$\beta_{i_1,\cdots,i_K} = \beta_{i'_1,\cdots,i'_K}$. In the first case, if $\beta_{i_1,\cdots,i_K} < \beta_{i'_1,\cdots,i'_K}$, then the set $A_{i'_1,\cdots,i'_K}$ is empty, hence the K-tuple $(i'_1, \cdots, i'_K)$ should be eliminated from $\mathcal{I}_{\mathcal{K}}$. Conversely, if $\beta_{i_1,\cdots,i_K} > \beta_{i'_1,\cdots,i'_K}$, then $(i_1, \cdots, i_K)$ should be eliminated from $\mathcal{I}_{\mathcal{K}}$. We can conclude that for the first case, one K-tuple has to be eliminated from the index set $\mathcal{I}_{\mathcal{K}}$. For the second case, we have $A_{i_1,\cdots,i_K} = A_{i'_1,\cdots,i'_K}$, hence one of the two K-tuples can also be eliminated from $\mathcal{I}_{\mathcal{K}}$. Therefore after removing from $\mathcal{I}_{\mathcal{K}}$ the K-tuples with identical $\alpha_{i_1,\cdots,i_K}$'s, $\mathcal{I}_{\mathcal{K}}$ has the property that for any two distinct K-tuples in $\mathcal{I}_{\mathcal{K}}$, we have $\alpha_{i_1,\cdots,i_K} \neq \alpha_{i_1,\cdots,i_K}$.

Let us order the K-tuples in $\mathcal{I}_K$ in increasing order of $\alpha_{i_1,\cdots,i_K}$. Now consider the mapping $\pi : \mathcal{I}_K \to \{1, 2, \cdots, N\}$ which maps each K-tuple $(i_1, \cdots, i_K)$ to an integer representing its position in this ordering. Hence π is characterized by the property that $\pi((i_1, \cdots, i_K)) < \pi((i'_1, \cdots, i'_K))$ if and only if $\alpha_{i_1,\cdots,i_K} < \alpha_{i'_1,\cdots,i'_K}$. Further we use the notation α_ℓ, β_ℓ and A_ℓ, respectively, as short forms for $\alpha_{i_1,\cdots,i_K}, \beta_{i_1,\cdots,i_K}$ and $A_{i_1,\cdots,i_K}$, respectively, where $\ell = \pi((i_1, \cdots, i_K))$. Also, for $m < n$, denote

$$x_{m,n} = \frac{\beta_n - \beta_m}{2(\alpha_n - \alpha_m)}. \tag{2.18}$$

Then we obtain

$$\begin{aligned} A_1 &= (-\infty, \min_{1<n\leq N} x_{1,n}], \\ A_N &= [\max_{1\leq \ell<N} x_{\ell,N}, \infty), \quad \text{and} \\ A_i &= [\max_{1\leq \ell<i} x_{\ell,i}, \min_{i<n\leq N} x_{i,n}], \text{ for } \quad 1 < i < N, \end{aligned} \tag{2.19}$$

with the convention that $[b, a] = \emptyset$ if $b > a$.

The cells in the optimal central partition can be formed directly by solving (2.19) in $O(N^2)$ time. Vaishampayan uses a more ingenious method to identify the empty sets, thus being able to decrease the amount of computations for some inputs. Very recently, a more efficient algorithm which runs in $O(N)$ time has been proposed in [9].

2.3 Index Assignment

Because Vaishampayan's algorithm can only guarantee a locally optimal solution, the choice of the initial encoder influences its performance. The problem of finding a good initial encoder is strongly related to the problem of finding a good IA.

It was shown in [45] that for an optimal MDSQ, the cells in the central partition must be convex in the case of squared error distortion measure. Therefore, the cells in the central partition are intervals, while those in the side partitions are unions of possibly disconnected intervals. The central partition can be described by a vector of thresholds $\mathbf{t} = (t_0, t_1, \cdots, t_N)$, where $t_0 = -\infty$, $t_N = \infty$ and each interval $(t_{\ell-1}, t_\ell]$ is a cell in the central partition. The IA can be formally defined as an one-to-one mapping h from the set of indices ℓ of each interval $(t_{\ell-1}, t_\ell]$, $1 \leq \ell \leq N$, to the set $\mathcal{I}_\mathcal{K}$ of K-tuples $(i_1, \cdots, i_K)$, hence $h : \{1, \cdots, N\} \rightarrow \mathcal{I}_\mathcal{K}$, where $\mathcal{I}_\mathcal{K} \subseteq \{1, \cdots, M_1\} \times \cdots \times \{1, \cdots, M_k\}$. Precisely, $h(\ell) = (i_1, \cdots, i_K)$ means that $(t_{\ell-1}, t_\ell] = A_{i_1, \cdots, i_K}$. The vector of thresholds $\mathbf{t}$ and the IA together fully describe the encoder.

Note that when designing the IA, there are two aspects to be taken into consideration: 1) the choice of the subset $\mathcal{I}_\mathcal{K} \subseteq \{1, \cdots, M_1\} \times \cdots \times \{1, \cdots, M_k\}$, and 2) the mapping h. In [45], good IA's are proposed for the case of two balanced descriptions.

In order to shed some light on the importance of the IA, we give an example to show how the central and side distortions can be influenced by IA in the case of two descriptions. Assume the source samples are uniformly distributed over the interval

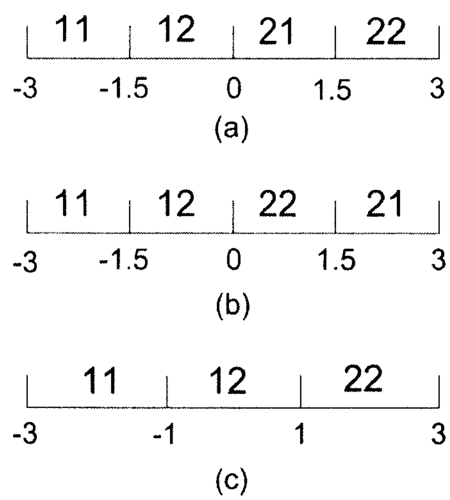


Figure 2.1: Three MDSQ's for a uniformly distributed source.

$[-3, 3]$ and that $r_1 = r_2 = 1$ bit per source sample. Consider the three MDSQ's illustrated in Fig. 2.1 with different central partition and IA. The two-digit number in each central cell denotes this cell's index pair: the first digit is the index generated by side encoder f_1, and the second digit is the index generated by side encoder f_2. Table 2.1 gives the central and side distortions of the three quantizers in Fig. 2.1.

Table 2.1: Central and Side Distortions

	$E(d^{12})$	$E(d^1)$	$E(d^2)$
(a)	0.1876	0.75	2.4376
(b)	0.1876	0.75	3
(c)	0.3333	1	1

Both (a) and (b) have the same central partition, hence their central distortions are the same. However, case (a) is superior to (b) because it has lower side distortion, which suggests that for the same set of index pairs, different mapping may result in different distortions. Then let us compare (a) with (c). (c) has 1 less central cell than (a), hence higher central distortion, but (c) has much lower average side distortion than (a). This example suggests that the number of cells (N) in the central partition induces the trade-off between the central and side distortions: as the number of central cells decreases, the central distortion will increase, and the average of side distortions will decrease.

The design of IA for the case of two balanced descriptions, proposed by Vaishampayan [45], includes two parts: the selection of the set of index pairs and the mapping of those pairs to the central cells. Assume the rate for each description is $r = \lceil \log_2 M \rceil$, and the number of central cells is $N (N \leq M^2)$. Let $\ell \min_p(n), p = 1, 2$, be the minimum value of the interval index ℓ that is mapped to an index pair whose pth element is equal to n. Let $\ell \max_p(n), p = 1, 2$ be the maximum value of the interval index ℓ that is mapped to an index pair whose pth element is equal to n. Then we define the spread of the nth cell of the pth side encoder

$$s_p(n) = \ell \max_p(n) - \ell \min_p(n) + 1, \quad p = 1, 2. \tag{2.20}$$

It has been proposed in [45] that to design a good IA, the spread $s_p(n)$ needs to be minimized, assuming that it is constant with respect to n and p. The set of index pairs $\mathcal{I}_{\mathcal{K}}$ can be identified with the elements of an $M \times M$ matrix (the pair (i_1, i_2)

corresponds to the element on the i_1th row and i_2th column). Then, the choice of $\mathcal{I}_\mathcal{K}$ can be regarded as selecting a set of elements of the matrix. Let us assume that the index pairs $(1,1),(2,2),\cdots,(M,M)$ are always in the set, and they are assigned to central cells from left to right. This way, the index pairs $(1,1),(2,2),\cdots,(M,M)$ form the main diagonal of the $M \times M$ matrix. To minimize the spread, we would exhaust all elements on a diagonal which lies closer to the main diagonal before moving to a diagonal which lies further away from the main diagonal. Hence we select a set of index pairs which lie on the main diagonal and on the $2k$ diagonals closest to the main diagonal. The mapping of the selected set of index pairs to the central cells could be interpreted as deciding a scanning sequence of this set to minimize the spread. Two families of IA's are presented in [45] for the case of two balanced descriptions: modified nested IA and modified linear IA. Fig. 2.2 shows examples proposed in [45] of these two IA families with $k = 3$, $M = 8$.

For the case of more descriptions, there is not a well-developed methodology for constructing good IA's. The work [2] presents an IA for MDSQ with more descriptions, but only when the side and central distortions are of interest (no other joint quantizers' distortions). [41] introduced a new method for MDSQ design, called the sequential design of MDSQ. However, good IA's for more than two descriptions are not yet well understood.

1	3	5	7				
2	8	10	12	14			
4	9	15	17	19	21		
6	11	16	22	23	25	27	
	13	18	24	29	30	32	34
		20	26	31	36	37	39
			28	33	38	41	43
				35	40	42	44

(a) Modified nested IA

1	2	4	7				
3	5	8	11	14			
6	9	12	15	18			
10	13	16	19	21	24	28	
	17	20		23	27	31	35
			22	26	30	34	38
			25	29	33	37	40
				32	36	39	41

(b) Modified linear IA

Figure 2.2: IA examples for k=3, M=8 [45].

Chapter 3

Symmetric MDSQ with Linear Joint Decoders

In this chapter, we first introduce the motivation of MDSQ with linear joint decoders and present the overview of the proposed optimal design algorithm. Next the step of decoder optimization is detailed in Section 3.3, and the complexity reduction in the encoder optimization step is analyzed in Section 3.4. In the last section, we describe the step of IA optimization.

3.1 Motivation

An MDSQ with K descriptions has $2^K - 1$ decoders, each with its own codebook. Moreover, the size of each codebook increases exponentially with the number of corresponding descriptions. For example, if the rate of each side quantizer is R, then the number of cells in a joint quantizer of $s, 2 \leq s \leq K$, descriptions is at most 2^{sR}. Thus, the total number of reconstruction values can amount to

$$\sum_{s=1}^{K} \binom{K}{s} (2^R)^s = (1 + 2^R)^K - 1. \tag{3.1}$$

As the number K of descriptions increases, storing all these values becomes an issue, especially in applications where the memory resources are scarce (for example, mobile devices).

In order to reduce the storage space at the decoder we need to compromise the decoding optimality. The solution we propose is to store a codebook for each side decoder and a few joint decoders, and generate the other codebooks as linear combinations of the side codebooks. Precisely, if $\mathcal{L} = \{l_1, \cdots, l_k, \cdots, l_s\}$ is a subset of descriptions for which the codebook is not stored, then its reconstruction values are computed according to

$$a^{\mathcal{L}}_{i_{l_1}, \cdots, i_{l_s}} = \frac{1}{s} \sum_{k=1}^{s} a^{(l_k)}_{i_{l_k}}, \tag{3.2}$$

for any $(i_1, \cdots, i_K) \in \mathcal{I}_K$. We call such an MDSQ, MDSQ with linear joint decoders, and use the acronym L-MDSQ. We refer to the MDSQ with all codebooks stored as Opt-MDSQ. As it will become apparent shortly, for L-MDSQ the joint decoders whose codebooks are stored can be individually optimized as in the Opt-MDSQ, while each of the other joint decoders will be individually suboptimal. Therefore, in order to increase the performance of the L-MDSQ, the stored codebooks should correspond to the subsets of descriptions with the highest probabilities of being received. In the case of symmetric, independent channels with probability of channel failure $q < 0.5$, the probability of receiving a particular set of descriptions increases with the set size. Therefore, guided by the principle outlined above we will store the codebooks of all subsets of at least $K_0 + 1$ descriptions, for a fixed value $K_0, 1 \le K_0 \le K$, and will use the linear rule (3.2) to generate the codebooks for subsets of 2 up to K_0 descriptions. The choice of the value K_0 should be done such that to strike a balance between the codebook storage needs and the MDSQ performance.

In the rest of this chapter we address the design of optimal symmetric L-MDSQ. As for general MDSQ, an L-MDSQ is called symmetric if the rates of all side encoders are equal and for any two sets with the same number of descriptions, the probability of being received is the same. We will denote by M the number of cells of each side encoder, and by $p(s)$ the probability of receiving some set of s descriptions, $0 \le s \le K$. For example, when the descriptions are transmitted over independent channels with the same failure rate q, we have $p(s) = (1-q)^s q^{K-s}$.

3.2 Optimal Design Algorithm for Symmetric L-MDSQ

We formulate the problem of optimal symmetric L-MDSQ design as the problem of minimizing its expected distortion $E(d)$. The expected distortion can be written as

$$E(d) = \sum_{s=0}^{K} p(s) \times \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} \sum_{(i_{l_1},\cdots,i_{l_s})\in\mathcal{I}_\mathcal{L}} \int_{\bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})} (x - a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}})^2 f_X(x)dx. \quad (3.3)$$

The design algorithm we propose is a generalized Lloyd algorithm to which we add an IA optimization step at each iteration. Precisely, the algorithm starts from an initial encoder and then proceeds to iterate over the following three steps: (1) decoder optimization, (2) encoder optimization, (3) IA optimization. After each iteration the cost function decreases, and since the value of the cost function is bounded below by zero, the sequence of the values is guaranteed to converge to a local minimum.

Encoder optimization step. At this step the decoder is fixed and the central partition is optimized. In principle, the same approach as in Opt-MDSQ design (described in Section 2.2.2) can be used here as well. However, we can reduce the computational cost by exploiting the fact that some joint codebooks are formed using linear combinations of the side codebooks. A detailed discussion of the complexity reduction is addressed in Section 3.4.

Decoder optimization step. Here the encoder is kept fixed and the stored codebooks are optimized. This step is more complex than in the design of Opt-MDSQ since some joint codebooks can no longer be optimized separately. Fortunately, as we show in Section 3.3, the problem turns out to be a convex quadratic optimization problem with a closed form solution (under certain conditions).

IA optimization step. At this step the decoder and the sets in the central partition are kept fixed, while the IA is optimized. We add this new step after the encoder optimization, to overcome the problem that good IA's for more than two descriptions are yet to be found. This step can be applied to both L-MDSQ and Opt-MDSQ. The detailed treatment of this step follows in Section 3.5.

3.3 Decoder Optimization Step

At this step the central partition and the IA are fixed, and the MDSQ codebook is optimized. In other words, the goal at this step is to choose the reconstruction values for the side codebooks and for the joint codebooks corresponding to at least $K_0 + 1$ descriptions, such that the expected distortion to be minimized. To this end, it is useful to write the expected distortion as follows $E(d) = E_1 + E_2$ where E_1 is the contribution due to decoding subsets of at most K_0 descriptions, while E_2 is the contribution due to decoding subsets of at least $K_0 + 1$ descriptions. Using (3.3) and (3.2) we obtain

$$E_1 = \sum_{s=1}^{K_0} p(s) \times \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} \sum_{(i_{l_1},\cdots,i_{l_s})\in\mathcal{I}_{\mathcal{L}}} \int_{\bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})} (x - \frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2 f_X(x)dx, \quad (3.4)$$

$$E_2 = \sum_{s=K_0+1}^{K} p(s) \times \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} \sum_{(i_{l_1},\cdots,i_{l_s})\in\mathcal{I}_{\mathcal{L}}} \int_{\bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})} (x - a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}})^2 f_X(x)dx. \quad (3.5)$$

Note that here we ignore the term corresponding to no description received because this is a constant for the optimization problem. It is clear that E_1 and E_2 can be optimized separately. Moreover, E_2 can be optimized by independently minimizing each integral in the summation, which leads to

$$a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}} = E[X|X \in \bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})] = \frac{\int_{\bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})} x f_X(x)dx}{\int_{\bigcap_{k=1}^{s} f_{l_k}^{-1}(i_{l_k})} f_X(x)dx}. \quad (3.6)$$

We are left now with the greater challenge of finding the side codebooks which minimize the expression E_1. Fortunately, as we will show next, this is a convex quadratic optimization problem. Precisely, we will prove that, if $\mathbf{y}$ denotes the MK dimensional vector which consists of all reconstruction values of the side codebooks:

$$\mathbf{y} = (a_1^{(1)}, a_2^{(1)}, \cdots, a_M^{(1)}, \cdots, a_1^{(j)}, \cdots, a_{i_j}^{(j)}, \cdots, a_M^{(j)}, \cdots, a_1^{(K)}, \cdots, a_M^{(K)})^T, \qquad (3.7)$$

then E_1 can be written as

$$E_1 = \mathbf{y}^T B \mathbf{y} + \mathbf{u}^T \mathbf{y} + r, \qquad (3.8)$$

where r is a real value, $\mathbf{u}$ is a constant KM dimensional vector and B is a constant $KM \times KM$ positive semi-definite matrix.

