

Thesis for the degree of Doctor of Philosophy

Source–Channel Coding in Networks

Niklas Wernersson



KTH Electrical Engineering

Communication Theory
School of Electrical Engineering
KTH (Royal Institute of Technology)

Stockholm 2008

Wernersson, Niklas
Source-Channel Coding in Networks

Copyright ©2008 Niklas Wernersson except where
otherwise stated. All rights reserved.

TRITA-EE 2008:028
ISSN 1653-5146
ISBN 978-91-7415-002-5

Communication Theory
School of Electrical Engineering
KTH (Royal Institute of Technology)
SE-100 44 Stockholm, Sweden
Telephone + 46 (0)8-790 7790

Abstract

The aim of source coding is to represent information as accurately as possible using as few bits as possible and in order to do so redundancy from the source needs to be removed. The aim of channel coding is in some sense the contrary, namely to introduce redundancy that can be exploited to protect the information when being transmitted over a nonideal channel. Combining these two techniques leads to the area of joint source–channel coding which in general makes it possible to achieve a better performance when designing a communication system than in the case when source and channel codes are designed separately. In this thesis four particular areas in joint source–channel coding are studied: analog (i.e. continuous) bandwidth expansion, distributed source coding over noisy channels, multiple description coding (MDC) and soft decoding.

A general analog bandwidth expansion code based on orthogonal polynomials is proposed and analyzed. The code has a performance comparable with other existing schemes. However, the code is more general in the sense that it is implementable for a larger number of source distributions.

The problem of distributed source coding over noisy channels is studied. Two schemes are proposed and analyzed for this problem which both work on a sample by sample basis. The first code is based on scalar quantization optimized for a certain channel characteristics. The second code is nonlinear and analog.

Two new MDC schemes are proposed and investigated. The first is based on sorting a frame of samples and transmitting, as side-information/redundancy, an index that describes the resulting permutation. In case that some of the transmitted descriptors are lost during transmission this side information (if received) can be used to estimate the lost descriptors based on the received ones. The second scheme uses permutation codes to produce different descriptions of a block of source data. These descriptions can be used jointly to estimate the original source data. Finally, also the MDC method multiple description coding using pairwise correlating transforms as introduced by Wang et al. is studied. A modification of the quantization in this method is proposed which yields a performance gain.

A well known result in joint source–channel coding is that the performance of a communication system can be improved by using soft decoding of the channel output at the cost of a higher decoding complexity. An alternative to this is to quantize the soft information and store the pre-calculated soft decision values in a

lookup table. In this thesis we propose new methods for quantizing soft channel information, to be used in conjunction with soft-decision source decoding. The issue on how to best construct finite-bandwidth representations of soft information is also studied.

Keywords: source coding, channel coding, joint source–channel coding, bandwidth expansion, distributed source coding, multiple description coding, soft decoding.

Acknowledgements

During my Ph.D. studies I have received a lot of help from various persons. Most important by far has been the help from my supervisor Professor Mikael Skoglund. Mikael has been supportive since the day I started at KTH and he has always been able to find time for our research discussions in his otherwise busy schedule. He has been actively interested in my work and always done his utmost in order to guide me. This has made him a great mentor and I would like to express my gratitude for all the help during these years.

I would further like to thank Professor Tor Ramstad for inviting me as a guest to NTNU. The visit at Tor's research group was a great experience.

I devote special thanks to Tomas I Andersson, Johannes Karlsson and Pål Anders Floor for work resulting in joint publications.

I would also like to thank all my current and past colleagues at KTH for all the valuable support, both in a scientific way as well as a social way. In particular I would like to thank: Joakim for many interesting talks at KTH-hallen. Karl for teaching me about good manners as well as teaming up with me in F.E.M. Mats, Erik and Peter H for valuable research discussions. Svante for educating me about Ransäter. Magnus J and Per for XC-skiing inspiration. David and George for teaching me a thing or two about Stockholm. Xi and Peter v. W for inspiring me to smuggle a bike. Lei for always being so helpful. Patrick for many interesting music conversations. Tùng for joining me in eating strange food in France. Henrik for letting me make fun of his lunch habits. Nina and Niklas J for great parties. Tomas I and Marc for philosophical conversations. Björn L for all the never ending information at bönebänken. And finally, Karin, Marie and Annika for spreading happiness around the office with their positive way.

I further wish to thank the faculty opponent and the grading committee for taking the time to be involved in the thesis defence.

Finally, I would like to thank my family back home (including grandmother Ingeborg and the Brattlöfs; Greta and the Nordling families) since they have managed to support me during these years, although I have been at some distance away. The support was highly appreciated. Last but certainly not least, thank you Barbara for everything. Without you this journey would have been a lot tougher and far less enjoyable.

Contents

| | |
|---|------------|
| Abstract | i |
| Acknowledgements | iii |
| Contents | v |
| | |
| I Introduction | 1 |
| Introduction | 1 |
| 1 Source Coding | 2 |
| 1.1 Lossless Coding | 3 |
| 1.2 Lossy Coding | 3 |
| 2 Channel Coding | 9 |
| 3 The Source–Channel Separation Theorem | 10 |
| 4 Analog Source–Channel Coding | 11 |
| 4.1 Analog Bandwidth Expansion | 12 |
| 4.2 Analog Bandwidth Compression | 13 |
| 5 Distributed Source Coding | 14 |
| 5.1 Theoretical Results | 14 |
| 5.2 Practical Schemes | 18 |
| 6 Multiple Description Coding | 19 |
| 6.1 Theoretical Results | 20 |
| 6.2 Practical Schemes | 21 |
| 7 Source Coding for Noisy Channels | 27 |
| 7.1 Scalar Source Coding for Noisy Channels | 28 |
| 7.2 Vector Source Coding for Noisy Channels | 29 |
| 7.3 Index Assignment | 31 |
| 8 Contributions of the Thesis | 31 |
| References | 36 |

II Included papers 45

| | |
|--|-----------|
| A Polynomial Based Analog Source–Channel Codes | A1 |
| 1 Introduction | A1 |
| 2 Problem Formulation | A2 |
| 3 Orthogonal Polynomials | A4 |
| 3.1 Definition | A4 |
| 3.2 The recurrence formula and zeros | A4 |
| 3.3 Gram–Schmidt | A5 |
| 3.4 The Christoffel–Darboux identity | A6 |
| 4 Encoding and Decoding | A6 |
| 5 Analysis | A7 |
| 5.1 Distance | A7 |
| 5.2 Stretch | A8 |
| 5.3 Dimensions | A8 |
| 6 Simulations | A9 |
| 6.1 Choosing \mathcal{I} | A9 |
| 6.2 Optimal companding and benchmark systems | A10 |
| 6.3 Results | A11 |
| 7 Conclusions | A13 |
| References | A13 |
| | |
| B Distributed Quantization over Noisy Channels | B1 |
| 1 Introduction | B1 |
| 2 Problem Formulation | B2 |
| 2.1 Binary Symmetric Channel | B2 |
| 2.2 Gaussian Channel | B3 |
| 3 Analysis | B4 |
| 3.1 Encoder for BSC | B4 |
| 3.2 Encoder for Gaussian Channel | B5 |
| 3.3 Decoder | B6 |
| 3.4 Design algorithm | B6 |
| 3.5 Optimal Performance Theoretically Attainable | B7 |
| 4 Simulations | B7 |
| 4.1 Structure of the Codebook - BSC | B9 |
| 4.2 Structure of the Codebook - Gaussian Channel | B9 |
| 4.3 Performance Evaluation | B12 |
| 5 Conclusions | B14 |
| References | B16 |

| | | |
|----------|--|-----------|
| C | Nonlinear Coding and Estimation for Correlated Data in Wireless Sensor Networks | C1 |
| 1 | Introduction | C1 |
| 2 | Problem Formulation | C2 |
| 3 | Discussion and Proposed Scheme | C3 |
| 3.1 | $\sigma_w^2 > 0$ and $\sigma_n^2 \rightarrow 0$ | C3 |
| 3.2 | $\sigma_w^2 = 0$ and $\sigma_n^2 > 0$ | C3 |
| 3.3 | Objective | C4 |
| 3.4 | Analysis | C5 |
| 3.5 | Proposed Scheme | C7 |
| 4 | Performance Analysis | C8 |
| 5 | Simulations | C10 |
| 6 | Conclusions | C12 |
| | Appendix | C13 |
| | References | C14 |
| D | Sorting-Based Multiple Description Quantization | D1 |
| 1 | Introduction | D1 |
| 2 | Sorting-Based MDQ | D1 |
| 3 | Analysis | D4 |
| 4 | Numerical Results | D5 |
| 5 | Conclusions | D8 |
| | Appendix | D9 |
| | References | D11 |
| | Addendum to the Paper | D12 |
| E | Multiple Description Coding using Rotated Permutation Codes | E1 |
| 1 | Introduction | E1 |
| 2 | Preliminaries | E2 |
| 3 | Proposed MDC Scheme | E2 |
| 3.1 | Calculating $E[\mathbf{x} i_j]$ | E3 |
| 3.2 | Calculating $E[\mathbf{x} i_1, \dots, i_J]$ | E4 |
| 3.3 | The Effect of the Generating Random Matrices | E6 |
| 4 | Simulations | E6 |
| 4.1 | Introducing More Design Parameters | E7 |
| 5 | Conclusions | E8 |
| | Appendix | E8 |
| | Acknowledgments | E9 |
| | References | E9 |
| F | Improved Quantization in Multiple Description Coding by Correlating Transforms | F1 |
| 1 | Introduction | F1 |
| 2 | Preliminaries | F2 |

| | | |
|----------|--|-----------|
| 3 | Improving the Quantization | F4 |
| 4 | Simulation Results | F6 |
| 5 | Conclusions | F7 |
| | References | F8 |
| G | On Source Decoding Based on Finite-Bandwidth Soft Information | G1 |
| 1 | Introduction | G1 |
| 2 | Problem formulation | G2 |
| 2.1 | Re-Quantization of a Soft Source Estimate | G3 |
| 2.2 | Vector Quantization of the Soft Channel Values | G4 |
| 2.3 | Scalar Quantization of Soft Channel Values | G5 |
| 3 | Implementation of the Different Approaches | G5 |
| 3.1 | Re-Quantization of a Soft Source Estimate | G6 |
| 3.2 | Vector Quantization of the Soft Channel Values | G6 |
| 3.3 | Scalar Quantization of Soft Channel Values | G7 |
| 4 | Simulation results | G8 |
| 4.1 | SOVQ | G8 |
| 4.2 | COVQ | G9 |
| 5 | Conclusions | G10 |
| | References | G10 |

Part I

Introduction

Introduction

In daily life most people in the world uses applications resulting from what today is known as the areas of source coding and channel coding. These applications may for instance be compact discs (CDs), mobile phones, MP3 players, digital versatile discs (DVDs), digital television, voice over IP (VoIP), videostreaming etc. This has further lead to a great interest in the area of joint source–channel coding, which in general makes it possible to improve the performance of source and channel coding by designing these basic building blocks jointly instead of treating them as separate units. Source and channel coding is of interest when for instance dealing with transmission of information, i.e. data. A basic block diagram of this is illustrated in Figure 1. Here $X^n = (X_1, X_2, \dots, X_n)$ is a sequence of source data, originating for instance from sampling a continuous signal, and information about this sequence is to be transmitted over a channel. In order to do so the information in X^n needs to be described such that it can be transmitted over the channel. We also want the receiver to be able to decode the received information and produce the estimate $\hat{X}^n = (\hat{X}_1, \hat{X}_2, \dots, \hat{X}_n)$ of the original data. The task of the *encoder* is to produce a representation of X^n and the task of the *decoder* is to produce the estimate \hat{X}^n based on what was received from the channel. In source–channel coding one is interested in how to design encoders as well as decoders.

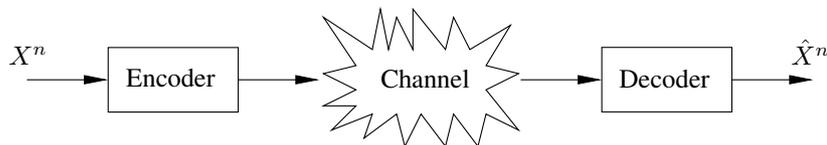


Figure 1: Basic block diagram of data transmission.

This thesis focuses on the area of joint source–channel coding and is based on the publications [2–12]. An introduction to the topic is provided and seven of the produced papers are included (papers A-G). The organization is as follows:

Part I contains an introduction where Section 1 explains the basics of source coding. Section 2 discusses channel coding which leads to Section 3 where the use of joint source–channel coding is motivated. In Section 4 one particular area of joint source–channel coding is discussed, namely analog bandwidth expansion which is also the topic of Paper A. The basics of distributed source coding is briefly summarized in Section 5 and Papers B–C deal with distributed source coding over noisy channels. Another example of joint source–channel coding is multiple description coding which is introduced in Section 6 and further developed in Papers D, E and F. Section 7 and Paper G consider source coding for noisy channels. In Section 8 the main contributions of Papers A–G will be summarized and finally, Part II of this thesis contains Papers A–G.

1 Source Coding

When dealing with transmission or storage of information this information generally needs to be represented using a discrete value. Source coding deals with how to represent this information as accurately as possible using as few bits as possible, casually speaking “compression.” The topic can be divided into two cases: *lossless coding*, which requires the source coded version of the source data to be sufficient for reproducing an identical version of the original data. When dealing with *lossy coding* this is no longer required and the aim here is rather to reconstruct an approximated version of the original data which is as good as possible.

How to define “good” is not a trivial question. In source coding this is solved by introducing some structured way of measuring quality. This measure is called *distortion* and can be defined in many ways depending on the context. See for instance [13] for a number of distortion measures applied to gray scale image coding. However, when dealing with the more theoretical aspects of source coding it is well-established practice to use the mean squared error (MSE) as a distortion measure. The dominance of the MSE distortion measure is more likely to arise from the fact that the MSE in many analytical situations can lead to nice and closed form expressions rather than its ability to accurately model the absolute truth about whether an approximation is good or bad. However, in many applications the MSE is a fairly good model for measuring quality and we will in this entire thesis use MSE as a distortion measure. Assuming the vector X^n contains the n source data values $\{X_i\}_{i=1}^n$ and the vector \hat{X}^n contains the n reconstructed values $\{\hat{X}_i\}_{i=1}^n$, the MSE is defined as

$$D_{\text{MSE}} = E \left[\frac{1}{n} \sum_{i=1}^n (X_i - \hat{X}_i)^2 \right]. \quad (1)$$

1.1 Lossless Coding

Assume X^n , where the X_i 's now are discrete values, in Figure 1 describes for example credit card numbers which are to be transmitted to some receiver. In this case it is crucial that the received information is sufficient to extract the exact original data since an approximated value of a credit card number will not be very useful. This is hence a scenario where it is important that no information is lost when performing source coding which requires lossless coding.

It seems reasonable that there should exist some kind of lower bound on how much the information in X^n can be compressed in the encoder. This bound does indeed exist and can be found by studying the *entropy rate* of the process that produces the random vector X^n . Let X be a discrete random variable with alphabet \mathcal{A}_X and probability mass function $p(x) = \Pr\{X = x\}, x \in \mathcal{A}_X$. The *entropy* $H(X)$ of X is defined as

$$H(X) = - \sum_{x \in \mathcal{A}_X} p(x) \log_2 p(x). \quad (2)$$

$H(X)$ is mainly interesting when studying independent identically distributed (i.i.d.) variables. When looking at non-i.i.d. stationary processes the *order- n entropy*

$$H_n(X^n) = -\frac{1}{n} \sum_{x^n \in \mathcal{A}_X^n} p(x^n) \log_2 p(x^n) \quad (3)$$

and the *entropy rate*

$$H_\infty(X) = \lim_{n \rightarrow \infty} H_n(X^n) \quad (4)$$

are of greater interest. Note that all these definitions measures entropy in bits which is not always the case, see e.g. [14]. It turns out that the minimum expected codeword length, L_n , per coded symbol X_i , when coding blocks of length n , satisfies

$$H_n(X^n) \leq L_n < H_n(X^n) + \frac{1}{n} \quad (5)$$

meaning that by increasing n , L_n can get arbitrary close to the entropy rate of a random stationary process. It can be shown that $H_{n+1}(X^{n+1}) \leq H_n(X^n) \forall n$ and hence, the entropy rate provides a lower bound on the average length of a uniquely decodable code. For the case of non-stationary processes the reader is referred to [15].

There are a number of coding schemes for performing lossless coding; Huffman coding, Shannon coding, Arithmetic coding and Ziv-Lempel coding are some of the most well known methods [14].

1.2 Lossy Coding

As previously stated when dealing with lossy coding we no longer have the requirement of reconstructing an identical copy of the original data X^n . Consider

for example the situation when we want to measure the height of a person; the *exact* length will be a real value meaning that there will be an infinite number of possible outcomes of the measurement. We therefore need to restrict the outcomes somehow, we could for instance assume that the person is taller than 0.5m and no taller than 2.5m. If we also assume that we do not need to measure the length more accurately than in centimeters the measurement can result in 200 possible outcomes which will approximate the exact length of the person. Approximations like this is done in source coding in order to represent (sampled) continuous signals like sound, video etc. These approximations are referred to as quantization. Furthermore, from (2) it seems intuitive that the smaller the number of possible outcomes, i.e. the coarser the measurement, the fewer bits are required to represent the measured data. Hence, there exists a fundamental tradeoff between the quality of the data (distortion) and the number of bits required per measurement (rate).

Scalar/Vector Quantization

Two fundamental tools in lossy coding are scalar and vector quantization. A scalar quantizer is a noninvertible mapping, \mathcal{Q} , of the real line, \mathbb{R} , onto a finite set of points, $\mathcal{C} = \{c_i\}_{i \in \mathcal{I}}$, where $c_i \in \mathbb{R}$ and \mathcal{I} is a finite set of indices,

$$\mathcal{Q} : \mathbb{R} \rightarrow \mathcal{C}. \quad (6)$$

The values in \mathcal{C} constitute the codebook for \mathcal{Q} . Assuming $|\mathcal{I}|$ gives the cardinality of \mathcal{I} the quantizer divides the real line into $|\mathcal{I}|$ regions \mathcal{V}_i (some of them may however be empty). These regions are called *quantization cells* and are defined as

$$\mathcal{V}_i = \{x \in \mathbb{R} : \mathcal{Q}(x) = c_i\}. \quad (7)$$

We think of i as the product of the encoder and c_i as the product of the decoder

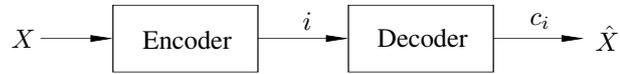


Figure 2: Illustration of an encoder and a decoder.

as shown in Figure 2. Vector quantization is a straightforward generalization of scalar quantization to higher dimensions:

$$\mathcal{Q} : \mathbb{R}^n \rightarrow \mathcal{C}. \quad (8)$$

with the modification that $c_i^n \in \mathbb{R}^n$. The quantization cells are defined as

$$\mathcal{V}_i = \{x^n \in \mathbb{R}^n : \mathcal{Q}(x^n) = c_i^n\}. \quad (9)$$

Vector quantization is in some sense the “ultimate” way to quantize a signal vector. No other coding technique exists that can do better than vector quantization for a given number of dimensions and a given rate. Unfortunately the computational complexity of vector quantizers grows exponentially with the dimension making it infeasible to use unstructured vector quantizers for high dimensions, see e.g. [16, 17] for more details on this topic.

Finally we also mention the term Voronoi region: if MSE is used as a distortion measure the scalar/vector quantizer will simply quantize the value x^n to the closest possible c_i^n . In this case the quantization cells \mathcal{V}_i are called Voronoi regions.

Rate/Distortion

As previously stated there seems to be a tradeoff between rate, R , and distortion, D , when performing lossy source coding. To study this we define the encoder as a mapping f such that

$$f : \mathcal{X}^n \rightarrow \{1, 2, \dots, 2^{nR}\} \quad (10)$$

and the decoder g

$$g : \{1, 2, \dots, 2^{nR}\} \rightarrow \mathcal{X}^n. \quad (11)$$

For a pair of f and g we get the distortion as

$$D = E \left[\frac{1}{n} d(X^n, g(f(X^n))) \right] \quad (12)$$

where $d(X^n, \hat{X}^n)$ defines the distortion between X^n and \hat{X}^n (the special case of MSE was introduced in (1)). A rate distortion pair (R, D) is *achievable* if there exist f and g such that

$$\lim_{n \rightarrow \infty} E \left[\frac{1}{n} d(X^n, g(f(X^n))) \right] \leq D. \quad (13)$$

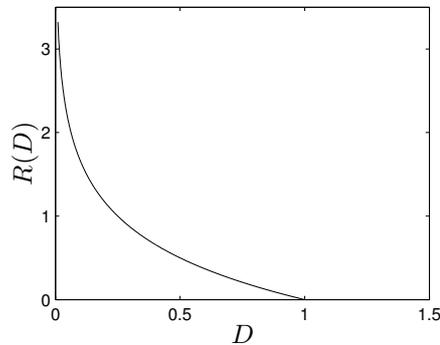


Figure 3: The rate distortion function for zero mean unit variance i.i.d. Gaussian source data.

Furthermore, the *rate distortion region* is defined by the closure of all achievable rate distortion pairs. Also, the *rate distortion function* $R(D)$ is given by the infimum of all rates R that achieve the distortion D . For a stationary and ergodic process it can be proved that [14]

$$R(D) = \lim_{n \rightarrow \infty} \frac{1}{n} \inf_{f(\hat{X}^n|X^n): E[\frac{1}{n}d(X^n, \hat{X}^n)] \leq D} I(X^n; \hat{X}^n). \quad (14)$$

This can be seen as a constrained optimization problem: find the $f(\hat{X}^n|X^n)$ that minimizes mutual information $I(X^n; \hat{X}^n)$ under the constraint that the distortion is less or equal to D . In Figure 3 the rate distortion function is shown for the well known case of zero mean unit variance i.i.d. Gaussian source data. The fundamental tradeoff between rate and distortion is clearly visible.

Permutation Coding

One special kind of lossy coding, which is used in Paper B of this thesis, is permutation coding which will be explained in this section. Permutation coding was introduced by Slepian [18] and Dunn [19] and further developed by Berger [20] which also is a good introduction to the subject. We will here focus on “Variant I” minimum mean-squared error permutation codes. There is also “Variant II” codes

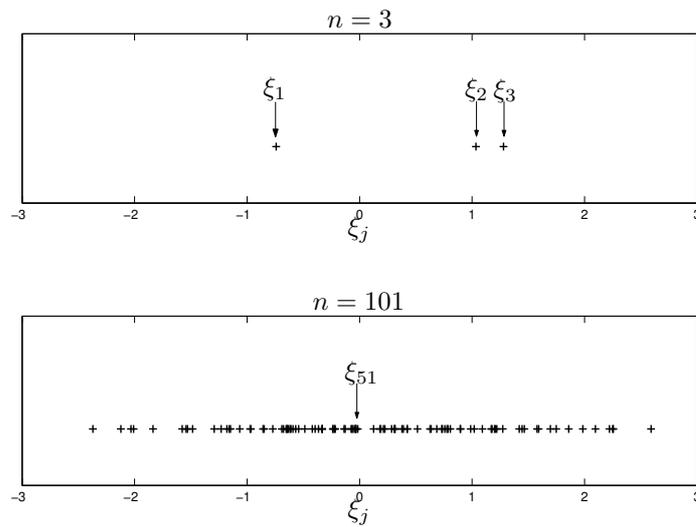


Figure 4: The magnitude of the samples in two random vectors containing zero mean Gaussian source data are shown. In the upper plot the dimension is 3 and the lower the dimension is 101.

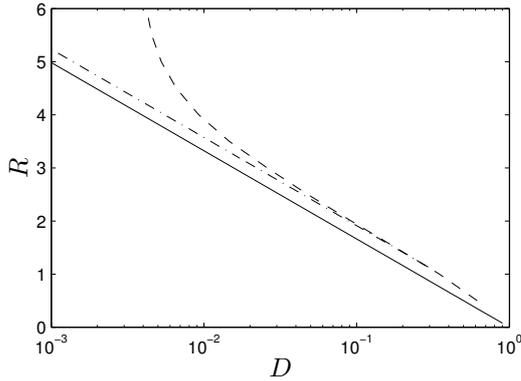


Figure 5: Performance of permutation codes (dashed line), entropy coded quantization (dash-dotted line) and rate distortion function (solid line) for unit variance i.i.d. Gaussian data. The permutation code has dimension $n = 800$.

but the theory of these is similar to the theory of “Variant I” codes and is therefore not considered.

Permutation coding is an elegant way to perform lossy coding with low complexity. Consider the case when we want to code a sequence of real valued random variables $\{X_i\}_{i=1}^{\infty}$. With permutation coding this sequence can be vector quantized in a simple fashion such that the block $X^n = (X_1, X_2, \dots, X_n)$ is quantized to an index $I \in \{1, \dots, M\}$. To explain the basic idea consider Figure 4 where an experiment where two random vectors X^n have been generated containing zero mean i.i.d. Gaussian source data. The magnitude of the different samples are plotted on the x -axis. The first vector has dimension 3 and the second has dimension 101. Furthermore, define $\xi_j, j = 1, \dots, n$, to be the j th smallest component of X^n and then consider the “mid sample,” ξ_2 in the first plot and ξ_{51} in the second. If we imagine that we would repeat the experiment by generating new vectors it is clear that ξ_{51} from this second experiment is likely to be close to ξ_{51} from the first experiment. This is also true for the the first plot when studying ξ_2 but we can expect a larger spread for this case. A similar behavior will also be obtained for all the other ξ_j 's.

Permutation coding uses this fact, namely that knowing the order of the samples in X^n can be used to estimate the value of each sample. Therefore, the order of the samples is described by the encoder. One of the main advantages of the method is its low complexity, $\mathcal{O}(n \log n)$ from sorting the samples, which makes it possible to perform vector quantization in high dimensions. In [21] it shown that permutation codes are equivalent to entropy coded scalar quantization in the sense that their rate versus distortion relation are identical when $n \rightarrow \infty$. Al-

though this result only holds when $n \rightarrow \infty$ the performance tends to be almost identical as long as intermediate rates are being used for high, but finite, n 's. For high rates this is no longer true. Typically there exists some level for the rate when increasing the rate no longer improves the performance. This saturation level depends on the size of n and increasing n moves the saturation level to a higher R , see [20]. This is illustrated in Figure 5 where the performance is shown of permutation codes (dashed line), entropy coded quantization (dash-dotted line) as well as the rate distortion function (solid line) for unit variance Gaussian data. For the permutation code $n = 800$ was used and the saturation effect starts to become visible around $R = 3.5$ bits/sample. In [22] it is shown that, somewhat contrary to intuition, there exist permutation codes with finite n 's possessing an even better performance than when $n \rightarrow \infty$ and hence also entropy coded scalar quantization. This effect does however depend on the source distribution.

When encoding and decoding in permutation coding there will exist one codeword, for instance corresponding to the first index, of the form

$$c_1^n = (\mu_1, \overset{\leftarrow{n_1}}{\dots}, \mu_1, \mu_2, \overset{\leftarrow{n_2}}{\dots}, \mu_2, \dots, \mu_K, \overset{\leftarrow{n_K}}{\dots}, \mu_K) \quad (15)$$

where μ_i satisfies $\mu_1 \leq \mu_2 \leq \dots \leq \mu_K$ and the n_i 's are positive integers satisfying $n_1 + n_2 + \dots + n_K = n$. All other codewords $c_2^n, c_3^n, \dots, c_M^n$ are constructed by creating all possible permutations of c_1^n meaning that there in total will be

$$M = \frac{n!}{\prod_{i=1}^K n_i!} \quad (16)$$

different codewords. If the components of X^n are i.i.d. all of these permutations are equally likely meaning that the entropy of the permutation index I will equal $\log_2 M$. It is a fairly straightforward task to map each of these permutations to a binary number corresponding to a Huffman code. Hence, the rate per coded symbol, X_i , is given from

$$R \gtrsim \frac{1}{n} \log_2 M \quad (17)$$

where “ \gtrsim ” means “close to from above”. Also, it turns out that the optimal encoding procedure, for a given set $\{(n_i, \mu_i)\}_{i=1}^K$, is to replace the n_1 smallest components of X^n by μ_1 , the next n_2 smallest components by μ_2 and so on. This further means that ordering the components of X^n also will decide the outcome of the vector quantization. This is an appealing property since sorting can be done with $\mathcal{O}(n \log n)$ complexity which is low enough for implementing permutation vector quantization in very high dimensions.

Now define $S_i = n_1 + n_2 + \dots + n_i$ and $S_0 = 0$ and assume the n_i 's to be fixed. The optimal choice of μ_i , for the MSE case, is then given from

$$\mu_i = \frac{1}{n_i} \sum_{j=S_{i-1}+1}^{S_i} E[\xi_j] \quad (18)$$

which can be used to design the different μ_i 's. Hence, the expected average of the n_1 smallest ξ_j 's creates μ_1 etc. When designing the n_i 's we instead assume the μ_i 's to be fixed. Defining

$$p_i = \frac{n_i}{n} \quad (19)$$

gives an approximation of the rate as

$$R \approx - \sum_{i=1}^K p_i \log_2 p_i. \quad (20)$$

Using this as a constraint when minimizing the expected distortion results in a constrained optimization problem giving the optimal choice of p_i as

$$p_i = \frac{2^{-\beta\mu_i^2}}{\sum_{j=1}^K 2^{-\beta\mu_j^2}} \quad (21)$$

where β is chosen such that (20) is valid. However, p_i will be a real value and n_i is required to be an integer meaning that we from (21) need to create an approximate optimal value of n_i . With these equations we can optimize (18) and (21) in an iterative fashion eventually converging in some solution for the parameters $\{\mu_i, n_i\}_{i=1}^K$. K is found by trying out this iteration procedure for different K 's (many of them can be ruled out) and the best K , i.e. the K producing the best distortion, is chosen. For a more detailed description of this procedure see [20].

2 Channel Coding

When performing source coding one aims to remove all redundancy in the source data, for channel coding the opposite is done; redundancy is introduced into the data in order to protect the data against channel errors. These channel errors can for instance be continuous valued, considered in Papers A–C, packet losses, considered in Papers D–F, or bit errors considered in Paper G. In Figure 6 the nonideality of the channel is modelled as discrete memoryless disturbance, $p(j|i)$, when transmitting the index i and receiving index j . Note that j is not necessarily equal to i . Channel coding tries to protect the system against these kinds of imperfections.

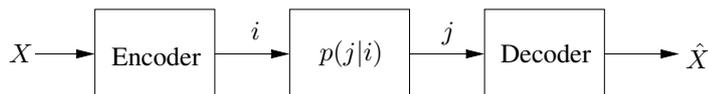


Figure 6: Model of transmission over a channel.

There exist theoretical formulas for how much information that, in theory, can be transmitted over a channel with certain statistics. This value is called capacity

and tells us the maximum number of bits per channel use that can be transmitted over the channel such that an arbitrary low error probability can be achieved. It should be noted that the capacity is an supremum which it not necessarily achievable itself, this will depend on the channel statistics. For stationary and memoryless channels the capacity is

$$C = \max_{p(x)} I(X; Y) \quad (22)$$

which is the well known formula for capacity originating from Shannon's groundbreaking paper [1]. X and Y are not required to be discrete in this formula but generally when dealing with continuous alphabets a constraint on $p(x)$ in (22) is introduced such that the power is restricted, i.e. $p(x) : E[X^2] \leq P$. Furthermore, if the channel has memory Dobrushin [23] derived the capacity for "information stable channels" (see e.g. [24] for explanation) and Verdú and Han [25] showed a general formula valid for any channel.

3 The Source–Channel Separation Theorem

It is now time to combine the results from the discrete source coding theorem, (5), and the channel capacity theorem, (22). The discrete source coding theorem states that the data X^n can be compressed to use arbitrarily close to $H_\infty(X)$ bits per coded source symbol and the channel capacity theorem states that arbitrarily close to C bits per channel use can be reliably transmitted over a given channel. Knowing these separate results the question about how to design the encoder/decoder in a system which needs to do both source and channel coding, as in Figure 6, arises. Since the discrete source coding theorem only depends on the statistical properties of the source and the channel coding theorem only depends on the statistical properties of the channel one might expect that a separate design of source and channel codes is as good as any other method. It turns out that for stationary and ergodic sources a source–channel code exist when $H_\infty(X) < C$ such that the error probability during transmission can be made arbitrary small. The converse, $H_\infty(X) > C$, implies that the error probability is bounded away from zero and it is not possible to achieve arbitrary small error probability. The case when $H_\infty(X) = C$ is left unsolved and will depend on the source statistics as well as the channel properties.

For nonstationary sources the source–channel separation coding theorem takes an other shape and we need to use concepts like "strictly dominating" and "domination." This was introduced and explained in [24, 26].

Based on these theoretical results it may appear as if source and channel codes could be designed separately. However, this is only true under the assumptions valid when deriving the results in (5) and (22). One of these assumptions is the use of infinitely long codes, i.e. $n \rightarrow \infty$. In practice this is not feasible, especially when dealing with real time applications like video streaming or VoIP. This

motivates the study of joint source-channel coding since for finite n 's it will be possible to design better source-channel codes jointly than done separately. This subject is the main focus in this thesis.

4 Analog Source-Channel Coding

The topic of analog source-channel coding deals with the problem illustrated in Figure 7 where k source samples are transmitted by using n orthogonal channels. The encoder maps a vector of source symbols $x^k \in \mathbb{R}^k$ to a vector $y^n \in \mathbb{R}^n$ which is transmitted over the channel. Hence,

$$f : \mathbb{R}^k \rightarrow \mathbb{R}^n \quad (23)$$

where a power constraint

$$E [\|Y^n\|^2] \leq nP \quad (24)$$

is invoked on the encoder. As can be seen from the figure the channel adds continuous valued noise on the transmitted values and $r^n \in \mathbb{R}^n$ is received by the decoder. The decoder estimates x^k as

$$g : \mathbb{R}^n \rightarrow \mathbb{R}^k \quad (25)$$

and the objective is to minimize the expected distortion.

When both the source and the noise is i.i.d. zero-mean Gaussian, the distortion is measured in MSE and $k = n$, it is well known that linear encoding is optimal, i.e. $f(x^k) = \sqrt{(P/\sigma_x^2)}x^k$, under the assumption that the decoder knows the source and noise variances, see e.g. [27]. However, we will focus on the case when $k \neq n$ and then linear encoding is, in general, not optimal. The challenge is to design encoders and decoders yielding the highest possible performance given some certain source and channel statistics. For the case when $k < n$ this problem is referred to as *bandwidth expansion* and for the opposite case, i.e. $k > n$, it is referred to as *bandwidth compression*.

The common solution for bandwidth expansion/compression is digital and is implemented by producing separate source and channel codes. In practice, this is

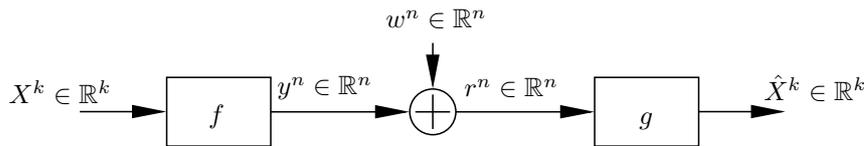


Figure 7: Bandwidth expansion ($k < n$) and compression ($k > n$).

generally done by quantizing the source followed by digital channel coding and transmission. Due to powerful source and channel coding techniques the performance of such systems can be very high when the channel quality is close to what the system has been designed for. There are, however, some disadvantages with the digital approach. In order to get a high performance long block lengths are required both for the source and channel code. This will therefore introduce delays into the system which may be undesirable, especially for a real time system. There is also a threshold effect associated with a digital system: if the channel quality goes below a certain level the channel code will break down and the system performance will deteriorate rapidly. On the other hand, if the channel quality is increased above this level the performance will not increase but rather reach a constant level which is due to the nonrepairable errors introduced by the quantizer.

In recent years analog, or at least partially analog, systems as an alternative to digital systems have received increased attention, see e.g. [28] and the references therein. Analog systems do, in general, not have the same disadvantages as digital systems. Hence, in some scenarios an analog approach may be more suitable than a digital one. On the other hand, in practice, the performance of a digital system is in general higher than for an analog system when being used for the channel quality that it has been designed for.

4.1 Analog Bandwidth Expansion

Analog bandwidth expansion was briefly discussed already in one of Shannon's early papers [29]. One of the reasons that linear encoding is suboptimal when $k < n$ is that a linear encoding function $f(x^k)$ uses only a k -dimensional subspace of the channel space. More efficient mappings would use a higher number of the available channel space dimensions. An example of this is illustrated in Figure 8 for $k = 1$ and $n = 2$. By using nonlinear encoding functions, illustrated by the solid 'S-shaped' curve $f(x)$, we are able to better fill the channel space than when using linear encoding functions, represented by the dashed curve. A longer curve essentially means a higher resolution when estimating x as long as we decode to the right fold of the curve, illustrated by sample x_1 in the figure. However, decreasing the SNR will at some point result in that different folds of the curve will lie too close to each other and the decoder will start making large decoding errors, illustrated by sample x_2 in the figure. Decreasing the SNR below this threshold will therefore significantly deteriorate the performance. We refer to these errors as 'small' and 'large' decoding errors. Increasing the SNR, on the other hand, will always improve the performance since the magnitude of the small decoding errors will decrease. This is one of the main advantages of analog systems compared to digital systems since the performance of a digital system will approach a saturation level when the SNR grows large.

The problem of designing analog source-channel codes is therefore a problem of finding nonlinear curves such that points far separated in the source space are also far separated in the channel space. Hence, we would like to 'stretch' the

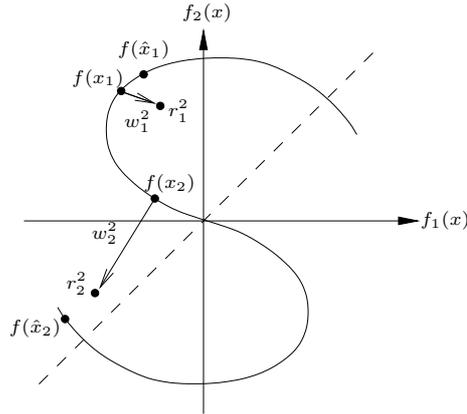


Figure 8: x_1 illustrates a 'small' decoding error and x_2 illustrates a 'large' decoding error.

curve as much as possible under the power constraint at the same time as we keep different folds of the curve separated enough for a given channel noise level.

Important publications on bandwidth expansion are [30, 31] where the performance of analog bandwidth expansion source-channel codes is analyzed for high SNR's. Furthermore, although linear expansions in general are suboptimal they are easy to analyze and optimize and this is done in [32]. Some ideas on how construct nonlinear codes are presented in e.g. [8, 33, 34] and more explicit codes are presented in [9, 12, 35–37]/Paper A.

4.2 Analog Bandwidth Compression

Analog bandwidth compression was studied in for instance [38, 39] where a few explicit codes were developed and analyzed. In particular, it was concluded that for a Gaussian source and an AWGN channel the Archimedes' spiral, illustrated in Figure 9, is appropriate for 2 : 1 compression for a large range of SNR's. In order to perform the compression the encoder maps a point (x_1, x_2) to the closest point on the spiral, i.e.

$$f(x_1, x_2) = \alpha \arg \min_x [(x_1 - \beta_1(x))^2 + (x_2 - \beta_2(x))^2] \quad (26)$$

where the spiral is described by $(\beta_1(x), \beta_2(x))$. α will control the output power and $f(x_1, x_2)$ is transmitted over the channel. Based on the received value $r = f(x_1, x_2) + w$ the decoder estimates (x_1, x_2) .

Another paper on the topic is [40] where bandwidth compression is studied for the relay channel.

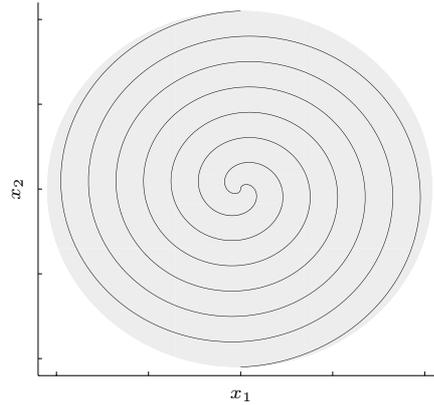


Figure 9: Archimedes' spiral used for bandwidth compression.

5 Distributed Source Coding

Distributed source coding is an important extension to the traditional point to point source coding discussed in Section 1. The main message in this topic is that in a situation with one decoder but many encoders, where each of them observes some random variable, there is a gain in performing distributed source coding if the random variables are correlated. This gain can be obtained even if the encoders do not communicate with each other. Good introductions to the topic are for instance [41, 42] and the references therein.

Correlated source data seems like a reasonable assumption in for instance wireless sensor networks where a high spatial density of sensor nodes potentially leads to correlation between different sensor measurements. Given that the sensors run on batteries it would be desirable to lower the amount of transmitted data since that could prolong the battery life time. In many applications also lowering the required bandwidth for a sensor network may be of interest. These observations, together with the increasing interest in wireless sensor networks, have fueled the research of distributed source coding in recent years. Another interesting application for distributed source coding has shown to be video coding, see e.g. [43] and the references therein.

5.1 Theoretical Results

The Slepian–Wolf Problem

One of the fundamental results for distributed source coding is the Slepian–Wolf theorem, published in [44] by Slepian and Wolf. We will briefly summarize and

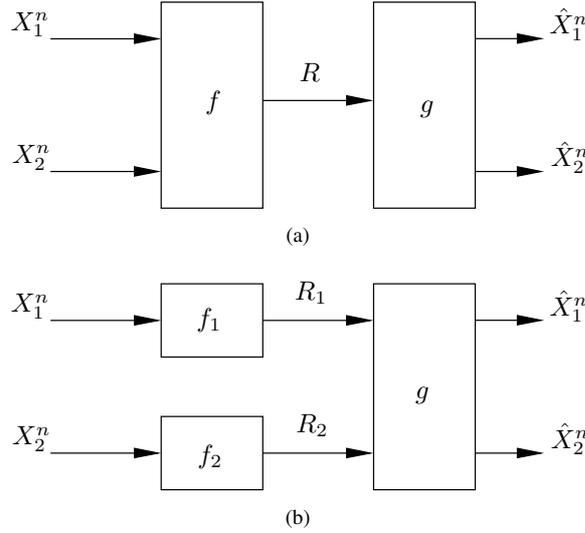


Figure 10: (a) One encoder and two sources. (b) Two encoders and two sources.

discuss this theorem.

Consider the situation in Figure 10(a): Two *discrete* i.i.d. random variables, X_1 and X_2 , are to be encoded by an encoder using rate R as

$$f : \mathcal{X}_1^n \times \mathcal{X}_2^n \rightarrow \{1, 2, \dots, 2^{nR}\} \quad (27)$$

and a decoder

$$g : \{1, 2, \dots, 2^{nR}\} \rightarrow \mathcal{X}_1^n \times \mathcal{X}_2^n \quad (28)$$

needs to reconstruct the encoded data such that $\hat{X}_1^n = X_1^n$ and $\hat{X}_2^n = X_2^n$ is ensured with arbitrary small error probability. This situation is essentially the same as the point to point source coding problem as discussed in Section 1 and we conclude that a rate R arbitrary close to $H(X_1, X_2)$ can be used, hence $R = H(X_1, X_2)$.

Now consider instead Figure 10(b). Again, two *discrete* i.i.d. random variables, X_1 and X_2 , are to be encoded but this time we use two separate encoders that do not communicate with each other, hence

$$f_1 : \mathcal{X}_1^n \rightarrow \{1, 2, \dots, 2^{nR_1}\}, \quad (29)$$

$$f_2 : \mathcal{X}_2^n \rightarrow \{1, 2, \dots, 2^{nR_2}\}. \quad (30)$$

For the decoder we have

$$g : \{1, 2, \dots, 2^{nR_1}\} \times \{1, 2, \dots, 2^{nR_2}\} \rightarrow \mathcal{X}_1^n \times \mathcal{X}_2^n \quad (31)$$

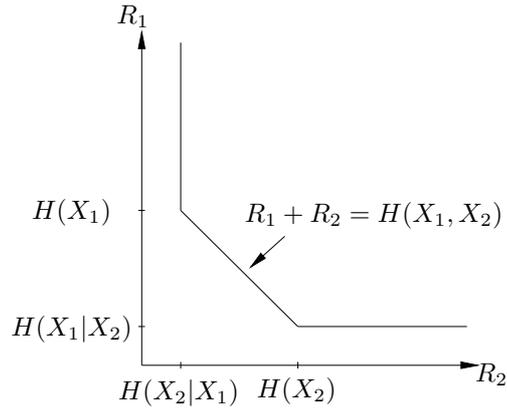


Figure 11: The Slepian–Wolf region.

and we also here need to reconstruct the encoded data such that $\hat{X}_1^n = X_1^n$ and $\hat{X}_2^n = X_2^n$ with arbitrary small error probability. According to the Slepian–Wolf theorem [44] the rate region illustrated in Figure 11 and described by

$$\begin{aligned} R_1 &\geq H(X_1|X_2) \\ R_2 &\geq H(X_2|X_1) \\ R_1 + R_2 &\geq H(X_1, X_2) \end{aligned} \quad (32)$$

is achievable. This result is somewhat nonintuitive since it means that the sum rate $R = R_1 + R_2 = H(X_1, X_2)$ is achievable also for this second situation. Hence, in terms of sum rate, there is no loss in using separate encoders compared to joint encoders. Therefore, in situations where we have separate encoders and correlated source data there is a gain in considering distributed source coding since $H(X_1, X_2) < H(X_1) + H(X_2)$.

Example of Slepian–Wolf Coding

In Figure 12 we give a simple example of Slepian–Wolf coding. Let us assume that we have a random source (X_1, X_2) with 16 possible outcomes, these outcomes are marked with circles in Figure 12 and they are all equally likely. Given that we need to encode these outcomes using the structure from Figure 10(a), hence encode X_1 and X_2 jointly, we would simply label the 16 possible outcomes with indexes $0, 1, \dots, 15$ which would require 4 bits. If we instead use the structure from Figure 10(b), hence encode X_1 and X_2 separately, one way to encode the variables would be to entropy code them using $R_1 = H(X_1)$ and $R_2 = H(X_2)$. This would however result in a higher sum rate than in the previous case. A more

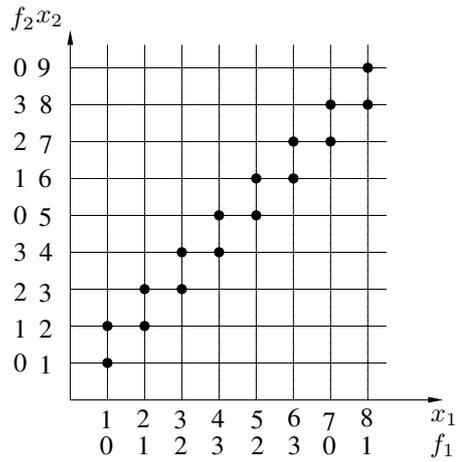


Figure 12: Model of transmission over a channel.

sophisticated way would be to use the index labelling $f_1(x_1)$, shown in the figure, when encoding X_1 and the labelling $f_2(x_2)$ when encoding X_2 . Hence, as can be seen we are labelling X_1 with indexes 0, 1, 2, 3 which will require 2 bits and the same is done for X_2 . In total there will be $4 \cdot 4 = 16$ possible outputs for $(f_1(x_1), f_2(x_2))$ and all of them will be uniquely decodable. Therefore, we will in total require $R_1 + R_2 = 4$ bits, just as in the first case when we did the encoding jointly.

The Wyner–Ziv Problem

The Slepian–Wolf theorem considers lossless source coding of discrete sources. In [45] Wyner and Ziv made a continuation on this result by considering lossy source coding with side information at the decoder as illustrated by Figure 13. It was shown that for a discrete stationary and ergodic source X^n with continuous

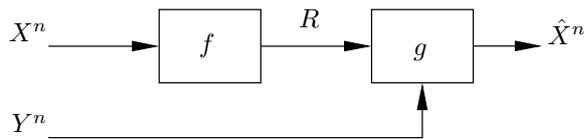


Figure 13: Source coding with side information at the receiver.

stationary and ergodic side information Y^n the lowest achievable rate satisfies

$$R(D) = \lim_{n \rightarrow \infty} \inf_{f(Z^n|X^n): E[\frac{1}{n}d(X^n, g(Y^n, Z^n))] < D} I(X^n; Z^n) - I(Y^n; Z^n) \quad (33)$$

for a given distortion D . This result was later also developed to the case of continuous sources X^n in [46]. Unlike Slepian–Wolf coding a rate loss is usually suffered when comparing Wyner–Ziv coding to the case when the side information is available to both the encoder and the decoder. One important exception to this is when X^n and Y^n are jointly Gaussian and MSE is used as a distortion measure. Here, the achievable rates are the same no matter if the side information is available to the encoder or not. Given that the covariance matrix, for this case, is

$$\begin{pmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{pmatrix}$$

the Wyner–Ziv rate distortion function is

$$R(D) = R_{X|Y}(D) = \frac{1}{2} \log^+ \left[\frac{\sigma_X^2(1 - \rho^2)}{D} \right] \quad (34)$$

where $\log^+ x = \max(\log x, 0)$.

5.2 Practical Schemes

Ideas on how to perform practical Slepian–Wolf coding are presented in [47, 48], allowing the use of powerful channel codes such as LDPC and Turbo codes in the context of distributed source coding, see e.g. [49, 50]. For the case with continuous sources, i.e. lossy coding, relevant references include [51, 52]. In general, all these methods require the use of long codes.

Alternative approaches are found in [53–58] where the distributed source coding problem is interpreted as a quantization problem. For wireless sensor networks it is also relevant to include noideal channels into the problem which is studied in for instance [59]. Practical schemes for this problem includes [6, 7, 10, 60, 61]. In [60] distributed detection over non-ideal channels is studied and in [61] quantization of correlated sources in a packet network is studied, resulting in a general problem including multiple description coding, see Section 6, as well as distributed source coding as special cases. [6, 7, 10]/Paper B designs and evaluates scalar quantizers for continuous channels.

Yet another approach for the distributed source coding problem with nonideal channels is to consider analog source–channel codes. This is studied in for instance [62, 63] where linear source–channel codes are proposed and analyzed. The linear approach is however suboptimal for the case with orthogonal channels, see e.g. [59] and compare to [64, 65] for the nonorthogonal case, and motivated by this [12]/Paper C proposes and analyzes an analog nonlinear approach.

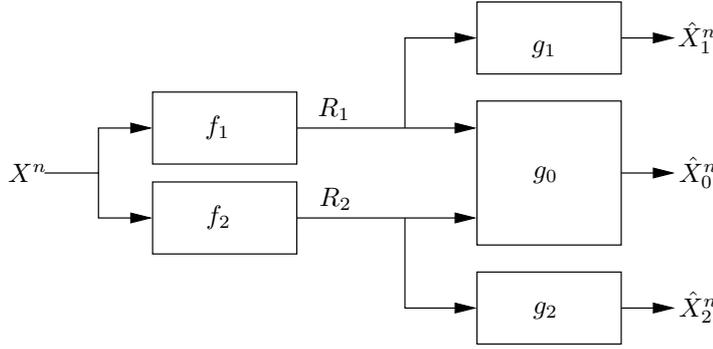


Figure 14: The MDC problem illustrated for two channels.

6 Multiple Description Coding

In multiple description coding (MDC) the total available rate for transmitting source data is split between a number of different channels. Each of these channels may be subject to failure, meaning that some of the transmitted data may be lost. The aim of MDC is then to reconstruct an approximated version of the source data even when only a subset of the used channels is in working state. The problem is illustrated in Figure 14 for two channels. Here f_1 and f_2 are the encoders used for channels 1 and 2 respectively and defined as

$$f_k : \mathcal{X}^n \rightarrow \{1, 2, \dots, 2^{nR_k}\} \quad \forall k \in \{1, 2\}. \quad (35)$$

Hence, the encoders will use R_1 and R_2 of the total available rate $R = R_1 + R_2$. There will exist three decoders: g_1 and g_2 used when only the information from one channel is received and g_0 used when the information from both channels are received, i.e. both channels are in working state. The decoders are defined as

$$g_k : \{1, 2, \dots, 2^{nR_k}\} \rightarrow \mathcal{X}^n \quad \forall k \in \{1, 2\} \quad (36)$$

$$g_0 : \{1, 2, \dots, 2^{nR_1}\} \times \{1, 2, \dots, 2^{nR_2}\} \rightarrow \mathcal{X}^n. \quad (37)$$

For the different decoders we define distortions as

$$D_k = E\left[\frac{1}{n}d(X^n, \hat{X}_k^n)\right] \quad \forall k \in \{0, 1, 2\}. \quad (38)$$

We call D_0 the *central distortion* and D_1 and D_2 *side distortions*.

As an example of MDC, consider the case when X is i.i.d. binary distributed taking values 0 and 1 with probability 1/2. Also assume the Hamming distance is used as a distortion measure, i.e. $d(1, 1) = d(0, 0) = 0$ and $d(0, 1) = d(1, 0) = 1$. Suppose that $D_0 = 0$ is required, $R_1 = R_2 = 0.5$ bits per symbol and the aim is

to minimize $D_1 = D_2 = D$. One intuitive approach would be to transmit half of the bits on one channel and the other half on the other channel. This would then give $D = 0.25$ (achieved by simply guessing the value of the lost bits). However, in [66] it is shown that one can do better and it is in fact, somewhat surprisingly, possible to achieve $D = (\sqrt{2} - 1)/2 \approx 0.207$.

The MDC literature is vast, theoretical results as well as practical schemes are presented in the sections below.

6.1 Theoretical Results

One of the first results in MDC was El Gamal and Cover's region of achievable quintuples $(R_1, R_2, D_0, D_1, D_2)$ [67]. This result states that $(R_1, R_2, D_0, D_1, D_2)$ is achievable if there exist random variables $\hat{X}_0, \hat{X}_1, \hat{X}_2$ jointly distributed with sample X from an i.i.d. source such that

$$R_1 > I(X; \hat{X}_1), \quad (39)$$

$$R_2 > I(X; \hat{X}_2), \quad (40)$$

$$R_1 + R_2 > I(X; \hat{X}_0, \hat{X}_1, \hat{X}_2) + I(\hat{X}_1; \hat{X}_2), \quad (41)$$

$$D_k \leq E[d(X, \hat{X}_k)] \quad \forall k \in \{0, 1, 2\}. \quad (42)$$

Ozarow [68] showed this bound to be tight for the case of Gaussian sources with variance σ_X^2 (although Ozarow uses $\sigma_X^2 = 1$ in his paper) and also derived closed form expressions for the achievable quintuples which satisfy

$$D_1 \geq \sigma_X^2 e^{-2R_1} \quad (43)$$

$$D_2 \geq \sigma_X^2 e^{-2R_2} \quad (44)$$

$$D_0 \geq \begin{cases} \sigma_X^2 e^{-2(R_1+R_2)} \frac{1}{1 - (\sqrt{\Pi} - \sqrt{\Delta})^2} & \text{if } \Pi \geq \Delta \\ \sigma_X^2 e^{-2(R_1+R_2)} & \text{otherwise} \end{cases} \quad (45)$$

where

$$\Pi = (1 - D_1/\sigma_X^2)(1 - D_2/\sigma_X^2) \quad (46)$$

$$\Delta = D_1 D_2 / \sigma_X^4 - e^{-2(R_1+R_2)}. \quad (47)$$

Studying these equations by setting $R_1 = R_2$ we see that there will be a tradeoff between the performance D_1, D_2 versus the performance D_0 . Decreasing D_1 and D_2 means that we need to increase D_0 and vice versa (can for instance be seen in Figure 4 of Paper E where $D_1 = D_2$).

Ahlsvede [69] showed that the El Gamal–Cover region is tight for the “no excess rate for the joint description” meaning the case when the best possible D_0 is achieved according to $R_1 + R_2 = R(D_0)$, where $R(D)$ is the rate distortion formula. In [70] Zhang and Berger constructed a counterexample which shows

that the El Gamal–Cover region is not tight in general. The problem of finding a bound that fully describes the achievable multiple description region for two descriptors is still unsolved.

In [71] Zamir shows that

$$\mathcal{D}^*(\sigma_X^2, R_1, R_2) \subseteq \mathcal{D}_X(R_1, R_2) \subseteq \mathcal{D}^*(P_X, R_1, R_2). \quad (48)$$

Here σ_X^2 is the variance of the source, $P_X = 2^{2h(X)}/2\pi e$ where $h(X)$ is the differential entropy of X . $\mathcal{D}^*(\sigma^2, R_1, R_2)$ denotes the set of achievable distortions (D_0, D_1, D_2) when using rates R_1 and R_2 on a Gaussian source with variance σ^2 . $\mathcal{D}_X(R_1, R_2)$ denotes the set of achievable distortions (D_0, D_1, D_2) for the source X .

In [72] outer and inner bounds on the achievable quintuples are achieved that relate to the El Gamal–Cover region. The multiple description problem has also been extended the K -channel case in [73] as well as in [74, 75] where the area of distributed source coding [76, 77] is used as a tool in MDC. Further results can be found in [78, 79].

6.2 Practical Schemes

Also the more practical area of MDC has received considerable attention, see e.g. [80]. Below are a few of the most well known MDC methods explained in brief.

Multiple Description Scalar Quantizers

In [81] Vaishampayan makes the first constructive attempt at designing a practical MDC scheme, motivated by the extensive information theory research summarized in the previous section. The paper considers designing scalar quantizers, for memoryless source data, as encoders (f_1, f_2) producing indices (i, j) . It is important to note that the quantization intervals of f_1 and f_2 can be disjoint intervals as shown in Figure 15. This will in turn lead to the existence of a virtual encoder f_0 created by the indices from f_1 and f_2 and an *index assignment matrix*, see examples in Figure 16. The index generated from f_1 is mapped to a row of the index assignment matrix and f_2 is mapped to a column. Hence, when both indices are received we know that the original source data must have been in the interval created by the intersection of the two quantization intervals described by f_1 and f_2 , i.e. $x \in \{x : (f_1(x) = i) \wedge (f_2(x) = j)\}$. The virtual encoder f_0 will therefore give rise to the central distortion D_0 which is illustrated in Figure 15 where the left index assignment matrix of Figure 16 is used.

Based on this idea the MDC system is created by optimizing the lagrangian function

$$L = E[d(X, \hat{X}_0)] + \lambda_1(E[d(X, \hat{X}_1)] - D_1) + \lambda_2(E[d(X, \hat{X}_2)] - D_2). \quad (49)$$

$$\begin{array}{rcccccc}
 f_2(x) : & 1 & 1 & 2 & 2 & 3 & 2 & \gg x \\
 f_1(x) : & 1 & 2 & 1 & 2 & 2 & 3 & \gg x \\
 f_0(x) : & 1 & 2 & 3 & 4 & 5 & 6 & \gg x
 \end{array}$$

Figure 15: Encoders f_1 and f_2 will together with the (left) index assignment matrix of Figure 16 create a third virtual encoder f_0 .

| | | | | | |
|-----|---|---|----|----|-----|
| 1 | 3 | | | | ... |
| 2 | 4 | 5 | | | |
| | 6 | 7 | 9 | | |
| | | 8 | 10 | 11 | |
| | | | 12 | 13 | ... |
| ... | | | | | ... |

| | | | | | |
|-----|---|----|----|----|-----|
| 1 | 3 | 5 | | | ... |
| 2 | 6 | 8 | 10 | | |
| 4 | 7 | 11 | 12 | 14 | |
| | 9 | 13 | 16 | 17 | |
| | | 15 | 18 | 21 | ... |
| ... | | | | | ... |

Figure 16: Two examples of index assignment matrices. The left matrix will enable more protection against packet losses and hence a lower performance on D_0 . The right matrix will on the other hand enable a higher performance on D_0 .

It is shown that this optimization problem results in a procedure where (i) for a fixed decoder, optimal encoders can be created, and (ii) for a fixed encoder, optimal decoders can be created. Alternating between these two optimization criteria will eventually converge to a solution just as in regular vector quantization training. By choosing low values for λ_1 and λ_2 the solution will converge to an MDC scheme with high performance on D_0 and low performance on D_1 and D_2 . Choosing high values for the λ_k 's will on the other hand yield a low performance on D_0 and a high performance on D_1 and D_2 . Hence, λ_1 and λ_2 can be used to design the system for different levels of error protection.

Furthermore, also the design of the index assignment matrix will impact the tradeoff between D_0, D_1 and D_2 . In order to optimize D_0 there should be no empty cells in the matrix leading to as many quantization regions for the virtual encoder f_0 as possible. On the other hand, if we are interested in only optimizing D_1 and D_2 there should only be nonempty cells along the diagonal of the index assignment matrix corresponding to transmitting the same description on both channels. In Figure 16 two examples of index assignment matrices are shown and since there are more nonempty cells in the right example using this matrix will make it possible to get a better performance on D_0 than if the left matrix was used.

The difficult problem on how to actually design the optimal index assignment

matrix is not solved in the paper, instead two heuristic methods to design these matrices are presented which are argued to have good performances.

This original idea of Vaishampayan has been investigated and improved in many papers since it was introduced in [81]. In [82] the method is extended to entropy-constrained MDSQ (ECMDSQ). Here, two additional terms are added in the Lagrangian function (49), similar to what is done in entropy-constrained quantization (see e.g. [83]), resulting in

$$L = E[d(X, \hat{X}_0)] + \lambda_1(E[d(X, \hat{X}_1)] - D_1) + \lambda_2(E[d(X, \hat{X}_2)] - D_2) \\ + \lambda_3(H(I) - R_1) + \lambda_4(H(J) - R_2) \quad (50)$$

where $H(I)$ and $H(J)$ are the entropy of the indices i and j generated by the encoders f_1 and f_2 . It is shown that introducing this modification still leads to an iterative way to optimize the system. Hence, also λ_3 and λ_4 can be used to control the convergence of the solution, increasing the values of these will try to force the solution to use a lower rate. A comparison between MDSQ and ECMDSQ can for instance be seen in Figure 4 of Paper B where $D_1 = D_2$.

In [84] a high-rate analysis of MDSQ and ECMDSQ is presented. Motivated by the fact that comparing different MDC schemes is hard due to the many parameters involved (central/side distortions, rates at the different channels) it is also proposed that the product $D_0 D_1$ for the balanced case, i.e. $R_1 = R_2$, is a good figure of merit when measuring performance. As a special case MDSQ/ECMDSQ are analyzed for the Gaussian case and then compared to the Ozarow bound (43-45). This resulted in an 8.69 dB/3.07 dB gap respectively compared to the theoretical bound. This result was later strengthened in [85].

Some improved results on the index assignment were obtained in [86] and this problem was also studied in [87] where an algorithm is found for designing an index assignment matrix when more than two channels are used.

Multiple Description Lattice Vector Quantization

The idea of multiple description lattice vector quantization (MDLVQ) is introduced in [88, 89]. This is in some sense an extension of the idea of MDSQ to the vector quantization case. However, when dealing with unconstrained vector quantization the complexity grows very quickly with the number of dimensions; in order to reduce this complexity the vector quantization can be constrained in some way which generally results in a decreased complexity at the cost of a sub-optimal performance. One example of this is lattice vector quantization where all codewords are from a lattice (or possibly a subset). This greatly simplifies the optimal encoding procedure and a lower encoding complexity is achieved. The high complexity of unconstrained vector quantizers implies that a pure multiple description vector quantization (MDVQ) scheme, suggested in e.g. [90-92], may be impractical for high dimensions and rates which motivates the use of lattice vector quantization in the framework of MDC.

In an n -dimensional MDLVQ two basic lattices are used: the fine lattice $\Lambda \subset \mathbb{R}^n$ and the coarser lattice $\Lambda' \subset \mathbb{R}^n$. Λ will constitute the codewords of the central decoder and Λ' will constitute the codewords of the side decoders. Furthermore, Λ' is chosen such that it is geometrically similar to Λ meaning that Λ' can be created by a rotation and a scaling of Λ . In addition, no elements of Λ should lie on the boundaries of the Voronoi regions of Λ' . An important parameter is the index

$$K = \left| \frac{\Lambda}{\Lambda'} \right| \quad (51)$$

which describes how many lattice points from Λ there exist in the Voronoi regions of Λ' (it is assumed that $K \geq 1$). The lower the value of K , the more error protection is put into the system.

Based on these lattices an index assignment mapping function ℓ , which is an injection, is created as a one-to-one mapping between a lattice point in Λ and two lattice points in $\Lambda' \times \Lambda'$. Hence,

$$\Lambda \xrightarrow{1-1} \ell(\Lambda) \subseteq \Lambda' \times \Lambda'. \quad (52)$$

The encoder will start quantizing a given vector X^n to the closest point $\lambda \in \Lambda$. By deriving $\ell(\lambda)$ the resulting point is mapped to $(\lambda_1, \lambda_2) \in \Lambda' \times \Lambda'$, i.e. two points in the coarser lattice. It should here be noted that the order of these points are of importance meaning that $\ell^{-1}(\lambda_1, \lambda_2) \neq \ell^{-1}(\lambda_2, \lambda_1)$. Descriptions of λ_1 and λ_2 are then transmitted over one channel each and if both descriptions are received the inverse mapping ℓ^{-1} is used to recover λ . If only one descriptor is received λ_1 , or λ_2 , is used as a reconstruction point. This means that the distance between λ and the λ_k 's will affect the side distortion and [89] considers the design of the index mapping for the symmetric case when producing equal side distortions from equal-rate channels (further investigated in [93] for the asymmetric case). An asymptotic analysis is also provided which reveals that the performance of MDLVQ can get arbitrarily close to the asymptotic multiple description rate distortion bound [94] when the rate and dimension approach infinity. Also [85] provides insight in the asymptotical behavior of MDLVQ.

In [95] a simple, yet powerful, modification of the encoder is introduced which makes it possible not only to optimize the encoding after the central distortion which was previously the case. This is done by instead of, as in the original idea, minimizing

$$\|X^n - \hat{X}_0^n\|^2 \quad (53)$$

minimize

$$\alpha \|X^n - \hat{X}_0^n\|^2 + \beta (\|X^n - \hat{X}_1^n\|^2 + \|X^n - \hat{X}_2^n\|^2). \quad (54)$$

Choosing a large value on β will decrease the side distortion at the cost of increasing the central distortion and vice versa.

Multiple Description Using Pairwise Correlating Transforms

Multiple description coding using pairwise correlating transforms (MDCPC) was introduced in [96–99]. The basic idea here is to create correlation between the transmitted information. This correlation can be exploited in the case of a packet loss on one of the channels since the received packet due to the correlation will contain information also about the lost packet. In order to do this a *piecewise correlating transform* \mathbf{T} is used such that

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \mathbf{T} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}, \quad (55)$$

where

$$\mathbf{T} = \begin{bmatrix} r_2 \cos \theta_2 & -r_2 \sin \theta_2 \\ -r_1 \cos \theta_1 & r_1 \sin \theta_1 \end{bmatrix}. \quad (56)$$

Here r_1 and r_2 will control the length of the basis vectors and θ_1 and θ_2 will control the direction. The transform is invertible so that

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \mathbf{T}^{-1} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}. \quad (57)$$

Based on the choice of $r_1, r_2, \theta_1, \theta_2$ a controlled amount of correlation, i.e. redundancy, will be introduced in Y_1 and Y_2 which are transmitted over the channels. The more redundancy introduced the lower side distortion will be obtained at the cost of an increased central distortion.

However, in their present form (55)-(57) use continuous values. In order to make the idea implementable quantization needs to be performed at some stage in these equations. This is solved by quantizing X_1 and X_2 and then finding an approximation of the transform \mathbf{T} which ensures that also Y_1 and Y_2 will be discrete. In [2]/Paper F of this thesis we propose to change the order of this such that the transformation is performed first and secondly Y_1 and Y_2 are quantized. This results in a performance gain.

To get some intuition about the behavior of the original method some of the theoretical results of [98, 100] are reviewed: For the case when $R_1 = R_2 = R$ and using high rate approximations it can be showed that when no redundancy is introduced between the packets, i.e \mathbf{T} equals the identity matrix, the performance will behave approximately as

$$D_0^* = \frac{\pi e}{6} \sigma_1 \sigma_2 2^{-2R} \quad (58)$$

$$D_s^* = \frac{1}{4}(\sigma_1^2 + \sigma_2^2) + \frac{\pi e}{12} \sigma_1 \sigma_2 2^{-2R} \quad (59)$$

where D_s^* is the average side distortion between the two channels. Using the transformation matrix

$$\mathbf{T} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (60)$$

we instead get

$$D_0 = \Gamma D_0^* \quad (61)$$

$$D_s = \frac{1}{\Gamma^2} \frac{1}{4} (\sigma_1^2 + \sigma_2^2) + \Gamma \frac{\pi e}{12} \sigma_1 \sigma_2 2^{-2R} \quad (62)$$

where

$$\Gamma = \frac{(\sigma_1^2 + \sigma_2^2)/2}{\sigma_1 \sigma_2}. \quad (63)$$

Hence, the central distortion is increased (assuming $\sigma_1 > \sigma_2$) and the constant term in the average side distortion is decreased at the same time as the exponential term is increased. Two conclusions can be drawn; firstly MDCPC is not of interest when the rate is very high, since D_s is bounded by a constant term. Secondly, the method also requires unequal variances for the sources X_1 and X_2 since otherwise $\Gamma = 1$. The method is however efficient in increasing robustness with a small amount of redundancy.

The method has been further developed in [100] where it is extended for using more than two channels. Some analytical results on the case of Gaussian source data are presented in [101].

Multiple Description Coding Using Frames

The idea of multiple description coding using frames has obvious similarities with ordinary block channel coding, see e.g. [102]. The idea presented in [103–106] is to multiply an n -dimensional source vector X^n with a rectangular matrix $\mathbf{F} \in \mathbb{R}^{m \times n}$ of rank n and with $m > n$;

$$Y^m = \mathbf{F}X^n. \quad (64)$$

Y^m will constitute an overcomplete representation of X^n and X^n is hence described by m descriptors which can be transmitted over one channel each in the ordinary MDC fashion. In the case that at least n descriptors are received X^n can be recovered by creating the (pseudo-)inverse matrix corresponding to the received descriptors in Y^m . In the case that $q < n$ descriptors are received the received data will describe and $(n - q)$ -dimensional subspace \mathcal{S}_q such that $X^n \in \mathcal{S}_q$. X^n can then be reconstructed as

$$\hat{x}^n = E[X^n | x^n \in \mathcal{S}_q] \quad (65)$$

if the aim is to minimize the MSE [107].

In the idea above we have not included the fact that quantization will be necessary of the descriptors Y^m . The basic idea is however still valid and the impact of the quantization is analyzed in [32].

Other Methods

An other interesting approach was introduced in [108] where entropy-coded dithered lattice quantizers are used. The method is shown to have a good performance but it is not asymptotic optimal. This work is carried on in [109, 110] which results in a scheme that can achieve the whole Ozarow region for the Gaussian case when the dimension becomes large.

In [5]/Paper D we study the possibility to use sorting as a tool to produce MDC and somewhat related to this we study the use of permutation codes in an MDC scheme in [4]/Paper E.