In order to prove this claim we first expand each integral in (3.4) as the summation of integrals over cells in the central partition. Recall that the sets in the central partition are $A_{i_1,\cdots,i_K} = \{x | f_1(x) = i_1, \cdots, f_K(x) = i_K\}$, for all $(i_1, \cdots, i_K) \in \mathcal{I}_{\mathcal{K}}$. In the sequel we will use the notation $\mathbf{i}$ for a K-tuple $(i_1, \cdots, i_K)$. Thus, we have

$$E_1 = \sum_{s=1}^{K_0} p(s) \times \sum_{\mathcal{L} \in \mathcal{K}, |\mathcal{L}|=s} \sum_{\mathbf{i} \in \mathcal{I}_{\mathcal{K}}} \int_{A_{\mathbf{i}}} (x - \frac{1}{s} \sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2 f_X(x) dx. \qquad (3.9)$$

Next we will rewrite conveniently each of the above integrals. For this we make first the following notations

$$c_{\mathbf{i}} = \int_{A_{\mathbf{i}}} x f_X(x) dx \qquad (3.10)$$

and

$$P_{\mathbf{i}} = \int_{A_{\mathbf{i}}} f_X(x) dx. \qquad (3.11)$$

Then we have

$$\begin{aligned}
&\int_{A_i} (x - \frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2 f_X(x)dx \\
&= \int_{A_i} \Big(x^2 - 2\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)} x + (\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2\Big) f_X(x)dx \\
&= \int_{A_i} x^2 f_X(x)dx - 2\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)} \int_{A_i} x f_X(x)dx + (\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2 \int_{A_i} f_X(x)dx \\
&= \int_{A_i} x^2 f_X(x)dx - 2\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)} c_i + (\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2 P_i
\end{aligned} \tag{3.12}$$

By substituting (3.12) in (3.9) we obtain that

$$E_1 = \sum_{s=1}^{K_0} p(s) \times \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} \Big(\sum_{i\in\mathcal{I}_\mathcal{K}} \int_{A_i} x^2 f_X(x)dx - 2\sum_{i\in\mathcal{I}_\mathcal{K}} \frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)} c_i + \sum_{i\in\mathcal{I}_\mathcal{K}} (\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2 P_i\Big), \tag{3.13}$$

Denote further

$$\mathcal{M} = \sum_{i\in\mathcal{I}_\mathcal{K}} \int_{A_i} x^2 f_X(x)dx = \int_{\mathcal{R}} x^2 f_X(x)dx. \tag{3.14}$$

Then E_1 becomes

$$\begin{aligned}
E_1 &= \sum_{s=1}^{K_0} p(s) \times \binom{K}{s}\mathcal{M} - 2\sum_{s=1}^{K_0} p(s) \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} \sum_{i\in\mathcal{I}_\mathcal{K}} \frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)} c_i \\
&+ \sum_{s=1}^{K_0} p(s) \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} \sum_{i\in\mathcal{I}_\mathcal{K}} (\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2 P_i \\
&= \sum_{s=1}^{K_0} p(s) \times \binom{K}{s}\mathcal{M} - 2\sum_{s=1}^{K_0} p(s) \sum_{i\in\mathcal{I}_\mathcal{K}} c_i \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} \frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)} \\
&+ \sum_{s=1}^{K_0} p(s) \sum_{i\in\mathcal{I}_\mathcal{K}} P_i \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} (\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2.
\end{aligned} \tag{3.15}$$

We have used the fact that the number of subsets of s descriptions equals $\binom{K}{s}$. Now note that the values $\mathcal{M}$, c_i and P_i depend only on the encoder, hence they are constants for our optimization problem. Therefore, from the above equality it is already clear that E_1 is a quadratic function of $\mathbf{y}$, in other words, that there exist the scalar r, the vector $\mathbf{u}$ and the matrix B such that (3.8) to hold. Precisely,

$$r = \sum_{s=1}^{K_0} p(s) \times \binom{K}{s} \mathcal{M}, \tag{3.16}$$

and $\mathbf{u}$ and B are chosen such that the following equalities to be satisfied

$$\mathbf{u}^T \mathbf{y} = -2 \sum_{s=1}^{K_0} p(s) \sum_{\mathrm{i} \in \mathcal{I}_{\mathcal{K}}} c_{\mathrm{i}} \underbrace{\sum_{\mathcal{L} \in \mathcal{K}, |\mathcal{L}|=s} \frac{1}{s} \sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)}}_{T_1}, \tag{3.17}$$

$$\mathbf{y}^T B \mathbf{y} = \sum_{s=1}^{K_0} p(s) \sum_{\mathrm{i} \in \mathcal{I}_{\mathcal{K}}} P_{\mathrm{i}} \underbrace{\sum_{\mathcal{L} \in \mathcal{K}, |\mathcal{L}|=s} (\frac{1}{s} \sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2}_{T_2(s)}. \tag{3.18}$$

In order to derive $\mathbf{u}$ and B we need first to write T_1 and $T_2(s)$ in a simpler form. Notice that by expanding T_1, each $a_{i_j}^{(j)}$, $1 \le j \le K$, appears $\binom{K-1}{s-1} = \frac{s}{K}\binom{K}{s}$ times in the summation. We found this value by counting the total number of subsets $\mathcal{L}$ which contain description j and other $s-1$ descriptions. Hence,

$$T_1 = \sum_{\mathcal{L} \in \mathcal{K}, |\mathcal{L}|=s} \frac{1}{s} \sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)} = \frac{1}{K} \binom{K}{s} \sum_{j=1}^{K} a_{i_j}^{(j)}. \tag{3.19}$$

In order to simplify $T_2(s)$ we have to treat separately the cases $s > 1$ and $s = 1$. When $s = 1$, we have clearly

$$T_2(1) = \sum_{j=1}^{K} (a_{i_j}^{(j)})^2. \tag{3.20}$$

When $s > 1$, notice that after expanding the summations, $T_2(s)$ contains terms of the form $(a_{i_j}^{(j)})^2$ and $2a_{i_j}^{(j)} a_{i_{j'}}^{(j')}$, $1 \le j < j' \le K$. Each $(a_{i_j}^{(j)})^2$ appears $\binom{K-1}{s-1} = \frac{s}{K}\binom{K}{s}$ times, and each $2a_{i_j}^{(j)} a_{i_{j'}}^{(j')}$ occurs $\binom{K-2}{s-2} = \binom{s}{2}\binom{K}{s}/\binom{K}{2}$ times. The latter quantity was computed by counting the total number of subsets $\mathcal{L}$ which contain descriptions j, j' and other $s-2$ descriptions. Then

$$\begin{aligned} T_2(s) &= \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} (\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2 \\ &= \frac{1}{Ks}\binom{K}{s}\sum_{j=1}^{K}(a_{i_j}^{(j)})^2 + \frac{2\binom{s}{2}}{\binom{K}{2}s^2}\binom{K}{s}\sum_{j=1}^{K-1}\sum_{j'=j+1}^{K} a_{i_j}^{(j)}a_{i_{j'}}^{(j')} \\ &= \frac{1}{Ks}\binom{K}{s}\sum_{j=1}^{K}(a_{i_j}^{(j)})^2 + \frac{2(s-1)}{Ks(K-1)}\binom{K}{s}\sum_{j=1}^{K-1}\sum_{j'=j+1}^{K} a_{i_j}^{(j)}a_{i_{j'}}^{(j')}. \end{aligned} \tag{3.21}$$

Relation (3.20) implies that the above simplified expression of $T_2(s)$ for $s > 1$, also holds for $s = 1$. Therefore, in the sequel we will use (3.21) as an equivalent form for $T_2(s)$, for all s.

Define a function $w(\imath)$, such that

$$w(\imath) = \begin{cases} \imath \mod M, & \imath \mod M \neq 0 \\ M, & \imath \mod M = 0 \end{cases}$$

Additionally, we denote by $C_n^{(j)}$ the sum of $c_{\mathbf{i}}$'s for all central cells $A_{\mathbf{i}}$ such that the jth component of the K-tuple $\mathbf{i}$ equals n, i.e.,

$$C_n^{(j)} = \sum_{\mathbf{i}\in\mathcal{I}_{\mathcal{K}}, i_j=n} c_{\mathbf{i}} \tag{3.22}$$

Now we can proceed to the construction of $\mathbf{u}$. Substituting (3.19) into (3.17), we have

$$\mathbf{u}^T\mathbf{y} = -2\sum_{s=1}^{K_0} p(s)\sum_{\mathbf{i}\in\mathcal{I}_{\mathcal{K}}} c_{\mathbf{i}}\frac{1}{K}\binom{K}{s}\sum_{j=1}^{K} a_{i_j}^{(j)}, \tag{3.23}$$

Now we are able to generate vector $\mathbf{u}$'s components $u_\imath$, $1 \le \imath \le KM$ as follows

$$u_\imath = -2\sum_{s=1}^{K_0} p(s)\frac{1}{K}\binom{K}{s} C_{w(\imath)}^{(\lceil \imath/M\rceil)}, \tag{3.24}$$

Further, we denote by $P_n^{(j)}$ the sum of probabilities of all central cells $A_{\mathbf{i}}$ such that the jth component of the K-tuple $\mathbf{i}$ equals n. Similarly, we denote by $P_{n,n'}^{(j,j')}$ the probability of the intersection of the cell in the side description j corresponding to index n, and the cell in the side description j' corresponding to index n'. Formally,

$$P_n^{(j)} = \sum_{\mathbf{i}\in\mathcal{I}_K, i_j=n} P_{\mathbf{i}} \tag{3.25}$$

$$P_{n,n'}^{(j,j')} = \sum_{\substack{\mathbf{i}\in\mathcal{I}_K \\ i_j=n, i_{j'}=n'}} P_{\mathbf{i}}. \tag{3.26}$$

Now we can proceed to the construction of matrix B. Substituting (3.20) and (3.21) into (3.18), we have

$$\mathbf{y}^T B\mathbf{y} = \sum_{s=1}^{K_0} p(s) \sum_{\mathbf{i}\in\mathcal{I}_K} P_{\mathbf{i}} \Big(\frac{1}{Ks}\binom{K}{s} \sum_{j=1}^{K} (a_{i_j}^{(j)})^2 + \frac{2(s-1)}{Ks(K-1)}\binom{K}{s} \sum_{j=1}^{K-1}\sum_{j'=j+1}^{K} a_{i_j}^{(j)} a_{i_{j'}}^{(j')} \Big). \tag{3.27}$$

Let b_{ij}, $1 \le i, j \le KM$, denote the element on the ith row and jth column of matrix B. Then we can obtain b_{ij} by

$$b_{ij} = \begin{cases} \sum_{s=1}^{K_0} p(s)\binom{K}{s}\frac{1}{Ks} P_{w(i)}^{(\lceil i/M\rceil)}, & i = j \\ \sum_{s=1}^{K_0} p(s)\binom{K}{s}\frac{(s-1)}{Ks(K-1)} P_{w(i),w(j)}^{(\lceil i/M\rceil,\lceil j/M\rceil)}, & i < j \\ \sum_{s=1}^{K_0} p(s)\binom{K}{s}\frac{(s-1)}{Ks(K-1)} P_{w(j),w(i)}^{(\lceil j/M\rceil,\lceil i/M\rceil)}, & i > j, \end{cases}$$

Matrix B is said to be positive semi-definite if and only if

$$\mathbf{y}^T B\mathbf{y} \ge 0, \tag{3.28}$$

for any KM dimensional vector $\mathbf{y}$ [3]. It is clear from (3.18) that the above inequality is satisfied since $p(s)$, $P_{\mathbf{i}}$ and $(\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2$ are all non-negative values. It follows that the problem of minimizing E_1 is an unconstrained convex quadratic problem, for which efficient solvers are known [4].

Additionally, if matrix B is nonsingular, a closed form of the solution could be derived by taking the derivative of $\mathbf{y}^T B\mathbf{y} + \mathbf{u}^T\mathbf{y} + r$ and setting it to zero, which implies

$$\begin{aligned} \frac{\partial}{\partial \mathbf{y}}(\mathbf{y}^T B\mathbf{y} + \mathbf{u}^T\mathbf{y}) = 0 \\ 2B\mathbf{y} + \mathbf{u} = 0, \end{aligned} \tag{3.29}$$

because B is a symmetric matrix, and further

$$\mathbf{y} = -\frac{1}{2}B^{-1}\mathbf{u}. \tag{3.30}$$

Next we show that if there is some $s_0, 1 \le s_0 \le K_0 - 1$ such that $p(s_0) \neq 0$, then matrix B is nonsingular. For this purpose, it is sufficient to prove that under the above condition, $\mathbf{y}^T B\mathbf{y} = 0$ implies $\mathbf{y} = 0$, in other words that matrix B is positive definite [3].

Assume that $\mathbf{y}^T B\mathbf{y} = 0$. Since $p(s_0) \neq 0$ and $P_{\mathbf{i}} \neq 0$, (3.18) implies that

$$\sum_{\mathcal{L}\in\mathcal{K}, |\mathcal{L}|=s_0} (\frac{1}{s_0}\sum_{k=1}^{s_0} a_{i_{l_k}}^{(l_k)})^2 = 0, \tag{3.31}$$

for all $\mathbf{i} \in \mathcal{I}_{\mathcal{K}}$. The above equality further implies that

$$\sum_{k=1}^{s_0} a_{i_{l_k}}^{(l_k)} = 0, \tag{3.32}$$

for any $\mathbf{i} \in \mathcal{I}_{\mathcal{K}}$ and $\mathcal{L} \in \mathcal{K}$ with $|\mathcal{L}| = s_0$.

Consider now an arbitrary $\mathbf{i} \in \mathcal{I}_{\mathcal{K}}$ and two arbitrary distinct descriptions $u \neq v$, $1 \leq u, v \leq K$. Pick a description set $\mathcal{L}_1$ such that it contains the uth description but not the vth description. Also pick a description set $\mathcal{L}_2$ such that $\mathcal{L}_2 = \mathcal{L}_1 \backslash u \cup v$. Subtracting the equation (3.32) corresponding to $\mathcal{L}_1$ and the equation (3.32) corresponding to $\mathcal{L}_2$, we obtain

$$a_{i_u}^{(u)} = a_{i_v}^{(v)}. \tag{3.33}$$

Since (3.33) holds for any u and v, it follows that

$$a_{i_1}^{(1)} = a_{i_2}^{(2)} = \cdots = a_{i_K}^{(K)} = 0. \tag{3.34}$$

Further, since (3.34) is valid for all $\mathbf{i} \in \mathcal{I}_{\mathcal{K}}$, it follows that $\mathbf{y} = 0$.□

3.4 Encoder Optimization Step. Complexity Reduction.

At this step the decoder is fixed and the encoder is optimized. We can handle this optimization step by computing first the reconstruction values for all decoders $g_{\mathcal{L}}$ corresponding to at most K_0 descriptions, i.e. $|\mathcal{L}| \leq K_0$, by using

$$a_{i_{l_1}, \cdots, i_{l_s}}^{\mathcal{L}} = \frac{1}{s} \sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)}, \tag{3.35}$$

for all $\mathbf{i} \in \mathcal{I}_{\mathcal{K}}$, and then proceeding as in the encoder optimization step for Opt-MDSQ described in Section 2.2.2. In other words, the values $\alpha_{\mathbf{i}}$ and $\beta_{\mathbf{i}}$, for all $\mathbf{i} \in \mathcal{I}_{\mathcal{K}}$ are computed next by applying

$$\begin{aligned} \alpha_{\mathrm{i}} &= \sum_{\mathcal{L}=\{l_1,\cdots,l_s\}\subseteq\mathcal{K}} p(|\mathcal{L}|)a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}} \\ \beta_{\mathrm{i}} &= \sum_{\mathcal{L}=\{l_1,\cdots,l_s\}\subseteq\mathcal{K}} p(|\mathcal{L}|)(a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}})^2, \end{aligned} \tag{3.36}$$

and further (2.19) is solved.

An alternative method for calculating the quantities α_{i} and β_{i}, of lower computational cost, is by deriving simplified expressions for α_{i} and β_{i} based only on the reconstruction values in the stored codebooks.

In order to highlight the reduction in complexity incurred by the second approach we will first evaluate the number of operations needed when the first approach is used. To compute the value of each α_{i}, since there are 2^K-1 terms in the summation, 2^K-1 multiplications and 2^K-2 additions are needed, hence $2^{K+1}-3$ operations overall. To compute the value of each β_{i}, since there are 2^K-1 terms in the summation, $2(2^K-1)$ multiplications and 2^K-2 additions are needed, hence $3\cdot 2^K-4$ operations overall.