7 Source Coding for Noisy Channels

In Section 1 we explained the basics of source coding. Here a random variable X^n was encoded into an index i which in turn is decoded to an estimate \hat{X}_i^n of X^n . However, when transmitting i over a nonideal channel imperfections in the channel may lead to that we receive a value j which not necessarily equals i . This effect needs to be taken into consideration when designing a source code that will be transmitted over a nonideal channel. We introduce the basic concepts by an example; consider a 2-dimensional source distributed as in Figure 17(a) where

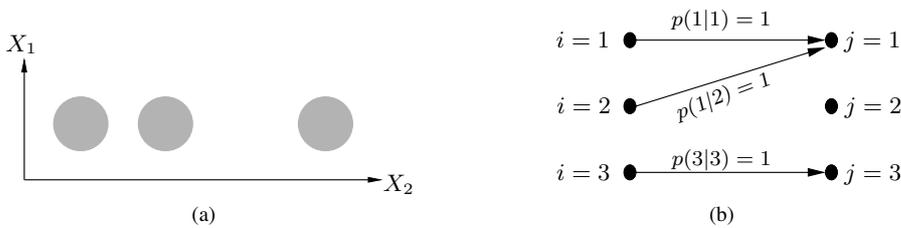


Figure 17: (a) Illustration of a 2-dimensional distribution and (b) the channel $p(j|i)$.

the source data is uniformly distributed over 3 regions. Further assume that we need to quantize this source to an index $i \in \{1, 2, 3\}$ which will be transmitted over a channel $p(j|i)$ described by Figure 17(b). In traditional source coding we would simply aim for minimizing the distortion without invoking channel knowledge in the design. This would, in the MSE case, produce quantization regions and reconstruction points as illustrated in Figure 18(a). Hence, a large contribution to the distortion occurs when transmitting the index $i = 2$ since the receiver will interpret this as if $i = 1$ was actually transmitted and reconstruct accordingly. However, invoking information about the channel statistics in the design would

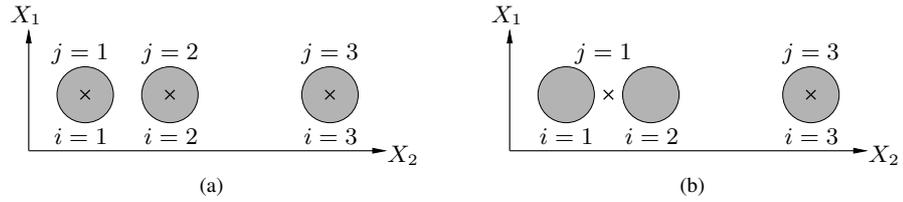


Figure 18: Using the MSE distortion measure: (a) Encoder/decoder optimized for an ideal channel and (b) encoder/decoder optimized for the non-ideal channel of Figure 17(b).

make it possible to optimize the system better. This will enable the system to better protect itself against channel failures resulting in the design illustrated in Figure 18(b) where a better overall MSE will be obtained than in the previous case. Note that in the encoding we need to consider both how to design the quantization regions as well as the index assignment, i.e. how to label the quantization regions with indices (changing the labelling affects the performance). The basic problems of source coding with noisy channels are clear from this example: quantization, index assignment and decoding. These problems are discussed in the following sections.

7.1 Scalar Source Coding for Noisy Channels

The first constructive work on scalar source coding for noisy channels was carried out by Fine in [111]. Here a communication system with a discrete noisy channel is considered and rules for creating encoders as well as decoders are presented. This work continues in [112] where optimal quantizers and decoders are developed for the case of binary channels, both for the unrestricted scalar quantizer as well as for uniform quantizers. In [113] Farvardin and Vaishampayan extended this result to other channels and they also introduced a procedure to improve the index assignment. These results are central in this topic and are summarized below.

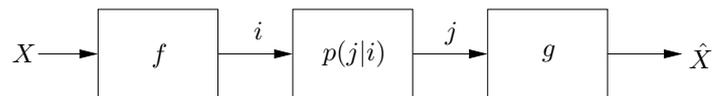


Figure 19: Model of a communication system.

The considered system is illustrated in Figure 19 where X is assumed to be an i.i.d. source and $i \in \{1, \dots, M\}$. It is also assumed that MSE is used as a

distortion measure. For a fixed decoder g the encoder is created as

$$f(x) = \arg \min_i \left(E[(x - \hat{X})^2 | I = i] \right). \quad (66)$$

This will result in optimal scalar quantizers for a given index assignment and a given decoder g since it will consider the fact that \hat{X} is a random variable. It is further shown that (66) can be developed such that the borders of the optimal scalar quantizer can be found in an analytical fashion. Altogether, these two steps will improve the performance of the encoder and the algorithm moves on to optimize the decoder under the assumption that the encoder f is fixed. The optimal encoder is given by

$$g(j) = E[X | J = j]. \quad (67)$$

These equations makes it possible to optimize the encoder and decoder in an iterative fashion and will result in a locally optimal solution for f and g , which not necessarily equals the global optimal solution.

7.2 Vector Source Coding for Noisy Channels

Early works that can be categorized as vector source coding for noisy channels include [114, 115] but the first more explicit work can be found in [116]. Here optimality conditions for encoders and decoders are formulated which results in *channel optimized vector quantization* (COVQ). Also [117, 118] studies this subject where [118] is good introduction to COVQ. The central results of these publications are summarized for the MSE case. The system considered is basically the same as in Figure 19 with X replaced by X^n and \hat{X} replaced by \hat{X}^n . For a fixed decoder g the encoder should perform its vector quantization as

$$f(x^n) = \arg \min_i \left(E[(x^n - \hat{X}^n)^2 | I = i] \right) \quad (68)$$

and for a fixed encoder f the decoder should be designed as

$$g(j) = E[X^n | J = j]. \quad (69)$$

Also here the design is done by alternating between optimizing the encoder, (68), and the decoder, (69). Although the algorithm usually tends to converge to a good solution the initialization of the index assignment usually affects the performance of the resulting design.

Further work is done in [119] where it is shown that once a COVQ system has been designed the complexity is no greater than ordinary vector quantization. This indeed motivates the use of COVQ. However, just as in regular vector quantization complexity will be an issue when the dimension and/or the rate is high. One solution to this problem is to somehow restrict the structure of the quantization regions by performing a multistage VQ, see e.g. [120, 121] for ideal channels and [122] for noisy channels. These quantizers will in general have a lower performance but also a lower complexity.

Soft Decoding

Most of the work done in joint source–channel coding uses a discrete channel model, $p(j^k|i^k)$, arising from an analog channel in conjunction with a hard decision scheme. This is illustrated in Figure 20 where i^k is transmitted and distorted by the analog channel such that r^k is received as

$$r^k = i^k + w^k \quad (70)$$

where w^k is additive (real–valued) noise. This value is then converted to a discrete value j^k which results in a discrete channel model. However, if it is assumed

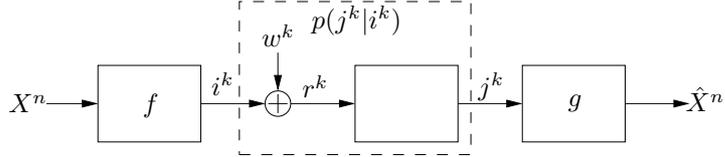


Figure 20: Model of a channel arising from an analog channel in conjunction with a hard decision scheme

that the receiver can access and process the analog (or soft) values r^k the decoder g could be based on r^k , instead on j^k . Since $i^k \rightarrow r^k \rightarrow j^k$ will constitute a Markov chain we can conclude that r^k will contain more, or possibly an equal amount of, information about i^k than j^k which means that decoding based on r^k should make it possible to get a better performance than decoding based on j^k . For instance in the MSE case the optimal decoder is given as

$$\hat{X}^n = E[X^n | R^k = r^k]. \quad (71)$$

Decoding based on soft channel values is often referred to as soft decoding and the idea originates from [123]. Further work in the area includes [124] where Hadamard–based optimal and suboptimal soft decoders are developed. In [125] a general treatment for soft decoding of noisy channels with finite memory is provided. One of the main results is that the complex algorithms resulting from this theory can be written in a recursive manner lowering the complexity. Also in [126] channels with memory is studied.

Using soft decoding results in a high decoding complexity. In [127, 128] this complexity is decreased by quantizing the soft channel values. The quantization produces a vector j^{qk} and is hence a compromise between using the complete soft information r^k and the commonly occurring coarse version, j^k , of the information. The design criteria used is to design uniform scalar quantizers for the r_m 's such that the mutual information between r_m and j_m^q is maximized. We further investigate how to create j^{qk} by presenting a different approach in [3]/Paper G of

this thesis where theory for designing nonuniform quantizers are developed. These quantizers are optimal in the MSE sense.

7.3 Index Assignment

When dealing with noisy channels the index assignment of the different codewords becomes an important issue to consider. Otherwise, codewords that interchange frequently may be far apart in the signal space which may cause a large contribution to the distortion. However, the index assignment problem is NP-hard and to find the optimal solution one would have to run a full search over the $M!$ possible orderings of the M codevectors (although some of these permutations can be ruled out). Conducting a full search is not feasible in most practical situations which have made the research focus on suboptimal algorithms, with limited complexity, to solve the problem.

Early publications about the index assignment problem are [129, 130] where heuristic algorithms are described. In [118, 131] simulated annealing is used to generate good index assignments and in [132] a greedy method called “Pseudo-Gray Coding” is developed which is shown to have a good performance. In [133, 134] the Hadamard transform is used as a tool to create, as well as to indicate the performance of, index assignments for binary symmetric channels. The method is demonstrated to have a high performance and a fairly low complexity. This work is extended to cover more general channel models in [135].

8 Contributions of the Thesis

In this thesis we focus on four topics:

- *Analog Bandwidth expansion.* In Paper A we introduce and analyze a new analog source–channel code based on orthogonal polynomials.
- *Distributed source coding over noisy channels.* In Papers B and C we present new joint source–channel schemes for distributed source coding over nonideal orthogonal channels.
- *Multiple description coding.* In papers D and E we introduce new MDC schemes. These schemes are analyzed as well as simulated. In paper F we improve the quantization in multiple description coding using pairwise correlation transforms, briefly explained in Section 6.2.
- *Soft decoding.* The main contribution in paper G is the study of how to quantize soft information for later use in a soft decision scheme.

The connections between these different topics and papers are illustrated in Figure 21. MDC can be seen as a packet based expansion which makes this problem somewhat similar to the analog bandwidth expansion problem. The same goes for

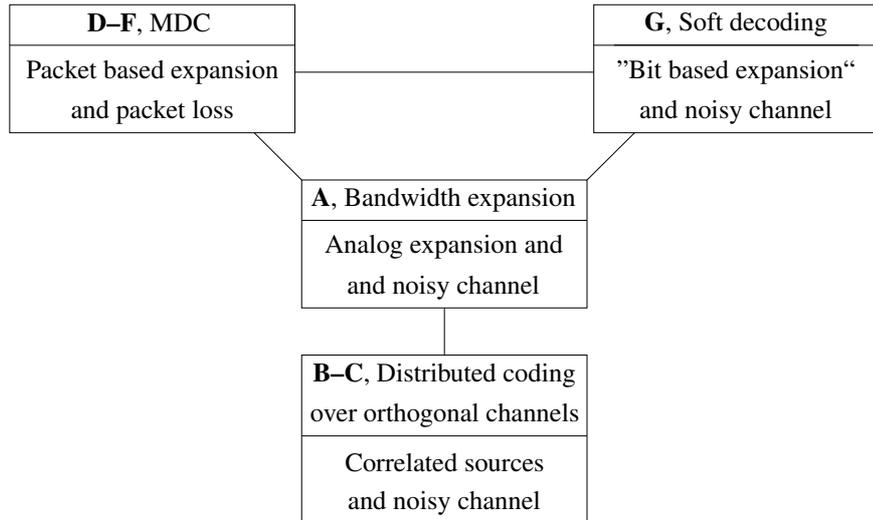


Figure 21: The connection between the different papers/topics in the thesis.

soft decoding, but here we transmit bits instead of packets and we can therefore think of this situation as a kind of "bit based expansion". Finally, the bandwidth expansion problem is a special case of distributed source coding over orthogonal channels which will be discussed more in Paper C. The main contributions made in each paper are summarized below.

Paper A: Polynomial Based Analog Source–Channel Codes [12]

We study the problem of bandwidth expansion, i.e. when one source sample, X , is transmitted over N orthogonal channels yielding a $1 : N$ expansion ratio. An analog source–channel code based on orthogonal polynomials is proposed and analyzed. Previous analog source–channel bandwidth expansion schemes have either focused on a uniform source distribution or otherwise a general source distribution but only for small N 's. Our main contribution in this paper is that we produce a code for a large number of source distributions implementable also for large N 's. The code can be generated using a Gram-Schmidt procedure, to fit virtually any source distribution. By simulations we show that the performance is comparable to other existing schemes. However, the proposed scheme is more general and can be implemented a larger class of source distributions.

Paper B: Distributed Quantization over Noisy Channels [10]

The problem of designing distributed quantizers for noisy channels is considered in a joint source–channel perspective. An algorithm for designing channel optimized distributed scalar quantizers for orthogonal channels is proposed and evaluated. The algorithm will produce a system operating on a sample by sample basis in a similar fashion as a channel optimized scalar quantizer (COSQ). In particular the cases of the binary symmetric channel as well as the additive white Gaussian noise channel are studied. It is demonstrated that the algorithm produces a well working system which is also robust against channel SNR mismatch. The work also results in some interesting visualizations which clarifies the connection between bandwidth expansion and distributed source coding over noisy channels. This connection is elaborated further in Paper C.

Paper C: Nonlinear Coding and Estimation for Correlated Data in Wireless Sensor Networks [11]

Consider the problem when a number of sensors access noisy observations of a random source. These observations need to be communicated over orthogonal noisy channels to a final destination where the source value is estimated. We demonstrate similarities between this problem and the problem of bandwidth expansion which motivates the use of analog source–channel codes similar to what is used for bandwidth expansion. This results in nonlinear analog source–channel codes for distributed estimation over noisy channels which are analyzed in a general setting. The conducted analysis reveals that there is a fundamental tradeoff when designing analog source–channel codes for this problem: either one aims for combating the sensor observation noise which will make the system more sensitive to the channel noise. On the other hand, if one instead would like to combat the channel noise the system will become more sensitive to the observation noise.

Based on this understanding an explicit source–channel code is proposed, analyzed and simulated. One appealing property of the proposed scheme is that it is implementable for many sources, contrary to most existing nonlinear distributed source–channel coding systems.

Paper D: Sorting–based Multiple Description Quantization [5]

A new method for performing multiple description coding is introduced. The scheme is based on sorting a frame of samples and transmitting, as side-information/redundancy, an index that describes the resulting permutation. In the case that some of the transmitted descriptors are lost this side information (if received) can be used to estimate the lost descriptors based on the received ones. This can be done since the side information describes the order of the descriptors within the frame and hence each received descriptor will narrow down the possible values of the lost ones. The side information is shown to be useful also in the case

when no descriptors are lost. For the case of a uniform i.i.d. source a closed form expression for the performance is derived making it possible to analytically optimize the choice of system parameters. Simulations conducted show the scheme to have a similar performance to multiple description scalar quantization. The main advantage of the suggested method is that it has virtually zero design complexity, making it easy to implement and adapt to varying loss probabilities. It also has the advantage of allowing straightforward implementation of high dimensional MDC.

Paper E: Multiple Description Coding using Rotated Permutation Codes [4]

The problem of designing multiple description source codes for J channels is addressed. Our proposed technique consist mainly of two steps; first J copies of the vector X^n , containing the source data $\{x_i\}_{i=1}^{i=n}$, are *rotated* by multiplication of J random unitary matrices, i.e. one rotation matrix is generated and used for each channel. Secondly each of the resulting vectors are, independently of the other vectors, vector quantized using permutation source coding. Furthermore, the decoding of the received information when only one channel is received is done by applying the inverse rotation to the result of standard permutation decoding; the optimum strategy for combining the information from multiple channels is a more difficult problem. Instead of implementing optimal combining we propose to simply average the decoded output of the individual channels and then adjust the length of the resulting vector based on a theoretical analysis valid for permutation codes. The choice of using permutation codes comes from the fact that their low complexity makes high dimensional vector quantization possible, i.e. large n 's, and our simulations have indicated that the random generation of rotation matrices works well when the dimension is high. For low dimensions different outcomes of the generated rotation matrices seem to yield quite different performance, meaning that the random design may not be as appropriate for this case. Hence, any vector quantization scheme able to perform quantization in high dimensions could potentially replace the permutation coding in the proposed scheme.

Using i.i.d. zero-mean Gaussian data for the x_i 's we evaluated the proposed scheme by comparing it to multiple description scalar quantization (MDSQ) as well as entropy-constrained MDSQ (ECMDSQ). It was shown that when using $J = 2$ and $R = 4$ bits/symbol the proposed system outperformed MDSQ. Compared to ECMDSQ a performance gain was achieved when optimizing the systems for receiving, close to, only one descriptor. Otherwise ECMDSQ had the better performance. The main advantages of the proposed method are its relatively low complexity and its ability to easily implement any number of descriptions.

Paper F: Improved Quantization in Multiple Description Coding by Correlating Transforms [2]

Multiple description using pairwise correlation transforms (MDCPC) is studied. Here a pairwise correlating transforms introduces correlation between different

bitstreams. In the case of a lost bitstream, this correlation can be used in order to get an estimate of a lost stream. In this paper we suggest a new approach for performing the quantization in MDCPC. Using the original method the data is quantized and then transformed by a matrix operator in order to increase the redundancy between descriptors. We suggest to reverse the order of these operations: first the data is transformed and then quantized. We show that this leads to a modification of the distortion measure. Using the generalized Lloyd algorithm when designing the quantization codebook also leads to a new way to update the codevectors. The modification makes it possible to improve the shape of the quantization cells and to tailor these to the employed transform. Our simulations indicates that the modified method performs better than the original one when smaller amounts of redundancy are introduced into the transmitted data. For the simulations conducted the modified method gave 2 dB signal-to-distortion gain compared to the original system when no descriptors were lost. The gain decreased to about 0.5-1 dB when the probability of lost descriptors was increased.

Paper G: On Source Decoding based on Finite-Bandwidth Soft Information [3]

Designing a communication system using joint source-channel coding in general makes it possible to achieve a better performance than when the source and channel codes are designed separately, especially under strict delay-constraints. The majority of work done in joint source-channel coding uses a discrete channel model, corresponding to an analog channel in conjunction with a hard decision scheme. The performance of such a system can however be improved by using soft decoding at the cost of a higher decoding complexity. An alternative is to quantize the soft information and store the pre-calculated soft decision values in a lookup table. In this paper we propose new methods for quantizing soft channel information, to be used in conjunction with soft-decision source decoding.

The issue on how to best construct finite-bandwidth representations of soft information is further studied and compared for three main approaches: 1) re-quantization of the soft decoding estimate; 2) vector quantization of the soft channel values, and; 3) scalar quantization of soft channel values. We showed analytically that 1) and 2) are essentially equivalent. Also, since 3) is a special case of 2) it can only yield similar or worse performance. However, we derived expressions that specify the optimal scalar quantizers, and when using designs based on these a performance close to that of approaches 1) and 2) was achieved. The gain of this suboptimality is a substantially lower complexity which makes the method interesting.

References

- [1] C. E. Shannon, "A mathematical theory of communication," *The Bell Systems Technical Journal*, vol. 27, no. 3 and 4, pp. 379–423 and 623–656, July and Oct. 1948.
- [2] N. Wernersson, T. Skölleremo, and M. Skoglund, "Improved quantization in multiple description coding by correlating transforms," in *IEEE MMSP 2004*, September 2004.
- [3] N. Wernersson and M. Skoglund, "On source decoding based on finite-bandwidth soft information," in *Proceedings IEEE Int. Symp. Information Theory*, September 2005, pp. 87–91.
- [4] —, "Multiple description coding using rotated permutation codes," in *Data Compression Conference*, March 2006.
- [5] —, "Sorting-based multiple description quantization," *IEEE Transactions on Communications*, vol. 54, no. 9, pp. 1521–1526, September 2006.
- [6] J. Karlsson, N. Wernersson, and M. Skoglund, "Distributed scalar quantizers for noisy channels," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2007, pp. 633–636.
- [7] N. Wernersson, J. Karlsson, and M. Skoglund, "Distributed scalar quantizers for gaussian channels," in *Proceedings IEEE Int. Symp. Information Theory*, June 2007, pp. 1741–1745.
- [8] P. A. Floor, T. A. Ramstad, and N. Wernersson, "Power constrained channel optimized vector quantizers used for bandwidth expansion," in *IEEE International Symposium on Wireless Communication Systems*, October 2007.
- [9] N. Wernersson, M. Skoglund, and T. Ramstad, "Analog source-channel codes based on orthogonal polynomials," in *Asilomar Conference on Signals, Systems and Computers*, November 2007.
- [10] N. Wernersson, J. Karlsson, and M. Skoglund, "Distributed quantization over noisy channels," *IEEE Transactions on Communications*, 2008, to appear.
- [11] N. Wernersson and M. Skoglund, "Nonlinear coding and estimation for correlated data in wireless sensor networks," *IEEE Transactions on Communications*, 2008, submitted.
- [12] N. Wernersson, M. Skoglund, and T. Ramstad, "Polynomial based analog source-channel codes," *IEEE Transactions on Communications*, 2008, submitted.
- [13] A. Eskicioglu and P. Fisher, "Image quality measures and their performance," *IEEE Transactions on Information Theory*, vol. 43, no. 12, pp. 2959–2965, dec 1995.
- [14] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley-Interscience, 1991.
- [15] T. S. Han and S. Verdú, "Approximation theory of output statistics," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 752–772, May 1993.
- [16] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Dordrecht, The Netherlands: Kluwer academic publishers, 1992.
- [17] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2325–2383, October 1998.

- [18] D. Slepian, "Permutation modulation," *Proc. IEEE*, vol. 53, pp. 228–236, March 1965.
- [19] J. Dunn, *Coding for continuous sources and channels, Ph.D. dissertation*. Columbia Univ., New York, 1965.
- [20] T. Berger, F. Jelinek, and J. Wolf, "Permutation codes for sources," *IEEE Transactions on Information Theory*, vol. 18, no. 1, pp. 160–169, January 1972.
- [21] T. Berger, "Optimum quantizers and permutation codes," *IEEE Transactions on Information Theory*, vol. 18, no. 6, pp. 759–765, November 1972.
- [22] V. Goyal, S. Savari, and W. Wang, "On optimal permutation codes," *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2961–2971, November 2001.
- [23] R. L. Dobrushin, "General formulation of Shannon's main theorem of information theory," *Usp. Math. Nauk.*, 1959, translated in *Am. Math. Soc. Trans.*, 33:323-438.
- [24] S. Vembu, S. Verdú, and Y. Steinberg, "The source-channel separation theorem revisited," *IEEE Transactions on Information Theory*, vol. 41, no. 1, pp. 44–54, January 1995.
- [25] S. Verdú and T. S. Han, "A general formula for channel capacity," *IEEE Transactions on Information Theory*, vol. 40, no. 4, pp. 1147–1157, July 1994.
- [26] S. Vembu, S. Verdú, and Y. Steinberg, "When does the source-channel separation theorem hold?" in *Proceedings IEEE Int. Symp. Information Theory*, June 1994, p. 198.
- [27] M. Gastpar, B. Rimoldi, and M. Vetterli, "To code, or not to code: lossy source-channel communication revisited," *IEEE Transactions on Information Theory*, vol. 49, no. 5, pp. 1147–1158, May 2003.
- [28] M. Skoglund, N. Phamdo, and F. Alajaji, "Hybrid digital-analog source-channel coding for bandwidth compression/expansion," *IEEE Transactions on Information Theory*, vol. 52, no. 8, pp. 3757–3763, August 2006.
- [29] C. E. Shannon, "Communication in the presence of noise," *Proc. IRE*, pp. 10–21, January 1949.
- [30] J. Ziv, "The behavior of analog communications systems," *IEEE Transactions on Information Theory*, vol. 16, no. 5, pp. 587–594, September 1970.
- [31] D. J. Sakrison, *Transmission of Waveforms and Digital Information*. Wiley, 1968.
- [32] V. Goyal, M. Vetterli, and N. Thao, "Quantized overcomplete expansions in \mathbb{R}^n : analysis, synthesis, and algorithms," *IEEE Transactions on Information Theory*, vol. 44, pp. 16–31, January 1998.
- [33] U. Timor, "Design of signals for analog communication," *IEEE Transactions on Information Theory*, vol. 16, no. 5, pp. 581–587, September 1970.
- [34] D. D. McRay, "Performance evaluation of a new modulation technique," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 431–445, August 1971.
- [35] V. Vaishampayan and S. I. R. Costa, "Curves on a sphere, shift-map dynamics, and error control for continuous alphabet sources," *IEEE Transactions on Information Theory*, vol. 49, no. 7, pp. 1658–1672, July 2003.

- [36] B. Chen and G. W. Wornell, "Analog error-correcting codes based on chaotic dynamical systems," *IEEE Transactions on Communications*, vol. 46, no. 7, pp. 881–890, July 1998.
- [37] M. Taheradeh and A. K. Khandani, "Robust joint source-channel coding for delay-limited applications," in *Proceedings IEEE Int. Symp. Information Theory*, June 2007, pp. 726–730.
- [38] A. Fuldseth and T. A. Ramstad, "Bandwidth compression for continuous amplitude channels based on vector approximation to a continuous subset of the source signal space," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Munich, Germany, April 1997, pp. 3093–3096.
- [39] S. Y. Chung, "On the construction of some capacity-approaching coding schemes," Ph.D. dissertation, MIT, 2000.
- [40] J. Karlsson and M. Skoglund, "Joint source-channel mappings for the relay channel," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, April 2008.
- [41] S. S. Pradhan, J. Kusuma, and K. Ramchandran, "Distributed compression in a dense microsensor network," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 51–60, March 2002.
- [42] Z. Xiong, A. D. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Processing Magazine*, vol. 21, no. 5, pp. 80–94, September 2004.
- [43] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71–83, January 2005.
- [44] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. IT-19, no. 4, pp. 471–480, July 1973.
- [45] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Transactions on Information Theory*, vol. IT-22, no. 1, pp. 1–10, January 1976.
- [46] A. D. Wyner, "The rate-distortion function for source coding with side information at the decoder-ii: General sources," *Information and Control*, vol. 38, pp. 60–80, 1978.
- [47] A. Wyner, "Recent results in the shannon theory," *IEEE Transactions on Information Theory*, vol. 20, no. 1, pp. 2–10, January 1974.
- [48] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Transactions on Information Theory*, vol. 49, no. 3, pp. 626–643, March 2003.
- [49] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Communications Letters*, vol. 6, no. 10, pp. 440–442, October 2002.
- [50] J. Garcia-Frias and Y. Zhao, "Compression of correlated binary sources using turbo codes," *IEEE Communications Letters*, vol. 5, no. 10, pp. 417–419, October 2001.
- [51] Z. Xiong, A. Liveris, S. Cheng, and Z. Liu, "Nested quantization and Slepian-Wolf coding: a Wyner-Ziv coding paradigm for i.i.d. sources," in *IEEE Workshop on Statistical Signal Processing*, September 2003, pp. 399–402.

- [52] S. Pradhan and K. Ramchandran, "Generalized coset codes for distributed binning," *IEEE Transactions on Information Theory*, vol. 51, no. 10, pp. 3457–3474, October 2005.
- [53] T. J. Flynn and R. M. Gray, "Encoding of correlated observations," *IEEE Transactions on Information Theory*, vol. 33, no. 6, pp. 773–787, November 1987.
- [54] W.M. Lam and A. R. Reibman, "Design of quantizers for decentralized estimation systems," *IEEE Transactions on Communications*, vol. 41, no. 11, pp. 1602–1605, November 1993.
- [55] D. Rebollo-Monedero, R. Zhang, and B. Girod, "Design of optimal quantizers for distributed source coding," in *Proc. IEEE Data Compression Conf.*, March 2003, pp. 13–22.
- [56] E. Tuncel, "Predictive coding of correlated sources," in *IEEE Information Theory Workshop*, October 2004, pp. 111–116.
- [57] M. Fleming, Q. Zhao, and M. Effros, "Network vector quantization," *IEEE Transactions on Information Theory*, vol. 50, no. 8, pp. 1584–1604, August 2004.
- [58] A. Saxena and K. Rose, "Distributed predictive coding for spatio-temporally correlated sources," in *Proceedings IEEE Int. Symp. Information Theory*, June 2007, pp. 1506–1510.
- [59] J. J. Xiao and Z. Q. Luo, "Multiterminal source-channel communication over an orthogonal multiple-access channel," *IEEE Transactions on Information Theory*, vol. 53, no. 9, pp. 3255–3264, September 2007.
- [60] B. Liu and B. Chen, "Channel-optimized quantizers for decentralized detection in sensor networks," *IEEE Transactions on Information Theory*, vol. 52, no. 7, pp. 3349–3358, July 2006.
- [61] A. Saxena, J. Nayak, and K. Rose, "On efficient quantizer design for robust distributed source coding," in *Proceedings IEEE Data Compression Conference*, March 2006, pp. 63–71.
- [62] S. Cui, J. Xiao, A. Goldsmith, Z. Luo, and V. Poor, "Estimation diversity and energy efficiency in distributed sensing," *IEEE Transactions on Signal Processing*, vol. 55, no. 9, pp. 4683–4695, September 2007.
- [63] Z. Luo, G. Giannakis, and S. Zhang, "Optimal linear decentralized estimation in a bandwidth constrained sensor network," in *Proceedings IEEE Int. Symp. Information Theory*, September 2005, pp. 1441–1445.
- [64] M. Gastpar and M. Vetterli, "Source-channel communication in sensor networks," in *Proc. 2nd Int. Workshop Information Processing in Sensor Networks (IPSN03) (Lecture Notes in Computer Science)*, L. J. Guibas and F. Zhao, Eds. Berlin, Germany: Springer-Verlag, 2003, pp. 162–177.
- [65] M. Gastpar, "Uncoded transmission is exactly optimal for a simple gaussian "sensor" network," in *Proc. 2nd Annu. Workshop on Information Theory and its Applications*, January 2007, pp. 177–182.
- [66] T. Berger and Z. Zhang, "Minimum breakdown degradation in binary source encoding," *IEEE Transactions on Information Theory*, vol. 29, no. 6, pp. 807–814, November 1983.

- [67] A. E. Gamal and T. Cover, "Achievable rates for multiple descriptions," *IEEE Transactions on Information Theory*, vol. 28, no. 6, pp. 851–857, November 1982.
- [68] L. Ozarow, "On a source coding problem with two channels and three receivers," *Bell Syst. Tech. J.*, vol. 59, no. 10, pp. 1909–1921, December 1980.
- [69] R. Ahlswede, "The rate-distortion region for multiple descriptions without excess rate," *IEEE Transactions on Information Theory*, vol. 31, no. 6, pp. 721–726, November 1985.
- [70] Z. Zhang and T. Berger, "New results in binary multiple descriptions," *IEEE Transactions on Information Theory*, vol. 33, no. 4, pp. 502–521, July 1987.
- [71] R. Zamir, "Gaussian codes and Shannon bounds for multiple descriptions," *IEEE Transactions on Information Theory*, vol. 45, no. 7, pp. 2629–2636, November 1999.
- [72] L. A. Lastras-Montano and V. Castelli, "Near sufficiency of random coding for two descriptions," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 681–695, February 2006.
- [73] R. Venkataramani, G. Kramer, and V. Goyal, "Multiple description coding with many channels," *IEEE Transactions on Information Theory*, vol. 49, no. 9, pp. 2106–2114, September 2003.
- [74] S. Pradhan, R. Puri, and K. Ramchandran, " n -channel symmetric multiple descriptions-part i: (n, k) source-channel erasure codes," *IEEE Transactions on Information Theory*, vol. 50, no. 1, pp. 47–61, January 2004.
- [75] R. Puri, S. Pradhan, and K. Ramchandran, " n -channel symmetric multiple descriptions-part ii: an achievable rate-distortion region," *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1377–1392, April 2005.
- [76] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 471–480, July 1973.
- [77] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Transactions on Information Theory*, vol. 22, no. 1, pp. 1–10, January 1976.
- [78] R. Ahlswede, "On multiple descriptions and team guessing," *IEEE Transactions on Information Theory*, vol. 32, no. 4, pp. 543–549, July 1986.
- [79] F. Fang-Wei and R. W. Yeung, "On the rate-distortion region for multiple descriptions," *IEEE Transactions on Information Theory*, vol. 48, no. 7, pp. 2012–2021, July 2002.
- [80] V. K. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Processing Magazine*, vol. 18, pp. 74–93, sep 2001.
- [81] V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 821–834, 1993.
- [82] V. A. Vaishampayan and J. Domaszewicz, "Design of entropy-constrained multiple-description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 40, pp. 245–250, January 1994.
- [83] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Entropy-constrained vector quantization," *IEEE Transactions on Signal Processing*, vol. 37, no. 1, pp. 31–42, January 1989.

- [84] V. A. Vaishampayan and J. C. Batllo, "Asymptotic analysis of multiple description quantizers," *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 278–284, January 1998.
- [85] C. Tian and S. S. Hemami, "Universal multiple description scalar quantization: analysis and design," *IEEE Transactions on Information Theory*, vol. 50, no. 9, pp. 2089–2102, September 2004.
- [86] J. Balogh and J. A. Csirik, "Index assignment for two-channel quantization," *IEEE Transactions on Information Theory*, vol. 50, no. 11, pp. 2737–2751, November 2004.
- [87] T. Y. Berger-Wolf and E. M. Reingold, "Index assignment for multichannel communication under failure," *IEEE Transactions on Information Theory*, vol. 48, no. 10, pp. 2656–2668, October 2002.
- [88] S. D. Servetto, V. A. Vaishampayan, and N. J. A. Sloane, "Multi description lattice vector quantization," in *Proceedings IEEE Data Compression Conference*, Snowbird, UT, March 1999, pp. 13–22.
- [89] V. A. Vaishampayan, N. J. A. Sloane, and S. D. Servetto, "Multiple description vector quantization with lattice codebooks: Design and analysis," *IEEE Transactions on Information Theory*, vol. 47, no. 5, pp. 1718–1734, 2001.
- [90] M. Fleming and M. Effros, "Generalized multiple description vector quantization," in *Proceedings IEEE Data Compression Conference*, March 1999, pp. 3–12.
- [91] N. Görtz and P. Leelapornchai, "Optimization of the index assignments for multiple description vector quantizers," *IEEE Transactions on Communications*, vol. 51, no. 3, pp. 336–340, March 2003.
- [92] P. Koulgi, S. Regunathan, and K. Rose, "Multiple description quantization by deterministic annealing," *IEEE Transactions on Information Theory*, vol. 49, no. 8, pp. 2067 – 2075, August 2003.
- [93] S. N. Diggavi, N. J. A. Sloane, and V. A. Vaishampayan, "Asymmetric multiple description lattice vector quantizers," *IEEE Transactions on Information Theory*, vol. 48, no. 1, pp. 174–191, January 2002.
- [94] V. A. Vaishampayan, A. R. Calderbank, and J. C. Batllo, "On reducing granular distortion in multiple description quantization," in *Proceedings IEEE Int. Symp. Information Theory*, August 1998, p. 98.
- [95] V. K. Goyal, J. A. Kelner, and J. Kovacević, "Multiple description vector quantization with a coarse lattice," *IEEE Transactions on Information Theory*, vol. 48, no. 3, pp. 781–788, March 2002.
- [96] Y. Wang, M. Orchard, and A. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," in *Proc. IEEE 1997 1st Workshop Multimedia Signal Processing (MMSP'97)*, Princeton, NJ, jun 1997, pp. 419–424.
- [97] M. T. Orchard, Y. Wang, V. A. Vaishampayan, and A. R. Reibman, "Redundancy rate-distortion analysis of multiple description coding using pairwise correlating transforms," in *Proc. IEEE International Conference on Image Processing*, 1997, pp. 608 – 611.