Next we will derive simpler expressions for α_{i} and β_{i}. Equality (3.36) implies that

$$\alpha_{\mathrm{i}} = \underbrace{\sum_{s=1}^{K_0} \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} p(s)a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}}}_{T_3} + \sum_{s=K_0+1}^{K} \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} p(s)a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}}. \tag{3.37}$$

Using (3.35) and (3.19), T_3 can be written as follows

$$\begin{aligned} T_3 &= \sum_{s=1}^{K_0} \sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} p(s)\frac{1}{s}\sum_{k=1}^{s} a^{(l_k)}_{i_{l_k}} \\ &= \sum_{s=1}^{K_0} p(s)\frac{1}{K}\binom{K}{s}\sum_{j=1}^{K} a^{(j)}_{i_j}. \end{aligned} \tag{3.38}$$

Substituting further in (3.37) we obtain

$$\alpha_{\mathrm{i}} = \sum_{s=1}^{K_0} p(s)\frac{1}{K}\binom{K}{s}\sum_{j=1}^{K} a_{i_j}^{(j)} + \sum_{s=K_0+1}^{K}\sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} p(s)a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}}. \tag{3.39}$$

Since the quantity $\sum_{s=1}^{K_0} p(s)\frac{1}{K}\binom{K}{s}$ can be pre-computed, rather than evaluated for each $\mathbf{i}$, the number of operations in the first term equals $K-1$ additions plus 1 multiplication. Hence K operations are needed for the first term. The number of operations in the second term is $2\sum_{s=K_0+1}^{K}\binom{K}{s}-1$, since there are $\sum_{s=K_0+1}^{K}\binom{K}{s}$ terms in the summation. Therefore, $K+2\sum_{s=K_0+1}^{K}\binom{K}{s}-1$ operations are needed to compute each α_{i}. When $K_0=K$, the number of operations is K. When $K_0=K-1$, the number of operations is $K+1$.

To simplify the expression of β_{i} note first that

$$\begin{aligned}\beta_{\mathrm{i}} &= \sum_{\mathcal{L}=\{l_1,\cdots,l_s\}\subseteq\mathcal{K}} p_{\mathcal{L}}(a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}})^2\\ &= \underbrace{\sum_{s=1}^{K_0}\sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} p(s)(a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}})^2}_{T_4} + \sum_{s=K_0+1}^{K}\sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} p(s)(a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}})^2,\end{aligned} \tag{3.40}$$

Using (3.35) and (3.21), T_4 can be written as

$$\begin{aligned}T_4 &= \sum_{s=1}^{K_0}\sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} p(s)(\frac{1}{s}\sum_{k=1}^{s} a_{i_{l_k}}^{(l_k)})^2\\ &= \sum_{s=1}^{K_0} p(s)\Big(\frac{1}{Ks}\binom{K}{s}\sum_{j=1}^{K}(a_{i_j}^{(j)})^2 + \frac{2(s-1)}{Ks(K-1)}\binom{K}{s}\sum_{j=1}^{K-1}\sum_{j'=j+1}^{K} a_{i_j}^{(j)}a_{i_{j'}}^{(j')}\Big)\\ &= \sum_{s=1}^{K_0} p(s)\frac{1}{Ks}\binom{K}{s}\sum_{j=1}^{K}(a_{i_j}^{(j)})^2 + \sum_{s=1}^{K_0} p(s)\frac{2(s-1)}{Ks(K-1)}\binom{K}{s}\sum_{j=1}^{K-1}\sum_{j'=j+1}^{K} a_{i_j}^{(j)}a_{i_{j'}}^{(j')}.\end{aligned} \tag{3.41}$$

Substituting further in (3.40) we obtain

$$\begin{aligned}\beta_{\mathbf{i}} =& \sum_{s=1}^{K_0} p(s)\frac{1}{Ks}\binom{K}{s}\sum_{j=1}^{K}(a_{i_j}^{(j)})^2 \\ &+ \sum_{s=1}^{K_0} p(s)\frac{2(s-1)}{Ks(K-1)}\binom{K}{s}\sum_{j=1}^{K-1}\sum_{j'=j+1}^{K} a_{i_j}^{(j)}a_{i_{j'}}^{(j')} \\ &+ \sum_{s=K_0+1}^{K}\sum_{\mathcal{L}\in\mathcal{K},|\mathcal{L}|=s} p(s)(a_{i_{l_1},\cdots,i_{l_s}}^{\mathcal{L}})^2,\end{aligned} \tag{3.42}$$

Since the quantities $\sum_{s=1}^{K_0} p(s)\frac{1}{Ks}\binom{K}{s}$ and $\sum_{s=1}^{K_0} p(s)\frac{2(s-1)}{Ks(K-1)}\binom{K}{s}$ can be pre-computed, rather than evaluated for each $\mathbf{i}$, the number of operations in the first term equals $K-1$ additions plus $K+1$ multiplications, hence $2K$ operations are needed. The number of operations in the second term equals $\frac{1}{2}(K^2-K)-1$ additions plus $\frac{1}{2}(K^2-K)+1$ multiplications, hence (K^2-K) operations are needed. The number of operations in the third term is $3\sum_{s=K_0+1}^{K}\binom{K}{s}-1$, since there are $\sum_{s=K_0+1}^{K}\binom{K}{s}$ terms in the summation. Therefore, $K^2+K+3\sum_{s=K_0+1}^{K}\binom{K}{s}-1$ operations are needed to compute each $\beta_{\mathbf{i}}$. When $K_0=K$, the number of operations is K^2+K. When $K_0=K-1$, the number of operations is K^2+K+2.

In conclusion, in MDSQ with linear joint decoders, the number of operations to obtain the $\alpha_{\mathbf{i}}$'s and $\beta_{\mathbf{i}}$'s, for all $\mathbf{i}\in\mathcal{I}_{\mathcal{K}}$, will be reduced. This reduction in computational cost becomes more significant as K_0 gets larger.

3.5 IA Optimization Step

Since the generalized Lloyd algorithm for MDSQ design is only guaranteed to converge to a locally optimum, the selection of the initial encoder is very important. Recall that an encoder is fully represented by the sets in the central partition and the IA, i.e., the assignment of K-tuples $(i_1, \cdots, i_K)$ to these sets. Unfortunately, the process of developing good IA's for more than two descriptions is not yet well understood. To overcome this disadvantage, we introduce in each iteration a step for IA optimization, immediately after the encoder optimization step.

At this step the reconstruction values $a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}}$ are fixed. The sets in the central partition are fixed as well, but they are not associated with K-tuples $(i_1, \cdots, i_K)$. The goal is to choose the IA such that the expected distortion is minimized.

Let $A_1, \cdots A_N$ denote the cells in the central partition. Then an IA can be defined as a mapping $h : \{1, 2, \cdots, N\} \to \mathcal{S}_{\mathcal{K}}$, such that for $\ell \neq \ell'$ we have $h(\ell) \neq h(\ell')$, where $\mathcal{S}_{\mathcal{K}}$ denotes the set of all possible K-tuples, i.e., $\mathcal{S}_{\mathcal{K}} = \{1, \cdots, M\}^K$. In order to formulate the optimization problem we assign a cost $c(\ell, (i_1, \cdots, i_K))$ to each pair $(\ell, (i_1, \cdots, i_K))$, where $\ell \in \{1, 2, \cdots, N\}$ and $(i_1, \cdots, i_K) \in \mathcal{S}_{\mathcal{K}}$, defined as follows

$$c(\ell, (i_1, \cdots, i_K)) = \begin{cases} \sum_{s=1}^{K} p(s) \sum_{\mathcal{L} \in \mathcal{K}, |\mathcal{L}|=s} \int_{A_\ell} (x - a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}})^2 f_X(x) dx, & \text{if } K_0 = K \\ \\ \sum_{s=1}^{K-1} p(s) \sum_{\mathcal{L} \in \mathcal{K}, |\mathcal{L}|=s} \int_{A_\ell} (x - a^{\mathcal{L}}_{i_{l_1},\cdots,i_{l_s}})^2 f_X(x) dx, & \text{if } K_0 < K. \end{cases}$$

We formulate the optimization problem as follows

$$\text{minimize}_h \quad \sum_{\ell=1}^{N} c(\ell, h(\ell)), \tag{3.43}$$

where the minimization is carried on over all mappings h defined as above. Note that (3.43) is a problem of minimum weight bipartite graph matching, which can be solved in polynomial time, precisely, $O(n^3)$ time, where $n = |\mathcal{S}_{\mathcal{K}}|$ [32].

The cost function in (3.43) for the case $K_0 = K$ corresponds to the expected distortion of the MDSQ, while for $K_0 < K$, it corresponds to the expected distortion from which the contribution due to the central distortion (when all descriptions are received) is excluded. The reason for this is that in the latter case the optimal reconstruction values can be used for the central decoder. These values depend only on the central partition cells and not on the IA. Therefore the central distortion is a constant in the optimization problem, hence it can be safely excluded from the cost function. The above considerations imply that after solving problem (3.43) the expected distortion of the MDSQ does not increase.

Note that the IA optimization step can as well be added to the generalized Lloyd algorithm for the design of Opt-MDSQ. In this case, the cost $c(\ell, (i_1, \cdots, i_K))$ is defined as in the case $K_0 < K$.

Note that when $s \geq K_0 + 1$, not all reconstruction values used in the cost function already exist in the current stored codebooks. This is because the codebooks are constructed using the current IA set $\mathcal{I}_{\mathcal{K}}$, while the cost function considers the set $\mathcal{S}_{\mathcal{K}}$ of all possible K-tuples. To solve this problem, we set the non-existing reconstruction values to some very large constant values (compared to the existing reconstruction values), so that they always produce a much higher cost than the existing reconstruction values, hence they will never be assigned to any central cell.

Chapter 4

Improved MD-UEP Framework

MD-UEP is a simple method to obtain symmetric multiple descriptions, by combining a successively refinable (i. e., progressively refinable or scalable, or embedded) source code with uneven erasure protection [34, 29]. One way to obtain a progressive code is through successively refinable quantization (SRQ). In this thesis we consider only MD-UEP based on SRQ. Such MD-UEP has been discussed in [19, 41]. MD-UEP is also a coding scheme with reduced storage space decoder. A detailed discussion of the storage space need of MD-UEP is presented at the end of Section 4.2.

We propose an improvement to the MD-UEP system. The main idea is to split first the source samples into K subsets and to code each subset separately using the SRQ. Then correlation is introduced between descriptions by applying uneven erasure protection with interleaved systematic Reed Solomon codes of length K across descriptions. We will show that the proposed MD-UEP strictly outperforms the old

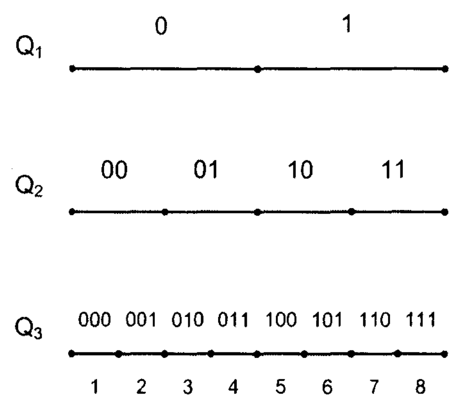


Figure 4.1: An SRQ of three refinement stages.

one. We will also assess the asymptotical improvement as the quantization block length and rate approach ∞.

In order to introduce the new MD-UEP we first need to present briefly the SRQ coding scheme and the traditional MD-UEP technique. Then we detail the novel MD-UEP framework followed by the asymptotical analysis of performance.

4.1 Successively Refinable Quantizer

An SRQ of K refinement stages consists of a sequence of K embedded quantizers, $Q^K = (Q_1, Q_2, \cdots, Q_K)$. The term embedded is used here to characterize the fact that the encoder partition of each Q_i is a refinement of the encoder partition of Q_{i-1}. The bitstream output by the SRQ is a concatenation of K layers, where the kth layer is output at the kth refinement stage. The source description provided by the first k layers is equivalent to the description provided by quantizer Q_k. Therefore, if

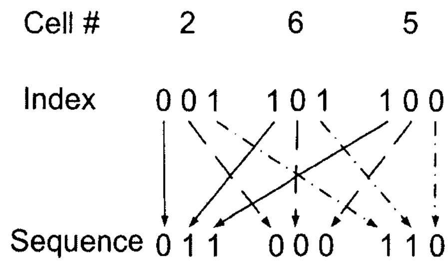


Figure 4.2: Output sequence of the SRQ.

$R(Q_k)$, denotes the rate of Q_k, then the aggregate rate of the first k refinement stages, equals $R(Q_k)$. The bitstream produced by the SRQ is progressively decodable and the distortion achieved after decoding the first k layers equals the distortion $D(Q_k)$ of quantizer Q_k. Fig. 4.1 shows an example of a three-stage SRQ designed for uniformly distributed source. Assume there are three source samples to be quantized, and they are in the second, sixth, and fifth cell in the partition of Q_3, respectively. The indices of those cells are 001, 101, and 110, respectively. Then the output sequence can be formed by first taking the most significant bit of the three indices, then the second significant bit, and finally, the least significant bit. Fig. 4.2 shows how the output sequence is formed.

4.2 Traditional MD-UEP

In the traditional MD-UEP scheme, for each $k, 1 \leq k \leq K$, the kth layer of the bitstream output by the SRQ is partitioned into groups of k consecutive symbols which are further encoded by a strict systematic (K, k) Reed Solomon (RS) code

[36]. We use the term *strict* systematic code in order to enforce the fact that the information symbols are placed at the beginning of the channel codeword. The effect of a (K, k) RS code is that, when at least k out of K descriptions are received, all K bits can be recovered correctly. Conversely, if $k' > k$ descriptions are lost, it is impossible to recover any information contained in the lost descriptions.

The K descriptions are formed across the channel codewords, such that for each j, the jth description contains the jth symbol of each channel codeword. Fig. 4.3(a) illustrates the previous MD-UEP scheme. Each row represents an RS codeword. Each column represents a description.

Note that the rate of each description equals

$$R = \sum_{i=1}^{K-1} \frac{1}{i(i+1)} R(Q_i) + \frac{R(Q_K)}{K}.$$

This system ensures that when only $k, 1 \leq k \leq K$, descriptions are available at the decoder, the first k layers of the source bitstream can be completely recovered. The missing source symbols from the rest of the layers will prevent the decoding of any available symbols from these layers. Therefore, only the first k layers can be decoded leading to the source reconstruction with distortion $D(Q_k)$.

To measure the performance of the MD-UEP, we use the expected distortion of the source reconstruction at the decoder when at least one description is received. We use this expected distortion instead of the expected distortion over all possibilities, in order to eliminate the term corresponding to no description received, which complicates the performance analysis. Assuming that $U(k)$ denotes the conditional probability that only k descriptions are available given that at least one is received, we obtain the following expression of the expected distortion:

$$\overline{D}_{MD-UEP} = \sum_{k=1}^{K} U(k) D(Q_k). \tag{4.1}$$

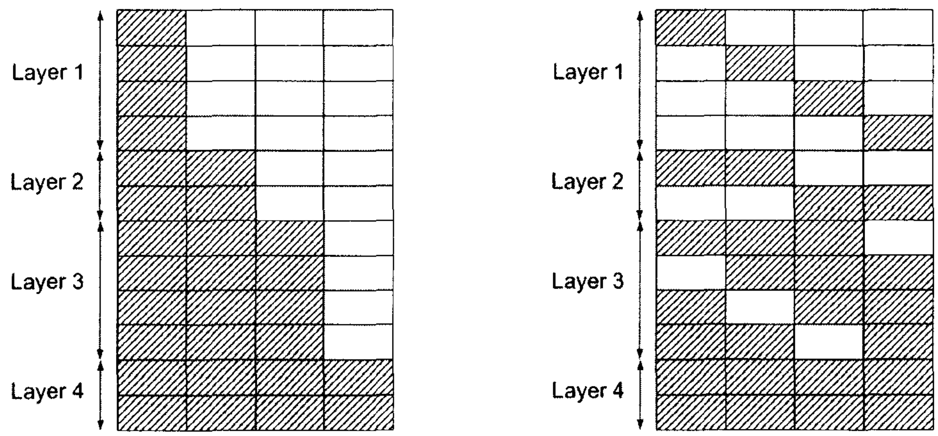
The problem of optimal design of a symmetric multiple description system can be formulated as the problem of minimizing the expected distortion at the decoder under a constraint on the rate of a description. In the case of MD-UEP, this translates to designing an SRQ which minimizes the expected distortion (4.1) under the constraint

$$\sum_{i=1}^{K-1} \frac{1}{i(i+1)} R(Q_i) + \frac{R(Q_K)}{K} = R, \tag{4.2}$$

where R is the target rate per description. Moreover, there is an additional constraint due to finite block size of the quantizer used. Precisely, if a fixed-rate vector quantizer of block length m is used, then $mR(Q_i)$ have to be positive integers. Hence, when the SRQ is formed of scalar quantizers ($m = 1$) the rates $R(Q_i)$ have to be positive integers.

Note that the solution to the optimization problem might require that $R(Q_i) = R(Q_{i+1})$ for some i, which implies that layer $i+1$ is empty. In this case the SRQ contains less than K distinct refinement stages.

It has been mentioned at the beginning of this chapter that, MD-UEP is also a coding scheme with reduced storage space decoder. Here we present a detailed analysis of the storage space need at the decoder. To decode the RS code, K matrices need to be stored, for there are K different RS codes used, hence K different generator



(a) Previous MD-UEP (b) New MD-UEP

Figure 4.3: Illustration of MD-UEP frameworks for $K = 4$. The shadowed rectangles represent source symbols, while white rectangles represent redundancy symbols. Each row symbolizes an RS codeword. Each column symbolizes a description. The portion marked as "Layer i" represents the $(4, i)$ RS codewords.

matrices. The total number of elements is $\sum_{s=1}^{K} sK = \frac{1}{2}K^2(K+1)$. Also needed to be stored at the decoder are the K sets of codebooks, one for each quantizer $Q_k, 1 \leq k \leq K$. Then the total size can amount to $K \cdot 2^{R(Q_K)}$. Therefore, there are in total at most $K \cdot 2^{R(Q_K)} + \frac{1}{2}K^2(K+1)$ elements needed to be stored at the decoder.

4.3 Novel MD-UEP

An obvious drawback of the old MD-UEP framework is the waste of information symbols of the decoder. As explained previously, when only $k < K$ descriptions are received any received source symbols belonging to layers $k+1$ through K, is not useful for decoding and therefore discarded. We propose a new MD-UEP which overcomes this drawback.

First the set of source samples is split into subsets $S_1, S_2, \cdots, S_K$, where each S_i contains the samples at time instants $i, K+i, 2K+i, \cdots$. If the source has memory, and a vector SRQ with block length m is used, then S_i should contain the ith, $(K+i)$th, $(2K+i)$th, etc., blocks of consecutive m samples.

Further, each subset of samples is separately encoded using the SRQ. Thus K independently decodable and progressively refinable source sub-streams are produced. Further, for each k, the source symbols belonging to the kth layer (from all sub-streams) are partitioned into groups of k, such that any two symbols in a group come from different sub-streams. It is easy to see that the kth layers of different sub-streams will have the same length, therefore such a partitioning is possible. Each such group is encoded by an (K, k) interleaved systematic RS code. We use the term *interleaved* systematic code in order to emphasize that the source symbols are not necessarily placed at the beginning of the codeword, but are interleaved with the parity symbols. Precisely, the RS channel codeword encoding a group $s_{i_1}, s_{i_2}, \cdots s_{i_k}$, where s_{i_j} belongs to sub-stream i_j, will have its i_jth channel symbol equal to s_{i_j}.

The descriptions are then formed across the channel codewords, such that for each j, the jth description contains the jth symbol of each channel codeword. Fig. 4.3(b) depicts the proposed MD-UEP.

To summarize, each description i contains the source symbols of sub-stream i

(i.e. the most refined descriptions provided by the SRQ for the source samples in set S_i) and additionally RS redundancy symbols which introduce correlation between descriptions. Since each sub-stream is independently decodable, no information symbols received at the decoder will be discarded. The rate of each description is the same as in the previous MD-UEP. To see this note that when the number of the source samples is large enough, the total number of source symbols in the kth layer over all sub-streams, is the same as the size of the kth layer if all samples were encoded by the SRQ, together. Therefore, the same number of (K, k) RS codewords will be used to encode these source symbols, hence the same number of generator matrices will be stored at the decoder. Additionally, since the sequence of rates $(R(Q_1), \cdots, R(Q_K))$ of the SRQ used in the novel framework is the same as of the SRQ used in the traditional framework, we can conclude that the novel framework has the same decoder storage space need as the traditional framework.

When only $k < K$ descriptions are received at the decoder, all the source symbols in the first k layers, from all sub-streams can be recovered, their decoding leading to a source reconstruction with average distortion $D(Q_k)$. Moreover, the source symbols in layers $k+1$ through K, from the received descriptions can also be decoded. Therefore the samples encoded by these descriptions will be reconstructed to the highest quality. In conclusion, a fraction of $\frac{k}{K}$ of the source samples are reconstructed with average distortion $D(Q_K)$, while the rest are reconstructed with average distortion $D(Q_k)$.

We conclude that the expected distortion at the decoder becomes

$$\begin{aligned}\overline{D}_{newMD-UEP} &= \sum_{k=1}^{K} U(k)(\frac{K-k}{K}D(Q_k) + \frac{k}{K}D(Q_K)) \\ &= \sum_{k=1}^{K-1} U(k)\frac{K-k}{K}D(Q_k) + \sum_{k=1}^{K} \frac{k}{K}U(k)D(Q_K). \end{aligned} \quad (4.3)$$

It is clear that the proposed system provides a net improvement over the previous MD-UEP when using the same SRQ. Moreover, to maximize its performance an optimized SRQ can be used, which minimizes the expected distortion (4.3) under the constraint (4.2).

4.4 Asymptotical Analysis of Performance of MD-UEP

In this section we evaluate the improvement in performance assuming a vector SRQ with block length approaching ∞, for a memoryless Gaussian source with variance σ^2. We assume the descriptions are sent over independent channels with the same breaking down probability q. Therefore the conditional probability of only k descriptions being received, given that at least one is received is

$$U(k) = \frac{1}{1-q^K}\binom{K}{k}(1-q)^k q^{(K-k)}, \quad (4.4)$$

$1 \leq k \leq K$. We consider the squared distance as distortion measure. Since the distortion-rate function of a memoryless Gaussian source is $D(R) = \sigma^2 2^{-2R}$ and since such a source is successively refinable [6], it follows that, given an increasing sequence of target rates $R_1 \le R_2 \le \cdots \le R_K$, there is a sequence of vector SRQ's of K refinement stages, with block-length approaching ∞, such that for each stage k, the aggregate rate approaches R_k, and the distortion approaches $\sigma^2 2^{-2R_k}$. Therefore, the expected distortion of the MD-UEP under the old framework approaches $\sigma^2 \sum_{k=1}^{K} U(k) 2^{-2R_k}$. The value D_{old} of the optimal expected distortion of the old MD-UEP, achievable asymptotically in the quantizer block-length, can be found by solving the following convex optimization problem

$$\begin{aligned} \text{minimize} \quad & \sigma^2 \sum_{k=1}^{K} U(k) 2^{-2R_k} \\ \text{subject to} \quad & 0 \le R_1 \le R_2 \le \cdots \le R_K \\ & \sum_{k=1}^{K-1} \frac{1}{k(k+1)} R_k + \frac{1}{K} R_K = R, \end{aligned} \tag{4.5}$$

where R is the rate of each description. Consider the set $\mathcal{P}$ of planar points $P_0 = (0,0)$, $P_k = (1 - 1/(k+1), \sum_{i=1}^{k} U(i))$, where $1 \le k \le K-1$, and $P_K = (1,1)$. The extreme points situated on the lower convex hull of this set are important for this optimization problem. This is because it can be shown along the lines of [34], that for any k such that P_k is not an extreme point, the solution has to satisfy $R_k = R_\ell$, where ℓ is the smallest index larger than k, such that P_ℓ is an extreme point. This allows for all variables R_k for which P_k is not an extreme point, to be eliminated. The new problem can be solved by taking into account only the equality constraint and its solution will satisfy the inequality constraints as well. Based on this idea we derive the solution for values of R high enough. In order to give the expression of the solution let us define first k_0 as the largest integer between 1 and K such that $U(k_0-1)(k_0-1) \le U(k_0)+U(k_0+1)+\cdots+U(K)$. Denote further

$$A = \sum_{k=1}^{k_0-1} \frac{1}{k(k+1)} \log_2(U(k)k(k+1)) + \frac{1}{k_0}(\log_2(k_0 \sum_{k=k_0}^{K} U(k)).$$

It can be shown that for $R \ge 1/2A - 1/2\log_2 U(1) - 1/2$ (derived from the constraint $R_1 \ge 0$), we have $D_{old} = \sigma^2 2^A 2^{-2R}$. A detailed discussion of the derivation procedure is included in Appendix.

Denote now

$$\begin{aligned} V(k) =& \frac{K-k}{K} U(k) \\ =& \frac{1}{1-q^K} \frac{K-k}{K} \frac{K!}{k!(K-k)!} (1-q)^k q^{(K-k)} \\ =& \frac{1}{1-q^K} \binom{K-1}{k} (1-q)^k q^{(K-k)}, \end{aligned} \tag{4.6}$$

for $1 \le k \le K-1$, and

$$\begin{aligned} V(K) =& \sum_{k=1}^{K} \frac{k}{K} U(k) \\ =& \sum_{k=1}^{K} \frac{1}{1-q^K} \frac{k}{K} \frac{K!}{k!(K-k)!} (1-q)^k q^{(K-k)} \\ =& \frac{1-q}{1-q^K}. \end{aligned} \tag{4.7}$$

Then the optimal expected distortion D_{new}, achievable asymptotically in the quantizer block length, by the proposed MD-UEP with rate R per description, can be found by solving the optimization problem

$$\text{minimize} \qquad \sigma^2 \sum_{k=1}^{K} V(k) 2^{-2R_k} \tag{4.8}$$

subject to the two constraints in problem (4.5). Observing that for $K \geq 3$ and $q \leq 1/K$, the set of planar points $\mathcal{P}$ corresponding to the optimization problem (4.8) is convex, and letting

$$B = \sum_{k=1}^{K-1} \frac{1}{k(k+1)} \log_2(V(k)k(k+1)) + \frac{1}{K}(\log_2(KV(K)),$$

we derive the solution when $K \geq 3$, $q \leq 1/K$ and $R \geq 1/2B - 1/2\log_2 V(1) - 1/2$, as $D_{new} = \sigma^2 2^B 2^{-2R}$.

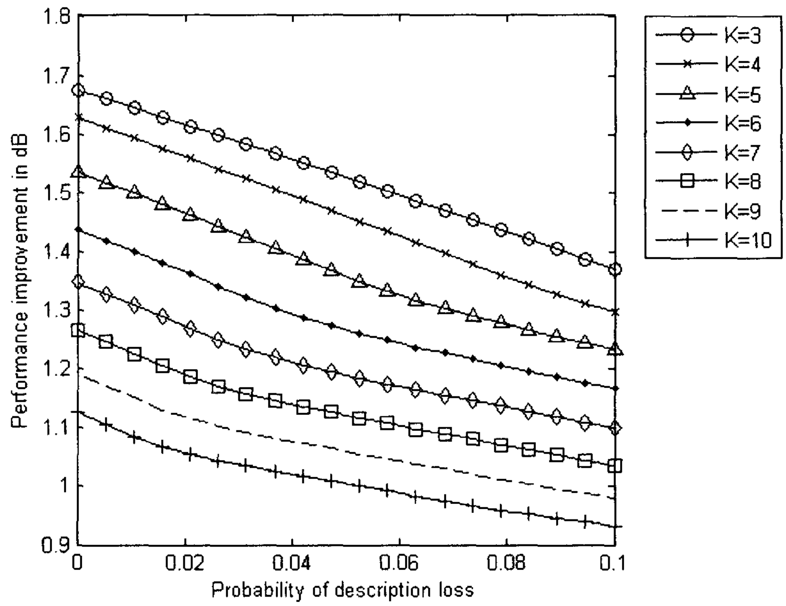
For clarity of the performance comparison between the two systems, we use the difference in dB defined as $\Delta_{new/old} = 10\log_{10}\frac{D_{old}}{D_{new}}$. Using the above formulae we obtain that for $K \geq 3$, $p \leq 1/K$ and R sufficiently high

$$\begin{aligned}\Delta_{new/old} =& 10(A-B)\log_{10} 2\\ =& 10[\sum_{k=1}^{k_0-1} \frac{1}{k(k+1)} \log_{10}\frac{K}{K-k} + \frac{1}{k_0}\log_{10}\left(k_0 \sum_{k=k_0}^{K} U(k)\right)\\ &- \frac{1}{K}\log_{10}(\frac{1-q}{1-q^K}K) - \sum_{k=k_0}^{K-1}\frac{1}{k(k+1)}\log_{10}\left(V(k)k(k+1)\right)].\end{aligned} \tag{4.9}$$

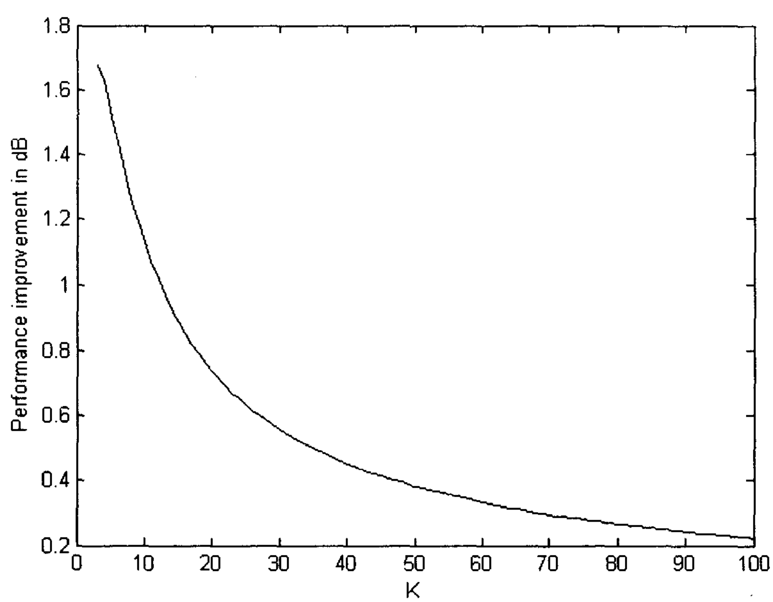
Note that when $q < 1/K^2$, we have $k_0 = K$. It follows that, as $q \to 0$ and $R \to \infty$, the following holds

$$\Delta_{new/old} \to 10\sum_{k=1}^{K-1}\frac{1}{k(k+1)}\log_{10}\frac{K}{K-k}. \tag{4.10}$$

Fig. 4.4(a) plots the difference in performance, in dB, between the two MD-UEP frameworks, for $R \to \infty$, $q \in (0, 0.1]$, and K ranging from 3 to 10, using (4.9). We see that the highest improvement is 1.68 dB, achieved for $K = 3$ and $q \to 0$, while the lowest is 0.93 dB, achieved when $K = 10$ and $q = 0.1$. Interestingly, for each value of



(a) $3 \le K \le 10$, $q \le 0.1$.



(b) $3 \le K \le 100$, $q \to 0$.

Figure 4.4: Asymptotical improvement in performance, of new MD-UEP vs. old MD-UEP.

q, the improvement is highest when $K = 3$ and decreases as K increases. Moreover, for each K the improvement in performance becomes smaller as q increases. The decreasing trend as the number of descriptions increases, is preserved for $q \to 0$ and $R \to \infty$, as it can be seen from Fig. 4.4(b), which plots the quantity (4.10).

For the case of two descriptions, problems (4.5) and (4.8) are easier to solve, without needing the assumption of high rate or the restriction on the probability of description loss q. Thus we obtain the following expressions of D_{old} and D_{new} when $K = 2$

$$D_{old} = \sigma^2(\tfrac{2q}{1+q} + \tfrac{1-q}{1+q}2^{-4R}) \text{ for } q \in (0, \tfrac{1}{1+2^{1+4R}}).$$

$$D_{old} = \sigma^2(\tfrac{2}{1+q}\sqrt{2q(1-q)}2^{-2R}) \text{ for } q \in [\tfrac{1}{1+2^{1+4R}}, \tfrac{1}{3}].$$

$$D_{old} = \sigma^2 2^{-2R} \text{ for } q \in (\tfrac{1}{3}, 1].$$

$$D_{new} = \sigma^2(\tfrac{q}{1+q} + \tfrac{1}{1+q}2^{-4R}) \text{ for } q \in (0, 2^{-4R}).$$

$$D_{new} = \sigma^2(\tfrac{2}{1+q}\sqrt{q}2^{-2R}) \text{ for } q \in [2^{-4R}, 1].$$

A detailed derivation is included in Appendix.

In Fig. 4.5 we plot the difference in performance between the new versus the old MD-UEP, in the case of two descriptions and small rates, $R = 0.5, 1, 2$. For each R, $\Delta_{new/old}$ is a unimodal function of q. For $R = 0.5$ its peak is ≈ 1 dB, achieved for $q \approx 0.2$. As R increases, its peak increases approaching 1.5 dB, and the probability where it is achieved approaches 0. Note that the performance of the two systems is identical when $q = 0$ and $q = 1$, as expected.