- [98] Y. Wang, M. Orchard, V. Vaishampayan, and A. Reibman, "Multiple description coding using pairwise correlation transforms," *IEEE Transactions on Image Processing*, vol. 10, no. 3, March 2001.
- [99] Y. Wang, A. Reibman, M. Orchard, and H. Jafarkhani, "An improvement to multiple description transform coding," *IEEE Transactions on Signal Processing*, vol. 50, no. 11, November 2002.
- [100] V. K. Goyal and J. Kovacević, "Generalized multiple description coding with correlating transforms," *IEEE Transactions on Information Theory*, vol. 47, no. 6, pp. 2199–2224, sep 2001.
- [101] S. S. Pradhan and K. Ramchandran, "On the optimality of block orthogonal transforms for multiple description coding of gaussian vector sources," *IEEE Signal Processing Letters*, vol. 7, no. 4, pp. 76 – 78, April 2000.
- [102] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: North-Holland, 1977.
- [103] V. K. Goyal, J. Kovacević, and M. Vetterli, "Multiple description transform coding: Robustness to erasures using tight frame expansion," in *Proceedings IEEE Int. Symp. Information Theory*, Cambridge, MA, August 1998, p. 408.
- [104] ———, "Quantized frame expansions as source-channel codes for erasure channels," in *Proceedings IEEE Data Compression Conference*, Snowbird, UT, March 1999, pp. 326–335.
- [105] V. K. Goyal, J. Kovacević, and J. A. Kelner, "Quantized frame expansions with erasures," *Applied and Computational Harmonic Analysis*, vol. 10, no. 3, pp. 203 – 233, May 2001.
- [106] J. Kovacević, P. L. Dragotti, and V. K. Goyal, "Filter bank frame expansions with erasures," *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1439 – 1450, June 2002.
- [107] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.
- [108] Y. Frank-Dayana and R. Zamir, "Dithered lattice-based quantizers for multiple descriptions," *IEEE Transactions on Information Theory*, vol. 48, no. 1, pp. 192 – 204, January 2002.
- [109] J. Chen, C. Tian, T. Berger, and S. S. Hemami, "A new class of universal multiple description lattice quantizers," in *Proceedings IEEE Int. Symp. Information Theory*, September 2005, pp. 1803 – 1807.
- [110] ———, "Multiple description quantization via gram-schmidt orthogonalization," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5197–5217, December 2006.
- [111] T. Fine, "Properties of an optimum digital system and applications," *IEEE Transactions on Information Theory*, vol. IT-10, pp. 443–457, October 1964.
- [112] A. Kurtenbach and P. Wintz, "Quantizing for noisy channels," *IEEE Trans. Commun. Techn.*, vol. COM-17, pp. 291–302, April 1969.
- [113] N. Farvardin and V. Vaishampayan, "Optimal quantizer design for noisy channels: An approach to combined source–channel coding," *IEEE Transactions on Information Theory*, vol. 33, no. 6, pp. 827–838, November 1987.

- [114] J. G. Dunham and R. M. Gray, "Joint source and noisy channel trellis encoding," *IEEE Transactions on Information Theory*, vol. 27, no. 4, pp. 516–519, July 1981.
- [115] J. D. Gibson and T. R. Fischer, "Alphabet-constrained data compression," *IEEE Transactions on Information Theory*, vol. 28, no. 3, pp. 443–457, May 1982.
- [116] H. Kumazawa, M. Kasahara, and T. Namekawa, "A construction of vector quantizers for noisy channels," *Electronics and engineering in Japan*, vol. 67-B, pp. 39–47, January 1984.
- [117] K. A. Zeger and A. Gersho, "Vector quantizer design for memoryless noisy channels," in *IEEE International Conference on Communications*, Philadelphia, USA, 1988, pp. 1593–1597.
- [118] N. Farvardin, "A study of vector quantization for noisy channels," *IEEE Transactions on Information Theory*, vol. 36, no. 4, pp. 799–809, July 1990.
- [119] N. Farvardin and V. Vaishampayan, "On the performance and complexity of channel-optimized vector quantizers," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 155–159, January 1991.
- [120] B. Juang and A. Gray, "Multiple stage vector quantization for speech coding," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 1982.
- [121] W. Y. Chan, S. Gupta, and A. Gersho, "Enhanced multistage vector quantization by joint codebook design," *IEEE Transactions on Communications*, vol. 40, no. 11, pp. 1693–1697, November 1992.
- [122] N. Phamdo, N. Farvardin, and T. Moriya, "A unified approach to tree-structured and multistage vector quantization for noisy channels," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 835–850, May 1993.
- [123] V. Vaishampayan and N. Farvardin, "Joint design of block source codes and modulation signal sets," *IEEE Transactions on Information Theory*, vol. 38, no. 4, pp. 1230–1248, July 1992.
- [124] M. Skoglund and P. Hedelin, "Hadamard-based soft decoding for vector quantization over noisy channels," *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 515–532, March 1999.
- [125] M. Skoglund, "Soft decoding for vector quantization over noisy channels with memory," *IEEE Transactions on Information Theory*, vol. 45, no. 4, pp. 1293–1307, May 1999.
- [126] N. Phamdo, F. Alajaji, and N. Farvardin, "Quantization of memoryless and Gauss-Markov sources over binary Markov channels," *IEEE Transactions on Communications*, vol. 45, no. 6, pp. 668–675, June 1997.
- [127] F. Alajaji and N. Phamdo, "Soft-decision COVQ for Rayleigh fading channels," *IEEE Communications Letters*, vol. 2, no. 6, pp. 162–164, June 1998.
- [128] N. Phamdo and F. Alajaji, "Soft-decision demodulation design for COVQ over white, colored and ISI Gaussian channels," *IEEE Transactions on Communications*, vol. 48, no. 9, pp. 1499–1506, September 2000.
- [129] K. A. Zeger and A. Gersho, "Zero redundancy channel coding in vector quantization," *IEEE Electronic Letters*, vol. 23, no. 12, pp. 654–655, June 1987.

- [130] J. DeMarca and N. Jayant, "An algorithm for assigning binary indices to the codevectors of a multi-dimensional quantizer," in *IEEE Internationell Conference on Communications*, Seattle, USA, 1987, pp. 1128–1132.
- [131] D. J. Goodman and T. J. Moulslley, "Using simulated annealing to design digital transmission codes for analogue sources," *IEE Electronic Letters*, vol. 24, no. 10, pp. 617–619, May 1988.
- [132] K. A. Zeger and A. Gersho, "Pseudo-Gray coding," *IEEE Transactions on Communications*, vol. 38, no. 12, pp. 2147–2158, December 1990.
- [133] P. Knagenhjelm and E. Agrell, "The Hadamard transform — A tool for index assignment," *IEEE Transactions on Information Theory*, vol. 42, no. 4, pp. 1139–1151, July 1996.
- [134] R. Hagen and P. Hedelin, "Robust vector quantization by a linear mapping of a block code," *IEEE Transactions on Information Theory*, vol. 45, no. 1, pp. 200–218, January 1999.
- [135] M. Skoglund, "On channel constrained vector quantization and index assignment for discrete memoryless channels," *IEEE Transactions on Information Theory*, vol. 45, no. 7, pp. 2615–2622, November 1999.

Part II

Included papers

Paper A

Polynomial Based Analog Source–Channel Codes

Niklas Wernersson, Mikael Skoglund and Tor Ramstad
Submitted to *IEEE Transactions of Communications*

Polynomial Based Analog Source–Channel Codes

Niklas Wernersson, Mikael Skoglund and Tor Ramstad

Abstract

In many communication applications one is interested in transmitting a time-discrete analog-valued (i.e. continuous alphabet) source over a time-discrete analog channel. We study this problem in the case of bandwidth expansion, in the sense that one source sample, X , is transmitted over N orthogonal channels. An analog source–channel code based on orthogonal polynomials is proposed and analyzed. The code can be generated using a Gram-Schmidt procedure, to fit virtually any source distribution.

Index Terms—Modulation, nonlinear functions, source coding, channel coding, error correction.

1 Introduction

We consider the problem of analog bandwidth expansion over a Gaussian memoryless channel. This means designing analog source–channel bandwidth expansion codes where a source sample X is transmitted using an AWGN channel N times, yielding a $1 : N$ expansion rate. (We will from now on refer to these codes as 'analog source–channel codes'.) The common solution for this problem is digital and is implemented by producing separate source and channel codes. In practice, this is generally done by quantizing the source followed by digital channel coding and transmission. Due to powerful source and channel coding techniques the performance of such a system can be very high when the channel quality is close to what the system has been designed for. There are, however, some disadvantages with the digital approach. In order to get a high performance long block lengths are required both for the source and channel code. This will therefore introduce delays into the system which may be undesirable, especially for a real time system. There is also a threshold effect associated with a digital system: if the channel quality goes below a certain level the channel code will break down and the system performance will deteriorate rapidly. On the other hand, if the channel quality is increased above this level the performance will not increase but rather reach a constant level which is due to the nonrepairable errors introduced by the quantizer.

In recent years analog, or at least partially analog, systems as an alternative to digital systems have received increased attention, see e.g. [1, 2] and the references therein. Analog systems do, in general, not have the same disadvantages as digital systems. Hence, in some scenarios an analog approach may be more suitable than a digital one. On the other hand, in

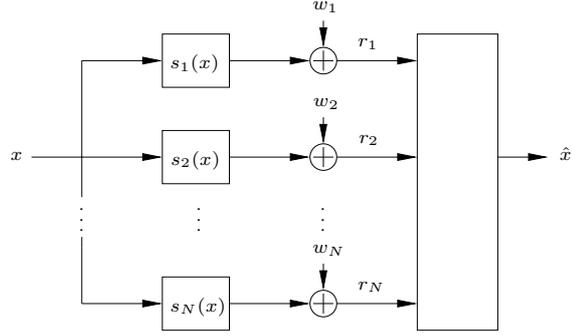


Figure 1: Structure of the system.

practice, the performance of a digital system is in general higher than for an analog system when being used for the channel quality that it has been designed for.

Analog bandwidth expansion was briefly discussed already in one of Shannon's early papers [5]. Furthermore [6] analyzes the performance of analog source-channel codes at high SNR's. The papers [7–9] present ideas for how to construct these codes and in [10, 11] it is concluded that the problem also relates to the problem of distributed estimation. However, to our knowledge, the only more explicit results on how to design analog source-channel codes are found in [2–4]. In all these papers a uniform source distribution is considered.

Our main contribution in this paper, which was partly presented in [12], is to present a method for designing well-performing analog source-channel codes assuming a general source distribution.

The paper is organized as follows. In Section 2 the problem is formulated. Section 3 provides a brief introduction to orthogonal polynomials. Section 4 will propose an analog source-channel code and Section 5 will present some analytical results for this code. Finally we present some simulation results and conclusions in Sections 6-7.

2 Problem Formulation

Consider the problem illustrated in Figure 1. An analog source value x , with variance σ_x^2 and pdf $f(x)$, is encoded by the function

$$\mathbf{s}(x) = (s_1(x), s_2(x), \dots, s_N(x))^T \quad (1)$$

and transmitted over N orthogonal AWGN channels created by using e.g. TDMA or FDMA. The different encoding functions perform analog mappings, that is $s_i : \mathbb{R} \rightarrow \mathbb{R}$, under the power constraints

$$E[s_i(X)^2] = \int_{\mathbb{R}} f(x) s_i(x)^2 dx \leq P \quad \forall i \in \{1, 2, \dots, N\}. \quad (2)$$

The decoder estimates x based on the received values

$$\mathbf{r} = \mathbf{s}(x) + \mathbf{w} \quad (3)$$

where \mathbf{w} is independent and identically distributed (i.i.d.) memoryless Gaussian distributed with covariance matrix $\sigma_w^2 I$. Hence, the decoding is performed as

$$\hat{x} = \hat{x}(\mathbf{r}) \quad (4)$$

and the objective is to minimize the expected mean square error (MSE) $E[(X - \hat{X})^2]$.

When the source is i.i.d. zero-mean Gaussian and $N = 1$, it is well known that *linear encoding* is optimal, i.e. $s_1(x) = \sqrt{(P/\sigma_w^2)}x$, under the assumption that the decoder knows the source and noise variances. However, when $N > 1$ this is no longer true and nonlinear encoding can have superior performance compared to linear encoding strategies, see e.g. [13]. One of the reasons for this is that a linear encoding function $\mathbf{s}(x)$ only uses a one dimensional subspace of the channel space. More efficient mappings would use a higher number of the available channel space dimensions. An example of this is illustrated in Figure 2 for $N = 2$. By using nonlinear encoding functions, illustrated by the solid 'S-shaped' curve $\mathbf{s}(x)$, we are able to better fill the channel space than when using linear encoding functions, represented by the dashed curve. A longer curve essentially means a higher resolution when estimating x as long as we decode to the right fold of the curve, illustrated by sample x_1 in the figure. However, decreasing the SNR will at some point result in that different folds of the curve will lie too close to each other and the decoder will start making large decoding errors, illustrated by sample x_2 in the figure. Decreasing the SNR below this threshold will therefore significantly deteriorate the performance. We refer to these errors as 'small' and 'large' decoding errors. Increasing the SNR, on the other hand, will always improve the performance since the magnitude of the small decoding errors will decrease. This is one of the main advantages of analog systems compared to digital systems since the performance of a digital system will approach a saturation level when the SNR grows large.

The problem of designing analog source–channel codes is therefore a problem of finding nonlinear curves $\mathbf{s}(x) : \mathbb{R} \rightarrow \mathbb{R}^N$ under the power constraints (2) such that points far separated in the source space also are far separated in the channel space, i.e.

$$\|\mathbf{s}(x) - \mathbf{s}(x + \delta)\|^2 > \Delta^2 \quad \forall \{(x, \delta) : f(x) > 0, |\delta| > \delta_0\} \quad (5)$$

where Δ and δ_0 are some positive constants. If this condition is true, no large (the value of δ_0 will define "large") decoding errors will occur if the probability of noise vectors $2\|\mathbf{w}\|^2 > \Delta^2$ can be neglected, see e.g. [14]. This will be true when the SNR reaches above some certain threshold. Furthermore, given that no large coding errors occur the expected MSE will only depend on the small decoding errors. In [15] it is shown that under these conditions the MSE will depend on the stretch of $\mathbf{s}(x)$ defined as

$$\mathcal{S}(x) = \left\| \frac{d}{dx} \mathbf{s}(x) \right\|, \quad (6)$$

which preferably should be as large as possible. The expected MSE is then well approximated by

$$E[(X - \hat{X})^2] = \sigma_w^2 \int_{\mathbb{R}} \frac{f(x)}{\left\| \frac{d}{dx} \mathbf{s}(x) \right\|^2} dx. \quad (7)$$

To summarize, we would like to 'stretch' the curve $\mathbf{s}(x)$ as much as possible under the constraints in (2) and (5). We will in this paper design such curves using orthogonal polynomials.

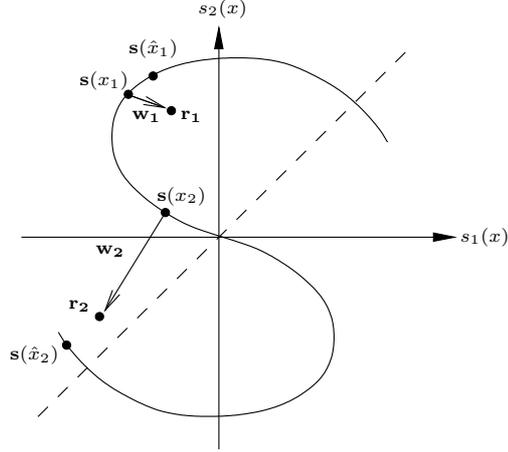


Figure 2: x_1 illustrates a 'small' decoding error and x_2 illustrates a 'large' decoding error.

3 Orthogonal Polynomials

The orthogonal polynomials literature is vast, see e.g. [16, 17] for an introduction to the area. We will here summarize some important results which will be used in the following sections.

3.1 Definition

A sequence of polynomials $\{P_n(x)\}_{n=0}^{\infty}$ is called *orthogonal* with respect to the weight function $f(x)$ if $P_n(x)$ is a polynomial of degree n and for all non-negative n and m

$$\int_{\mathbb{R}} f(x) P_n(x) P_m(x) dx = \delta_{nm} K_n, \quad K_n \neq 0 \quad (8)$$

where δ_{nm} is Kronecker's delta symbol. Furthermore, if $K_n = 1 \forall n$ the sequence of polynomials $\{P_n(x)\}_{n=0}^{\infty}$ is *orthonormal*. We will use the notation $p_n(x)$ when referring to orthonormal polynomials, hence

$$p_n(x) = \sqrt{1/K_n} P_n(x). \quad (9)$$

3.2 The recurrence formula and zeros

It can be shown that given a sequence of orthogonal polynomials $\{P_n(x)\}_{n=0}^{\infty}$ there exist constants c_n and λ_n such that

$$P_n(x) = (x - c_n) P_{n-1}(x) - \lambda_n P_{n-2}(x), \quad n > 0 \quad (10)$$

where $P_0(x) = 1$ and $P_{-1}(x) = 0$. These constants will depend of the weight function $f(x)$. The zeros of $P_n(x)$ are real and simple. We will denote the k :th smallest zero of

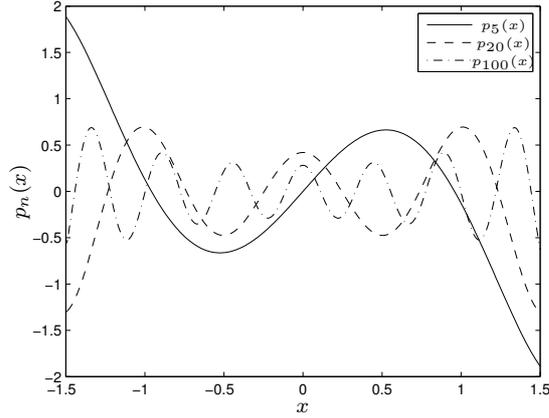


Figure 3: Orthonormal polynomials for a Gaussian source ($\sigma^2 = 0.5$).

$P_n(x)$ as $x_{k,n}$. It can be shown that the zeros of $P_n(x)$ and $P_{n+1}(x)$ mutually separate each other such that

$$x_{k,n+1} < x_{k,n} < x_{k+1,n+1}, \quad k = 1, 2, \dots, n. \quad (11)$$

These properties explain the well known fact that orthogonal polynomials have an oscillating behavior, see e.g. [18]. Due to that all zeros of $P_n(x)$ are real and simple we can write

$$P_n(x) \propto (x - x_{1,n})(x - x_{2,n}) \cdots (x - x_{n,n}), \quad (12)$$

meaning that as x increases $P_n(x)$ will alternate sign when the different zeros are passed. In fact, the polynomial $P_n(x)$ usually behave quite similar to a sine if studied in a fairly short interval where $f(x)$ is strong, at least when n grows large. This is illustrated in Figure 3 where orthonormal polynomials for a Gaussian source with variance 0.5 is plotted. As it turns out, the oscillating property will be important for our proposed analog source-channel code.

3.3 Gram-Schmidt

Given a weight function $f(x) \geq 0$ orthonormal polynomials $\{p_n(x)\}_{n=0}^{\infty}$ will exist. These polynomials can be derived by a Gram-Schmidt procedure described briefly below.

1. Set $P_0(x) = 1$ and create $p_0(x)$ from (9). Set $n = 0$.
2. Create

$$P_{n+1}(x) = x^{n+1} - \sum_{k=0}^n a_k p_k(x) \quad (13)$$

where by choosing

$$a_k = \int_{\mathbb{R}} f(x) x^{n+1} p_k(x) dx \quad (14)$$

it is ensured that $P_{n+1}(x)$ is orthogonal to $\{p_l(x)\}_{l=0}^n$ since

$$\begin{aligned} \int_{\mathbb{R}} f(x)P_{n+1}(x)p_l(x)dx &= \int_{\mathbb{R}} f(x)x^{n+1}p_l(x)dx - \int_{\mathbb{R}} f(x) \sum_{k=0}^n a_k p_k(x)p_l(x)dx \\ &= a_l - a_l = 0 \end{aligned} \quad (15)$$

if $l < n + 1$.

3. Create $p_{n+1}(x)$ by (9) and set $n = n + 1$. Go to Step 2.

Hence, given a certain pdf $f(x)$ the procedure above will produce the corresponding orthonormal polynomials.

3.4 The Christoffel–Darboux identity

Finally we state the Christoffel–Darboux identity,

$$\sum_{n=0}^N p_n(x)p_n(u) = \sqrt{\frac{K_{N+1}}{K_N}} \frac{p_{N+1}(x)p_N(u) - p_N(x)p_{N+1}(u)}{x - u}. \quad (16)$$

4 Encoding and Decoding

We propose a nonlinear analog source–channel code as follows. Given $\mathcal{I} = \{i_1, i_2, \dots, i_N\}$ the analog source–channel code $\mathbf{s}_{\mathcal{I}}(x)$ is created by choosing

$$s_j(x) = \sqrt{P}p_{i_j}(x) \quad \forall j \in \{1, 2, \dots, N\}. \quad (17)$$

For example, given a source distribution $f(x)$ and $\mathcal{I} = \{1, 2, 3, 4\}$ we will use $\sqrt{P}p_1(x)$ as encoding function for the first channel, $\sqrt{P}p_2(x)$ for the second channel and so on. Here the $p_n(x)$'s are orthonormal with respect to the weight function $f(x)$. Hence, the weight function equals the source pdf meaning that the power constraint in (2) will be fulfilled since

$$E[s_j(X)^2] = \int_{\mathbb{R}} f(x)[\sqrt{P}p_{i_j}(x)]^2 dx = P. \quad (18)$$

Optimal decoding, in the minimum MSE sense, is given as [19]

$$\hat{x}(\mathbf{r}) = E[x|\mathbf{r}] \quad (19)$$

but this decoding function would be very complex to implement. We therefore choose to instead implement the suboptimal maximum likelihood decoder

$$\hat{x}(\mathbf{r}) = \arg \max_x g(\mathbf{r}|x) = \arg \min_x \|\mathbf{s}_{\mathcal{I}}(x) - \mathbf{r}\|^2 = \arg \min_x \left[\sum_{i_j \in \mathcal{I}} (\sqrt{P}p_{i_j}(x) - r_j)^2 \right], \quad (20)$$

where $g(\cdot|\cdot)$ denotes the channel transition pdf. Hence, the decoding procedure is equivalent to minimizing a $2i_N$ -degree polynomial (assuming i_N is the largest index in \mathcal{I}). Efficient algorithms exist for minimizing univariate polynomials which makes this feasible.¹ It should be emphasized that (20), i.e. the receiver, does not need to know σ_w^2 .

Finally, note that the special case $\mathcal{I}_L \triangleq \{1, 1, \dots, 1\}$ means that the encoding is linear as discussed in Section 2 and in Figure 2. We will later use this system as a benchmark system.

5 Analysis

We will here discuss the three desirable properties for analog source–channel codes discussed in Section 2: *A*) distance properties as in (5), *B*) stretch properties as in (7) and finally, *C*) the ability to use up to all available channel space dimensions. We will show that the proposed scheme possesses properties *A* and *C* and we further conjecture that it also possesses property *B* for a large class of pdf:s $f(x)$. For such pdf:s the suggested scheme will therefore procedure a working source–channel code which consequently can be generated using the Gram–Schmidt procedure from Section 3.

5.1 Distance

Assume that four consecutive orthonormal polynomials are used as an analog source–channel code, hence $\mathcal{I} = \{n-3, n-2, n-1, n\}$. We will here show that this implies (5).

Proof: If (5) is not valid for some x and $|\delta| > 0$ we have $\|\mathbf{s}_{\mathcal{I}}(x) - \mathbf{s}_{\mathcal{I}}(x + \delta)\| = 0$ meaning $P_i(x) = P_i(x + \delta)$ for $i \in \mathcal{I}$. If we use this in the recurrence equation (10) we get

$$P_n(x + \delta) = (x + \delta - c_n)P_{n-1}(x + \delta) - \lambda_n P_{n-2}(x + \delta)$$

and hence

$$P_n(x) = (x - c_n)P_{n-1}(x) + \delta P_{n-1}(x) - \lambda_n P_{n-2}(x).$$

Comparing this last equation with (10) we realize that this can be true only if: *i*) $\delta = 0$ or *ii*) $P_{n-1}(x) = 0$. Hence, x must equal one of $P_{n-1}(x)$'s zeros $x_{k,n-1}$. Repeating this derivation for polynomials $\{n-3, n-2, n-1\}$ and studying $x = x_{k,n-1}$ we get

$$P_{n-1}(x_{k,n-1}) = (x_{k,n-1} - c_{n-1})P_{n-2}(x_{k,n-1}) + \delta P_{n-2}(x_{k,n-1}) - \lambda_{n-1} P_{n-3}(x_{k,n-1}).$$

Hence, also $P_{n-2}(x_{k,n-1})$ needs to equal 0 which can not be true according to (11). Hence, there can be no point where $\mathbf{s}_{\mathcal{I}}(x) = \mathbf{s}_{\mathcal{I}}(x + \delta)$ if $|\delta| > 0$ and $\mathcal{I} = \{n-3, n-2, n-1, n\}$ meaning that $\Delta^2 > 0$ will exist such that (5) is fulfilled. ■

In Figure 4 we illustrate the distance properties for a few different \mathcal{I} 's by plotting the function $\|\mathbf{s}_{\mathcal{I}}(x) - \mathbf{s}_{\mathcal{I}}(x_0)\|^2$ in the case of a uniform source distribution, i.e. $f(x) = 1/2$ for $x \in (-1, 1)$. All these codes have appropriate distance properties.

¹It should here be pointed out that when simulating the scheme for large i_N 's our computer's default accuracy for float values was not high enough making the minimization procedure to fail due to round off errors. This therefore needed to be considered. For our simulations in Section 6 the effect became visible for i_N 's somewhere between 20 and 25.

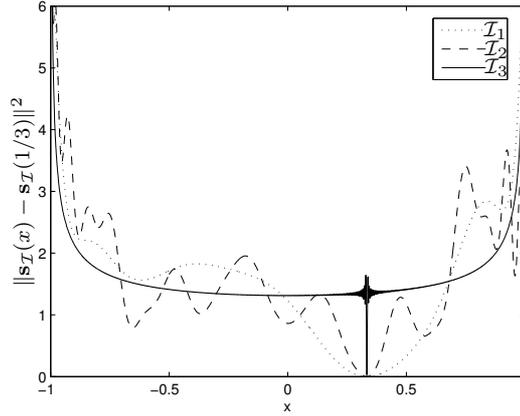


Figure 4: The distance $\|\mathbf{s}_{\mathcal{I}}(x) - \mathbf{s}_{\mathcal{I}}(x_0)\|^2$ for $\mathcal{I}_1 = \{1, 2, 3, 4, 5, 6\}$, $\mathcal{I}_2 = \{1, 2, 5, 6, 17, 18\}$ and $\mathcal{I}_3 = \{1, 2, \dots, 1000\}$ when $x_0 = 1/3$. The powers have been normalized so that each channel uses $P = 1/N$. Similar results were obtained also when other values for x_0 are chosen.

5.2 Stretch

Given that the SNR is high the MSE is well approximated by (7). We conjecture that given a smooth and continuous $f(x)$ for $\mathcal{I} = \{1, 2, \dots, N\}$, where N is large enough, we have

$$\int_{\mathbb{R}} \frac{f(x)}{\|\frac{d}{dx} \mathbf{s}_{\mathcal{I}}(x)\|^2} dx < \int_{\mathbb{R}} \frac{f(x)}{\|\frac{d}{dx} \mathbf{s}_{\mathcal{I}_L}(x)\|^2} dx. \quad (21)$$

In fact, for all the different $f(x)$:s we have evaluated, for instance the ones in the Section 6, the inequality has been true already for $N = 2$. Consequently, for high SNR's the proposed code will produce a better MSE than a linear code.

5.3 Dimensions

As earlier commented on a weakness with the linear code $\mathbf{s}_{\mathcal{I}_L}(x)$ is that only a one dimensional subspace of the channel space is used for transmission. We will here show that given $f(x_{k, i_{N+1}}) \neq 0 \forall k$ and $\mathcal{I} = \{i_1, i_2, \dots, i_N\}$ where $i_1 < i_2 < \dots < i_N$ all dimensions in the channel space will be used.

Proof: If we use $\bar{\mathcal{I}} = \{0, 1, \dots, i_N\}$ to create the $(i_N + 1) \times (i_N + 1)$ matrix

$$\bar{A} = [\mathbf{s}_{\bar{\mathcal{I}}}(x_{1, i_{N+1}}) \quad \mathbf{s}_{\bar{\mathcal{I}}}(x_{2, i_{N+1}}) \quad \dots \quad \mathbf{s}_{\bar{\mathcal{I}}}(x_{i_N+1, i_{N+1}})]$$

all columns will be orthogonal. This comes from the fact that the right-hand-side of (16) will always equal 0 if $x = x_{i, i_{N+1}}$ and $u = x_{j, i_{N+1}}$ and $i \neq j$. The left-hand-side of (16)

describes the dot product of column vectors i and j in \bar{A} which hence are orthogonal. This further means that \bar{A} is full rank. If we instead create the matrix

$$A = [\mathbf{s}_{\mathcal{I}}(x_{1,i_{N+1}}) \ \mathbf{s}_{\mathcal{I}}(x_{2,i_{N+1}}) \ \cdots \ \mathbf{s}_{\mathcal{I}}(x_{i_{N+1},i_{N+1}})]$$

it is a version of \bar{A} where a number of rows are removed. Since \bar{A} is full rank also A will be full rank. Hence, $\mathbf{s}_{\mathcal{I}}(x)$ will use all N available dimensions in the channel space for transmission since we have assumed $f(x_{k,i_{N+1}}) \neq 0 \ \forall k$. ■

To summarize, the curve $\mathbf{s}_{\mathcal{I}}(x)$ is not contained in a subspace of the N -dimensional channel space.

6 Simulations

6.1 Choosing \mathcal{I}

In Section 5 we motivated that orthogonal polynomials have properties making them appropriate as analog source–channel codes as proposed in Section 4. However, the question on how to choose which indexes to include in \mathcal{I} remains. We will here present a strategy for this based on results derived in [2]. Only the case with symmetric pdf's will be considered and we will leave the asymmetric case as a future topic.

As discussed in Section 3 orthonormal polynomials have an oscillating property. In this sense our proposed scheme is similar to the scheme in [2] where sine and cosine functions with different frequencies are used to create analog source–channel codes for a uniformly distributed source $x \in (-1, 1)$. Furthermore, [2] carries out a rigorous analysis in order to optimize the choice of which frequencies to use which leads to the encoding structure

$$\left. \begin{aligned} s_{1+2i}(x) &= \alpha \cos(\pi a^i x) \\ s_{2+2i}(x) &= \alpha \sin(\pi a^i x) \end{aligned} \right\} \forall i \in \{0, 1, \dots, N/2 - 1\} \quad (22)$$

where $a \in \mathbb{N}$. It is shown that a higher a leads to a higher stretch but also that different folds of the curve is packed closer to each other. If we study $(s_1(x), s_2(x))$ we note that this pair of functions will create a circle in the plane when x is increased from -1 to 1 . Furthermore, for $(s_{1+2i}(x), s_{2+2i}(x))$ the same behavior is obtained with the difference that the function will go a^i laps around the circle. Our proposed structure will approximate this behavior if

$$\mathcal{I} = \{2a^0 - 1, 2a^0, 2a^1 - 1, 2a^1, 2a^2 - 1, 2a^2, \dots\}. \quad (23)$$

This comes from the fact that $p_{2a^k}(x)$ will have $2a^k$ zeros and hence a^k 'periods' (oscillations) before the polynomial starts approaching $\pm\infty$. Due to the distribution of zeros (11) there will exist 'phase shifts' between $p_i(x)$ and $p_{i+1}(x)$ and we can roughly think of every other polynomial as a sine and every other as a cosine, at least for symmetric pdf's. The curve $(p_{2a^i-1}(x), p_{2a^i}(x))$ will therefore approximate the circles created by the system in (22). This is illustrated in Figure 5 where $(p_{49}(X), p_{50}(X))$ have been simulated for 50000 Gaussian source samples X , darker points in the plot means more occurring points. Most of the source samples are mapped to an area approximately looking like a circle around the origin although we also note that the samples in the tails of the pdf $f(x)$ will be mapped to regions further out. Therefore the results obtained in [2] motivates the choice of \mathcal{I} as in (23). This choice has also been supported by simulations conducted where other choices of \mathcal{I} have been evaluated.

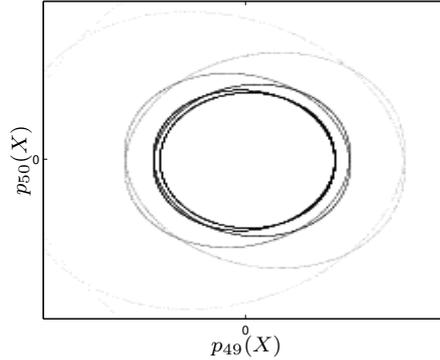


Figure 5: $p_{49}(x)$ versus $p_{50}(x)$ simulated for a Gaussian source. The darker point, the more occurrences.

6.2 Optimal companding and benchmark systems

As previously noted, to our knowledge, the only existing schemes of analog source-channel codes are [2–4] which all are designed for a uniform distribution. We will use [2, 3] as benchmark systems and from now on refer to these as the V&C and C&W codes. A weakness with these is that $s(-1) = s(1)$. Hence, they are sensitive for large decoding errors when transmitting values close to the ends of the support region for the distribution. This is solved by instead encoding βx , rather than x , where β is some appropriate constant less than 1 in order to prevent large decoding errors. However, this property makes it hard to naturally extend the ideas to other distributions, for instance the Gaussian distribution which does not have a limited support. This was briefly commented on in [2] and it was suggested that optimal companding, see e.g. [15], could be used in conjunction with the codes in order to create codes for other distributions. In order to get some benchmark systems for nonuniform sources we therefore use optimal companding together with the V&C and C&W codes. The compander will map the source x to $y(x) \in (-1, 1)$ and then transmit $s(y(x))$.² From [15] we get the optimal compander, in the minimum MSE sense, as

$$y(x) = 2 \frac{\int_{-\infty}^x f(u)^{\frac{1}{3}} du}{\int_{-\infty}^{\infty} f(u)^{\frac{1}{3}} du} - 1. \quad (24)$$

The receiver will estimate \hat{y} and the inverse companding operation is performed such that \hat{x} is found fulfilling $\hat{y} = y(\hat{x})$.

²Since $y(x)$ produced by the compander not is uniformly distributed the individual power constraints of (2) will be violated. We allow this and instead require the total power consumption to be less than NP for the benchmark systems.

6.3 Results

In order to evaluate the proposed code we carried out simulations for three different source distributions: Uniform, Gaussian and a Gaussian mixture (GM). The pdf's for these can be written

$$\begin{aligned} f_U(x) &= 1/2 \quad x \in (-1, 1) \\ f_G(x, \sigma^2) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \\ f_{GM}(x, \sigma^2) &= \sum_i q_i f_G(x - \mu_i, \sigma^2) \end{aligned}$$

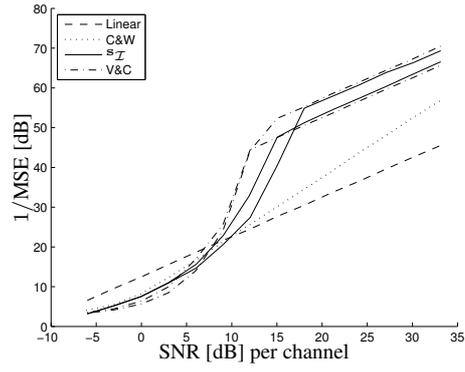
where we used $\sigma^2 = 0.5$ for the Gaussian case. This results in two of the most famous orthogonal polynomials for the Uniform and Gaussian source, namely the Legendre and Hermite polynomials which both are well documented and explicit formulas for the polynomials exists. For the GM distribution we used $\sigma^2 = 1$, $\mathbf{q} = 1/6 \cdot [1, 1, 2, 1, 1]$ and $\mu = [-7, -4, 0, 4, 7]$ which is not a well documented weight function. Therefore this simulation tested the usefulness of the Gram–Schmidt procedure from Section 3 which was used to generate the corresponding analog source–channel code.