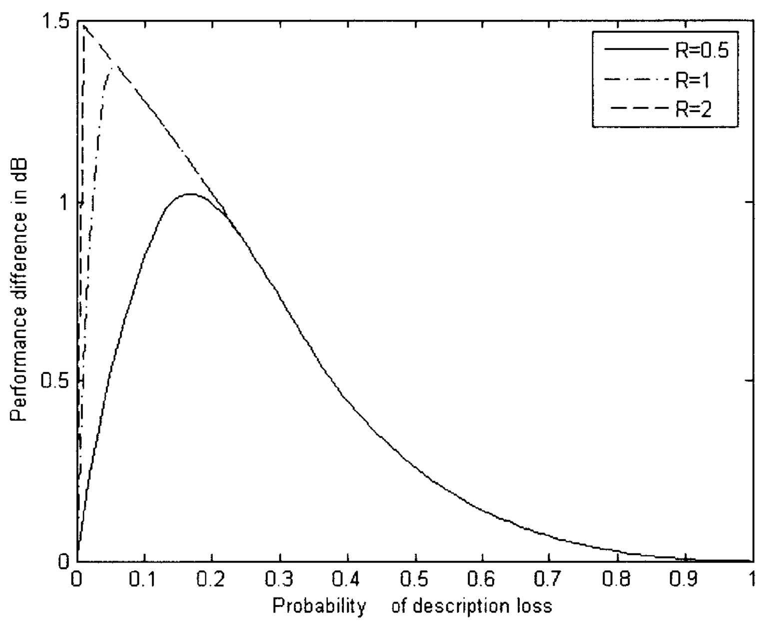


Figure 4.5: Asymptotical difference in performance, between the new and old MD-UEP, for $K = 2$, and $R = 0.5, 1, 2$.

Chapter 5

Experiments and Discussions

The purpose of this chapter is to assess the performance in practice of the proposed MDC techniques in comparison with the traditional schemes. Our tests are performed on a zero mean, unit variance, memoryless Gaussian source. The number K of descriptions considered in our experiments is between 2 and 4. Although the motivation given in this thesis for the proposed techniques is valid for higher number of descriptions, the experimental results for smaller K can be used to predict the behavior as K increases. In all cases we consider transmission over independent channels with the same probability q of failure. Consequently, the probability $p(s)$ of receiving a particular set of s descriptions is

$$p(s) = (1-q)^s q^{(K-s)}, \tag{5.1}$$

where $0 \leq s \leq K$ is the number of descriptions received at the decoder. The range of channel failure probabilities considered in our experiments is $q \in [0.001, 0.3]$. The performance of each MDC technique is measured in using the expected distortion in dB, i.e., $10\log_{10}(Expected \quad Distortion)$.

This chapter contains two sections. The experimental results in the first section highlight the impact of the IA optimization step in the design of L-MDSQ and Opt-MDSQ. The second section performs a comparison between, L-MDSQ, Opt-MDSQ, traditional MD-UEP and proposed MD-UEP. In the implementation of the MD-UEP schemes a scalar SRQ was used for fairness of comparison with the MDSQ.

5.1 Impact of IA Optimization

We have introduced the IA optimization step in the design of the proposed MDSQ's in order to overcome a possibly bad choice of initial IA. As we have pointed out in Section 3.2, this step can be used in the Opt-MDSQ design as well.

Fig. 5.1-5.4 plot the performance of Opt-MDSQ with and without IA optimization in the design algorithm, versus q. Each figure illustrates one of the following cases 1) $K = 2$ and $R = 2$, 2) $K = 2$ and $R = 3$, 3) $K = 3$ and $R = 2$, 4) $K = 4$ and $R = 1$. In the case of two descriptions, we use the modified linear IA proposed in [45], Fig. 2.2 shows the IA when rate is 3 bits/description. As it can be seen from Fig. 5.1 and Fig. 5.2, although the modified linear IA is very good already, there is still some room for improvement. The use of the IA optimization provides an improvement of up to 0.82 dB.

When it comes to the case of three or four descriptions, since good IA's are not

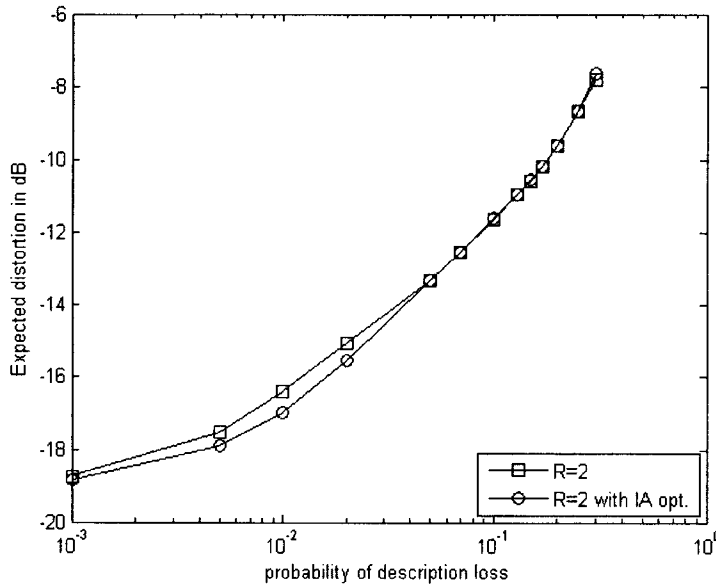


Figure 5.1: Improvement by IA optimization for Opt-MDSQ, K=2, R=2.

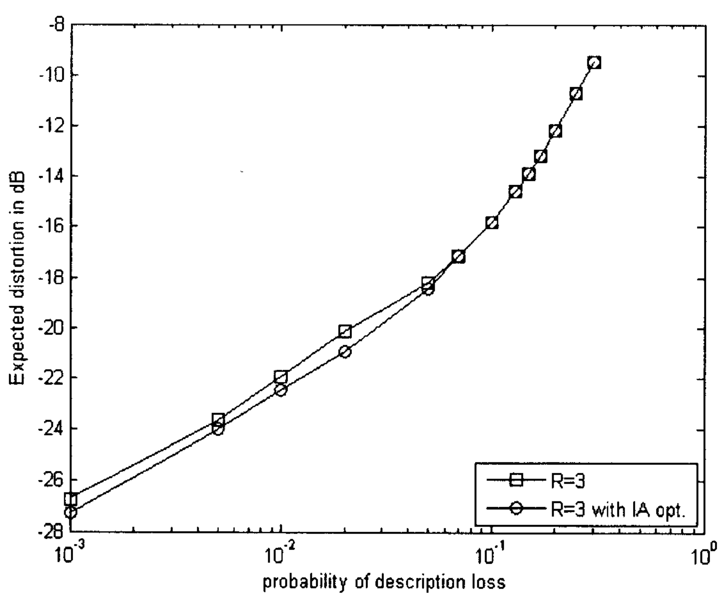


Figure 5.2: Improvement by IA optimization for Opt-MDSQ, K=2, R=3.

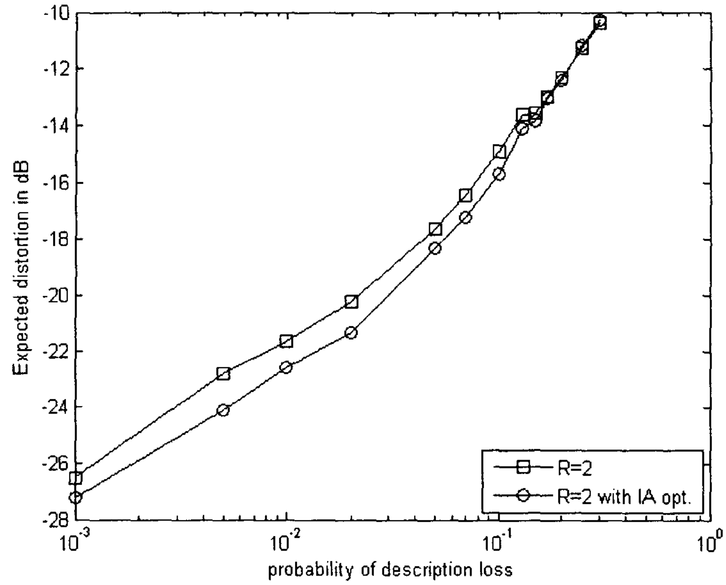


Figure 5.3: Improvement by IA optimization for Opt-MDSQ, K=3, R=2.

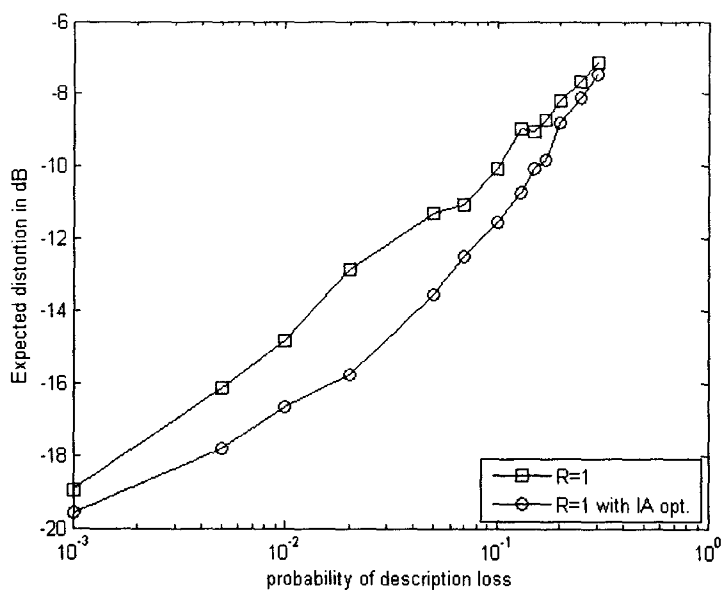


Figure 5.4: Improvement by IA optimization for Opt-MDSQ, K=4, R=1.

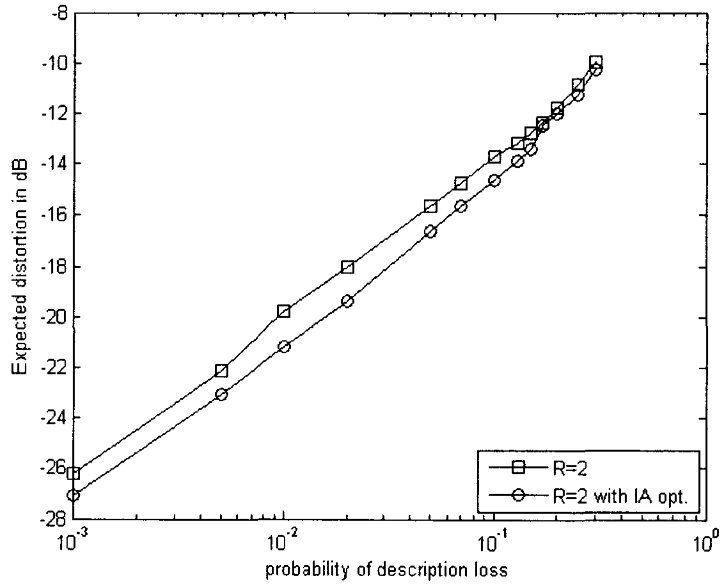


Figure 5.5: Improvement by IA optimization for L-MDSQ, K=3, K_0=2, R=2.

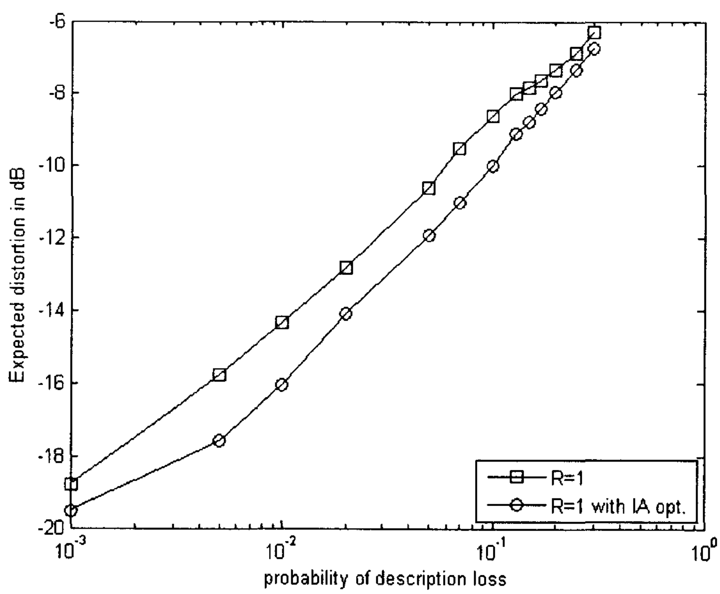


Figure 5.6: Improvement by IA optimization for L-MDSQ, K=4, K_0=3, R=1.

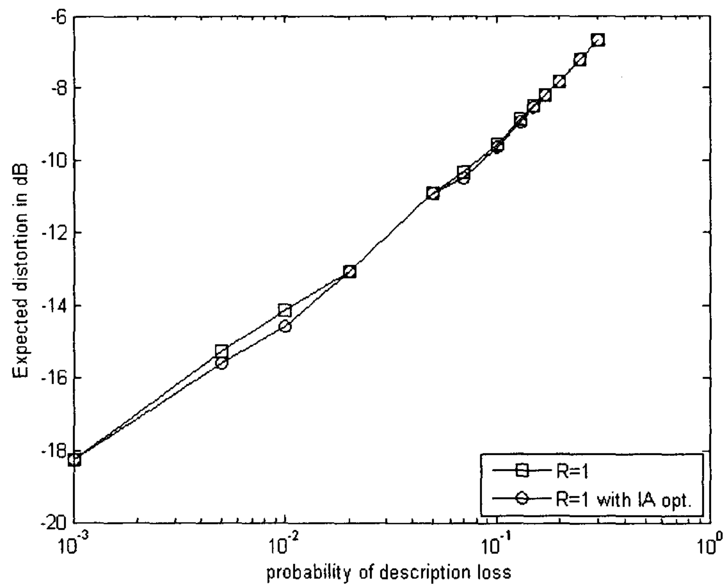


Figure 5.7: Improvement by IA optimization for L-MDSQ, K=4, K_0=4, R=1.

yet known, we use a method inspired by [41]. Consider the K-tuples whose indices in all K dimensions are the same. We call these K-tuples the diagonal elements. If the rate for each description is R, then there are 2^R such elements. For two K-tuples $(i_1, \cdots, i_K)$ and $(j_1, \cdots, j_K)$, we denote by l the distance between them defined as: $l = \sum_{k=1}^{K} |i_k - j_k|$. Further, for each K-tuple, we denote by l_{min} the minimum of the distances to every diagonal elements. We only select the K-tuples whose l_{min} is less than or equal to some threshold. For instance, if $K = 3$ and $R = 2$, there are $2^{RK} = 2^6 = 64$ K-tuples, and the diagonal elements are $(1,1,1)$, $(2,2,2)$, $(3,3,3)$ and $(4,4,4)$. If we set the threshold of l_{min} to be 2, then the K-tuple $(1,1,4)$ will not be included in the set $\mathcal{I}_K$, for its l_{min} is 3. Next we partition the set $\mathcal{I}_K$ into 2^R sets in such a way that, the kth set contains the K-tuples whose l_{min} is associated with the diagonal element $(k, \cdots, k)$. We begin with the first set, in which every K-tuple is closer (measured by l) to $(1, \cdots, 1)$ than to any other diagonal element. Then we proceed to the second, third,..., 2^Kth set. In each set, the K-tuples are randomly arranged.

Fig. 5.1-5.4 show that the improvement due to the use of IA optimization step increases with K, reaching up to 1.29 dB in the case of 3 descriptions, and 2.92 dB in the case of 4 descriptions, respectively. This observation is in accordance with the intuition that, as K increases it is more difficult to construct a good IA.

Fig. 5.5-5.6 illustrate the improvement in performance when IA optimization is applied to L-MDSQ with $K_0 = K - 1$, in other words, with an optimal central codebook. The number of descriptions considered are $K = 3$ and $K = 4$, and the rates are $R = 2$ when $K = 3$, and $R = 1$ when $K = 4$. Considerable gain is achieved here as well, reaching up to 1.42 dB for 3 descriptions, and 1.81 dB for 4 descriptions, respectively.

However, in the case of L-MDSQ with $K_0 = K$ (only side codebooks are stored), the IA optimization step does not make a significant difference. We have tested several cases: $R = 2$ and $R = 3$ when $K = 2$, $R = 1$ and $R = 2$ when $K = 3$, $R = 1$ when $K = 4$, and observed that the impact of IA optimization is not significant in all the cases. Here we only present the effect of IA optimization when it is applied to the case of 4 descriptions, $R = 1$ in Fig. 5.7. It can be seen that the improvement is less than 0.5 dB.

In conclusion, our tests show that the impact of IA optimization step in L-MDSQ design increases as the number of stored joint codebooks increases, culminating in the case of Opt-MDSQ (where all codebooks are stored).

5.2 Performance Evaluation of the New Techniques

In this section we present the experimental comparison between the proposed MDC techniques and two existing ones (Opt-MDSQ and previous MD-UEP).

As it was discussed in Section 3.1, L-MDSQ has lower decoding storage needs than Opt-MDSQ, but at the cost of decrease in performance. One purpose of the experiments presented here is to assess how significant this loss is in performance.

As justified in Section 4.4, the new MD-UEP always outperforms the traditional MD-UEP theoretically. Another goal of our experiments is to validate this conclusion and to assess the magnitude of this improvement.

Intuitively MD-UEP (both schemes) does not perform as well as MDSQ, since the former has a higher amount of redundancy. Therefore, another objective of our tests is to compare the practical performance of (traditional/new) MD-UEP with the MDSQ, and ultimately compare the proposed L-MDSQ with the new MD-UEP scheme.

In our tests we have used optimized MD-UEP based on scalar SRQ. Before presenting the tests results we first briefly describe our method for MD-UEP optimization.

5.2.1 Optimization of MD-UEP

First note that the only difference between the optimization problems for new and old MD-UEP consists in the different weights assigned to the component distortions of the SRQ in the cost function. Therefore, the same method can be used to design the optimal solution in both cases, with the appropriate choice of the weights. Therefore, in the sequel we specifically address the design of previous MD-UEP technique.

As discussed in Section 4.2, the problem of optimal MD-UEP design can be formulated as the problem of constructing an SRQ which minimizes the cost function (4.1) under the constraint (4.2). Since a scalar SRQ is under consideration, there is the additional constraint that each $R(Q_i), 1 \le i \le K$, in (4.2) is an integer. Note that the number of K-tuples of rates $(R(Q_1), R(Q_2), \cdots, R(Q_K))$ which satisfy the above mentioned constraints is finite. We call any such K-tuple a qualifying K-stage rate allocation. We emphasize the requirement that $R(Q_i) \le R(Q_{i+1})$ imposed by the definition of the SRQ. Also note that we allow equality between the rates of consecutive stages.

Our strategy in solving the MD-UEP optimization problem consists of three steps: 1) list all qualifying K-stage rate allocations; 2) for each K-stage rate allocation, design a K-stage scalar SRQ achieving these rates, of minimal expected distortion (4.1), by using the algorithm proposed in [6]; 3) select the SRQ which has the minimum expected distortion.

The major challenge is that as the rate, or the number of descriptions increases, the

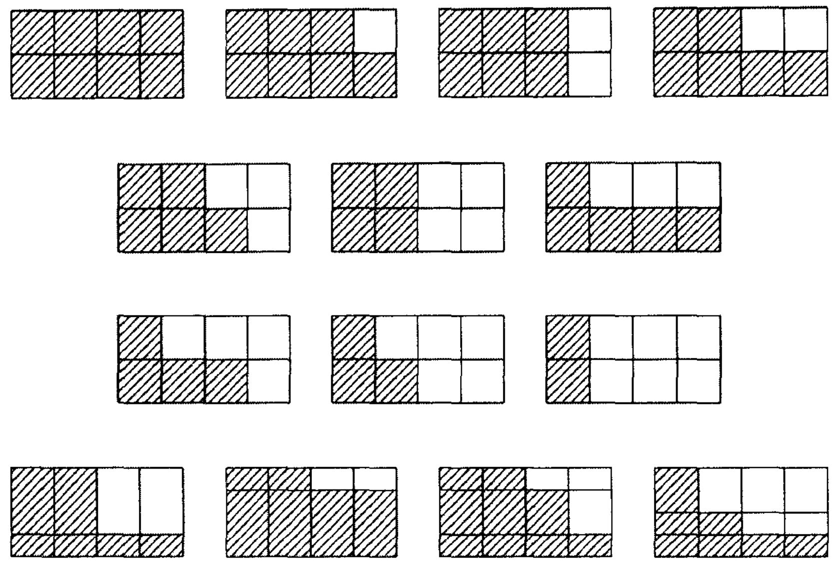


Figure 5.8: All possible frameworks for K=4, R=2. The shaded area represents the source symbols, the white area represents the protection symbols.

number of qualifying K-stage rate allocations becomes very large, hence our method will become very time-consuming. Here we only consider 2, 3 and 4 descriptions, with rates 2 and 3 for 2 descriptions, and rate 1 and 2 for 3 and 4 descriptions. Among all these cases, 4 descriptions with rate 2 has the largest number of qualifying K-stage rate allocations. Fig. 5.8 illustrates the MD-UEP rate allocation for all fourteen possibilities for this case. The shaded area represents the source symbols, the white area represents the protection symbols. There are only fourteen possible frameworks, hence the optimization is still tructable.

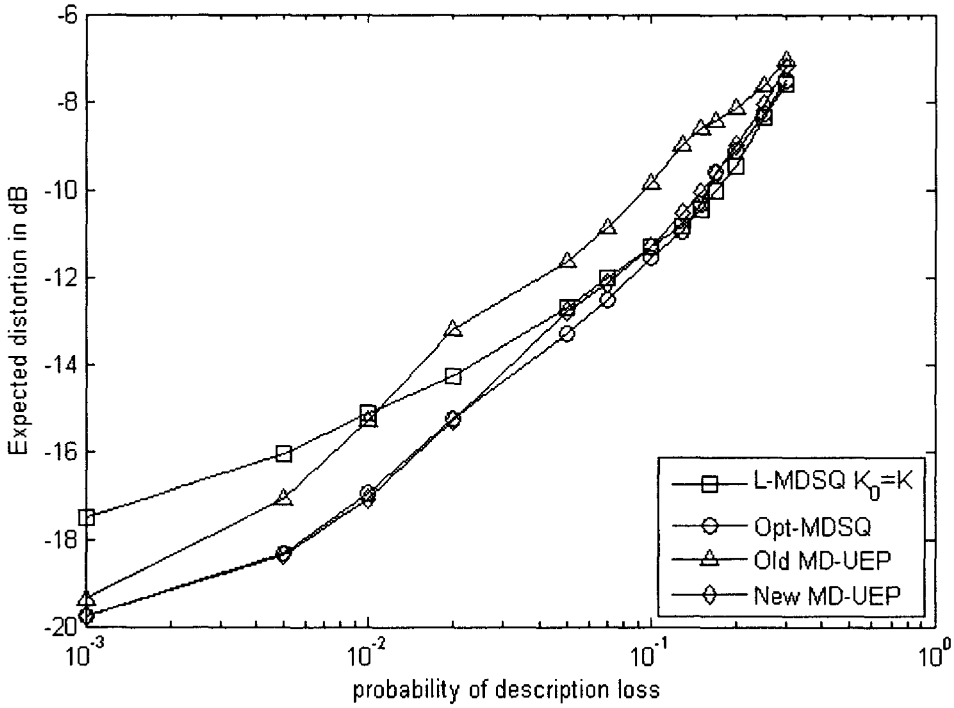


Figure 5.9: Comparison between MDSQ and MD-UEP, K=2, R=2.

5.2.2 Experimental Results

In this part we present the comparison between, L-MDSQ (two variants: $K_0 = K$ and $K_0 = K - 1$), Opt-MDSQ, traditional MD-UEP, and new MD-UEP, for 2 to 4 descriptions, and the following rates: $R = 2$ and $R = 3$ when $K = 2$, $R = 1$ and $R = 2$ when $K = 3$, $R = 1$ and $R = 2$ when $K = 4$.

Fig. 5.9,5.10 illustrate the performance of L-MDSQ ($K_0 = K$), Opt-MDSQ, old MD-UEP and new MD-UEP with 2 descriptions, rate 2 and rate 3, respectively, versus q, the probability of description loss. Fig. 5.11-5.14 illustrate the performance of L-MDSQ ($K_0 = K$, $K_0 = K - 1$), Opt-MDSQ, old MD-UEP and new MD-UEP with 3 and 4 descriptions, rate 1 and rate 2, respectively, versus q. The performance is measured by expected distortion in dB. IA optimization is applied to all MDSQ frameworks, except for the case of 4 descriptions, rate 2 (Fig. 5.14).

We will structure our analysis by discussing the comparison between the following pairs of MDC schemes: 1) L-MDSQ versus Opt-MDSQ; 2) traditional MD-UEP versus proposed MD-UEP; 3) MDSQ versus proposed MD-UEP.

L-MDSQ versus Opt-MDSQ. Most of the time the performance of L-MDSQ is worse than Opt-MDSQ. When $K_0 = K$, the performance gap between L-MDSQ and Opt-MDSQ becomes smaller as the probability q of description loss gets larger. This is in accordance with the

intuition that, as q increases, the central codebook becomes less important, hence the degradation in performance is not as significant as when q is very small. L-MDSQ

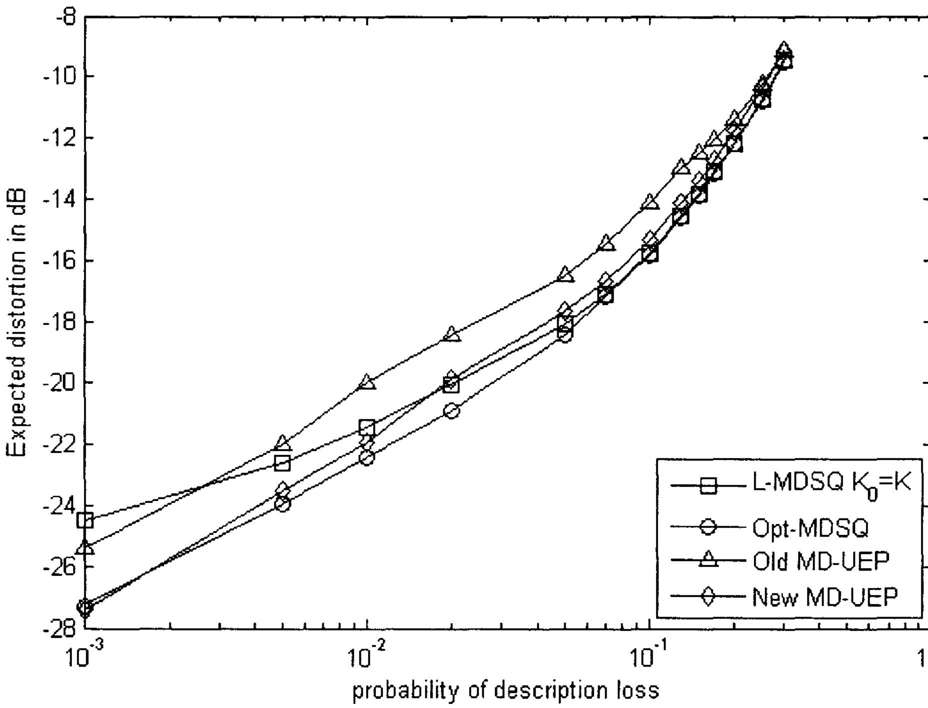


Figure 5.10: Comparison between MDSQ and MD-UEP, K=2, R=3.

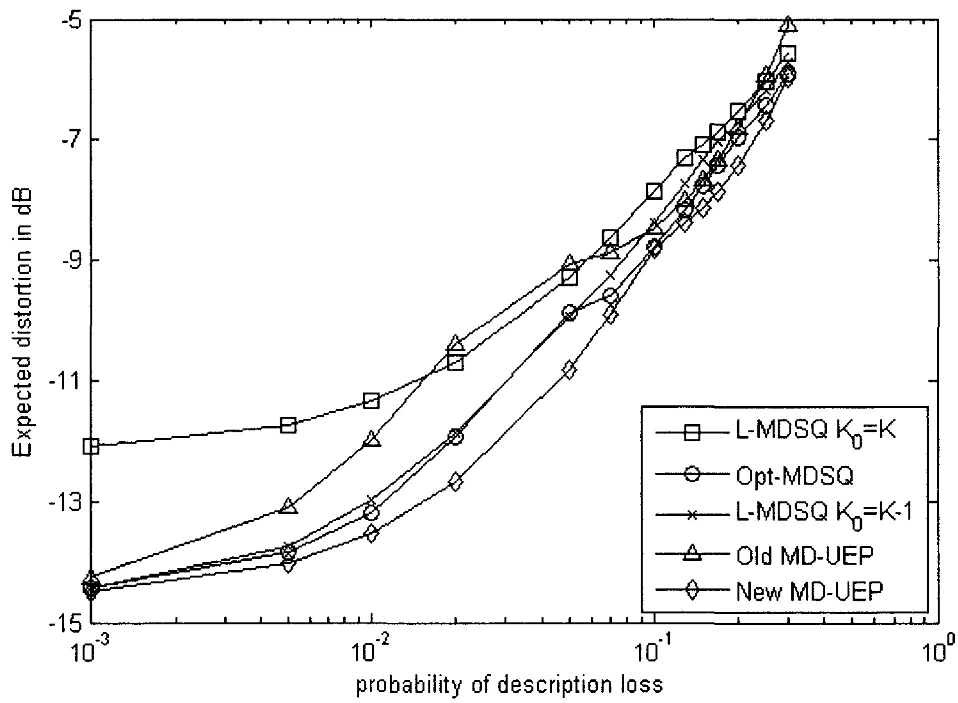


Figure 5.11: Comparison between MDSQ and MD-UEP, K=3, R=1.

with $K_0 = K - 1$ always performs better than L-MDSQ with $K_0 = K$, and worse than Opt-MDSQ. Note that this relationship in performance is just the reverse order of the relationship of the size of storage space needed at the decoder. When q is very small ($q = 0.001$), under most cases the performance of L-MDSQ with $K_0 = K - 1$ is very close to Opt-MDSQ. This can be justified by the fact that when q is very small, the central codebook becomes very important, and both schemes have an optimal central codebook stored.

Traditional MD-UEP versus new MD-UEP. The experimental results validate our conclusion that new MD-UEP strictly outperforms the traditional MD-UEP.

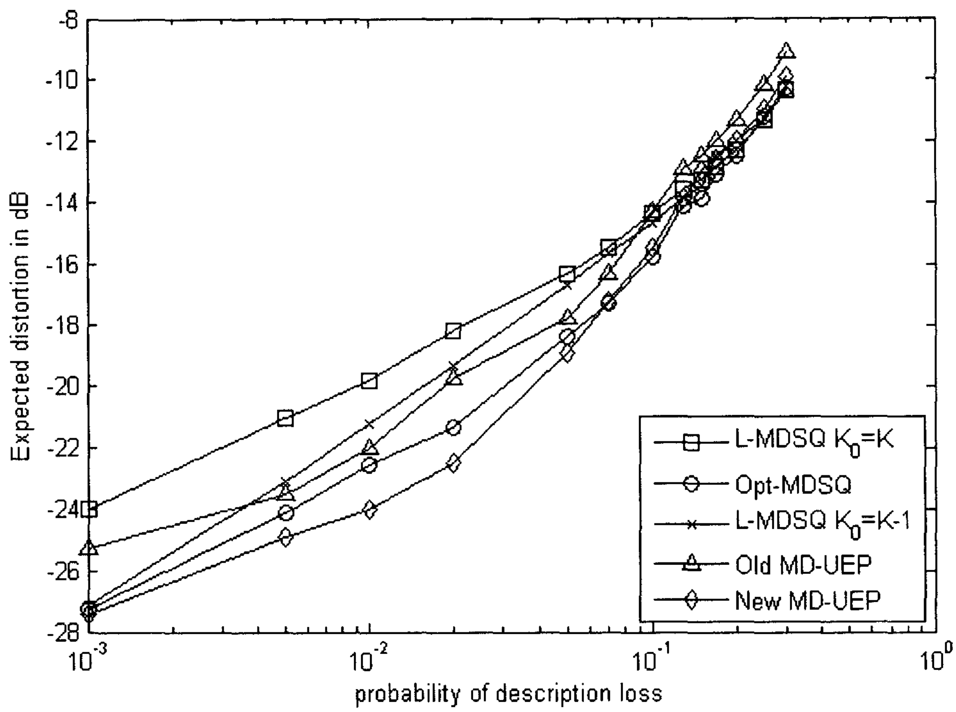


Figure 5.12: Comparison between MDSQ and MD-UEP, K=3, R=2.

The maximum difference is 2.1 dB for 2 descriptions, 2.7 dB for 3 descriptions, and 3.05 dB for 4 descriptions.

MDSQ versus new MD-UEP. For the case of 2 descriptions, the performances of Opt-MDSQ and new MD-UEP are quite close, and when q is more than 0.1, the performances of L-MDSQ ($K_0 = K$), Opt-MDSQ and new MD-UEP are very close. For the case of 3 and 4 descriptions, most of the times new MD-UEP outperforms L-MDSQ (both $K_0 = K$ and $K_0 = K - 1$) and Opt-MDSQ. Only when q is very high, L-MDSQ performs slightly better than new MD-UEP. Note that our tests show that the new MD-UEP could outperform Opt-MDSQ. This is contrary to the intuition that

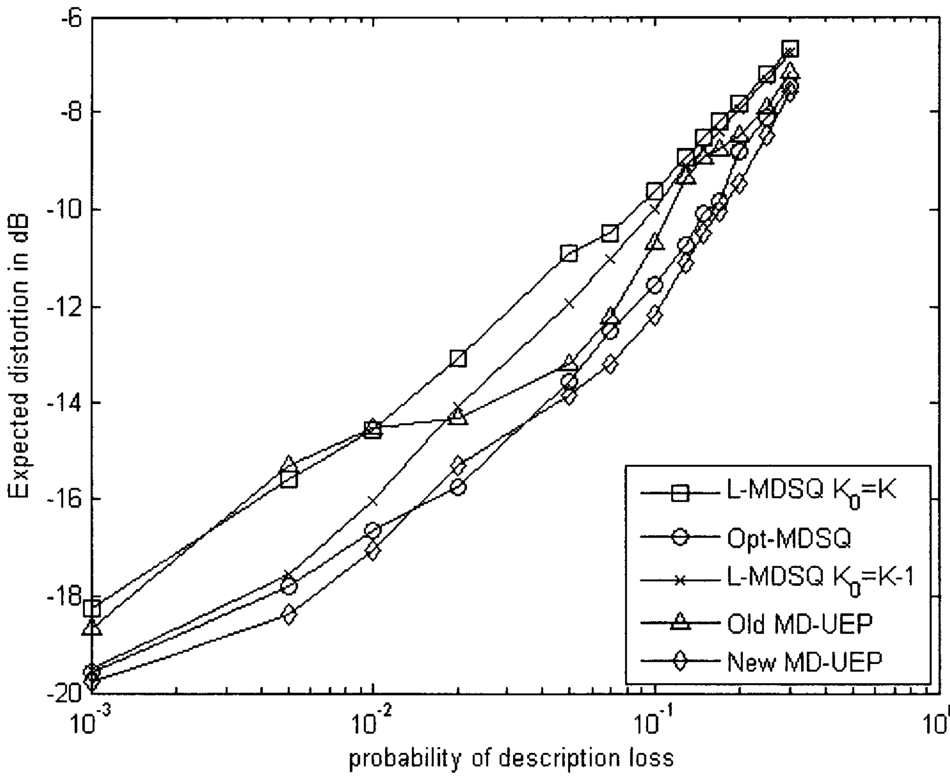


Figure 5.13: Comparison between MDSQ and MD-UEP, K=4, R=1.