In Figure 6 simulation results are presented for the case of 1 : 6 expansion. Performance is measured in terms of $1/\text{MSE}$ versus SNR per channel, i.e. $\text{SNR} \triangleq P/\sigma_w^2$. All curves show similar behavior, for low SNR's large decoding errors occur and the performance is low. Increasing the SNR from this point leads to an increase in performance and eventually the threshold where large decoding errors stop occurring is reached and the performance starts behaving according to (7).

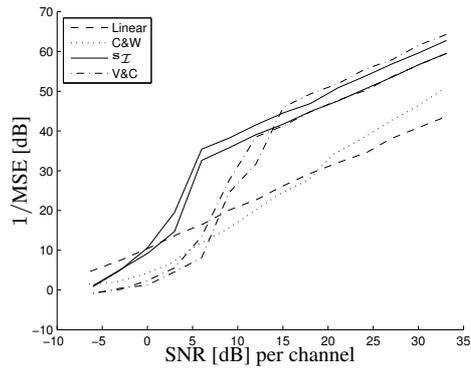
Figure 1.6(a) shows results for the uniform source. This is the case for which both the V&C and C&W codes were constructed. We experienced that the V&C code perform better than our proposed scheme. It appears as if when the stretch for both codes are chosen such that the performance is similar above the threshold the V&C code reaches its threshold at a lower SNR than our code. The C&W code on the other hand is quite far behind in terms of performance. The slope of the performance curve for the C&W code is on the other hand steeper which was also explained in [3]. The linear code, $\mathbf{s}_{\mathcal{L}}(x)$, is the best choice only for very bad channels.

In Figure 1.6(b) the Gaussian source is simulated and optimal companding has been implemented for the V&C and C&W codes. The companding (24) and the inverse companding operations, which for this case can be expressed in terms of the Q–function, will naturally add complexity. Here the proposed scheme has the superior performance.

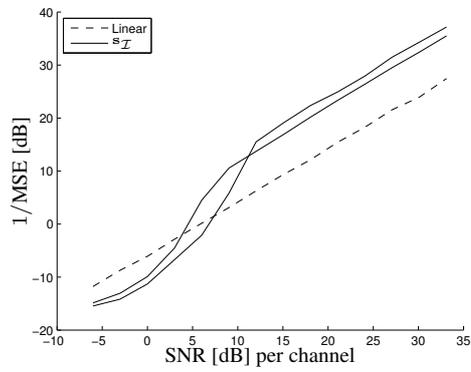
Finally, in Figure 1.6(c) we have also simulated the GM source. Deriving the companding integral (24) for this pdf is difficult and we can therefore only compare to linear coding. From the figure we see that the Gram–Schmidt generated polynomials produces a well working analog source–channel code. This illustrates the power of the proposed scheme and the Gram–Schmidt procedure in order to design analog source–channel codes for general distributions.



(a) Uniform



(b) Gaussian



(c) Gaussian mixture

Figure 6: Simulations for different source distributions. In (a)–(b) we use $a \in \{3, 4\}$ for V&C and $a \in \{4, 5\}$ for $s_I(x)$. In (c) we use $a \in \{2, 5\}$ for $s_I(x)$.

7 Conclusions

We have presented an analog source-channel coding scheme for bandwidth expansion and a Gram-Schmidt procedure for generating the code for a general source has been described. The scheme has a similar performance to the the best previous existing scheme but is more general and can be implemented for a larger class of source distributions.

References

- [1] M. Skoglund, N. Phamdo, and F. Alajaji, "Hybrid digital-analog source-channel coding for bandwidth compression/expansion," *IEEE Transactions on Information Theory*, vol. 52, no. 8, pp. 3757–3763, August 2006.
- [2] V. Vaishampayan and S. I. R. Costa, "Curves on a sphere, shift-map dynamics, and error control for continuous alphabet sources," *IEEE Transactions on Information Theory*, vol. 49, no. 7, pp. 1658–1672, July 2003.
- [3] B. Chen and G. W. Wornell, "Analog error-correcting codes based on chaotic dynamical systems," *IEEE Transactions on Communications*, vol. 46, no. 7, pp. 881–890, July 1998.
- [4] M. Taheradeh and A. K. Khandani, "Robust joint source-channel coding for delay-limited applications," in *Proceedings IEEE Int. Symp. Information Theory*, June 2007, pp. 726–730.
- [5] C. E. Shannon, "Communication in the presence of noise," *Proc. IRE*, pp. 10–21, January 1949.
- [6] J. Ziv, "The behavior of analog communications systems," *IEEE Transactions on Information Theory*, vol. 16, no. 5, pp. 587–594, September 1970.
- [7] U. Timor, "Design of signals for analog communication," *IEEE Transactions on Information Theory*, vol. 16, no. 5, pp. 581–587, September 1970.
- [8] D. D. McRay, "Performance evaluation of a new modulation technique," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 431–445, August 1971.
- [9] P. A. Floor, T. A. Ramstad, and N. Wernersson, "Power constrained channel optimized vector quantizers used for bandwidth expansion," in *IEEE International Symposium on Wireless Communication Systems*, October 2007.
- [10] N. Wernersson, J. Karlsson, and M. Skoglund, "Distributed quantization over noisy channels," *IEEE Transactions on Communications*, 2008, to appear.
- [11] N. Wernersson and M. Skoglund, "Nonlinear coding and estimation for correlated data in wireless sensor networks," *IEEE Transactions on Communications*, 2008, submitted.
- [12] N. Wernersson, M. Skoglund, and T. Ramstad, "Analog source-channel codes based on orthogonal polynomials," in *Asilomar Conference on Signals, Systems and Computers*, November 2007.
- [13] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering*. Wiley, 1965.

- [14] L. P. Seidman, "An upper bound on average estimation error in nonlinear systems," *IEEE Transactions on Information Theory*, vol. 14, no. 2, pp. 243–250, March 1968.
- [15] D. J. Sakrison, *Transmission of Waveforms and Digital Information*. Wiley, 1968.
- [16] T. S. Chihara, *An introduction to orthogonal polynomials*. Mathematics and its Applications, 1978.
- [17] G. Szegő, *Orthogonal polynomials*, 4th ed. American Mathematical Society, 1975.
- [18] A. Mate, P. Nevai, and V. Totik, "Oscillatory behavior of orthogonal polynomials," *Proceedings of the American Mathematical Society*, vol. 96, no. 2, pp. 261–268, February 1986.
- [19] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.

Paper B

Distributed Quantization over Noisy Channels

Niklas Wernersson, Johannes Karlsson and Mikael Skoglund
To appear in *IEEE Transactions of Communications*

Distributed Quantization over Noisy Channels

Niklas Wernersson, Johannes Karlsson and Mikael Skoglund

Abstract

The problem of designing simple and energy-efficient sensor nodes in a wireless sensor network is considered from a joint source–channel coding perspective. An algorithm for designing distributed scalar quantizers for orthogonal channels is proposed and evaluated. In particular the cases of the binary symmetric channel as well as the additive white Gaussian noise channel are studied. It is demonstrated that correlation between sources can be useful in order to reduce quantization distortion as well as protecting data when being transmitted over non-ideal channels. It is also demonstrated that the obtained system is robust against channel SNR mismatch.

Index Terms—Source coding, quantization, channel coding, correlation.

1 Introduction

Wireless sensor networks are expected to play an important role in tomorrow's sensing systems. One important property in these networks is that there may be a high correlation between different sensor measurements due to high spatial density of sensor nodes. This motivates source coding of correlated sources, which has been analyzed in for instance [1] where the well known Slepian–Wolf theorem is stated. Ideas on how to perform practical Slepian–Wolf coding are presented in [2, 3], allowing the use of powerful channel codes such as LDPC and Turbo codes in the context of distributed source coding, see e.g. [4, 5]. For the case with continuous sources, i.e. lossy coding, relevant references include [6, 7]. In general, these methods require the use of long codes and the encoding complexity will require some data processing in the sensor nodes. This will therefore counteract one of the desired design criteria in sensor network design, namely low cost and energy efficient sensor nodes. In addition, in many applications for example in networked control, a low delay is essential, preventing the use of long codes.

An alternative is therefore to design sensor nodes of very low complexity and low delay. This can be accomplished by interpreting the distributed source coding problem as a quantization problem. Previously, quantization of correlated sources has been studied in [8–13]. Our work is however targeted towards wireless sensor networks and introducing noisy channels is necessary in order to make the system more realistic. For non-ideal channels related previous work includes [14] which considers the problem of distributed detection over non-ideal channels. In [15] quantization of correlated sources in a packet network is studied,

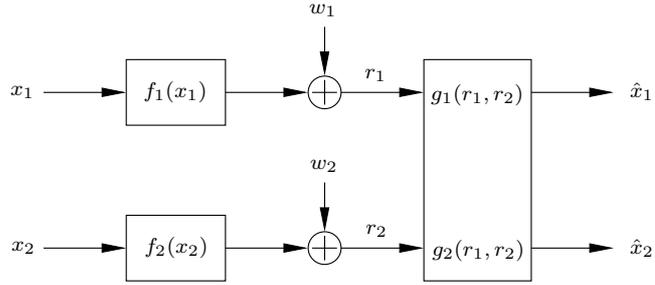


Figure 1: Structure of the system.

resulting in a general problem including multiple description coding as well as distributed source coding as special cases.

We will in this paper summarize and continue the work carried out in [16, 17] where distributed scalar quantizers were designed for different channel models. In what follows, we propose a design algorithm that results in sensor nodes operating on a sample by sample basis in a similar fashion as a channel optimized scalar quantizer (COSQ) [18].

2 Problem Formulation

We consider the problem of distributed joint source–channel coding illustrated in Figure 1. Two correlated random variables X_1 and X_2 are to be encoded by two encoders separated in space preventing cooperation between the encoders. To achieve low-complexity and low-delay encoding, the mappings f_1 and f_2 work in the following manner: f_1 and f_2 will first scalar quantize X_1 and X_2 to indexes i_1 and i_2 according to

$$q_k : \mathcal{X}_k \rightarrow \mathcal{I}_k \in \{0, 1, \dots, N - 1\} \quad \forall k \in \{1, 2\} \quad (1)$$

and these indexes are then transmitted over an additive white Gaussian noise (AWGN) channel. Two different transmission methods will be studied resulting in two different channel models. The first model is created by using BPSK on the AWGN channel in conjunction with hard decision decoding. This results in a binary symmetric channel (BSC) with some given bit error probability. Hence, in the first model each index is transmitted using the BSC $R \triangleq \log_2 N$ times. In the second model we will transmit each index by mapping the quantization index to a symbol in an N pulse amplitude modulated (N-PAM) signal. We will refer to this case as the ‘Gaussian channel’. We explain these two cases in greater detail below.

2.1 Binary Symmetric Channel

For the case of the BSC the quantization index i_k from (1) will be mapped to its binary representation as

$$\mathbf{f}_k : \mathcal{X}_k \xrightarrow{q_k} \mathcal{I}_k \rightarrow \{-1, 1\}^R \quad \forall k \in \{1, 2\}. \quad (2)$$

Hence, \mathbf{f}_k will use q_k to create the index i_k which is then represented binary. These bits are transmitted over a Gaussian channel using BPSK resulting in

$$\mathbf{r}_k = \mathbf{f}_k(x_k) + \mathbf{w}_k \quad \forall k \in \{1, 2\} \quad (3)$$

where \mathbf{w} is zero mean i.i.d. Gaussian noise with covariance matrix $\sigma_w^2 \mathbf{I}$. For each of these R received values a hard decision decoding rule is applied such that

$$j_k(m) = \text{sign}(r_k(m)) \quad m = 1, 2, \dots, R \quad (4)$$

where

$$\text{sign}(x) = \begin{cases} 1, & x \geq 0 \\ -1, & x < 0. \end{cases} \quad (5)$$

Given that -1 was transmitted, and letting $Q(\cdot)$ denote the Q-function, this will result in a bit error probability

$$\epsilon = \int_0^\infty \frac{1}{\sqrt{2\pi\sigma_w^2}} e^{-\frac{(r+1)^2}{2\sigma_w^2}} dr = Q\left(\frac{1}{\sigma_w}\right), \quad (6)$$

which is also, due to the symmetry, the total bit error probability.

Denoting the decimal representation of \mathbf{j}_k as j_k the decoding will be performed as

$$\hat{x}_k = g_k(j_1, j_2) \quad \forall k \in \{1, 2\}. \quad (7)$$

Hence, the decoding is based on both j_1 and j_2 .

Given this system, we define the mean squared error (MSE) as

$$D = \frac{1}{2}(D_1 + D_2) = \frac{1}{2} \left(E[(X_1 - \hat{X}_1)^2] + E[(X_2 - \hat{X}_2)^2] \right) \quad (8)$$

and our objective is to design the encoders and the decoder in order to minimize the MSE.¹

2.2 Gaussian Channel

For the Gaussian channel each of the indexes (i_1, i_2) are mapped to an N pulse amplitude modulated (N-PAM) signal such that

$$f_k(x_k) = \alpha(2q_k(x_k) - N + 1) \quad \forall k \in \{1, 2\}. \quad (9)$$

Here α is a constant such that the power constraints

$$E[f_k(X_k)^2] \leq P \quad \forall k \in \{1, 2\} \quad (10)$$

are satisfied. The two PAM signals are then transmitted over two orthogonal channels, created by using e.g. TDMA or FDMA, resulting in the received values

$$r_k = f_k(x_k) + w_k \quad \forall k \in \{1, 2\} \quad (11)$$

¹As pointed out by one of the reviewers one could also define a weighted MSE as $D(\rho) = \rho D_1 + (1 - \rho)D_2$ and adopt our derived equations accordingly. One interesting case would be $D(1)$ meaning that the second observation x_2 only serves as side information when estimating x_1 . However, we will only study the case $D(0.5)$, i.e. as in (8).

where the noise terms w_k are independent zero-mean Gaussian distributed with variance σ_w^2 . The decoder will have access to both r_1 and r_2 and forms its estimate of the original source data as

$$\hat{x}_k = g_k(r_1, r_2) \quad \forall k \in \{1, 2\}. \quad (12)$$

Here the objective is to design the encoders and the decoder in order to minimize the MSE from (8) under the power constraints given in (10).

3 Analysis

As in traditional Lloyd-Max training [19] we will optimize each part in the system in an iterative fashion keeping the other parts fixed. Note that the system contains three parts: two encoders and one decoder, although the decoder contains two decoding functions. We will in this section consider the design of these parts under the assumption that

$$X_k = Y + Z_k \quad \forall k \in \{1, 2\} \quad (13)$$

where Y , Z_1 and Z_2 are independent zero-mean Gaussian distributed random variables with variances σ_Y^2 , $\sigma_{Z_1}^2 = \sigma_{Z_2}^2 = \sigma_Z^2$. Hence, X_1 and X_2 are correlated which can be exploited in the encoding as well as the decoding.

For this jointly Gaussian distribution we get the conditional pdf

$$p(x_2|x_1) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\left(x_2 - \frac{\sigma_Y^2}{\sigma_Y^2 + \sigma_Z^2}x_1\right)^2}{2\sigma^2}\right) \quad (14)$$

where

$$\sigma^2 = \frac{\sigma_Z^4 + 2\sigma_Y^2\sigma_Z^2}{\sigma_Y^2 + \sigma_Z^2}. \quad (15)$$

Without loss of generality we will further assume that $E[X_1^2] = E[X_2^2] = 1$, hence $\sigma_Y^2 + \sigma_Z^2 = 1$.

3.1 Encoder for BSC

Only the design of \mathbf{f}_1 will be considered since \mathbf{f}_2 can be designed in the same fashion. Given that the encoder \mathbf{f}_1 observes x_1 and produces index i_1 it can derive the expected distortions for D_1 and D_2 as

$$D_1(x_1, i_1) = \sum_{j_1} \sum_{j_2} P(j_1|i_1)P(j_2|x_1) [x_1 - g_1(j_1, j_2)]^2 \quad (16)$$

$$D_2(x_1, i_1) = \int \sum_{j_1} \sum_{j_2} P(j_1|i_1)p(x_2|x_1)P(j_2|q_2(x_2)) [x_2 - g_2(j_1, j_2)]^2 dx_2, \quad (17)$$

where the integral is taken from $-\infty$ to ∞ and

$$P(j_2|x_1) = \sum_{i_2} P(j_2|i_2)P(i_2|x_1) \quad (18)$$

where

$$P(i_2|x_1) = \int_{x_2:q_2(x_2)=i_2} p(x_2|x_1)dx_2. \quad (19)$$

The other transition probabilities $P(\cdot|\cdot)$ are straightforward to derive, see e.g. [20]. In order to minimize the distortion (8) the quantizer $q_1(x_1)$ should be designed according to

$$q_1(x_1) = \arg \min_{i_1} (D_1(x_1, i_1) + D_2(x_1, i_1)). \quad (20)$$

In [18] the case of a single source was studied. In this case, the solution resulted in encoder regions which were intervals, and analytical expressions for finding the endpoints of these intervals were derived. However, (20) does in general not result in a similar solution and the encoder regions will in general *not be intervals*, but rather unions of separated intervals (this will be illustrated in Section 4).

3.2 Encoder for Gaussian Channel

For the Gaussian channel D_1 and D_2 can be expressed as

$$D_1(x_1, i_1) = \iint p(r_1|i_1)p(r_2|x_1) [x_1 - g_1(r_1, r_2)]^2 dr_2 dr_1 \quad (21)$$

$$D_2(x_1, i_1) = \iiint p(r_1|i_1)p(x_2|x_1)p(r_2|q_2(x_2)) [x_2 - g_2(r_1, r_2)]^2 dr_2 dx_2 dr_1, \quad (22)$$

where the integrals are taken from $-\infty$ to ∞ . In order to minimize the distortion (8) under the power constraint (10) the quantizer $q_1(x_1)$ should be designed according to

$$q_1(x_1) = \arg \min_{i_1} (D_1(x_1, i_1) + D_2(x_1, i_1) + \lambda(2i_1 - N + 1)^2). \quad (23)$$

Here, the first two terms aim at minimizing the distortion introduced by the quantizer whereas the third term will allow us to control the power consumption by choosing a value for the Lagrangian multiplier λ , see e.g. [21]. Unfortunately the integrals in (21)–(22) are difficult to evaluate since they contain $g_1(r_1, r_2)$ and $g_2(r_1, r_2)$ which vary with r_1 and r_2 . In order to get around this problem we use the technique of prequantizing r_1 and r_2 according to

$$h : (\mathcal{R}_1, \mathcal{R}_2) \rightarrow (\mathcal{J}_1, \mathcal{J}_2) \in \{1, 2, \dots, M\}^2 \quad (24)$$

which will produce the decoding functions

$$\hat{x}_k = g_k(h(r_1, r_2)) = g_k(j_1, j_2) \quad \forall k \in \{1, 2\}. \quad (25)$$

Furthermore, in this work we choose $M = N$ and let $h(r_1, r_2)$ simply map (r_1, r_2) to the closest possible output from the encoders defined by $(f_1(x_1), f_2(x_2))$. Hence

$$h(r_1, r_2) = \arg \min_{(j_1, j_2)} ((r_1 - \alpha(2j_1 - N + 1))^2 + (r_2 - \alpha(2j_2 - N + 1))^2). \quad (26)$$

The decoding functions will now be piecewise linear over r_1 and r_2 which greatly simplifies the derivation of (21)–(22) and we get the same equations as in (16)–(17) (although the transition probabilities are different). Using (23) together with (16)–(17) will therefore define the optimal quantizer $q_1(x_1)$ under the assumption that the decoder and the second encoder are fixed.

3.3 Decoder

Assuming fixed encoders it is a well known fact from estimation theory that the optimal, in minimum MSE sense, estimates of x_1 and x_2 are given as

$$\hat{x}_k = g_k(j_1, j_2) = E[x_k | j_1, j_2] \quad \forall k \in \{1, 2\}. \quad (27)$$

Hence, (27) is used to derive the decoders for both considered transmission methods.

3.4 Design algorithm

Based on the developed equations (20), (23) and (27) it will be possible to optimize the encoders and the decoder. A natural order to optimize these is : 1) the first encoder, 2) the decoder, 3) the second encoder, 4) the decoder. Each step in the iteration will guarantee the distortion to decrease and the training is repeated until the solution converges. Just as in the case of the Lloyd-Max algorithm this will result in a locally optimal system which is not necessarily the global optimum.

One problem with the suggested training above is that the obtained local optimum produced will depend greatly on the initialization of the decoder and encoders. In fact, in our simulations we experienced that very poor local optima were often found using the approach suggested above. This problem has also been encountered in [18, 22–24] where the method of noisy channel relaxation was introduced. The idea is essentially that it is easier to find a good local optimum for channels with high noise energy than for channels with low noise energy. Therefore a system is first designed for a very bad channel. Next, the channel quality is gradually improved and a new system is designed in each step. For each design, a full iterative training algorithm is executed using the reconstruction codebook from the previous design as initialization for the current design. We incorporate this idea by starting designing a system for a noise variance $\sigma_w'^2 \gg \sigma_w^2$. When this is completed $\sigma_w'^2$ is decreased with a stepsize σ_Δ^2 and a new system is designed. This is repeated L times. The algorithm is summarized below.

1. Initialize encoders and optimize the decoder by using (27).
2. Set values for L and σ_Δ^2 . Create $\sigma_w'^2 = \sigma_w^2 + L\sigma_\Delta^2$.
3. Design a system for the channel noise $\sigma_w'^2$ according to:
 - (a) Set the iteration index $k = 0$ and $D^{(0)} = \infty$.
 - (b) Set $k = k + 1$.
 - (c) Find the optimal quantizer q_1 by using (20) (or (23)).
 - (d) Find the optimal decoder by using (27).
 - (e) Find the optimal quantizer q_2 by using $q_2(x_2)$'s equivalence to (20) (or (23)).
 - (f) Find the optimal decoder by using (27).
 - (g) Evaluate the distortion $D^{(k)}$ for the system. If the relative improvement of $D^{(k)}$ compared to $D^{(k-1)}$ is less than some threshold $\delta > 0$ go to Step 4. Otherwise go to Step (b).
4. If $\sigma_w'^2 = \sigma_w^2$ stop the iteration. Otherwise create $\sigma_w'^2 = \sigma_w'^2 - \sigma_\Delta^2$ and go to Step 3 using the current encoders and decoder when initializing the next iteration.

We also experienced that when searching for a good local optima a small improvement was sometimes obtained by also performing a noise relaxation procedure for the correlation, i.e. varying σ_z^2 . However, the main improvement was obtained by the algorithm above.

3.5 Optimal Performance Theoretically Attainable

Recently the rate region for the quadratic two-terminal source coding problem has been completely characterized in [25]. Furthermore, in [26] it is shown that separating the source and channel code design, when the block lengths are approaching infinity, will be asymptotically optimal for the problem we are considering. Hence, by simply studying the channel capacity of the different orthogonal channels we get rate constraints, R_1 and R_2 , on the source code since these rates can be safely communicated to the decoder. Assuming that we have access to a capacity achieving channel code for the BSC we get

$$\beta_{\text{BSC}} = R_1 = R_2 = RC_{\text{BSC}} = R(1 + \epsilon \log_2 \epsilon + (1 - \epsilon) \log_2(1 - \epsilon)) \quad (28)$$

where C_{BSC} is the capacity of the BSC [27]. For the Gaussian channel we note that both encoders have the same power constraint (10) and that both channels have the same noise power. This gives

$$\beta_{\text{AWGN}} = R_1 = R_2 = C_{\text{AWGN}} = \frac{1}{2} \log_2 \left(1 + \frac{P}{\sigma_w^2} \right) \quad (29)$$

where C_{AWGN} is the capacity of the AWGN channel [27]. Using the appropriate β , from (28) or (29), and simplifying the expressions in [25] (remember the assumption $\sigma_Y^2 + \sigma_Z^2 = 1$) gives

$$D_1 D_2 \geq 2^{-4\beta} (1 - \sigma_Y^4) + \sigma_Y^4 2^{-8\beta}. \quad (30)$$

Since D_1 is inversely proportional to D_2 the total distortion in (8) will be minimized by setting $D = D_1 = D_2$. This gives the optimal performance theoretically attainable (OPTA) according to

$$D = \sqrt{2^{-4\beta} (1 - \sigma_Y^4) + \sigma_Y^4 2^{-8\beta}}. \quad (31)$$

That is, D in (31) is the lowest possible achievable distortion for this problem.

4 Simulations

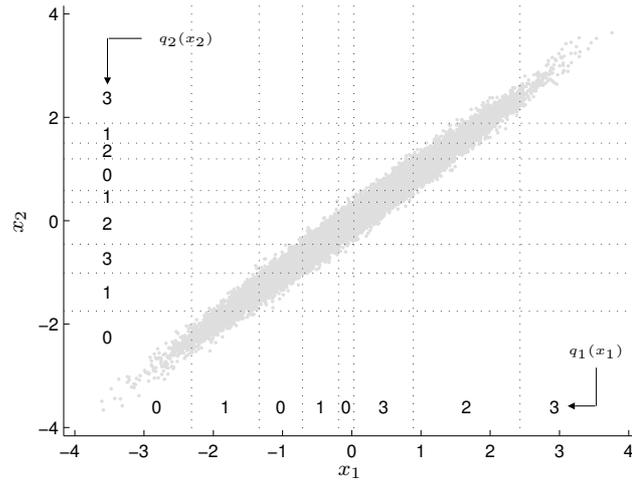
We will here visualize the structure of the encoders obtained when using the design algorithm presented in Section 3.4. The performance of a designed system is also compared to the OPTA derived in Section 3.5. In order to do so we measure the signal-to-distortion ratio (SDR) defined as

$$\text{SDR} = 10 \log_{10} \left(\frac{E[X_1^2] + E[X_2^2]}{E[(X_1 - \hat{X}_1)^2] + E[(X_2 - \hat{X}_2)^2]} \right) \quad (32)$$

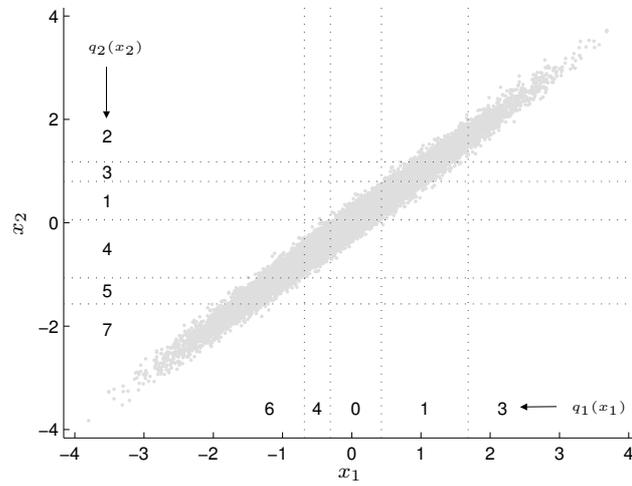
and we also define the correlation SNR as

$$\text{CSNR} = 10 \log_{10} \left(\frac{\sigma_Y^2}{\sigma_Z^2} \right). \quad (33)$$

Hence, $\text{CSNR} = -\infty$ dB means that X_1 and X_2 are uncorrelated and $\text{CSNR} = \infty$ dB means that they are fully correlated. We use the term SNR when referring to the channel SNR defined as $10 \log_{10}(P/\sigma_w^2)$. As initial encoders we used uniform quantizers and for the case of BSC the folded binary code [28] was used as initial codeword assignment.



(a)



(b)

Figure 2: Encoder structures for systems with CSNR = 20 dB and $R = 2$ bits/sample, $\epsilon = 0$ in (a) and $R = 3$ bits/sample, $\epsilon = 0.05$ in (b). The small dots in the background show a sample distribution of (X_1, X_2) and the dashed lines show the boundaries for the quantization regions.

4.1 Structure of the Codebook - BSC

In Figure 2 systems have been designed for $\text{CSNR} = 20$ dB and the resulting encoders are illustrated for different bit error probabilities. Starting with Figure 1.2(a), where $\epsilon = 0$ and $R = 2$ bits per sample and source, a number of source data samples are marked by the grayish distribution. These samples are spread out along the diagonal due to the correlation between x_1 and x_2 . In the plot the different quantization intervals for q_1 and q_2 are marked by the dashed lines. The representation of codewords produced by the quantizers in the different intervals are also marked. It is here interesting to note that many of the codewords are used for more than one quantization region. For example the codeword $i_2 = 1$ is used for 3 separated intervals such that $q_2(x_2) = 1$ when x_2 belongs (approximately) to the set $\{(-1.7, -1.0) \cup (0.4, 0.6) \cup (1.5, 1.9)\}$. With information from only one of the channels it is not possible to identify which of these different intervals x_2 belongs to. However, with help from i_1 (or rather j_1) this can be accomplished since $i_1 = 0$ or 1 is highly likely when $x_2 \in (-1.7, -1.0)$, $i_1 = 3$ is highly likely when $x_2 \in (0.4, 0.6)$, and so on. Hence, i_1 will indicate which of the separated intervals x_2 belongs to. In this way the distributed coding is used to decrease the quantization distortion. It is noteworthy that the sets of separated intervals are created by the design algorithm despite the fact that the initial encoders are regular quantizers where all quantization regions are single intervals.

When the bit error probability increases the encoders will be more restrictive in using all possible codewords since they will be more likely to be decoded incorrectly. In Figure 1.2(b) a system has been designed for $\epsilon = 0.05$ and $R = 3$ bits per sample and source. As can be seen only a subset of the codewords are now used by the encoders and these codewords have been placed with an appropriate index assignment.

4.2 Structure of the Codebook - Gaussian Channel

In order to illustrate the characteristics of the resulting system for the Gaussian channel a simple system with $N = 8$ has been designed and used for $\text{SNR} = 10$ dB and $\text{CSNR} = 20$ dB. The resulting quantizers are shown in Figure 1.3(a) and in Figure 1.3(b) it is illustrated how the quantization indexes are mapped to the channel space.

Starting with Figure 1.3(a) we once again see that the codewords will be reused as discussed in the previous section. See for instance the codeword $i_1 = 5$ which is used both when x_1 belongs (approximately) to the set $(-0.8, -0.3) \cup (1.7, 2.1)$. With help from i_2 (or rather r_2) the decoder will be able to distinguish between these two intervals since $i_2 = 2$ or 3 is highly likely if x_1 belongs to the first interval and otherwise $i_2 = 5$ or 6 will be highly likely.

Let us now consider what will happen when the source data is quantized by q_1 and q_2 and mapped to the signal space by f_1 and f_2 as described by (9). Both f_1 and f_2 uses 8-PAM, resulting in 64 possible combinations at the encoder output. However, many of these combinations are very unlikely to occur and for the simulation conducted the occurred outputs are marked by circles in Figure 1.3(b). Furthermore, when transmitting these output values the channels will add noise to the outputs creating a distribution of (r_1, r_2) which is indicated by the grayish distribution in Figure 1.3(b).

Finally, some extra source data values were created in Figure 1.3(a) where $x_1 = x_2 = x$, hence $\sigma_Z^2 = 0$, and we let x increase from $-\infty$ to ∞ . These values are marked by the line along the diagonal. The reason for adding these extra fully correlated values is that studying how this line is mapped to the channel signal space will give insight in the

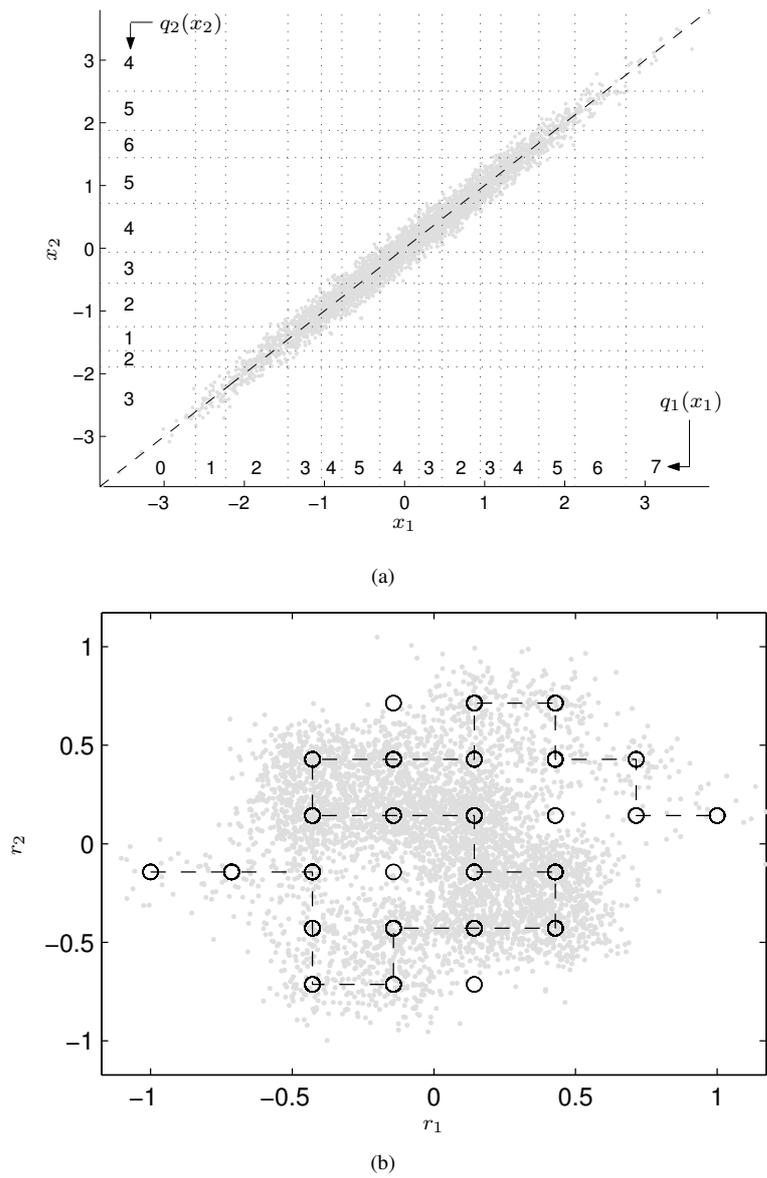
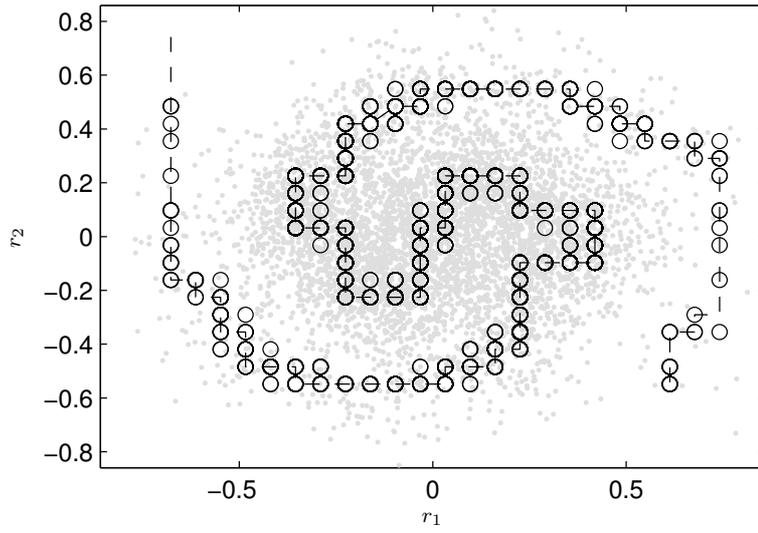
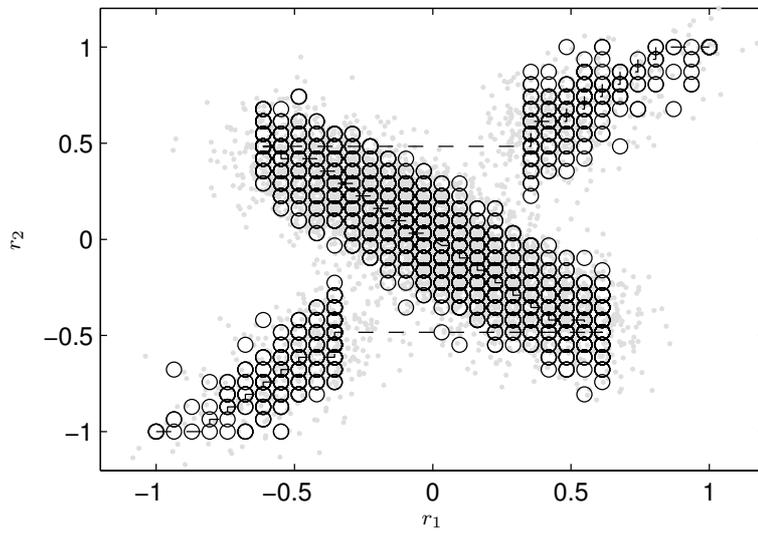


Figure 3: (a) Quantizers and (b) the corresponding mapping to the channel space for a system designed and used for $N = 8$, SNR = 10 dB and CSNR = 20 dB.



(a)



(b)

Figure 4: Other mappings to the channel space illustrated for (a) $N = 32$, SNR = 7.2 dB and CSNR = 30 dB and (b) $N = 32$, SNR = 10 dB and CSNR = 13 dB.

mapping procedure. By connecting the outputs created, when encoding this extra source data, we see how the line is mapped to the channel signal space (marked by a dashed line in Figure 1.3(b)). From this we note that, in general, samples far apart in the source signal space are also far apart in the channel signal space and vice versa. The power constraint will also focus the outputs in the area around the origin as much as possible in order to keep down the power consumption.

In Figures 1.4(a)–1.4(b) we present two other illustrations of the channel space, this time for $N = 32$. Figure 1.4(a) represents the case of high CSNR whereas Figure 1.4(b) represents low CSNR. From, for instance, Figure 1.4(a) we can imagine an underlying continuous curve $(\tilde{f}_1(x), \tilde{f}_2(x))$ which would be a good choice if we let $N \rightarrow \infty$. Furthermore, the curves created by $(f_1(x), f_2(x))$ appear to, especially for the high CSNR case, relate to what is often referred to as the bandwidth expansion problem mentioned already in one of Shannon’s first papers [29]. This is the resulting problem when $\text{CSNR} = \infty$ dB, i.e. $x_1 = x_2$, meaning that one is allowed to use a channel twice in order to transmit one source sample. It is well known that optimal encoding functions f_1 and f_2 will be nonlinear for this case, see e.g. [30, 31] and the references therein.

The connection between the bandwidth expansion problem and distributed source coding is an interesting insight and we draw the conclusion that if an analog system is to be used for distributed source coding linear operations for f_1 and f_2 are not necessarily appropriate. We have elaborated on this further in [32]. It is interesting to note that the curves $(f_1(x), f_2(x))$ are not necessarily continuous when $N \rightarrow \infty$ which also seems to be indicated by Figure 1.4(b).

Finally we comment on the fact that the number of used encoder outputs from f_1 and f_2 are not the same. For instance, in Figure 1.3(b) f_1 uses 8 encoder outputs whereas f_2 only uses 6. The curves $(f_1(x), f_2(x))$ created will have two properties, the first is that the distance between different folds of the curve will be high enough to combat the channel noise. The second property is that the created curves will place the most commonly occurring encoder outputs in the center where the power consumption is low. Less common encoder outputs will be placed further out and the curves will therefore grow outwards. However, due to the power constraint the power consumption will at some stage become too high and the algorithm will prevent the curve to grow any further. This will therefore cause the encoders to use different numbers of outputs.

4.3 Performance Evaluation

We begin with evaluating a system designed for the BSC with $R = 3$ bits per source sample, $\epsilon = 0.01$ which is equivalent to a channel with $\text{SNR} = 7.3$ dB (using the inverse of (6)) and $\text{CSNR} = 13$ dB. In Figure 5 we study the performance of the system (dashed line) when the SNR is varied. We have also included the OPTA (solid line) as well as a reference method (dotted line) in the plot. The reference method is traditional COSQ [18] where two independent COSQ’s are designed for $R = 3$ bits per sample and $\text{SNR} = 7.3$ dB, hence the correlation is not taken into consideration in the design. At the design SNR the gap to the OPTA curve is about 8 dB. Here it should be emphasized that achieving the OPTA requires infinite block-lengths, while our system works without delay on a sample by sample basis. Also, achieving OPTA will require that the system is optimized for each specific SNR whereas our simulated system is designed for one particular SNR but used for all simulated SNR’s. By comparing to the reference method we can see that the gain of

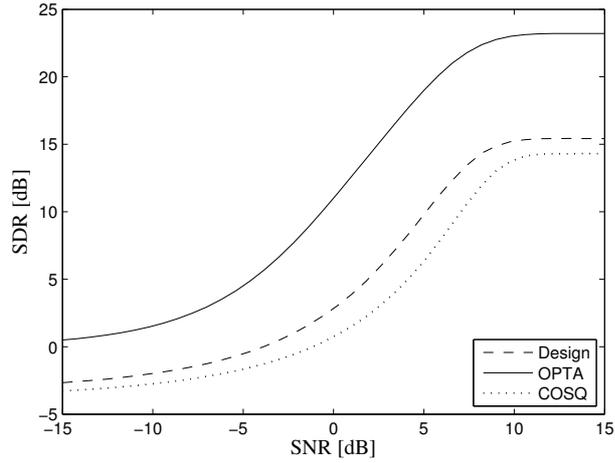


Figure 5: Evaluating the effect of varying the SNR when CSNR = 13 dB for a system designed for $R = 3$ bits per sample, SNR = 7.3 dB and CSNR = 13 dB.

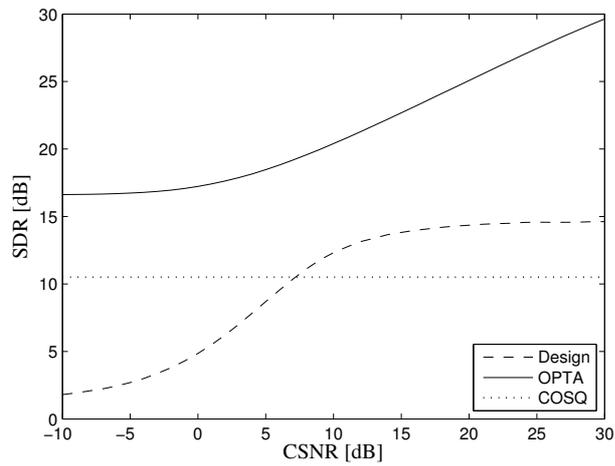


Figure 6: Evaluating the effect of varying the CSNR when SNR = 7.3 dB for a system designed for $R = 3$ bits per sample, SNR = 7.3 dB and CSNR = 13 dB.

utilizing the source correlation in the encoders and the decoder is about 3 dB at the design

SNR. When the SNR is increased above 10 dB the main contribution to the distortion comes from quantization which is limited by $R = 3$ bits per source sample, increasing the SNR above this point will therefore only have a small influence on the performance.

Next we keep the SNR fixed at 7.3 dB and look at the effect of a CSNR mismatch. That is, we evaluate the performance of the same system as above, which is designed for CSNR = 13 dB, when the true source correlation is varied. The result is shown in Figure 6 where we can see that the system is quite sensitive to a too low CSNR whereas a higher CSNR only gives a slight improvement in the performance. The designed system is however better than the reference method as long as the CSNR is above 7 dB. The reference method will not depend on the correlation and therefore has a constant performance.

In Figures 7–8 we present similar simulation results for the Gaussian channel. The simulated system is the same system as shown in Figure 1.4(b) designed for $N = 32$, SNR = 10 dB, CSNR = 13 dB and $\lambda = 0.01$. We have also here included the OPTA as well as a reference method, traditional COSQ, in the plot. In Figure 7 we let the CSNR equal 13 dB, hence what the system is designed for, but we vary the true SNR in order to study the effects of SNR mismatch. From the figure we see that in the area around SNR = 10 dB we are about 4 dB away from the OPTA (the additional figure is a magnification of the region around SNR = 10 dB). Increasing the SNR from this point will naturally increase the performance of OPTA and lowering the SNR will decrease the performance. It is therefore interesting to note that the designed system is able to follow the OPTA curve with essentially a constant 4 dB distance in the interval SNR $\in [5$ dB, 15 dB]. The system is hence robust to a too low SNR and at the same time it is able to exploit a high SNR in order to increase the performance. Comparing the system to the reference method we see that there is about a 1 dB performance gain when the SNR is above 5 dB.

In Figure 8 we instead let SNR=10 dB and study the effect of a mismatch in CSNR. Here it appears as the system is, just as in the BSC case, more sensitive to a too low CSNR. It can tolerate some mismatch but the performance will quite soon start decreasing rapidly. A too high CSNR only gives a slight improvement in performance and a saturation level is reached after only a few dB increase. Hence, for a high CSNR the proposed method has the better performance and vice versa.

5 Conclusions

A design algorithm for joint source–channel optimized distributed scalar quantizers is presented and evaluated. The resulting system works on a sample by sample basis yielding a very low encoding complexity, at an insignificant delay. Due to the source correlation, the resulting quantizers use the same codeword for several separated intervals in order to reduce the quantization distortion. Furthermore, the resulting quantization indexes are mapped to the channel signal space in such a way that source samples far from each other in the source signal space are well separated also in the channel signal space, and vice versa. This gives systems robust against channel SNR mismatch which was shown when comparing designed systems to the optimal performance theoretically attainable. The proposed main application of these quantizers is in low-complexity and energy-efficient wireless sensor nodes.

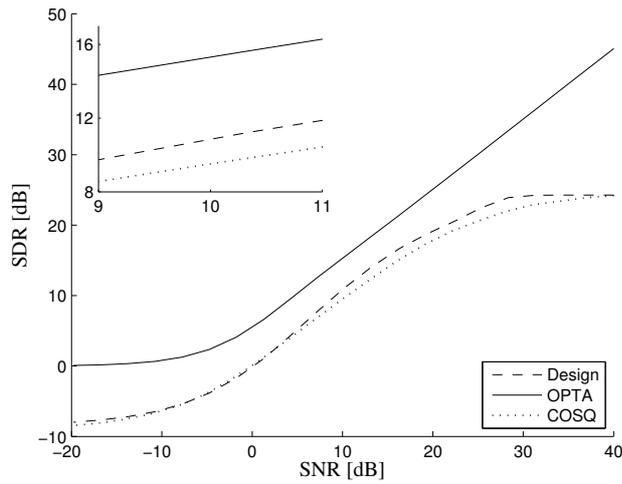


Figure 7: Evaluating the effect of varying the SNR when CSNR = 13 dB for a system designed for SNR = 10 dB and CSNR = 13 dB. The upper left plot shows a magnification of the area around SNR = 10 dB.

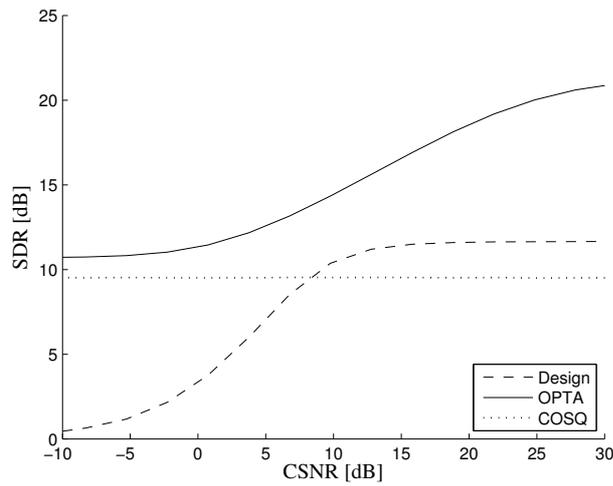


Figure 8: Evaluating the effect of varying the CSNR when SNR = 10 dB for a system designed for SNR = 10 dB and CSNR = 13 dB.

References

- [1] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 471–480, July 1973.
- [2] A. Wyner, "Recent results in the Shannon theory," *IEEE Transactions on Information Theory*, vol. 20, no. 1, pp. 2–10, January 1974.
- [3] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Transactions on Information Theory*, vol. 49, no. 3, pp. 626–643, March 2003.
- [4] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Communications Letters*, vol. 6, no. 10, pp. 440–442, October 2002.
- [5] J. Garcia-Frias and Y. Zhao, "Compression of correlated binary sources using turbo codes," *IEEE Communications Letters*, vol. 5, no. 10, pp. 417–419, October 2001.
- [6] Z. Xiong, A. Liveris, S. Cheng, and Z. Liu, "Nested quantization and Slepian-Wolf coding: a Wyner-Ziv coding paradigm for i.i.d. sources," in *IEEE Workshop on Statistical Signal Processing*, September 2003, pp. 399–402.
- [7] S. Pradhan and K. Ramchandran, "Generalized coset codes for distributed binning," *IEEE Transactions on Information Theory*, vol. 51, no. 10, pp. 3457–3474, October 2005.
- [8] T. J. Flynn and R. M. Gray, "Encoding of correlated observations," *IEEE Transactions on Information Theory*, vol. 33, no. 6, pp. 773–787, November 1987.
- [9] W.M. Lam and A. R. Reibman, "Design of quantizers for decentralized estimation systems," *IEEE Transactions on Communications*, vol. 41, no. 11, pp. 1602–1605, November 1993.
- [10] D. Rebollo-Monedero, R. Zhang, and B. Girod, "Design of optimal quantizers for distributed source coding," in *Proc. IEEE Data Compression Conf.*, March 2003, pp. 13–22.
- [11] E. Tuncel, "Predictive coding of correlated sources," in *IEEE Information Theory Workshop*, October 2004, pp. 111–116.
- [12] M. Fleming, Q. Zhao, and M. Effros, "Network vector quantization," *IEEE Transactions on Information Theory*, vol. 50, no. 8, pp. 1584–1604, August 2004.
- [13] A. Saxena and K. Rose, "Distributed predictive coding for spatio-temporally correlated sources," in *Proceedings IEEE Int. Symp. Information Theory*, June 2007, pp. 1506–1510.
- [14] B. Liu and B. Chen, "Channel-optimized quantizers for decentralized detection in sensor networks," *IEEE Transactions on Information Theory*, vol. 52, no. 7, pp. 3349–3358, July 2006.
- [15] A. Saxena, J. Nayak, and K. Rose, "On efficient quantizer design for robust distributed source coding," in *Proceedings IEEE Data Compression Conference*, March 2006, pp. 63–71.
- [16] J. Karlsson, N. Wernersson, and M. Skoglund, "Distributed scalar quantizers for noisy channels," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2007, pp. 633–636.

- [17] N. Wernersson, J. Karlsson, and M. Skoglund, "Distributed scalar quantizers for gaussian channels," in *Proceedings IEEE Int. Symp. Information Theory*, June 2007, pp. 1741–1745.
- [18] N. Farvardin and V. Vaishampayan, "Optimal quantizer design for noisy channels: An approach to combined source–channel coding," *IEEE Transactions on Information Theory*, vol. 33, no. 6, pp. 827–838, November 1987.
- [19] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Dordrecht, The Netherlands: Kluwer academic publishers, 1992.
- [20] J. G. Proakis, *Digital Communications*, 4th ed. McGraw-Hill, 2001.
- [21] V. Vaishampayan and N. Farvardin, "Joint design of block source codes and modulation signal sets," *IEEE Transactions on Information Theory*, vol. 38, no. 4, pp. 1230–1248, July 1992.
- [22] P. Knagenhjelm, "A recursive design method for robust vector quantization," in *Proc. Int. Conf. on Signal Processing Applications and Technology*, November 1992, pp. 948–954.
- [23] S. Gadkari and K. Rose, "Noisy channel relaxation for VQ design," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 1996, pp. 2048–2051.
- [24] A. Fuldseth and T. A. Ramstad, "Bandwidth compression for continuous amplitude channels based on vector approximation to a continuous subset of the source signal space," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Munich, Germany, April 1997, pp. 3093–3096.
- [25] A. B. Wagner, S. Tavildar, and P. Viswanath, "The rate region of the quadratic gaussian two-terminal source-coding problem," *arXiv:cs.IT/0510095*, 2005.
- [26] J. J. Xiao and Z. Q. Luo, "Multiterminal source-channel communication over an orthogonal multiple-access channel," *IEEE Transactions on Information Theory*, vol. 53, no. 9, pp. 3255–3264, September 2007.
- [27] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley-Interscience, 1991.
- [28] A. Mehes and K. Zeger, "Binary lattice vector quantization with linear block codes and affine index assignments," *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 79–94, January 1998.
- [29] C. E. Shannon, "Communication in the presence of noise," *Proc. IRE*, pp. 10–21, January 1949.
- [30] V. Vaishampayan and S. I. R. Costa, "Curves on a sphere, shift-map dynamics, and error control for continuous alphabet sources," *IEEE Transactions on Information Theory*, vol. 49, no. 7, pp. 1658–1672, July 2003.
- [31] N. Wernersson, M. Skoglund, and T. Ramstad, "Analog source-channel codes based on orthogonal polynomials," in *Asilomar Conference on Signals, Systems and Computers*, November 2007.
- [32] N. Wernersson and M. Skoglund, "Nonlinear coding and estimation for correlated data in wireless sensor networks," *IEEE Transactions on Communications*, 2008, submitted.

Paper C

Nonlinear Coding and Estimation for Correlated Data in Wireless Sensor Networks

Niklas Wernersson and Mikael Skoglund
Submitted to *IEEE Transactions of Communications*

Nonlinear Coding and Estimation for Correlated Data in Wireless Sensor Networks

Niklas Wernersson and Mikael Skoglund

Abstract

The problem of designing simple and energy-efficient nonlinear distributed source-channel codes is considered. By demonstrating similarities between this problem and the problem of bandwidth expansion, a structure for source-channel codes is presented and analyzed. Based on this analysis an understanding about desirable properties for such a system is gained and used to produce an explicit source-channel code which is then analyzed and simulated. One of the main advantages of the proposed scheme is that it is implementable for many sources, contrary to most existing nonlinear distributed source-channel coding systems.

1 Introduction

Wireless sensor networks are expected to play an important role in tomorrow's sensing systems. One important property in these networks is that there may be a high correlation between different sensor measurements due to high spatial density of sensor nodes. This motivates distributed source and channel coding of correlated sources, as analyzed in for instance [1–3], and optimal source-channel codes for this problem often require that nonlinear operations are carried out by the encoder on the source data (see e.g. [4, 5] for a case where this is not the case).

The research in finding practical codes for distributed source and channel coding can roughly be divided into two groups. The first contains work which considers *nonlinear* source-channel codes, and has produced numerous papers, for instance [6, 7] leading to a powerful, high delay, source coding algorithm termed DISCUS, which can be combined with traditional channel codes in order to produce high performing source-channel codes, see also [8–11]. On the other hand, [12, 13] use an alternative approach, with low delay, in order to design nonlinear source-channel codes. The drawback with all these approaches is that, although straightforward in theory, they are in general hard to implement for a large number of sources. In fact, there are very few papers considering a case where more than two correlated sources are involved.

This has motivated the research within the other group where analog *linear* source-channel codes are used, see e.g. [14–16] and references therein. The use of linear codes

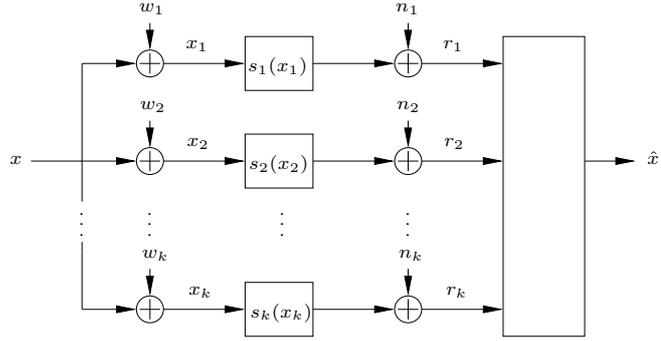


Figure 1: Structure of the system.

will in general lead to systems that are possible to optimize and implement for a large number of correlated sources. The drawback is that the linear approach in many situations is suboptimal. Hence, it should be possible to improve these systems by allowing nonlinear operations.

Our main contribution in this paper is that we consider an approach for analog nonlinear distributed source–channel coding, with better performance than linear codes, which can be implemented in the case of a large number of sources.

2 Problem Formulation

Consider the problem illustrated in Figure 1. An analog, i.e. continuous-valued, random source sample X with variance σ_x^2 is observed by k separate encoders (sensors) through the noisy observations

$$x_i = x + w_i, \quad 1 \leq i \leq k \tag{1}$$

where the W_i 's are independent identically distributed (i.i.d.) zero mean Gaussian with variance σ_w^2 . Each encoder encodes its own observation x_i by performing an analog mapping, that is $s_i : \mathbb{R} \rightarrow \mathbb{R}$, under the power constraint

$$E[s_i(X_i)^2] \leq P. \tag{2}$$

The encoded values,

$$\mathbf{s}(\mathbf{x}) \triangleq (s_1(x_1), s_2(x_2), \dots, s_k(x_k))^T \tag{3}$$

are transmitted over k orthogonal AWGN channels, created by using e.g. TDMA, FDMA or CDMA, and the decoder estimates x based on the received values

$$\mathbf{r} = \mathbf{s}(\mathbf{x}) + \mathbf{n} \tag{4}$$

where \mathbf{N} is i.i.d. memoryless Gaussian distributed with covariance matrix $\sigma_n^2 I$. Hence, the decoding is performed as

$$\hat{x} = \hat{x}(\mathbf{r}) \tag{5}$$

and the objective is to minimize the expected mean square error (MSE) $E[(X - \hat{X})^2]$. The main focus of this paper is on how to design $\mathbf{s}(\mathbf{x})$.

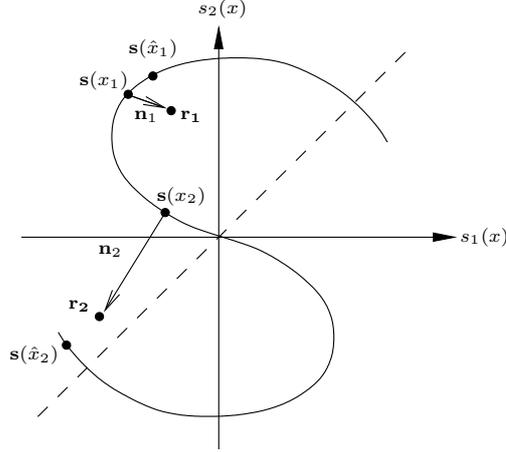


Figure 2: x_1 illustrates a 'small' decoding error and x_2 illustrates a 'large' decoding error.

3 Discussion and Proposed Scheme

We will in Subsections 3.A–B discuss two important special cases of the problem setting described in Figure 1. Understanding for these special cases leads to insight about how to design the encoding function $\mathbf{s}(\mathbf{x})$ for the general case. Based on this insight we present and analyze a structure for $\mathbf{s}(\mathbf{x})$ in Subsections 3.C–D. Finally, based on the derived results we propose an explicit scheme for $\mathbf{s}(\mathbf{x})$ in Subsection 3.E.

3.1 $\sigma_w^2 > 0$ and $\sigma_n^2 \rightarrow 0$

For the case when $\sigma_w^2 > 0$ and $\sigma_n^2 \rightarrow 0$ the AWGN channels are approaching ideal, i.e. noiseless, channels and r_i will approach $s_i(x_i)$. This means that given the *linear encoding* strategy,

$$s_i(x_i) = \sqrt{\frac{P}{\sigma_x^2 + \sigma_w^2}} x_i, \quad 1 \leq i \leq k, \quad (6)$$

the decoder will get access to the noisy observations $\{x_i\}_{i=1}^k$ (since the channel is close to perfect). Given these observations we could theoretically perform the best possible estimation based on the noisy observations as given by the Cramer–Rao lower bound, see e.g. [17]. It is clear that there will be no way to obtain a better performance, since that would require better sensor observations, and we can therefore conclude that the linear coding strategy described above is approaching the optimal strategy when $\sigma_n^2 \rightarrow 0$.

3.2 $\sigma_w^2 = 0$ and $\sigma_n^2 > 0$

In the case when $\sigma_w^2 = 0$ and $\sigma_n^2 > 0$ we will get $x_i = x \forall i$ which we will write as $\mathbf{s}(\mathbf{x}) = \mathbf{s}(x)$. This problem is equivalent to the problem often referred to as the *band-*

width expansion problem mentioned already in one of Shannon's first papers [18], see also [19–21] and the references therein. It is well known for this problem that when the source is i.i.d. zero-mean Gaussian and $k = 1$, linear encoding is optimal under the assumption that the decoder knows the source and noise variances. However, when $k > 1$ this is no longer true and nonlinear encoding can have superior performance compared to linear encoding strategies, see e.g. [22]. One of the reasons for this is that a linear encoding function $s(x)$ uses only a one dimensional subspace of the available channel space. More efficient mappings would use a higher number of the available channel space dimensions. An example of this is illustrated in Figure 2 where $k = 2$ is assumed. By using nonlinear encoding functions, illustrated by the solid 'S-shaped' curve $s(x)$, we are able to better fill the channel space than when using linear encoding functions, represented by the dashed curve. A longer curve essentially means a higher resolution when estimating x as long as we decode to the right fold of the curve, illustrated by sample x_1 in the figure. However, decreasing the SNR will at some point result in that different folds of the curve will lie too close to each other and the decoder will start making large decoding errors, illustrated by sample x_2 in the figure. Decreasing the SNR below this threshold will therefore significantly deteriorate the performance.

3.3 Objective

Based on the intuition from these two special cases we can conclude that for the problem considered in this paper, where both $\sigma_w^2 > 0$ and $\sigma_n^2 > 0$, good encoding functions $s(\mathbf{x})$ should take both these aspects into consideration. We illustrate the use of a nonlinear encoding function $s(x)$ in Figure 3. Again, $s(x)$ will be a point on the 'S-shaped' curve. However, we are not encoding x but \mathbf{x} which will create the point $s(\mathbf{x})$. Hence, the distribution of points $s(\mathbf{x})$ will be spread around the curve, as illustrated by the dotted region, and

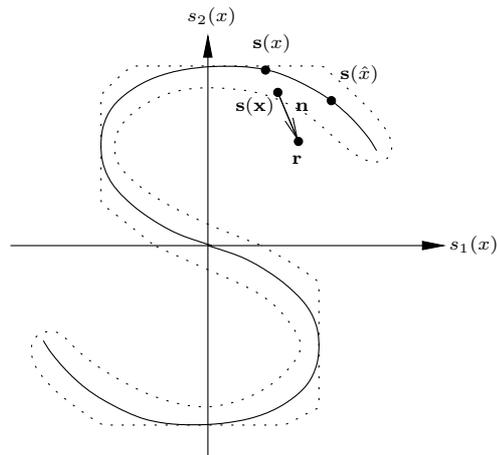


Figure 3: Using a nonlinear encoding function $s(\mathbf{x})$. The dotted region illustrates the distribution of the encoded values $s(\mathbf{x})$.

the amount of spread will depend on the observation noise variance σ_w^2 . When the encoded value is transmitted over the channels we also get a noise contribution \mathbf{n} from the channel and based on the received \mathbf{r} the decoder creates the estimate \hat{x} corresponding to some $\mathbf{s}(\hat{x})$. Our objective in this paper is to analyze and design nonlinear encoding functions $\mathbf{s}(\mathbf{x})$ of the type illustrated in Figure 3.

3.4 Analysis

In order to gain understanding for the use of nonlinear encoding functions $\mathbf{s}(\mathbf{x})$ we make a performance analysis under the assumptions that σ_w^2 and σ_n^2 are small. We also assume, in this subsection, that all encoding functions $s_i(x_i)$ are continuous and differentiable and that the curve $\mathbf{s}(x)$ is appropriately designed such that no large decoding errors occur under the assumed noise variances.

Let us start by studying a certain encoded observation x and the resulting estimate $\hat{x} = x + z$, with z representing the estimation error. Under the above assumptions, i.e. small noise variances, also z will be small and hence

$$\mathbf{s}(\hat{x}) = \mathbf{s}(x + z) \approx \mathbf{s}(x) + z\mathbf{s}'(x). \quad (7)$$

where

$$\mathbf{s}'(x) \triangleq \left(\frac{d}{dx} s_1(x), \frac{d}{dx} s_2(x), \dots, \frac{d}{dx} s_k(x) \right)^T. \quad (8)$$

Now consider the decoder. It is well known that in order to minimize the MSE the decoder should be implemented as

$$\hat{x}(\mathbf{r}) = E[X|\mathbf{r}]. \quad (9)$$

This function will however, in general, be difficult to implement. We will therefore consider the suboptimal maximum likelihood (ML) decoder. Since σ_w^2 is small, $\mathbf{s}(\mathbf{x})$ can be linearly approximated as

$$\mathbf{s}(\mathbf{x}) \approx \mathbf{s}(x) + \text{diag}(\mathbf{s}'(x))\mathbf{w} \quad (10)$$

which gives

$$\mathbf{r} = \mathbf{s}(\mathbf{x}) + \mathbf{n} \approx \mathbf{s}(x) + \text{diag}(\mathbf{s}'(x))\mathbf{w} + \mathbf{n}. \quad (11)$$

We approximate¹ the ML decoder as

$$\hat{x}(\mathbf{r}) = \arg \max_x p(\mathbf{r}|x) \approx \arg \min_x \|\mathbf{s}(x) - \mathbf{r}\|^2 \quad (12)$$

where $p(\cdot|\cdot)$ denotes the transition pdf from x to \mathbf{r} . Hence, the decoding function corresponds to decoding \mathbf{r} to the closest point on the curve $\mathbf{s}(\hat{x})$. However, for small values of $|x - \hat{x}|$ the curve $\mathbf{s}(\hat{x})$ is approximately linear and parallel to $\mathbf{s}'(x)$. This means that the decoder will remove the noise contributions orthogonal to $\mathbf{s}'(x)$ and we get

$$\mathbf{s}(\hat{x}) \approx \mathbf{s}(x) + \frac{(\text{diag}(\mathbf{s}'(x))\mathbf{w} + \mathbf{n}) \cdot \mathbf{s}'(x)}{\|\mathbf{s}'(x)\|} \frac{\mathbf{s}'(x)}{\|\mathbf{s}'(x)\|} \quad (13)$$

¹The true ML decoder will be difficult to analyze since the variances of the resulting noise terms in (11) will depend on x .

where the dot product describes the projection of the added noise onto the vector $\mathbf{s}'(x)$. Expanding the dot product we get

$$(\text{diag}(\mathbf{s}'(x))\mathbf{w} + \mathbf{n}) \cdot \mathbf{s}'(x) = (\text{diag}(\mathbf{s}'(x))\mathbf{w}) \cdot \mathbf{s}'(x) + \mathbf{n} \cdot \mathbf{s}'(x) \triangleq w + n \quad (14)$$

where

$$W \sim \mathcal{N}(0, \sigma_w^2 \sum_{i=1}^k s'_i(x)^4)$$

$$N \sim \mathcal{N}(0, \sigma_n^2 \sum_{i=1}^k s'_i(x)^2).$$

From (7) and (13) we identify

$$z \approx \frac{w + n}{\|\mathbf{s}'(x)\|^2} \quad (15)$$

and hence

$$E[(x - \hat{X})^2] = E[Z^2|x] \approx \frac{E[W^2|x]}{\|\mathbf{s}'(x)\|^4} + \frac{E[N^2|x]}{\|\mathbf{s}'(x)\|^4} = \sigma_w^2 \frac{\sum_{i=1}^k s'_i(x)^4}{\|\mathbf{s}'(x)\|^4} + \sigma_n^2 \frac{1}{\|\mathbf{s}'(x)\|^2}. \quad (16)$$

From this we conclude

$$E[(X - \hat{X})^2] \approx \int f(x) \left[\sigma_w^2 \frac{\sum_{i=1}^k s'_i(x)^4}{\|\mathbf{s}'(x)\|^4} + \sigma_n^2 \frac{1}{\|\mathbf{s}'(x)\|^2} \right] dx \quad (17)$$

where $f(x)$ is the pdf of x .

The second term in (17) is the MSE contribution from the channel noise. (This term was also derived in [23] for the bandwidth expansion case.) It tells us that we should aim for stretching the curve as much as possible, like stretching a rubber band, keeping in mind the constraint (2) at the same time as we also keep a high enough distance between different folds of the curve preventing large decoding errors. In order to stretch the curve it needs to turn in different directions which occurs when $s'_i(x) \neq s'_j(x)$ for some $i \neq j$. If we instead study the first part of (17), which is the MSE contribution from the observation noise, it is minimized when

$$\frac{\partial}{\partial s'_I(x)} \left(\frac{\sum_{i=1}^k s'_i(x)^4}{\|\mathbf{s}'(x)\|^4} \right) = 0 \quad (18)$$

for all I . By deriving this partial derivative it is straightforward to show that there is a global minimum at

$$s'_I(x) = \pm \sqrt{\frac{\sum_{i=1}^k s'_i(x)^4}{\sum_{i=1}^k s'_i(x)^2}}. \quad (19)$$

Neglecting the \pm -sign, which is of no importance here, this is only true when $s'_1(x) = s'_2(x) = \dots = s'_k(x)$. This indicates a linear system.

Hence, from (17) we understand the tradeoff between optimizing the system for being robust to the channel noise and for being robust to the observation noise: If we want to combat the channel noise we should create nonlinear curves $\mathbf{s}(\mathbf{x})$. On the other hand, if we want to combat the observation noise linear encoding functions will be more appropriate. (This is also clearly visible from the results in [13].)

3.5 Proposed Scheme

Based on the analysis in Section 3.4 we concluded that linear functions $s(\mathbf{x})$ are good with respect to the observation noise but inefficient with respect to the channel noise. Therefore, in order to produce a system which is able to handle both observation and channel noise, we propose a piecewise linear encoding function $s(\mathbf{x})$ as follows

$$s_i(x_i) = \begin{cases} \alpha x_i & 1 \leq i \leq k_0 \\ \alpha(x_i - \Delta \lfloor \frac{x_i}{\Delta} \rfloor) & k_0 < i \leq k \end{cases} \quad (20)$$

where $\lfloor \cdot \rfloor$ denotes rounding to the nearest integer and α will control the power usage. Hence, we allow noncontinuous and nondifferentiable functions which was not the case in the analysis. The reason is that this results in a system where $s'_1(x) = s'_2(x) = \dots = s'_k(x)$, except at the discontinuities, which is desirable with respect to the observation noise. At the same time we get a nonlinear system, better able to use the available channel space, which is desirable with respect to the channel noise. The drawback is that the approximation in (11) is violated making the decoder (12) inefficient. We therefore modify the decoding function as follows:

1. Create the ML estimate of x based on the linear encoding functions:

$$\hat{x}_{k_0} = \frac{1}{k_0} \sum_{i=1}^{k_0} \frac{r_i}{\alpha}. \quad (21)$$

2. Assume that $|\hat{x}_{k_0} - x_i - n_i/\alpha| \leq \Delta/2$ for $k_0 < i \leq k$ and create the ML estimates

$$\hat{x}_i(r_i) = \arg \min_{x_i} ((s_i(x_i) - r_i)^2 | x_i \in \{\hat{x}_{k_0} - x_i \leq \Delta/2\}). \quad (22)$$

This function tries to predict the removed part $\Delta \lfloor \frac{x_i}{\Delta} \rfloor$ from (20) based on the derived \hat{x}_{k_0} . The operation is illustrated in in Figure 4 for a few examples.

3. Based on this, create the final estimate of x as

$$\hat{x} = \frac{1}{k} \left(\sum_{i=1}^{k_0} \frac{r_i}{\alpha} + \sum_{i=k_0+1}^k \hat{x}_i(r_i) \right). \quad (23)$$

Let us now analyze the power consumption. Note that the power used by the nonlinear encoding functions will be less than the power used by the linear encoding functions. We define the normalized average power consumption as

$$P(\Delta, k_0) = \frac{1}{k\alpha^2} (k_0 E[s_I(X_I)^2] + (k - k_0) E[s_J(X_J)^2]) \quad (24)$$

where we assume $I \leq k_0 < J \leq k$. (The reason for dividing with α^2 is that we want $P(\Delta, k_0)$ to represent the change in power consumption due to Δ and k_0 and not due to the scaling factor α .) By performing timesharing the sensors could use the linear encoding function for a fraction k_0/k of the available time slots and then use the nonlinear encoding functions the rest of the time. Hence, $P(\Delta, k_0)$ can be seen as the average power used by each sensor when $\alpha = 1$.