MD-UEP should be worse than Opt-MDSQ (since the former has a higher amount of redundancy). We attribute this result to the sub-optimality of the design of Opt-MDSQ, which is due to the imperfection of the IA: we have already known good IA's for the case of 2 descriptions, hence the performance of Opt-MDSQ is good and sometimes is better than new MD-UEP; good IA's are not yet known for 3 and 4 descriptions, this could justify why most of the time the performance of Opt-MDSQ is worse than new MD-UEP. This is more obvious in Fig. 5.14. In this case IA

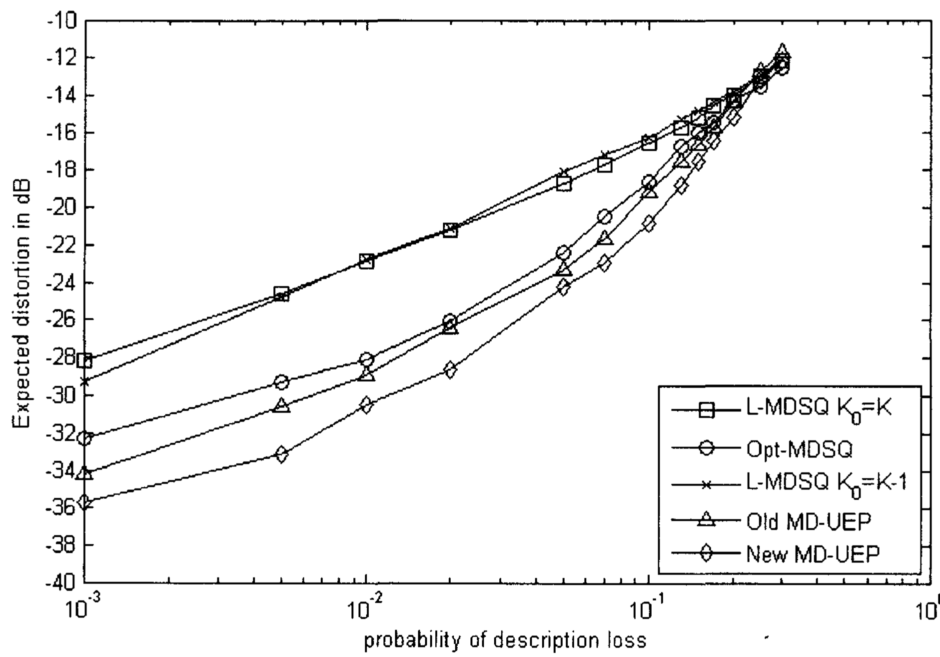


Figure 5.14: Comparison between MDSQ and MD-UEP, K=4, R=2.

optimization is not applied, and the gap between Opt-MDSQ and new MD-UEP reaches as much as 3.81 dB, which is bigger than in any other case.

Conclusion. When the probability of description loss is small, L-MDSQ with several joint codebooks stored is very competitive with Opt-MDSQ. New MD-UEP, surprisingly, performs very well, not only under the aspect of reducing the storage space, but also under the aspect of producing low reconstruction distortion. Its performance is close to Opt-MDSQ in general.

Chapter 6

Conclusion

High storage space needs of an MDC scheme could be an issue, especially in applications where the memory resources are scarce. Such applications motivate the study of MDC with reduced storage space decoder. In this work we proposed two techniques for symmetric MDC with reduced storage space decoder.

One of the two techniques is MDSQ with linear joint decoders. For such an MDSQ, we store all side codebooks and a few joint codebooks, and generate the other joint codebooks as linear combinations of the side codebooks. This way, the storage space is reduced, but under the cost of decreased performance. We addressed the problem of optimal design of such systems. The algorithm we proposed is a generalized Lloyd algorithm, similar to the one introduced by Vaishampayan [45], to which we added an IA optimization step at each iteration, to overcome the problem that good IA's for more than two descriptions are yet to be found. We also solved an additional challenge in the decoder optimization by proving that the problem is a convex optimization problem with a closed form solution (under some mild conditions). Our experimental results reveal that the impact of IA optimization step increases as more joint codebooks are stored at the decoder. Additionally, we observe that when the probability of description loss is very small, the performance of such MDSQ with side codebooks and a central codebook stored is very close to the performance of traditional MDSQ with all codebooks stored.

The other technique we proposed is an improvement to the traditional MD-UEP scheme. MD-UEP is also an MDC method of reduced storage space decoder. The new scheme strictly outperforms the traditional one. We evaluated and compared the asymptotical performance of both schemes for a Gaussian memoryless source. Our analysis shows that the improvement can reach up to 1.68 dB in the case of three descriptions and is over 1 dB for up to 10 descriptions and low probability of description loss.

We also performed an experimental study of the proposed MDC techniques and compared them with traditional MDSQ and MD-UEP. We observed that the new MD-UEP, surprisingly, performs very well, not only under the aspect of reducing the storage space, but also under the aspect of producing low reconstruction distortion. Traditional MDSQ and new MD-UEP have close performance in general. This is contrary to the intuition that MD-UEP should be worse than Opt-MDSQ, since the former has a higher amount of redundancy. We attribute this result to the sub-optimality of the design of the traditional MDSQ, which is due to the imperfection of the IA.

Bibliography

[1] T. Berger. *Rate Distortion Theory: A Mathematical Basis for Data Compression.* Prentice-Hall, Englewood Cliffs, 1971.

[2] T. Y. Berger-Wolf and E. M. Reingold. Index assignment for multichannel communication under failure. *IEEE Trans. Information Theory*, vol. 48, no. 10, pp. 2656-2668, Oct. 2002.

[3] R. Bhatia. *Positive Definite Matrices.* Princeton University Press, New Jersey, 2007.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, Cambridge, 2004.

[5] J. D. Bruce. Optimum quantization. *Sc. D. Thesis*, M. I. T., May 14, 1964.

[6] H. Brunk and N. Farvardin. Fixed-rate successively refinable scalar quantizers. In *Proc. IEEE Data Compression Conf.*, pp. 250-259, Mar. 1996.

[7] P. A. Chou, S. Mehrotra, and A. Wang. Multiple description decoding of overcomplete expansions using projections onto convex sets. In *Proc. IEEE Data Compression Conf.*, pp. 72-81, Mar. 1999.

[8] T. Cover and J. Thomas. *Elements of Information Theory.* Wiley and Sons, New York, 1991.

[9] S. Dumitrescu. Speed-up of encoder optimization step in multiple description quantizer design. In *Proc. IEEE Data Compression Conf.*, pp. 382-391, Mar. 2008.

[10] S. Dumitrescu and X. Wu. Optimal multiresolution quantization for scalable multimedia coding. *Information Theory Workshop*, pp. 139-142, Oct. 2002.

[11] S. Dumitrescu and X. Wu. Optimal two-description scalar quantizer design. *Algorithmica*, vol. 41, no. 4, pp. 300. 269-287, Feb. 2005.

[12] S. Dumitrescu and X. Wu. On global optimaility of gradient descent algorithms for fixed-rate scalar multiple description quantizer design. In *Proc. IEEE Data Compression Conf.*, pp. 388-397, Mar. 2005.

[13] S. Dumitrescu and X. Wu. Lagrangian Optimization of Two-description Scalar Quantizers. *IEEE Trans. Information Theory*, vol. 53, no. 11, pp. 3990-4012, Nov. 2007.

[14] S. Dumitrescu, X. Wu, and Z. Wang. Globally optimal uneven error-protected packetization of scalable code streams. *IEEE Trans. on Multimedia*, vol. 6, no. 2, pp.230-239, Apr. 2004.

[15] S. Dumitrescu and T. Zheng. Improved multiple description framework based on successively refinable quantization and uneven erasure protection. In *Proc. IEEE Data Compression Conf.*, pp. 514-514, Mar. 2008.

[16] M. Effros and D. Dugatkin. Multiresolution vector quantization. *IEEE Trans. Information Theory*, vol. 50, no. 12, pp. 3130-3145, Dec. 2004.

[17] P. E. Fleischer. Sufficient Conditions for Achieving Minimum Distortion in A Quantizer. *IEEE Int. Conv. Rec.*, part 1, pp. 104-111, 1964.

[18] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression.* Kluwer Academic Publishers, 1992.

[19] V. K Goyal. Multiple description coding: compression meets the network. *IEEE Signal Processing Magazine*, pp. 74-93, Sept. 2001.

[20] V. K Goyal and J. Kovačević. Optiaml multiple description transform coding of Gaussian vectors. In *Proc. IEEE Data Compression Conf.*, pp. 388-397, Mar. -Apr., 1998.

[21] V. K Goyal and J. Kovačević. Generalized multiple description coding with correlating transforms. *IEEE Trans. Information Theory*, vol. 47, Sept. 2001.

[22] V. K Goyal, J. Kovačević, and M. Vetterli. Multiple description transform coding: Robustness to erasures using tight frame expansions. In *Proc. IEEE Int. Symp. Information Theory*, pp. 408, Aug. 1998.

[23] V. K Goyal, J. Kovačević, and M. Vetterli. Quantized frame expansions as source-channel codes for erasure channels. In *Proc. IEEE Data Compression Conf.*, pp. 326-335, Mar. 1999.

[24] V. K Goyal, M. Vetterli, and N. T. Thao. Quantized overcomplete expansions in $\mathbb{R}^N$: Analysis, synthesis, and algorithms. *IEEE Trans. Information Theory*, vol. 44, pp. 16-31, Jan. 1998.

[25] H. Jafarkhani, H. Brunk, and N. Farvardin. Entropy constrained successively refinable scalar quantization. In *Proc. IEEE Data Compression Conf.*, Snowbird, UT, pp. 337-346, Mar. 1997.

[26] J. C. Kieffer. Uniqueness of Locally Optimal Quantizer for Log-Concave Density and Convex Error Weighting Function. *IEEE Trans. Information Theory*, vol. IT-29, no. 1, pp. 42-47, Jan. 1983.

[27] S. P. Lloyd. Least squares quantization in PCM. *IEEE Trans. Information Theory*, IT-28, pp. 129-137, 1982.

[28] J. Max. Quantizing for minimum distortion. *IEEE Trans. Information Theory*, vol. 6, pp. 7-12, Mar. 1960.

[29] A. E. Mohr, E. A. Riskin, and R. E. Ladner. Graceful degradation over packet erasure channels through forward error correction. In *Proc. IEEE Data Compression Conf.*, pp. 92-101, Mar. 1999.

[30] D. Muresan and M. Effros. Quantization as histogram segmentation: globally optimal scalar quantizer design in network systems. In *Proc. IEEE Data Compression Conf.*, pp. 302-311, Apr. 2002.

[31] L. Ozarow. On a source-coding problem with two channels and three receivers. *Bell Syst. Tech. J.*, 59:1909-1921, Dec.1980.

[32] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization Algorithms and Complexity.* Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982.

[33] R. Puri, S. S. Pradhan, and K. Ramchandran. n-channel symmetric multiple descriptions - part ii: an achievable rate-distortion region. *IEEE Trans. Information Theory*, vol. 51, pp. 1377-1392, Apr. 2005.

[34] R. Puri and K. Ramchandran. Multiple description source coding through forward error correction codes. *Proc. 33rd Asilomar Conf. Signals, Systems, and Computers*, vol. 1, pp. 342-346, Oct. 1999.

[35] S. Rangan and V. K Goyal. Recursive consistent estimation with bounded noise. *IEEE Trans. Information Theory*, vol. 47, pp. 457-464, Jan. 2001.

[36] L. Rizzo. Effective erasure codes for reliable computer communication protocols. *Comput. Commun. Rev.*, vol. 27, pp. 24-36, Apr. 1997.

[37] S.D. Servetto, V. A. Vaishampayan, and N. J. A. Sloane. Multiple description lattice vector quantization. In *Proc. IEEE Data Compression Conf.*, pp. 13-22, Mar. 1999.

[38] C. E. Shannon. Coding theorems for a discrete source with a fidelity criterion. *IRE National Convention Record*, 7(4):142-163, 1959.

[39] D. K. Sharma. Design of absolutely optimal quantizers for a wide class of distortion measures. *IEEE Trans. Information Theory*, vol. 24, pp. 693-702, Nov. 1978.

[40] V. Stankovic, R. Hamzaoui, and Z. Xiong. Efficient channel code rate selection algorithms for forward error correction of packetized multimedia bitstreams in varying channels. *IEEE Trans. Multimedia, Special Issue on Streaming Media*, vol. 6, no.2, pp. 240-248, Apr. 2004.

[41] C. Tian and S. S. Hemami. Sequential design of multiple description scalar quantizers. In *Proc. IEEE Data Compression Conf.*, pp. 32-41, Mar. 2004.

[42] C. Tian, S. Mohajer, and S. Diggavi. On the symmetric Gaussian multiple description rate-distortion function. In *Proc. IEEE Data Compression Conf.*, pp. 402-411, Mar. 2008.

[43] A. V. Trushkin. Sufficient conditions for uniqueness of a locally optimal quantizer for a class of convex error weighting functions. *IEEE Trans. Information Theory*, vol. 28, no. 2, pp. 187-198, Mar. 1982.

[44] A. V. Trushkin. Monotony of Lloyd's Method II for Log-Concave Density and Convex Error Weighting Function. *IEEE Trans. Information Theory*, vol. 30, no. 2, pp. 380-383, Mar. 1984.

[45] V. A. Vaishampayan. Design of multiple description scalar quantizers. *IEEE Trans. Information Theory*, vol. 39, pp. 821-834, May 1993.

[46] V. A. Vaishampayan and J. C. Batllo. Asymptotic analysis of multiple description quantizers. *IEEE Trans. Information Theory*, vol. 44, no. 1, pp. 278-284, Jan. 1998.

[47] V. A. Vaishampayan and J. Domaszewicz. Design of entropy-constrained multiple-description scalar quantizers. *IEEE Trans. Information Theory*, vol. 40, pp. 245-250, Jan. 1994.

[48] V. A. Vaishampayan, N. J. A. Sloane, and S. D. Servetto. Multiple description vector quantization with lattice codebooks: Design and analysis. *IEEE Trans. Information Theory*, 47(5):1718-1734, July 2001.

[49] Y. Wang, M. T. Orchard, and A. R. Reibman. Multiple description image coding for noisy channels by pairing tranform coefficients. In *Proc. IEEE Workshop on Multimedia Signal Processing*, Princeton, NJ, pp. 419-424, June 1997.

[50] H. Wang and P. Viswanath. Vector Gaussian multiple description with two levels of receivers. *IEEE Trans. Information Theory*, submitted for publication, Nov. 2006.

[51] H. Wang and P. Viswanath. Vector Gaussian multiple description with individual and central receivers. *IEEE Trans. Information Theory*, vol. 53, no. 6, pp. 2133-2153, Jun. 2007.

[52] X. Wu. Optimal quantization by matrix searching. *Journal of Algorithms*, vol. 12, no. 4, pp. 663-673, Dec. 1991.

[53] X. Wu and K. Zhang. Quantizer monotonicities and glabally optimal scalar quantizer design. *IEEE Trans. Information Theory*, vol. 39, no. 3, pp. 1049-1053, May 1993.

Appendix A

Derivation of Asymptotical Performance Analysis of MD-UEP

A.1 General Case

Here we present some details of the derivation of the asymptotical analysis of Section 4.4. This derivation is inspired by the work of Puri *et al.* [34].

Let k_0 be the largest integer such that

$$slope(P_{k_0-2}P_{k_0-1}) \leq slope(P_{k_0-1}P_K). \tag{A.1}$$

In other words, k_0 is the largest integer between 1 and K such that $U(k_0-1)(k_0-1) \leq U(k_0) + U(k_0+1) + \cdots + U(K)$. Then we show that the extreme points of the curve $\mathcal{P}$ are $P_0, P_1, \cdots, P_{k_0-1}, P_K$.