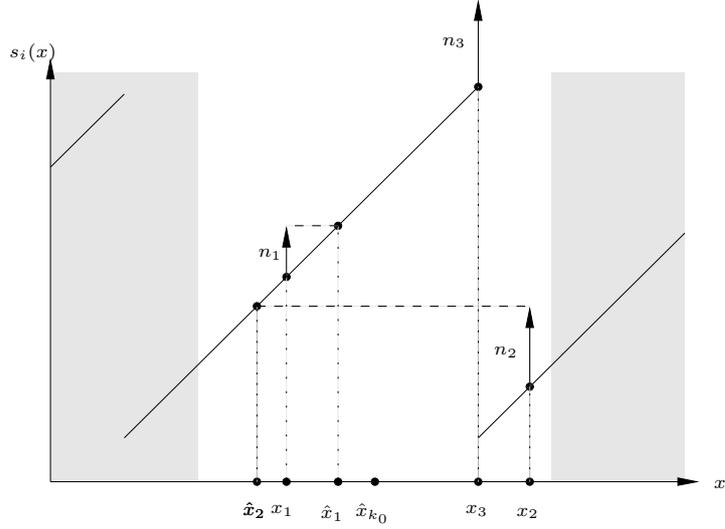


Figure 4: Illustration of the decoding function for the nonlinear encoding function $s_i(x)$. The decoder assumes that $|\hat{x}_{k_0} - x_i - n_i/\alpha| \leq \Delta/2$ and therefore limits the estimate \hat{x}_i to lie within this interval, illustrated with the white area in the figure. Three examples are shown, for x_1 the assumption made is correct and the estimation of the nonlinear part $\Delta \lfloor \frac{x_i}{\Delta} \rfloor$ is correct. For x_2 the assumption made is incorrect and the estimated nonlinear part will be wrong. Finally, for x_3 the assumption is actually wrong but the estimated nonlinear part will still be decoded correctly resulting in $\hat{x}_3 = x_3$.

4 Performance Analysis

It is clear that the performance of the system proposed in (20), given a certain P , will depend on Δ and k_0 . In order to optimize these parameters we will here derive the MSE as a function of Δ and k_0 under the assumption that x is i.i.d. zero-mean Gaussian. Hence, evaluating this function for different choices of Δ and k_0 will provide a possibility to design the system for a certain set of noise variances.

For a given Δ and k_0 we should choose

$$\alpha = \sqrt{\frac{P}{P(\Delta, k_0)}} \quad (25)$$

in order to get the power consumption P . In order to simplify the following equations we will instead study the equivalent system where $\alpha = 1$ and the channel noise variance equals $P(\Delta, k_0)/P\sigma_n^2$.

Given a certain \mathbf{r} and \mathbf{x} the decoder in (23) will produce

$$\hat{x}(\mathbf{r}, \mathbf{x}) \approx \underbrace{\frac{1}{k} \sum_{i=1}^k (x_i + n_i)}_{\triangleq \hat{x}_a} + \underbrace{\sum_{i=k_0+1}^k \frac{\Delta}{k} \left\lfloor \frac{\hat{x}_{k_0} - x_i - n_i}{\Delta} \right\rfloor}_{\triangleq \hat{x}_b} \quad (26)$$

where \hat{x}_b is a penalty term arising if $|\hat{x}_{k_0} - x_i - n_i| > \Delta/2$ meaning that the assumption made in (22) is incorrect. This situation is illustrated with sample x_2 in Figure 4. The reason for using an approximation sign in (26) is the kind of situation illustrated by sample x_3 in Figure 4. For this example $|\hat{x}_{k_0} - x_i - n_i| > \Delta/2$ is true but the estimated nonlinear part is still correct. Hence, in some specific cases the decoder (23) will perform better than what is described by (26).

For the expected MSE we get

$$\begin{aligned} \text{MSE}(\Delta, k_0) &= E[(X - \hat{X})^2] = E[(X - \hat{X}_a - \hat{X}_b)^2] \\ &= E[(X - \hat{X}_a)^2] + E[(\hat{X}_b)^2] + 2E[(X - \hat{X}_a)\hat{X}_b] \end{aligned} \quad (27)$$

where we directly see

$$E[(X - \hat{X}_a)^2] = \frac{\sigma_w^2 + P(\Delta, k_0)/P\sigma_n^2}{k}. \quad (28)$$

Here we see the role played by Δ and k_0 : decreasing Δ and/or k_0 will decrease $P(\Delta, k_0)$ and hence lower the effective channel noise variance. At the same time we can expect the second term in (27) to increase since the assumption made in (22) will get more likely to be incorrect. For the second term we get

$$\begin{aligned} E[\hat{X}_b^2] &= (k - k_0) \frac{\Delta^2}{k^2} \left(E \left[\left[\left\lfloor \frac{\hat{X}_{k_0} - X_i - N_i}{\Delta} \right\rfloor \right]^2 \right] \right. \\ &\quad \left. + (k - k_0 - 1) E \left[\left[\left\lfloor \frac{\hat{X}_{k_0} - X_i - N_i}{\Delta} \right\rfloor \left\lfloor \frac{\hat{X}_{k_0} - X_j - N_j}{\Delta} \right\rfloor \right] \right] \right) \\ &= (k - k_0) \frac{\Delta^2}{k^2} \left(E \left[\left[\frac{\xi_i}{\Delta} \right]^2 \right] + (k - k_0 - 1) E \left[\left[\frac{\xi_i}{\Delta} \right] \left[\frac{\xi_j}{\Delta} \right] \right] \right) \end{aligned} \quad (29)$$

where $i \neq j$, $\xi_i \triangleq \hat{X}_{k_0} - X_i - N_i$ and

$$\hat{X}_{k_0} - X_i - N_i \sim \mathcal{N}\left(0, \left(\frac{1}{k_0} + 1\right)(\sigma_w^2 + P(\Delta, k_0)/P\sigma_n^2)\right). \quad (30)$$

It is straightforward to calculate the first expected value in (29) by using the Q -function. For the second expected value we need to consider that the resulting Gaussian terms ξ_i and ξ_j are correlated since they both contain \hat{x}_{k_0} . The expression can be evaluated numerically or otherwise well approximated by approximating the jointly Gaussian distribution $f(\xi_i, \xi_j)$ as a Gaussian Mixture

$$f(\xi_i, \xi_j) \approx \frac{1}{4\pi\sigma^2} e^{-\frac{(\xi_1 - \mu)^2}{2\sigma^2}} e^{-\frac{(\xi_2 - \mu)^2}{2\sigma^2}} + \frac{1}{4\pi\sigma^2} e^{-\frac{(\xi_1 + \mu)^2}{2\sigma^2}} e^{-\frac{(\xi_2 + \mu)^2}{2\sigma^2}} \quad (31)$$

where $\sigma^2 = (\frac{1}{k_0} + 1)(\sigma_w^2 + P(\Delta, k_0)/P\sigma_n^2)$ and $\mu = \sqrt{\sigma^2/k_0}$. With this approximation also the second expected value in (29) is straightforward to derive using the Q-function.

Finally, in the appendix we show that

$$E[(X - \hat{X}_a)\hat{X}_b] = 0. \quad (32)$$

5 Simulations

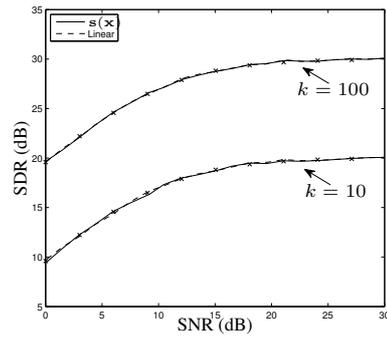
In Figures 5–6 we present results from three simulations. The purpose of the two first simulations is to investigate the behavior and performance of the proposed code for the two special cases discussed in Section 3. Finally, the third simulation deals with the more general situation which is the main focus of this paper. We will use the linear system (6) as reference system.

In all the simulations we used a zero-mean i.i.d. Gaussian distribution for X with variance σ_x^2 and without loss of generality we choose $\sigma_x^2 + \sigma_w^2 = 1$. We measure the performance in SDR $\triangleq (\sigma_x^2 + \sigma_w^2)/E[(X - \hat{X})^2]$ versus SNR $\triangleq P/\sigma_n^2$ and the correlation is measured as $\rho \triangleq \sigma_x^2/\sigma_w^2$. In order to design the systems for different noise levels we have used (27)–(32) to optimize the choices of Δ and k_0 . Since the derivation of (27) is fast, when using the suggested approximations, the optimization is done by simply deriving $\text{MSE}(\Delta, k_0)$ for a large number of combinations (Δ, k_0) . We then choose the best one out of these. The different choices are shown in Table 1.

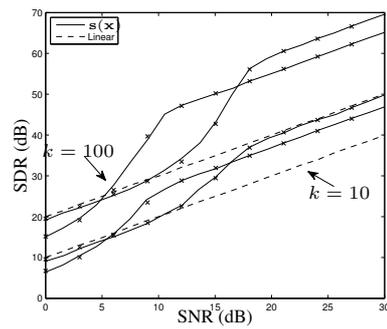
The first simulation is shown in Figure 1.5(a). This system has been optimized for an environment corresponding to the case described in Section 3.1, more precisely, $\rho = 9.5$ dB and SNR = 70 dB. Our discussion concluded that a linear system would be efficient for this case, since the channel is close to perfect, and the search over $\text{MSE}(\Delta, k_0)$ did in fact produce a linear system, i.e. $k_0 = k$. In the simulation we let $\rho = 9.5$ dB, hence what the system was designed for, but we vary the SNR in order to study the effects of SNR mismatch. We present results for two cases: $k = 10$ and $k = 100$. The performance of the linear system is shown by the dashed lines and the performance of the proposed scheme $s(\mathbf{x})$ is shown by the solid lines. Since the designed system here is linear the dashed and solid lines will naturally coincide. Finally, $\text{MSE}(\Delta, k_0)$ is evaluated for a few points and illustrated by the crosses. As can be seen, the approximations made in (27)–(32) produce a good approximation of the true MSE.

| k | SNR (dB) | ρ (dB) | Δ | k_0 |
|-----|----------|-------------|----------|-------|
| 10 | 70 | 9.5 | ∞ | 10 |
| 100 | 70 | 9.5 | ∞ | 100 |
| 10 | 10 | 70 | 1.18 | 1 |
| 100 | 10 | 70 | 0.50 | 1 |
| 10 | 20 | 70 | 0.36 | 1 |
| 100 | 20 | 70 | 0.12 | 1 |
| 10 | 10 | 20 | 1.56 | 2 |
| 100 | 10 | 20 | 1.08 | 4 |

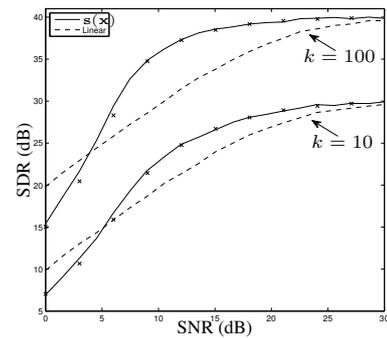
Table 1: The different choices of (Δ, k_0) .



(a)

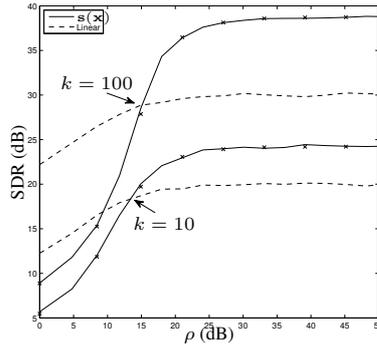


(b)



(c)

Figure 5: Systems for $k = 10$ and $k = 100$ designed for (a) $\rho = 9.5\text{dB}$ and $\text{SNR} = 70\text{dB}$ (b) $\rho = 70\text{dB}$ and $\text{SNR} = 10\text{dB}$ as well as $\text{SNR} = 20\text{dB}$ (c) $\rho = 20\text{dB}$ and $\text{SNR} = 10\text{dB}$. In all simulations the 'true' ρ has been used but the SNR is varied.



(a)

Figure 6: Systems for $k = 10$ and $k = 100$ designed for $\rho = 20$ dB and SNR = 10dB. In both simulations the 'true' SNR has been used but the ρ is varied.

In Figure 1.5(b) the systems have been optimized for an environment corresponding to the case described in Section 3.2, i.e. close to bandwidth expansion which occurs when ρ has a high value. We have here included two designs for $\rho = 70$ dB, namely SNR = 10 dB and SNR = 20 dB. Hence, one of the systems is optimized for a better channel meaning that different folds of $\mathbf{s}(x)$ should be packed closer to each other. This will give a better performance for high SNR's but the code will also break down faster when the SNR is decreased. This is clearly visible in the figure where we again simulate for the designed ρ but vary the SNR. The linear systems represented by the dashed curves are the better systems only for low SNR's.

Finally, we consider the, for this paper, most interesting case where we optimize a system for both observation and channel noise. In Figure 1.5(c) we have designed systems for $\rho = 20$ dB and SNR = 10 dB. Also here we simulate for the designed ρ but we vary the SNR. It is clear that there will be a large gain over the linear system except at low SNR's, where the source-channel code breaks down, and at high SNR's, where there is no gain in using nonlinear codes. In Figure 6 we instead evaluate the effects of correlation mismatch for the same system. That is, we evaluate the performance of the same system as above but this time we simulate for the designed SNR and let ρ vary. As can be seen the system is quite sensitive to a too low ρ whereas a higher ρ only gives a slight improvement in performance.

6 Conclusions

We have explained the similarities between the problem of distributed source-channel coding and the problem of bandwidth expansion. Based on this we have presented and analyzed a suitable structure for distributed source-channel codes. This analysis gives us insight into desirable properties for such a system. Based on this understanding an explicit code is presented, analyzed and simulated. The code is implementable for many sources, contrary to

most existing schemes, and it is concluded that a large gain is obtained from the nonlinear code compared to the linear code.

Appendix

We will here show that the third term in (27) equals 0. When doing so we will use the variable

$$U_i = W_i + N_i$$

with pdf $f(u_i)$ and variance $\sigma_u^2 = \sigma_w^2 + P(\Delta, k_0)/P\sigma_n^2$. Since both W_i and N_i are i.i.d. Gaussian also U_i will naturally be i.i.d. Gaussian. We will also use the variables I and J such that $I \leq k_0 < J$. We get

$$\begin{aligned} E[(X - \hat{X}_a)\hat{X}_b] &= E \left[\left(X - \frac{1}{k} \sum_{i=1}^k (X_i + N_i) \right) \left(\sum_{j=k_0+1}^k \frac{\Delta}{k} \left[\frac{\hat{X}_{k_0} - X_j - N_j}{\Delta} \right] \right) \right] \\ &= \frac{\Delta}{k^2} E \left[\left(\sum_{i=1}^k U_i \right) \left(\sum_{j=k_0+1}^k \left[\frac{\frac{1}{k_0} \sum_{i=1}^{k_0} U_i - U_j}{\Delta} \right] \right) \right] \\ &= \frac{\Delta(k - k_0)}{k^2} E \left[\left(\sum_{i=1}^k U_i \right) \left[\frac{\frac{1}{k_0} \sum_{i=1}^{k_0} U_i - U_J}{\Delta} \right] \right] \\ &= \frac{\Delta(k - k_0)}{k^2} \left(k_0 E \left[U_I \left[\frac{\frac{1}{k_0} \sum_{i=1}^{k_0} U_i - U_J}{\Delta} \right] \right] + \right. \\ &\quad \left. E \left[U_J \left[\frac{\frac{1}{k_0} \sum_{i=1}^{k_0} U_i - U_J}{\Delta} \right] \right] \right) \\ &= \frac{\Delta(k - k_0)}{k^2} \left(k_0 E \left[U_I \left[\frac{U_I + \xi_1}{\Delta} \right] \right] - E \left[U_J \left[\frac{U_J + \xi_2}{\Delta} \right] \right] \right) \end{aligned} \quad (33)$$

where

$$\begin{aligned} \xi_1 &\sim \mathcal{N} \left(0, \left(\frac{k_0 - 1}{k_0^2} + 1 \right) \sigma_u^2 \right) \\ \xi_2 &\sim \mathcal{N} \left(0, \frac{1}{k_0} \sigma_u^2 \right). \end{aligned}$$

Studying the first resulting expected value in (33), with the expectation taken over U_I , we get

$$\begin{aligned} E \left[U_I \left[\frac{U_I + \xi_1}{\Delta} \right] \right] &= \sum_{i=-\infty}^{\infty} i \int_{k_0(i\Delta - \frac{\Delta}{2} - \xi_1)}^{k_0(i\Delta + \frac{\Delta}{2} - \xi_1)} u_I f(u_I) du_I \\ &= \sqrt{\frac{\sigma_u^2}{2\pi}} \sum_{k=-\infty}^{k=\infty} e^{-\frac{k_0^2(i\Delta - \frac{\Delta}{2} - \xi_1)^2}{2\sigma_u^2}}. \end{aligned} \quad (34)$$

The second expected value in (33) can be derived in the same manner and produces a similar result. This gives

$$E[(X - \hat{X}_a)\hat{X}_b] = \frac{\Delta(k - k_0)}{k^2} \sqrt{\frac{\sigma_u^2}{2\pi}} \sum_{i=-\infty}^{\infty} E \left[k_0 e^{-\frac{k_0^2(i\Delta - \frac{\Delta}{2} - \xi_1)^2}{2\sigma_u^2}} - e^{-\frac{(i\Delta - \frac{\Delta}{2} - \xi_2)^2}{2\sigma_u^2}} \right] \quad (35)$$

where the expectation is on ξ_1 and ξ_2 . Straightforward derivations, for a general case covering both the expected values in (35), gives

$$E \left[e^{-\gamma(\beta - \xi)^2} \right] = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma_\xi^2}} e^{(-\gamma(\beta - \xi)^2)} e^{-\frac{\xi^2}{2\sigma_\xi^2}} d\xi = \frac{e^{-\frac{\gamma\beta^2}{2\gamma\sigma_\xi^2 + 1}}}{\sqrt{2\gamma\sigma_\xi^2 + 1}} \quad (36)$$

and by substituting γ , β and σ_ξ^2 with the corresponding coefficients from (35), followed by a few manipulations, it turns out that

$$E \left[k_0 e^{-\frac{k_0^2(i\Delta - \frac{\Delta}{2} - \xi_1)^2}{2\sigma_u^2}} \right] = E \left[e^{-\frac{(i\Delta - \frac{\Delta}{2} - \xi_2)^2}{2\sigma_u^2}} \right]. \quad (37)$$

Therefore, we can conclude, from (35), that

$$E[(X - \hat{X}_a)\hat{X}_b] = 0. \quad (38)$$

References

- [1] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 471–480, July 1973.
- [2] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Transactions on Information Theory*, vol. IT-22, no. 1, pp. 1–10, January 1976.
- [3] J. J. Xiao and Z. Q. Luo, "Multiterminal source-channel communication over an orthogonal multiple-access channel," *IEEE Transactions on Information Theory*, vol. 53, no. 9, pp. 3255–3264, September 2007.
- [4] M. Gastpar and M. Vetterli, "Source-channel communication in sensor networks," in *Proc. 2nd Int. Workshop Information Processing in Sensor Networks (IPSN03) (Lecture Notes in Computer Science)*, L. J. Guibas and F. Zhao, Eds. Berlin, Germany: Springer-Verlag, 2003, pp. 162–177.
- [5] M. Gastpar, "Uncoded transmission is exactly optimal for a simple gaussian "sensor" network," in *Proc. 2nd Annu. Workshop on Information Theory and its Applications*, January 2007, pp. 177–182.
- [6] A. Wyner, "Recent results in the shannon theory," *IEEE Transactions on Information Theory*, vol. 20, no. 1, pp. 2–10, January 1974.

- [7] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Transactions on Information Theory*, vol. 49, no. 3, pp. 626–643, March 2003.
- [8] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Communications Letters*, vol. 6, no. 10, pp. 440–442, October 2002.
- [9] J. Garcia-Frias and Y. Zhao, "Compression of correlated binary sources using turbo codes," *IEEE Communications Letters*, vol. 5, no. 10, pp. 417–419, October 2001.
- [10] Z. Xiong, A. Liveris, S. Cheng, and Z. Liu, "Nested quantization and Slepian-Wolf coding: a Wyner-Ziv coding paradigm for i.i.d. sources," in *IEEE Workshop on Statistical Signal Processing*, September 2003, pp. 399–402.
- [11] S. Pradhan and K. Ramchandran, "Generalized coset codes for distributed binning," *IEEE Transactions on Information Theory*, vol. 51, no. 10, pp. 3457–3474, October 2005.
- [12] A. Saxena and K. Rose, "Distributed predictive coding for spatio-temporally correlated sources," in *Proceedings IEEE Int. Symp. Information Theory*, June 2007, pp. 1506–1510.
- [13] N. Wernersson, J. Karlsson, and M. Skoglund, "Distributed quantization over noisy channels," *IEEE Transactions on Communications*, 2008, to appear.
- [14] Z. Luo, G. Giannakis, and S. Zhang, "Optimal linear decentralized estimation in a bandwidth constrained sensor network," in *Proceedings IEEE Int. Symp. Information Theory*, September 2005, pp. 1441–1445.
- [15] I. Bahceci and A. K. Khandani, "Energy-efficient estimation of correlated data in wireless sensor networks," in *Annual Conference on Information Sciences and Systems*, March 2006, pp. 973–978.
- [16] S. Cui, J. Xiao, A. Goldsmith, Z. Luo, and V. Poor, "Estimation diversity and energy efficiency in distributed sensing," *IEEE Transactions on Signal Processing*, vol. 55, no. 9, pp. 4683–4695, September 2007.
- [17] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.
- [18] C. E. Shannon, "Communication in the presence of noise," *Proc. IRE*, pp. 10–21, January 1949.
- [19] B. Chen and G. W. Wornell, "Analog error-correcting codes based on chaotic dynamical systems," *IEEE Transactions on Communications*, vol. 46, no. 7, pp. 881–890, July 1998.
- [20] V. Vaishampayan and S. I. R. Costa, "Curves on a sphere, shift-map dynamics, and error control for continuous alphabet sources," *IEEE Transactions on Information Theory*, vol. 49, no. 7, pp. 1658–1672, July 2003.
- [21] N. Wernersson, M. Skoglund, and T. Ramstad, "Analog source-channel codes based on orthogonal polynomials," in *Asilomar Conference on Signals, Systems and Computers*, November 2007.
- [22] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering*. Wiley, 1965.
- [23] D. J. Sakrison, *Transmission of Waveforms and Digital Information*. Wiley, 1968.

Paper D

Sorting-Based Multiple Description Quantization

Niklas Wernersson and Mikael Skoglund

Published in *IEEE Transactions of Communications*, 2006

©2006 IEEE
The layout has been revised

Sorting-Based Multiple Description Quantization

Niklas Wernersson and Mikael Skoglund

Abstract

We introduce a new method for multiple description quantization (MDQ), based on sorting a frame of samples and transmitting, as side-information/redundancy, an index that describes the resulting permutation. The sorting-based approach has a similar performance to multiple description scalar quantization and a flexible structure, providing straightforward implementation of multi-dimensional MDQ.

1 Introduction

Packet networks have gained in importance in recent years, for instance by the wide-spread use and importance of the Internet. Unfortunately, packet losses in these systems can in general not be neglected and this has to be taken into account in robust source-channel coding for multimedia communications. One way to deal with packet losses is to use multiple description coding/quantization where the objective is to code one source of data into multiple bitstreams. The coding is done in such a way that multiple levels of quality is achieved. This means that even if one or a few of the bitstreams are lost, the received bits should make it possible to get an approximated version of the original data. MDQ will therefore enhance the reliability of the communication system. Multiple-description coding has received considerable attention, see e.g. [1–5].

In this letter we present an MDQ method which uses sorting in the transmitter in order to produce a low-quality description of a source frame. This description, if received, can be used in the receiver either to produce a low-quality estimate or otherwise to improve an existing estimate based on other descriptions.

The letter is organized as follows. In Section 2 sorting-based MDQ is introduced. In Section 3 a basic analysis of the the new method is presented. Finally, we provide numerical results and conclusions in Sections 4–5.

2 Sorting-Based MDQ

The proposed method is illustrated in Figure 1. Consider independent identically distributed (i.i.d) random variables $X_1, X_2, \dots, X_M, X_i \in \mathbb{R}$, with probability density function $p(x)$ of support $[L, U]$, $-\infty \leq L < U \leq \infty$. These variables are to be quantized and transmitted

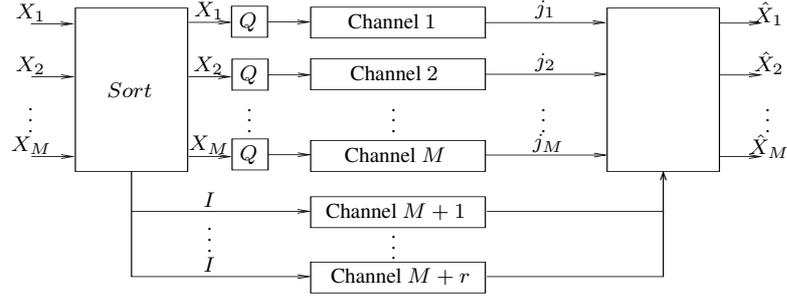


Figure 1: Sorting-based MDQ. A frame of random source samples are sorted in the transmitter. Quantized versions of these observations are then transmitted over different channels, as well as the index I containing information about how the observations were sorted. The received information is then used to estimate the original observations.

over a network using $M + r$ packets. We assume that the transmitted packets will be independently lost during the transmission meaning that it may be necessary to reconstruct the original data from only a subset of the transmitted packets. The suggested approach is then based on sorting¹ the observed random variables in the transmitter. The sorting procedure will rearrange a copy of the original values in ascending order. With $\mathcal{I}_M = \{1, \dots, M\}$, let $\pi : \mathcal{I}_M \rightarrow \mathcal{I}_M$ describe the corresponding permutation of the indices $1, 2, \dots, M$ such that

$$X_{\pi(1)} \leq X_{\pi(2)} \leq \dots \leq X_{\pi(M)}. \quad (1)$$

That is, $\pi(1)$ denotes the original index of the smallest X_i , etc. There are $M!$ possible permutations. Let $I \in \{1, \dots, M!\}$ be an integer that uniquely describes π , see e.g. [6] for details on how to determine such an index based on π . The index I can be represented in binary form using

$$f(M) \gtrsim \log_2(M!) \quad (2)$$

bits (with “ \gtrsim ” meaning “close to from above”). After sorting, X_1, X_2, \dots, X_M are quantized using M identical memoryless scalar quantizers. Letting $R_j = [l_j, u_j)$, $L \leq l_j < u_j \leq U$, denote the j th quantization region, the encoding produces the indices j_1, \dots, j_M as $X_i \in R_{j_i} \Rightarrow j_i = j$. These indices are then transmitted over M channels together with r identical copies of the index I , using r additional channels.

Let \hat{X}_i be the reconstructed version of X_i . In this letter we assume that the mean-square error (MSE), $E[(X_i - \hat{X}_i)^2]$, is used as a distortion measure. We will also assume that the different channels independently lose descriptors (packets) with probability p . Hence, due to the imperfections in the network some of the transmitted descriptors may be lost which leads to essentially four different cases when computing $\hat{X}_{\pi(i)}$:

¹There are numerous sorting algorithms in the literature which can be used in the transmitter, for instance Heap sort, having an $\mathcal{O}(M \log M)$ complexity which, for our application, is tolerable.

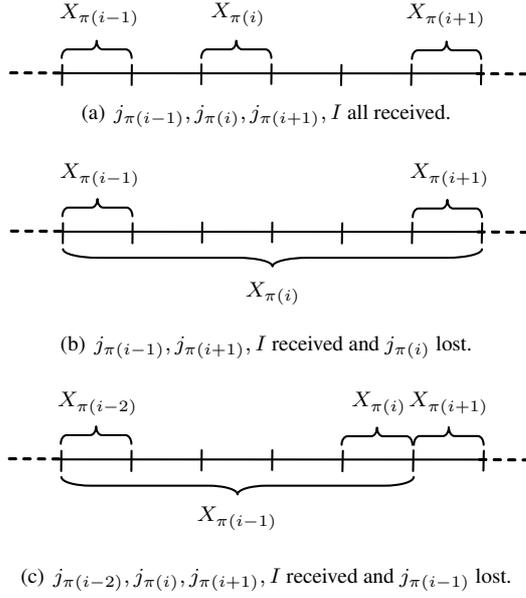


Figure 2: Three different cases that may occur due to packet losses. Quantization intervals are illustrated and possible intervals for some X_i 's are indicated based on the received indices.

1) At least the three descriptors $j_{\pi(i-1)}, j_{\pi(i)}, j_{\pi(i+1)}$ as well as the sorting index I received. In this case (Figure 2(a)) $\hat{X}_{\pi(i)} = E[X_{\pi(i)} | j_{\pi(i)}]$ constitutes a possible estimate. However, if two or more of the received j -values are equal, the index I will contain information that can be utilized to improve the estimate. Assume for instance that $j_{\pi(s+1)} = j_{\pi(s+2)} = \dots = j_{\pi(s+z)} = j$, for some s and z such that $1 \leq s+1 < s+z \leq M$, meaning that $X_{\pi(s+1)} \leq \dots \leq X_{\pi(s+z)}$. Hence, in order to minimize the MSE we should use

$$\begin{aligned} \hat{X}_{\pi(s+m)} &= E[X_{\pi(s+m)} | j_{\pi(s+1)}, \dots, j_{\pi(s+z)}, I] \\ &= E[X_{\pi(s+m)} | l_j \leq X_{\pi(s+1)} \leq \dots \leq X_{\pi(s+z)} < u_j] \end{aligned} \quad (3)$$

for all $m \in \{1, \dots, z\}$. The same situation occurs when any subset of the descriptors $j_{\pi(s+2)}, \dots, j_{\pi(s+z-1)}$ is lost, since we based on I and the received j -values will be able to conclude that $j_{\pi(s+1)} = \dots = j_{\pi(s+z)}$.

2) At least one of the descriptors $j_{\pi(i-1)}, j_{\pi(i)}, j_{\pi(i+1)}$ lost and I received. Here $j_{\pi(i)}$ is either lost (Figure 2(b)) or adjacent to a lost descriptor (Figure 2(c)). The information in I together with the received j -values can be exploited to produce $\hat{X}_{\pi(i)}$. Assume for instance that a number w of the j -values are lost, corresponding to the w consecutive source samples $X_{\pi(s+1)}, \dots, X_{\pi(s+w)}$, for some s , and assuming $j_{\pi(s-1)} < j_{\pi(s)}$ and

$j_{\pi(s+w+1)} < j_{\pi(s+w+2)}$, for simplicity. The estimate $\hat{X}_{\pi(s+m)}$ can then be computed as

$$\begin{aligned}\hat{X}_{\pi(s+m)} &= E[X_{\pi(s+m)} | j_{\pi(s)}, j_{\pi(s+w+1)}, I] \\ &= E[X_{\pi(s+m)} | l_{j_{\pi(s)}} \leq X_{\pi(s)} \leq X_{\pi(s+1)} \leq \cdots \\ &\quad \cdots \leq X_{\pi(s+w)} \leq X_{\pi(s+w+1)} < u_{j_{\pi(s+w+1)}}, \\ &\quad l_{j_{\pi(s)}} \leq X_{\pi(s)} < u_{j_{\pi(s)}}, l_{j_{\pi(s+w+1)}} \leq X_{\pi(s+w+1)} < u_{j_{\pi(s+w+1)}}] \quad (4)\end{aligned}$$

for $m = 0, \dots, w+1$. Note that this formula is used also for $m = 0$ and $m = w+1$ corresponding to received j -values. However, three special cases have to be considered, leading to modifications. These occur when $s+1 = 1$ and/or $s+w = M$ meaning that no lower/upper bound of the lost sequence is received, in which case we need to use L and/or U to bound the sequence. If $s+1 = 1$ we get

$$\begin{aligned}\hat{X}_{\pi(m)} &= E[X_{\pi(m)} | L \leq X_{\pi(1)} \leq \cdots \leq X_{\pi(w)} \leq X_{\pi(w+1)} < u_{j_{\pi(w+1)}}, \\ &\quad l_{j_{\pi(w+1)}} \leq X_{\pi(w+1)} < u_{j_{\pi(w+1)}}] \quad (5)\end{aligned}$$

for $m = 1, \dots, w+1$. Similarly, if $s+w = M$

$$\hat{X}_{\pi(s+m)} = E[X_{\pi(s+m)} | l_{j_{\pi(s)}} \leq X_{\pi(s)} \leq \cdots \leq X_{\pi(M)} \leq U, l_{j_{\pi(s)}} \leq X_{\pi(s)} < u_{j_{\pi(s)}}] \quad (6)$$

for $m = 0, \dots, w$. If both cases occur simultaneously we will get a similar formula to (3), where the source data are bounded by L and U and the index I alone will give a description of the sequence $\{X_i\}_{i=1}^M$.

3) *The descriptor j_i received and I lost.* Here no additional information about X_i is received, so we use

$$\hat{X}_i = E[X_i | j_i]. \quad (7)$$

4) *Both j_i and I lost.* In this case we do not receive any information at all about X_i , hence we use

$$\hat{X}_i = E[X_i]. \quad (8)$$

We note here that the structure proposed in Figure 1 transmits r identical copies of I , while better performance can actually be obtained for $r > 1$ if the coordinate system of the source data is rotated before producing each copy of I , resulting in $I_1 \neq I_2 \neq \dots \neq I_r$. These indices, if received, can be used jointly to produce a better descriptor. In this letter, designing such rotations is however left as a topic for future work.

3 Analysis

Here we will further analyze the reconstruction formulas (3)–(4), under the assumption that X_i is uniformly distributed on $[L, U] = [0, 1]$. We start by expanding (3) assuming a number z of the X -values are quantized to the same quantization cell, with index j . A uniform distribution gives

$$\hat{X}_{\pi(s+m)} = E[X_{\pi(s+m)} | l_j \leq X_{\pi(s+1)} \leq \cdots \leq X_{\pi(s+z)} < u_j] = l_j + \frac{m}{z+1}(u_j - l_j) \quad (9)$$

for all $m \in \{1, \dots, z\}$, stating the quite intuitive result that the reconstruction points should be spread out with an equal distance on the interval bounded by the j th quantization region. For the expression (4) we get

$$\begin{aligned} \hat{X}_{\pi(s+m)} &= E[X_{\pi(s+m)} | j_{\pi(s)}, j_{\pi(s+w+1)}, I] = l_{j_{\pi(s)}} + \\ &K \frac{(m+1)(1 - (1-u)^{w+3} - l^{w+3} + (l-u)^{w+3})}{1 - (1-u)^{(w+2)} - l^{(w+2)} + (l-u)^{(w+2)}} + \\ &K \frac{u(w+3)((l-u)^{(w+2)} - (1-u)^{(w+2)})}{1 - (1-u)^{(w+2)} - l^{(w+2)} + (l-u)^{(w+2)}}, \end{aligned}$$

where $K = \frac{u_{j_{\pi(s+w+1)}} - l_{j_{\pi(s)}}}{w+3}$, $u = \frac{u_{j_{\pi(s)}} - l_{j_{\pi(s)}}}{u_{j_{\pi(s+w+1)}} - l_{j_{\pi(s)}}}$, $l = \frac{l_{j_{\pi(s+w+1)}} - l_{j_{\pi(s)}}}{u_{j_{\pi(s+w+1)}} - l_{j_{\pi(s)}}}$

(10)

for $m = 0, \dots, w+1$, assuming w consecutive descriptors have been lost. In a similar manner we get

$$\begin{aligned} \hat{X}_{\pi(m)} &= E[X_{\pi(m)} | j_{\pi(w+1)}, I] \\ &= \frac{(u_{j_{\pi(w+1)}})}{w+2} \frac{m(1 - (l_{j_{\pi(w+1)}}/u_{j_{\pi(w+1)}})^{w+2})}{1 - (l_{j_{\pi(w+1)}}/u_{j_{\pi(w+1)}})^{w+1}}, \text{ for } m = 1, \dots, w+1 \end{aligned}$$
(11)

$$\begin{aligned} \hat{X}_{\pi(s+m)} &= E[X_{\pi(s+m)} | j_{\pi(s)}, I] \\ &= l_{j_{\pi(s)}} + \frac{(1 - l_{j_{\pi(s)}})}{w+2} \frac{(m+1)(1 - (1-u)^{w+2}) - u(w+2)(1-u)^{w+1}}{1 - (1-u)^{w+1}} \end{aligned}$$

where $u = \frac{u_{j_{\pi(s)}} - l_{j_{\pi(s)}}}{1 - l_{j_{\pi(s)}}}$ for $m = 0, \dots, w$

(12)

from (5) and (6), corresponding to the cases when we do not receive any descriptors producing a lower/upper limit for the lost sequence.