Proof.

$$slope(P_k P_{k+1}) \le slope(P_{k+1} P_{k+2}) \tag{A.2}$$

is equivalent to

$$\frac{U(k+1)}{(1-\frac{1}{k+2})-(1-\frac{1}{k+1})} \le \frac{U(k+2)}{(1-\frac{1}{k+3})-(1-\frac{1}{k+2})}, \tag{A.3}$$

where $0 \le k \le K-3$. By substituting (4.4) into the above inequality and after some math manipulations, we obtain

$$\frac{k+1}{K-k-1}q - \frac{k+3}{k+2}(1-q) \le 0. \tag{A.4}$$

Define

$$f(k) = \frac{1}{q}k^2 + (K-1+\frac{4-K}{q})k + 3K - 1 + \frac{3-3K}{q}, \tag{A.5}$$

where $0 \le k \le K-3$. Then (A.4) is equivalent to

$$f(k) \le 0. \tag{A.6}$$

$f(k)$ is a quadratic function, and since $\frac{1}{q} > 0$, the parabola opens upward, and

$$\begin{aligned}\Delta =&(qK+4-K-q)^2 - 4(3+3qK-3K-q)\\ =&(q-1)^2K^2 + (-2q^2-2q+4)K + (q-2)^2 > 0.\end{aligned}$$

To ensure $f(0) \le 0$, K must satisfy the constraint $K \ge \frac{3-q}{3-3q}$. When $q \le \frac{1}{2}$ and any $K \ge 2$, $f(0) \le 0$. If $f(K-3) \le 0$, K must satisfy the constraint

$$K \le \frac{2+q+\sqrt{-7q^2+4q+4}}{2q}.$$

When $k = K - 2$, (A.2) becomes

$$\frac{U(K-1)}{(1-\frac{1}{K})-(1-\frac{1}{K-1})} \leq \frac{U(K)}{1-(1-\frac{1}{K})}. \tag{A.7}$$

To satisfy this condition, it is needed that

$$K \leq \frac{q+\sqrt{-3q^2+4q}}{2q}.$$

Therefore, when $q \leq \frac{1}{2}$ and

$$K \in [2, \min(\frac{2+q+\sqrt{-7q^2+4q+4}}{2q}, \frac{q+\sqrt{-3q^2+4q}}{2q})], \tag{A.8}$$

we have $f(k) \leq 0$ for all k, hence the curve is convex, which implies that all points are extreme points. According to the definition of k_0, $k_0 = K$ and the claim is proved. If (A.8) is not satisfied, it follows that the quadratic function $f(k)$ has a zero value x_0 in the interval $[0, K-2)$. It follows that for all $0 \leq k \leq \lfloor x_0 \rfloor$, $f(k) \leq 0$, and for all $\lfloor x_0 \rfloor < k \leq K-2$, $f(k) > 0$. This implies that there is some $k_1 \leq \lfloor x_0 \rfloor$ such that all points $P_1, \cdots, P_{k_1}$ are extreme points and all points $P_{k_1+1}, \cdots, P_{K-1}$ are not. Since k_1 must satisfy $slope(P_{k_1-1}P_{k_1}) \leq slope(P_{k_1}P_K)$ in order to be an extreme point, it follows that $k_1 = k_0 - 1$ and the claim is proved. □

According to the observation of [34] (Page 345), when $R_1 \geq 0$ is satisfied, the problem (4.5) is equivalent to the following

$$\text{minimize} \quad \sigma^2 \sum_{k=1}^{k_0-1} U'(k) 2^{-2R_k} + \sigma^2 U'(k_0) 2^{-2R_{k_0}} \tag{A.9}$$

$$\text{subject to} \quad \sum_{k=1}^{k_0-1} M'(k) R_k + M'(k_0) R_{k_0} = R, , \tag{A.10}$$

where $R_{k_0} = R_{k_0+1} = \cdots = R_K$; $U'(k) = U(k)$ for $1 \leq k \leq k_0 - 1$ and $U'(k_0) = \sum_{i=k_0}^{K} U(i)$; and $M'(k) = \frac{1}{k(k+1)}$ for $1 \leq k \leq k_0 - 1$ and $M'(k_0) = \sum_{i=k_0}^{K-1} \frac{1}{i(i+1)} + \frac{1}{K}$. The above problem is a convex problem with one equality constraint [1]. To solve this problem, we introduce the Lagrangian

$$\mathcal{L}(R_1, \cdots, R_k, \lambda) = \sigma^2 \sum_{k=1}^{k_0} U'(k) 2^{-2R_k} + \lambda(\sum_{k=1}^{k_0} M'(k) R_k - R), \tag{A.11}$$

where λ is some nonnegative value. By setting all partial derivatives of the Lagrangian function with respect to $R_1, \cdots, R_{k_0}$ to zero, we obtain

$$-2 \ln 2\sigma^2 U'(k) 2^{-2R_k} + \lambda M'(k) = 0, \quad k = 1, \cdots, k_0. \tag{A.12}$$

Then we can get the values of $R_1, \cdots, R_{k_0}$ by solving the above equation:

$$R_k = -\frac{1}{2} \log_2 \frac{\lambda M'(k)}{2 \ln 2\sigma^2 U'(k)}, \quad k = 1, \cdots, k_0. \tag{A.13}$$

Because $(\frac{U'(k)}{M'(k)})_{1 \leq k \leq k_0}$, is an increasing sequence, it follows that $R_1 \leq R_2 \leq \cdots \leq R_{k_0}$. Next we substitute (A.13) into (A.10), to get the value of λ:

$$\sum_{k=1}^{k_0-1} \frac{1}{k(k+1)} (-\frac{1}{2}) \log_2 \frac{\lambda}{2 \ln 2\sigma^2 k(k+1) U(k)} + \frac{1}{k_0} (-\frac{1}{2}) \log_2 \frac{\lambda}{2 \ln 2\sigma^2 k_0 \sum_{k=k_0}^{K} U(k)} = R,$$

it follows that

$$\lambda = 2\sigma^2 \cdot 2^A \cdot 2^{-2R} \ln 2, \tag{A.14}$$

where

$$A = \sum_{k=1}^{k_0-1} \frac{1}{k(k+1)} \log_2(k(k+1)U(k)) + \frac{1}{k_0} \log_2(k_0 \sum_{k=k_0}^{K} U(k)). \tag{A.15}$$

[1] As proved in [34] the solution satisfies the inequality constraints of (4.5).

Finally, we obtain the expression of D_{old} by substituting (A.13) and (A.14) into (A.9):

$$\begin{aligned} D_{old} =& \sigma^2 \sum_{k=1}^{k_0-1} U(k) 2^{-2R_k} + \sigma^2 U'(k_0) 2^{-2R_{k_0}} \\ =& \sigma^2 \sum_{k=1}^{k_0-1} U(k) \frac{\lambda}{2 \ln 2 \sigma^2 k(k+1) U(k)} \\ &+ \sigma^2 (\sum_{k=k_0}^{K} U(k)) \frac{\lambda}{2 \ln 2 \sigma^2 k_0 \sum_{k=k_0}^{K} U(k)} \\ =& \sum_{k=1}^{k_0-1} \frac{\lambda}{2 \ln 2 k(k+1)} + \frac{\lambda}{2 \ln 2 k_0} \\ =& \frac{\lambda}{2 \ln 2} \\ =& \sigma^2 2^A 2^{-2R}. \end{aligned} \tag{A.16}$$

For the new framework, we will prove that when $K \geq 3$ and $q \leq \frac{1}{K}$, all points are extreme points.

Proof.

$$slope(P_k P_{k+1}) \leq slope(P_{k+1} P_{k+2}) \tag{A.17}$$

is equivalent to

$$\frac{V(k+1)}{(1 - \frac{1}{k+2}) - (1 - \frac{1}{k+1})} \leq \frac{V(k+2)}{(1 - \frac{1}{k+3}) - (1 - \frac{1}{k+2})}, \tag{A.18}$$

where $0 \leq k \leq K-3$. By substituting (4.6) and (4.7) into the above inequality and after some math manipulations, we obtain

$$(k+1)(k+2)q \leq (k+3)K - k - 2(1-q). \tag{A.19}$$

Define

$$f(k) = k^2 + (qK + 5 - K - 2q)k + 3Kq + 6 - 3K - 4q, \tag{A.20}$$

where $0 \le k \le K-3$. Then (A.19) is equivalent to

$$f(k) \le 0. \tag{A.21}$$

$f(k)$ is a quadratic function, with the parabola opening upward, and

$$\begin{aligned}\Delta =&(qK+5-K-2q)^2-4(6+3qK-3K-4q)\\ =&(q-1)^2K^2-2(2q^2-q-1)K+(2q-1)^2>0.\end{aligned}$$

To ensure $f(0) \le 0$, K must satisfy the constraint $K \ge \frac{6-4q}{3-3q}$. When $q \le \frac{1}{2}$ and $K \ge 3$, $f(0) \le 0$. If $q \le \frac{1}{K}$, we have $f(K-3) \le 0$.

When $k = K-2$, (A.2) becomes

$$\frac{V(K-1)}{(1-\frac{1}{K})-(1-\frac{1}{K-1})} \le \frac{V(K)}{1-(1-\frac{1}{K})}. \tag{A.22}$$

By substituting (4.6) and (4.7) into the above inequality and after some math manipulations, we obtain

$$(K-1)(1-q)^{K-2}q \le 1. \tag{A.23}$$

It is straightforward that if $q \le \frac{1}{K}$, the above inequality is true. Therefore, if $K \ge 3$ and $q \le \frac{1}{K}$, $f(k) \le 0$ for all k, hence the curve is convex, which implies that all points are extreme points, hence the claim is proved. □

Now we proceed to compute the value of D_{new}. When $K \ge 3$ and $q \le \frac{1}{K}$, the problem (4.8) is equivalent to the following (according to [34]).

$$\text{minimize} \quad \sigma^2 \sum_{k=1}^{K} V(k) 2^{-2R_k} \tag{A.24}$$

$$\text{subject to} \quad \sum_{k=1}^{K} M(k) R_k = R, \tag{A.25}$$

where $V(k) = \frac{K-k}{K}U(k)$ for $1 \leq k \leq K-1$ and $V(K) = \frac{1-q}{1-q^K}$; and $M(k) = \frac{1}{k(k+1)}$ for $1 \leq k \leq K-1$ and $M(K) = \frac{1}{K}$. The above problem is a convex problem with one equality constraint. To solve this problem, we introduce the Lagrangian

$$\mathcal{L}(R_1, \cdots, R_k, \lambda) = \sigma^2 \sum_{k=1}^{K} V(k) 2^{-2R_k} + \lambda (\sum_{k=1}^{K} M(k) R_k - R), \tag{A.26}$$

where λ is some nonnegative value. By setting all partial derivatives of the Lagrangian function with respect to $R_1, \cdots, R_K$ to zero, we obtain

$$-2 \ln 2 \sigma^2 V(k) 2^{-2R_k} + \lambda M(k) = 0, \quad k = 1, \cdots, K. \tag{A.27}$$

Then we can get the values of $R_1, \cdots, R_K$ by solving the above equation:

$$R_k = -\frac{1}{2} \log_2 \frac{\lambda M(k)}{2 \ln 2 \sigma^2 V(k)}, \quad k = 1, \cdots, K. \tag{A.28}$$

Next we substitute (A.28) into (A.25), to get the value of λ:

$$\sum_{k=1}^{K-1} \frac{1}{k(k+1)} (-\frac{1}{2}) \log_2 \frac{\lambda}{2 \ln 2 \sigma^2 k(k+1) V(k)} + \frac{1}{K} (-\frac{1}{2}) \log_2 \frac{\lambda}{2 \ln 2 \sigma^2 K V(K)} = R,$$

it follows that

$$\lambda = 2\sigma^2 \cdot 2^B \cdot 2^{-2R} \ln 2, \tag{A.29}$$

where

$$B = \sum_{k=1}^{K-1} \frac{1}{k(k+1)} \log_2(k(k+1)V(k)) + \frac{1}{K} \log_2(KV(K)). \tag{A.30}$$

Finally, we obtain the expression of D_{new} by substituting (A.28) and (A.29) into

(A.24):

$$\begin{aligned}
D_{new} =& \sigma^2 \sum_{k=1}^{K} V(k) 2^{-2R_k} \\
=& \sigma^2 \sum_{k=1}^{K} V(k) \frac{\lambda M(k)}{2 \ln 2 \sigma^2 V(k)} \\
=& \sigma^2 \frac{\lambda}{2 \ln 2 \sigma^2} \sum_{k=1}^{K} M(k) \\
=& \frac{\lambda}{2 \ln 2} \\
=& \sigma^2 2^B 2^{-2R}.
\end{aligned} \tag{A.31}$$

A.2 Two-Description UEP

For the case of two descriptions, problem (4.5) can be written as

$$\text{minimize} \quad \sigma^2 \frac{2q}{1+q} 2^{-2R_1} + \sigma^2 \frac{1-q}{1+q} 2^{-2R_2} \tag{A.32}$$

$$\text{subject to} \quad 0 \le R_1 \le R_2 \tag{A.33}$$

$$\frac{1}{2} R_1 + \frac{1}{2} R_2 = R. \tag{A.34}$$

We first simplify the problem by eliminating the variable R_2. Note that (A.34) implies $R_2 = 2R - R_1$. By replacing R_2 in (A.32) and (A.33), the optimization problem becomes

$$\text{minimize} \quad \sigma^2 \frac{2q}{1+q} 2^{-2R_1} + \sigma^2 \frac{1-q}{1+q} 2^{-2(2R-R_1)} \tag{A.35}$$

$$\text{subject to} \quad 0 \le R_1 \le R. \tag{A.36}$$

Further, by performing the change of variable $y = 2^{2R_1}$, we obtain the following equivalent optimization problem:

$$\text{minimize} \quad f(y) \tag{A.37}$$

$$\text{subject to} \quad 1 \leq y \leq 2^{2R}, \tag{A.38}$$

where

$$f(y) = \sigma^2 \frac{2q}{1+q}\frac{1}{y} + \sigma^2 \frac{1-q}{1+q} 2^{-4R} y. \tag{A.39}$$

By taking the derivative of $f(y)$ and setting it to zero, we obtain

$$y^* = \sqrt{\frac{2q}{1-q}} 2^{2R},$$

and

$$f(y^*) = \sigma^2 \frac{2}{1+q} \sqrt{2q(1-q)} 2^{-2R}.$$

When $q \in [\frac{1}{1+2^{4R+1}}, \frac{1}{3}]$, the value of y^* satisfies the constraint (A.38), so the optimal solution is y^*, and the optimal value is $f(y^*)$. When $q \in (0, \frac{1}{1+2^{1+4R}})$, $f(y)$ is monotonically increasing in $[1, 2^{2R}]$, so the optimal value is $f(1)$. When $q \in (\frac{1}{3}, 1]$, $f(y)$ is monotonically decreasing in $[1, 2^{2R}]$, so the optimal value is $f(2^{2R})$. Hence, it follows that

$D_{old} = \sigma^2(\frac{2q}{1+q} + \frac{1-q}{1+q} 2^{-4R})$ for $q \in (0, \frac{1}{1+2^{1+4R}})$.

$D_{old} = \sigma^2(\frac{2}{1+q}\sqrt{2q(1-q)} 2^{-2R})$ for $q \in [\frac{1}{1+2^{1+4R}}, \frac{1}{3}]$.

$D_{old} = \sigma^2 2^{-2R}$ for $q \in (\frac{1}{3}, 1]$.

For the new framework, the same idea can be applied. First we write the optimization problem as

$$\text{minimize} \quad \sigma^2 \frac{q}{1+q} 2^{-2R_1} + \sigma^2 \frac{1}{1+q} 2^{-2R_2} \tag{A.40}$$

$$\text{subject to} \quad 0 \leq R_1 \leq R_2 \tag{A.41}$$

$$\frac{1}{2} R_1 + \frac{1}{2} R_2 = R. \tag{A.42}$$

Let $y = 2^{2R_1}$, and form a function $f(y)$ of y

$$f(y) = \sigma^2 \frac{q}{1+q} \frac{1}{y} + \sigma^2 \frac{1}{1+q} 2^{-4R} y. \tag{A.43}$$

By taking the derivative of $f(y)$ and setting it to zero, we obtain

$$y^* = \sqrt{q} 2^{2R},$$

and

$$f(y^*) = \sigma^2 \frac{2}{1+q} \sqrt{q} 2^{-2R}.$$

Note that from the constraint (A.41), y has to be inside the region $[1, 2^{2R}]$. When $q \in [2^{-4R}, 1]$, $y^* \in [1, 2^{2R}]$, hence is the optimal solution, and the optimal value is $f(y^*)$. When q is less than 2^{-4R}, $f(y)$ is monotonically increasing in $[1, 2^{2R}]$, so the optimal value is $f(1)$. Hence, it follows that

$D_{new} = \sigma^2 (\frac{q}{1+q} + \frac{1}{1+q} 2^{-4R})$ for $q \in (0, 2^{-4R})$.

$D_{new} = \sigma^2 (\frac{2}{1+q} \sqrt{q} 2^{-2R})$ for $q \in [2^{-4R}, 1]$.