We also introduce a restriction on the rate R_X that can be used per quantizer. Let R denote the total rate per symbol X_i . In the case $M = 1$ and no sorting index, we would get $R_X = R$. However, increasing M and transmitting the sorting index will cost an extra $f(M)$, from (2), bits per M source samples. Hence, fixing the total rate per transmitted symbol we require the quantizers to restrict their rate to

$$R_X = R - M^{-1} f(M). \quad (13)$$

From (2)–(13) a closed-form analytical expression for the MSE when using sorting-based MDQ for a uniform source distribution can be derived, as shown in the appendix. The MSE will depend on 4 parameters, namely r – the retransmission index, M – the frame size, p – the packet loss probability and finally R – the rate per symbol. Using this formula, assuming R fixed and p known, we can optimize the choice of (M, r) .

4 Numerical Results

We investigate the performance of sorting-based MDQ for i.i.d uniform X_i 's by comparing to two other methods: 1) A scheme based on forward error correction (FEC) using

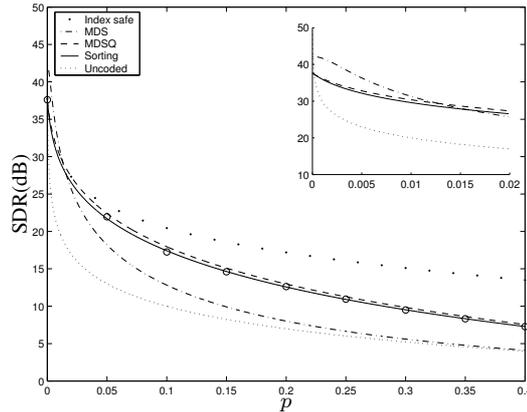


Figure 3: We compare sorting-based MDQ (solid line), MDS coding (dashed/dot), MDSQ (dashed) and an uncoded system (small dots). The circles show simulated results of a sorting-based MDQ system, coinciding with the analytical curve. We also illustrate the performance of sorting-based MDQ when the sorting index is transmitted over an error-free channel (larger dots). The systems are optimized for $R = 8$ and $p = 0.002$ (giving $(M, r) = (7, 1)$ for sorting-based MDQ).

maximum distance separable (MDS) codes [7] (MDS codes have the maximum possible minimum distance between codewords). In order to make the comparison fair we required the FEC to produce $M + r$ descriptors when encoding k X_i 's, that is $M + r$ channels are used. Such an MDS code will be able to correct $M + r - k$ losses. Reference [8] presents a formula to calculate an approximate value of the MSE when a given number of descriptors are lost, in a system with scalar quantization and FEC based on an MDS code. This applies directly to our setup, and can easily be extended to give an analytical MSE for a given packet loss probability, p ; 2) We also compare to a practical MDQ system, namely multiple description scalar quantization (MDSQ) [1]. This method will use fewer channels than the proposed method but on the other hand it has a higher design complexity. In Figures 3–4 the different systems are compared when optimized² for different loss probabilities and $R = 8$. In Figure 3 the systems are optimized for $p = 0.002$ and in Figure 4 for $p = 0.01$. Performance is studied by plotting the signal-to-distortion ratio (SDR), $E[X^2]/E[(X - \hat{X})^2]$, versus p . Magnified versions of the plots are also included to show the region with low loss probability in more detail. The solid line shows the performance of sorting-based MDQ based on the analytical formula (14) presented in the appendix. The circles are from a sim-

²For the MDS code the best k is found. For sorting-based MDQ (14) is minimized. However, since MDSQ is specified by the two design parameters $n \in \{0, \dots, 2^{\frac{R}{2}} - 1\}$ and $\lambda \in \mathbb{R}^+$, there are an infinite number of possible systems. We designed all possible combinations of n and $\lambda \in \{0, 0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ corresponding to a wide range of systems with no protection to high protection. The best of these was chosen.

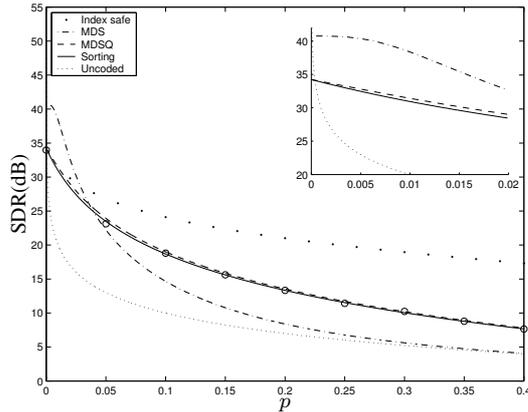


Figure 4: We compare sorting-based MDQ (solid line), MDS coding (dashed/dot), MDSQ (dashed) and an uncoded system (small dots). The circles show simulated results of a sorting-based MDQ system, coinciding with the analytical curve. We also illustrate the performance of sorting-based MDQ when the sorting index is transmitted over an error-free channel (larger dots). The systems are optimized for $R = 8$ and $p = 0.01$ (giving $(M, r) = (12, 1)$) for sorting-based MDQ.

ulated version of this system, and as shown these coincide with the curve for the analytical expression. The dash/dotted line shows the performance of the FEC/MDS-based scheme. This latter system outperforms the other systems in the region where they were optimized. It should however be pointed out that this is a theoretical result assuming we have access to a certain MDS code, finding this code may however be a nontrivial task [7]. The dashed line corresponds to MDSQ which has a similar performance to the proposed method. As a reference, the performance of a system using no channel coding is also evaluated, corresponding to $M = 1$ and $R_X = R$ for sorting-based MDQ, as shown by the (lower) dotted line. Finally, in order to study further how I impacts the performance, we also evaluate a sorting-based MDQ system where the sorting index I is assumed to be transmitted over an error-free channel, that is, the probability of losing the index I is 0. This system's performance is shown by the (upper) dotted line.

We are also interested in further exploring the effect of using the analytical expression (14) when choosing M and r . This is done in Figure 5, where the solid line represents the performance when the choice of (M, r) has been optimized for each loss probability. This system is compared to the same sorting-based MDQ system as in Figure 4 (dashed line), as well as the system without any channel coding (dotted line). It is clear that the choice of (M, r) affects the performance. The optimized choices of (M, r) are also shown in Table 1 for a few different loss probabilities. As can be seen from the table, for low loss probabilities $r = 1$ and M is fairly small. When the loss probability is increased the optimal M increases. At some stage, however, it becomes more beneficial to choose a smaller M

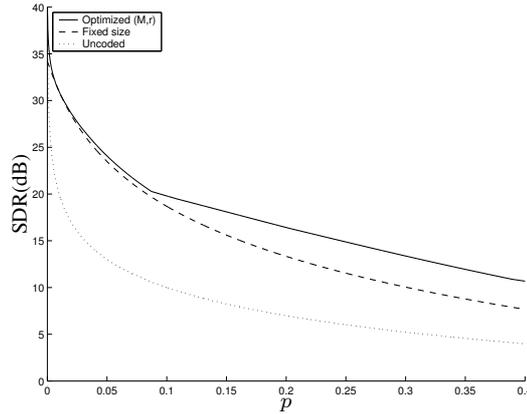


Figure 5: The performance of a sorting-based MDQ system (solid line) where the choice of (M, r) is optimized, a sorting-based MDQ system using the fixed size $M = 12$ and $r = 1$ (dashed line) and a system using no channel coding (dotted line).

and instead transmit I twice, i.e. $r = 2$, which occurs at $p = 0.088$. This point is easy to distinguish in Figure 5.

Table 1: The optimal (M, r) for different loss probabilities.

| p | 0 | 0.001 | 0.002 | 0.003 | ... | 0.087 | 0.088 |
|-----|---|-------|-------|-------|-----|-------|-------|
| r | 1 | 1 | 1 | 1 | ... | 1 | 2 |
| M | 1 | 6 | 7 | 8 | ... | 25 | 9 |

5 Conclusions

We introduced sorting-based multiple description quantization. The new MDQ method is based on sorting a set of source samples, and transmitting as side-information/redundancy an index that describes the resulting permutation. We also provided analysis of the performance holding for i.i.d uniform samples. The new technique was compared numerically to a scheme that uses FEC with maximum distance separable codes as well as to MDSQ, when these were designed for a certain packet loss probability. It was found that the FEC-based scheme had the better performance and that the proposed method had a similar performance to MDSQ. However, compared to the FEC and MDSQ schemes, the suggested method has virtually zero design complexity, making it easy to implement and adapt to varying loss probabilities. It also has the advantage over MDSQ to allow straightforward implementation of more than two descriptions.

Appendix

We will here derive an analytical approximation of the MSE for the proposed system, assuming the quantizers can use $\lfloor 2^{R_X} \rfloor$ quantization cells. We get

$$\text{MSE} = p^r \text{MSE}_{\text{lost}} + (1 - p^r) \text{MSE}_{\text{sort}} \quad (14)$$

where MSE_{lost} corresponds to the MSE when all r copies of I are lost. The probability of this event is p^r and leads to using (7)–(8) for reconstruction, making the calculation of MSE_{lost} straightforward. Using the well known formula for quantization noise of a uniform distribution gives

$$\text{MSE}_{\text{lost}} = p \frac{1}{12} + (1 - p) \frac{1}{12 \lfloor 2^{R_X} \rfloor^2}. \quad (15)$$

MSE_{sort} corresponds to the event when at least one copy of I is received, and is given as

$$\text{MSE}_{\text{sort}} = \sum_{k=0}^M P(k|M) \frac{1}{M} (\text{MSE}_r(k) + \text{MSE}_l(k)). \quad (16)$$

Here k describes how many of the M scalar quantization indices that were lost, and

$$P(k|M) = \binom{M}{k} p^k (1 - p)^{M-k} \quad (17)$$

is the probability of this event. We use (10) to reconstruct the lost values and (9) for the received values. With k losses, the resulting MSE's are denoted $\text{MSE}_l(k)$ (lost) and $\text{MSE}_r(k)$ (received), we get:

1) $\text{MSE}_l(k)$: Assuming that the w consecutive values $\{j_{\pi(s+m)}\}_{m=1}^{m=w}$ are lost (10) is used in order to estimate $\{X_{\pi(s+m)}\}_{m=0}^{m=w+1}$. However, unless $2^{R_X} \ll M$ we can expect $\hat{X}_{\pi(s)} \approx E[X|j_{\pi(s)}]$ and $\hat{X}_{\pi(s+w+1)} \approx E[X|j_{\pi(s+w+1)}]$, that is, as if (7) were used. The interval $(E[X|j_{\pi(s)}], E[X|j_{\pi(s+w+1)}])$ therefore bounds the lost values and $\{X_{\pi(s+m)}\}_{m=1}^{m=w}$ can be estimated using (9). Taking quantization noise, approximately distributed as $q_i \sim \mathcal{U}[-\frac{1}{2 \lfloor 2^{R_X} \rfloor}, \frac{1}{2 \lfloor 2^{R_X} \rfloor}]$ (with variance σ_q^2), into account gives

$$\hat{X}_{\pi(s+m)} \approx (X_{j_{\pi(s)}} + q_1) + \frac{m}{w+1} ((X_{j_{\pi(s+w+1)}} + q_2) - (X_{j_{\pi(s)}} + q_1)). \quad (18)$$

Using this formula we get

$$\begin{aligned} \mu(m|w) &= E[(\hat{X}_{\pi(i+m)} - X_{\pi(i+m)})^2 | 0 \leq \dots \leq X_{\pi(i)} \leq \dots \leq X_{\pi(i+w+1)} \leq \dots \leq 1] \\ &\approx \frac{1}{(M+1)(M+2)} \frac{m}{w+1} (w+1-m) + \sigma_q^2 \left[\left(1 - \frac{m}{w+1}\right)^2 + \left(\frac{m}{w+1}\right)^2 \right], \end{aligned} \quad (19)$$

where $\mu(m|w)$ is the expected MSE for $\hat{X}_{\pi(s+m)}$ in a lost sequence of length w .

We also need to know how likely a lost sequence of w consecutive samples is when k quantization indices are lost. In total there will be $\binom{M}{k}$ equally likely permutations for a given k . When $M-1 \leq k$ it is trivial to calculate how many of these that contain w consecutive lost samples. Otherwise, let $\alpha(w, s)$ denote the total number of permutations

where a lost sequence of length w starts at $X_{\pi(s)}$. Two cases need to be considered. The first is when $s = 1$ giving

$$\alpha(w, 1) = \binom{M-w-1}{k-w}. \quad (20)$$

The same expression is obtained when studying lost sequences of length w that end in $X_{\pi(M)}$. In between there will be $(M-1-w)$ possible locations to start a sequence of length w , in these cases

$$\alpha(w, s) = \binom{M-w-2}{k-w}. \quad (21)$$

Hence, the total number of lost sequences of length w , $\alpha_{tot}(w) = \sum_{s=1}^{s=M} \alpha(w, s)$, will be

$$\alpha_{tot}(w) = 2 \binom{M-w-1}{k-w} + (M-w-1) \binom{M-w-2}{k-w}. \quad (22)$$

Normalizing (22) with the total number of permutations gives

$$E[\alpha_{tot}(w)|k] = \frac{2 \binom{M-w-1}{k-w} + (M-w-1) \binom{M-w-2}{k-w}}{\binom{M}{k}} \quad (23)$$

which is the expected number of lost sequences of length w when k quantization indices have been lost (assuming $M-1 > k$). If $M-1 \leq k$ we get

$$\alpha_{tot}(w) = \begin{cases} 2 - 2\delta[M-w] & \text{when } M = k+1 \\ \delta[M-w] & \text{when } M = k \end{cases} \quad (24)$$

(with $\delta[\cdot]$ being the Kronecker delta) and normalizing gives $E[\alpha_{tot}(w)|k]$. (19), (23)–(24) then give

$$\text{MSE}_l(k) = \sum_{w=1}^k \left(E[\alpha_{tot}(w)|k] \sum_{m=1}^w \mu(m|w) \right). \quad (25)$$

2) $\text{MSE}_r(k)$: We are now interested in the number $z =$ number of X_i 's that were quantized to a given quantization cell. Assuming that we receive $j_{\pi(s+1)} = \dots = j_{\pi(s+z)}$ a similar derivation to (19) gives

$$\begin{aligned} \nu(m|z) &= E[(\hat{X}_{s+m} - X_{s+m})^2 | l_{j_{\pi(s+1)}} \leq X_{j_{\pi(s+1)}} \leq \dots \leq X_{j_{\pi(s+z)}} \leq u_{j_{\pi(s+z)}}] \\ &= \frac{m(z+1-m)}{[2^{R_X}]^2 (z+1)^2 (z+2)} \end{aligned} \quad (26)$$

where $\nu(m|z)$ is the expected MSE when estimating $\hat{X}_{\pi(s+m)}$. Then we study the probability of a certain z given that k samples were lost. Denoting $\beta_{tot}(z)$ the number of permutations yielding z (this time position is of no interest) gives

$$E[\beta_{tot}(z)|k] = \frac{[2^{R_X}] \binom{M-k+[2^{R_X}]-2-z}{M-k-z}}{\binom{M-k+[2^{R_X}]-1}{M-k}}. \quad (27)$$

In this calculation we have neglected the case when for instance $j_{\pi(2)}$ is lost and $j_{\pi(1)} = j_{\pi(3)}$ is received, since this makes it possible to conclude $j_{\pi(1)} = j_{\pi(2)}$. Finally we use (26)–(27) to get

$$\text{MSE}_r(k) = \sum_{z=1}^{M-k} \left(E[\beta_{\text{tot}}(z)|k] \sum_{m=1}^z \nu(m|z) \right). \quad (28)$$

References

- [1] V. A. Vaishampayan, “Design of multiple description scalar quantizers,” *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 821–834, 1993.
- [2] V. A. Vaishampayan, N. J. A. Sloane, and S. D. Servetto, “Multiple description vector quantization with lattice codebooks: Design and analysis,” *IEEE Transactions on Information Theory*, vol. 47, no. 5, pp. 1718–1734, 2001.
- [3] V. K. Goyal, J. Kovacević, and M. Vetterli, “Multiple description transform coding: Robustness to erasures using tight frame expansion,” in *Proceedings IEEE Int. Symp. Information Theory*, Cambridge, MA, August 1998, p. 408.
- [4] V. Goyal, M. Vetterli, and N. Thao, “Quantized overcomplete expansions in \mathbb{R}^n : analysis, synthesis, and algorithms,” *IEEE Transactions on Information Theory*, vol. 44, pp. 16–31, January 1998.
- [5] V. K. Goyal, J. Kovacević, and J. A. Kelner, “Quantized frame expansions with erasures,” *Applied and Computational Harmonic Analysis*, vol. 10, no. 3, pp. 203 – 233, May 2001.
- [6] T. Cover, “Enumerative source encoding,” *IEEE Transactions on Information Theory*, vol. 19, pp. 73–77, jan 1973.
- [7] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: North-Holland, 1977.
- [8] V. K. Goyal, J. Kovacević, and M. Vetterli, “Quantized frame expansions as source-channel codes for erasure channels,” in *Proceedings IEEE Data Compression Conference*, Snowbird, UT, March 1999, pp. 326–335.

Addendum to the Paper

In the original paper the details on how to develop the expressions (4)–(6) to (10)–(12) were left out due to lack of space. These derivations are summarized in this section which was not included in the original paper. To develop (4) given on a form similar to

$$\hat{\xi}_j = E[\xi_j | 0 \leq \xi_1 \leq \xi_2 \leq \dots \leq \xi_k \leq 1, 0 \leq \xi_1 \leq u, l \leq \xi_k \leq 1] \quad (29)$$

we define

$$\begin{aligned} \mathcal{V}_1 &= \{\xi_1, \xi_2, \dots, \xi_k : 0 \leq \xi_1 \leq \xi_2 \leq \dots \leq \xi_k \leq 1\} \\ \mathcal{V}_2 &= \{\xi_1, \xi_2, \dots, \xi_k : u \leq \xi_1 \leq \xi_2 \leq \dots \leq \xi_k \leq 1\} \\ \mathcal{V}_3 &= \{\xi_1, \xi_2, \dots, \xi_k : 0 \leq \xi_1 \leq \xi_2 \leq \dots \leq \xi_k \leq l\} \\ \mathcal{V}_4 &= \{\xi_1, \xi_2, \dots, \xi_k : u \leq \xi_1 \leq \xi_2 \leq \dots \leq \xi_k \leq l\} \end{aligned}$$

and further

$$\begin{aligned} \mathcal{V} &= \{\xi_1, \xi_2, \dots, \xi_k : 0 \leq \xi_1 \leq \xi_2 \leq \dots \leq \xi_k \leq 1, 0 \leq \xi_1 \leq u, l \leq \xi_k \leq 1\} \\ &= ((\mathcal{V}_1 \setminus \mathcal{V}_2) \cup \mathcal{V}_4) \setminus \mathcal{V}_3. \end{aligned}$$

This makes it possible, using the notation $\xi = \{\xi_1, \xi_2, \dots, \xi_k\}$, to write (29) as

$$\begin{aligned} \hat{\xi}_i &= E[\xi_i | \xi \in \mathcal{V}] = \frac{\int_{\mathcal{V}} f(\xi) \xi_i d\xi}{\int_{\mathcal{V}} f(\xi) d\xi} = \frac{\int_{\mathcal{V}} \xi_i d\xi}{\int_{\mathcal{V}} 1 d\xi} \\ &= \frac{\int_{\mathcal{V}_1} \xi_i d\xi - \int_{\mathcal{V}_2} \xi_i d\xi - \int_{\mathcal{V}_3} \xi_i d\xi + \int_{\mathcal{V}_4} \xi_i d\xi}{\int_{\mathcal{V}_1} 1 d\xi - \int_{\mathcal{V}_2} 1 d\xi - \int_{\mathcal{V}_3} 1 d\xi + \int_{\mathcal{V}_4} 1 d\xi} \end{aligned} \quad (30)$$

where each of the integrals is straightforward to derive and doing so will result in (10). In the same manner we get (11)–(12) from (5)–(6).

Paper E

Multiple Description Coding using Rotated Permutation Codes

Niklas Wernersson and Mikael Skoglund

Article summarized in *Proceedings of IEEE Data Compression Conference 2006*

Full article published as *Technical Report TRITA-EE 2006:007*,
KTH, School of Electrical Engineering 2006

Multiple Description Coding using Rotated Permutation Codes

Niklas Wernersson and Mikael Skoglund

Abstract

We introduce a new method for multiple description coding based on producing different permutation coded descriptions of the original source data. The proposed method is of relatively low complexity and scales easily to any number of descriptions. Performance gains compared to resolution-constrained multiple description scalar quantization are demonstrated. The new approach is also able to match the performance of entropy-constrained multiple description scalar quantization.

1 Introduction

Packet networks have gained in importance in recent years, for instance via the wide-spread use and importance of the Internet. Unfortunately, packet losses in these systems can in general not be neglected and this has to be taken into account in robust source-channel coding for multimedia communications. One way to deal with packet losses is to use multiple description coding (MDC) where the objective is to code one source of data into multiple bitstreams. The coding is done in such a way that multiple levels of quality is achieved. This means that even if one or a few of the bitstreams are lost, the received bits should make it possible to get an approximated version of the original data. MDC will therefore enhance the reliability of the communication system. Multiple-description source coding has received considerable attention: see e.g. [1–6] for practical coding schemes and [7, 8] for theoretical bounds.

The contribution of the present paper is to study the possibility of using permutation coding, see e.g. [9–11], as a basic building block in a new MDC scheme. We produce multiple descriptions by utilizing permutation coding to code randomly rotated versions of a source sequence. The result is a low-complexity approach to MDC that easily scales to any number of descriptions.

The paper is organized as follows. In Section 2 a brief presentation of permutation codes is given. In Section 3 the proposed MDC scheme is presented. Finally, we provide numerical results and conclusions in Sections 4–5.

2 Preliminaries

There have been numerous papers written on permutation codes. We will in this paper focus on “Variant I” minimum mean-squared error permutation codes meaning that we will use the popular mean squared error (MSE) as a distortion measure. A good introduction to these codes and how to design them can be found in [10] which will be briefly summarized in this section.

Consider the case when we want to code a sequence of real valued random variables $\{x_i\}_{i=1}^{\infty}$. With permutation coding this sequence can be block quantized in a simple fashion such that the block $\mathbf{x} = (x_1, x_2, \dots, x_N)^T$ is quantized to an index $I \in \{1, \dots, M\}$. There will exist one codeword, for instance corresponding to the first index, of the form

$$\mathbf{c}_1 = (\overset{\leftarrow n_1 \rightarrow}{\mu_1, \dots}, \overset{\leftarrow n_2 \rightarrow}{\mu_1, \mu_2, \dots}, \mu_2, \dots, \overset{\leftarrow n_K \rightarrow}{\mu_K, \dots}, \mu_K) \quad (1)$$

where μ_i satisfies $\mu_1 \leq \mu_2 \leq \dots \leq \mu_K$ and the n_i 's are positive integers satisfying $n_1 + n_2 + \dots + n_K = N$. All other codewords $\mathbf{c}_2, \mathbf{c}_3, \dots, \mathbf{c}_M$ are constructed by creating all possible permutations of \mathbf{c}_1 meaning that there in total will be

$$M = N! / \prod_{i=1}^K n_i! \quad (2)$$

different codewords. If the components of \mathbf{x} are independent identically distributed (i.i.d) all of these permutations are equally likely meaning the the entropy of I will equal $\log_2 M$. It is a fairly straightforward task to map each of these permutations to a binary number corresponding to a Huffman code. Hence, the rate per coded symbol, x_i , is given from

$$R \simeq \frac{1}{N} \log_2 M. \quad (3)$$

Also, it turns out that the optimal encoding procedure, for a given set $\{(n_i, \mu_i)\}_{i=1}^K$, is to replace the n_1 smallest components of \mathbf{x} by μ_1 , the next n_2 smallest components by μ_2 and so on. This further means that ordering the components of \mathbf{x} also will decide the outcome of the block quantization. This is an appealing property since sorting can be done with $\mathcal{O}(N \log N)$ complexity which is low enough such that block quantization can be implemented for very high dimensions. For details on how to design $\{(n_i, \mu_i)\}_{i=1}^K$ see [10].

3 Proposed MDC Scheme

As briefly mentioned in Section 2 permutation codes have some quite appealing properties, for instance their ability to perform high dimensional block quantization at low complexity. In this section we will present an MDC scheme based on permutation codes making it possible to perform high dimensional MDC. This scheme is illustrated in Figure 1.

Consider the random vector $\mathbf{x} = (x_1, \dots, x_N)^T$ where the x_i 's are i.i.d zero-mean Gaussian random variables for $i = 1, \dots, N$. This vector is to be quantized and transmitted over a network by the use of J descriptors (or packets). We assume that these descriptors are transmitted over J channels that independently either loses or otherwise recovers the descriptor perfectly at the receiver side.

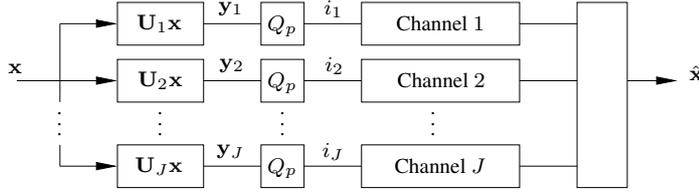


Figure 1: Proposed scheme for MDC using rotated permutation codes.

When designing the system in Figure 1 we start by generating J random unitary $N \times N$ matrices, $\{\mathbf{U}_j\}_{j=1}^J$, with eigenvectors uniformly distributed on an N -dimensional hypersphere¹. Both the encoder and decoder will have access to these matrices. Then, for each channel the vector

$$\mathbf{y}_j = \mathbf{U}_j \mathbf{x} \quad (4)$$

is produced. All these vectors will also be zero-mean Gaussian since the multiplication of a unitary matrix corresponds to a rotation in the N -dimensional space. The vectors \mathbf{y}_j are then block-quantized using a permutation code (denoted Q_p in the figure). Note that only the parameters of one permutation code need to be determined since $\mathbf{y}_1, \dots, \mathbf{y}_J$ all have the same pdf. Hence, once one permutation code has been designed, this code can be used in all the quantizers, Q_p . The generated index from each quantized version of \mathbf{y}_j is then transmitted over the corresponding channel. The receiver will receive either all or a subset of the transmitted indices depending on whether indices were lost or not. Let the symbol ' \times ' denote "descriptor lost", i.e., a received value $i_j = \times$ means the j :th channel failed. Using MSE as distortion measure the optimal reconstruction point, $\hat{\mathbf{x}}$, is then given as

$$\hat{\mathbf{x}} = E[\mathbf{x}|i_1, \dots, i_J] = r^* \mathbf{v}^* \quad (5)$$

(with $i_j \in \{1, \dots, M, \times\}$) where r^* is the length of this vector and \mathbf{v}^* is the normalized direction such that $\|\mathbf{v}^*\|^2 = 1$.

3.1 Calculating $E[\mathbf{x}|i_j]$

Consider the case when only one descriptor i_j is received. One intuitive way to estimate \mathbf{x} is to set $\hat{\mathbf{x}} = \mathbf{U}_j^{-1} \hat{\mathbf{y}}$ where $\hat{\mathbf{y}} = E[\mathbf{y}|i_j]$ is given when decoding the permutation code. This gives the distortion

$$\begin{aligned} D_{\mathbf{x}} &= \frac{1}{N} E\|\mathbf{x} - \hat{\mathbf{x}}\|^2 = \frac{1}{N} E\|\mathbf{x} - \mathbf{U}_j^{-1} \hat{\mathbf{y}}\|^2 = \\ &= \frac{1}{N} E\|\mathbf{U}_j(\mathbf{x} - \mathbf{U}_j^{-1} \hat{\mathbf{y}})\|^2 = \frac{1}{N} E\|\mathbf{y} - \hat{\mathbf{y}}\|^2 = D_{\mathbf{y}}. \end{aligned} \quad (6)$$

Hence, the distortion $D_{\mathbf{y}}$, which is given when reconstructing \mathbf{y} with its optimal reconstruction point $E[\mathbf{y}|i_j]$, equals the distortion $D_{\mathbf{x}}$. Furthermore, since \mathbf{x} and \mathbf{y} have the

¹This can be done by for instance conducting a QR-factorization of a matrix containing zero-mean Gaussian variables.

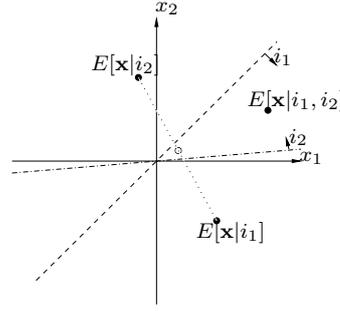


Figure 2: Illustration of the basic idea. Each received descriptor will describe a possible N -dimensional volume that \mathbf{x} belonged to. In this case we have 2 dimensions.

same distribution we conclude that also $\hat{\mathbf{x}}$ must have been an optimal reconstruction point. Hence,

$$E[\mathbf{x}|i_j] = \mathbf{U}_j^{-1} E[\mathbf{y}|i_j]. \quad (7)$$

3.2 Calculating $E[\mathbf{x}|i_1, \dots, i_J]$

We will now think of the received indices as descriptions of volumes in an N -dimensional space. To explain this idea, consider the example, illustrated in Figure 2, with a permutation code using the parameters $N = 2$, $n_1 = n_2 = 1$. Also assume $J = 2$ and $\mathbf{U}_1 = \mathbf{I}_2$ where \mathbf{I}_2 is the 2-unitary matrix. Receiving i_1 will in this case describe whether $x_1 \leq x_2$ or $x_2 < x_1$ corresponding to a volume which we will denote \mathcal{V}_1 . Hence, i_1 tells us that $\mathbf{x} \in \mathcal{V}_1$ which gives the optimal reconstruction point as $E[\mathbf{x}|i_1] = E[\mathbf{x}|\mathbf{x} \in \mathcal{V}_1]$. In the same manner also i_2 will describe a volume, \mathcal{V}_2 , such that $\mathbf{x} \in \mathcal{V}_2$. This volume will however be different from \mathcal{V}_1 due to the random rotation. In the case that both i_1 and i_2 are received we know that $\mathbf{x} \in \mathcal{V}_1 \cap \mathcal{V}_2$ and we get the optimal reconstruction point $E[\mathbf{x}|i_1, i_2] = E[\mathbf{x}|\mathbf{x} \in \mathcal{V}_1 \cap \mathcal{V}_2]$. In general, each received index i_j will describe a volume \mathcal{V}_j and since all channels use the same permutation code all of these volumes will have equal volume (except in the case of a lost descriptor; $i_j = \times$ gives $\mathcal{V}_j = \mathbb{R}^N$ and $E[\mathbf{x}|i_j] = \mathbf{0}$). This gives

$$E[\mathbf{x}|i_1, \dots, i_J] = \frac{\int_{\mathcal{V}_1 \cap \dots \cap \mathcal{V}_J} \mathbf{x} f(\mathbf{x}) d\mathbf{x}}{\int_{\mathcal{V}_1 \cap \dots \cap \mathcal{V}_J} f(\mathbf{x}) d\mathbf{x}} \approx \frac{1}{J'} \sum_{j=1}^J E[\mathbf{x}|i_j] = r\mathbf{v} \quad (8)$$

where J' is the number of received (i.e. “nonlost”) packets, r is the length of the resulting vector and \mathbf{v} is the normalized direction such that $\|\mathbf{v}\|^2 = 1$. The approximation done by averaging between all different reconstruction points makes intuitive sense, in Figure 2 the resulting point is symbolized by a circle. This approximation is further motivated by the results presented in the appendix where we, for the example illustrated in Figure 2, derive upper and lower bounds on the length r^* of the optimal reconstruction point in (5). Furthermore, for the general case we have managed to show an upper bound and we conject

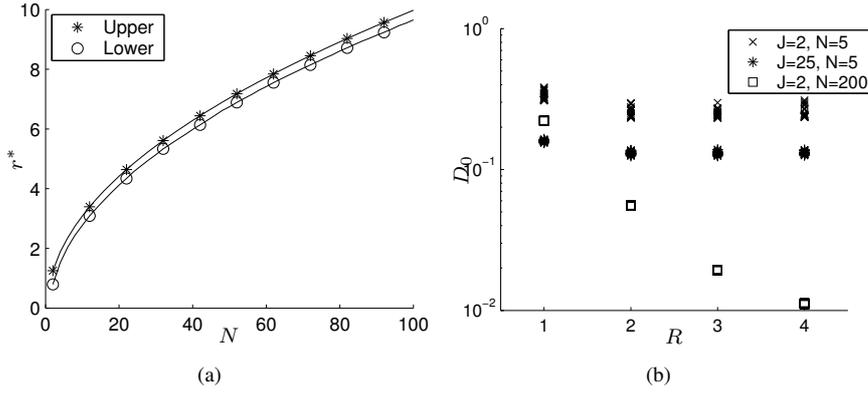


Figure 3: (a) The upper and lower limits for r^* when $R = 2.5$ bits/symbol/channel. (b) The effect of the random matrix generation for $J = 2, N = 5$ (crosses), $J = 25, N = 5$ (asterisk) and $J = 2, N = 200$ (boxes).

a lower bound accordingly to

$$\sqrt{\sum_{i=1}^K n_i \mu_i^2} \leq r^* \leq \frac{\sqrt{2}\sigma\Gamma\left(\frac{N+1}{2}\right)}{\Gamma\left(\frac{N}{2}\right)} \quad (9)$$

where $\Gamma(\cdot)$ is the gamma function. In Figure 3(a) we show these bounds when $R = 2.5$ bits/symbol/channel. Increasing the rate will make the gap between the lower and upper bound smaller and decreasing the rate will increase the gap. We also note that

$$\begin{aligned} r^2 &= \left\| \frac{1}{J} \sum_{j=1}^J E[\mathbf{x}|i_j] \right\|^2 = \frac{1}{J^2} \left| \sum_{j_1=1}^J \sum_{j_2=1}^J E[\mathbf{x}|i_{j_1}]^T E[\mathbf{x}|i_{j_2}] \right| \leq \\ &\leq \frac{1}{J^2} \sum_{j_1=1}^J \sum_{j_2=1}^J \left| E[\mathbf{x}|i_{j_1}]^T E[\mathbf{x}|i_{j_2}] \right| \leq \sum_{i=1}^K n_i \mu_i^2 \end{aligned} \quad (10)$$

meaning that the length of the vector yielded by (8) is shorter, or equal, then the lower bound of the optimal length in (9). We therefore replace the length r by the lower bound giving the final formula for estimating $\hat{\mathbf{x}}$ as

$$E[\mathbf{x}|i_1, \dots, i_J] \approx \left(\sum_{i=1}^K n_i \mu_i^2 \right)^{\frac{1}{2}} \mathbf{v}. \quad (11)$$

It was also verified in our simulations that this modification of the vector length gave a small increase in performance.

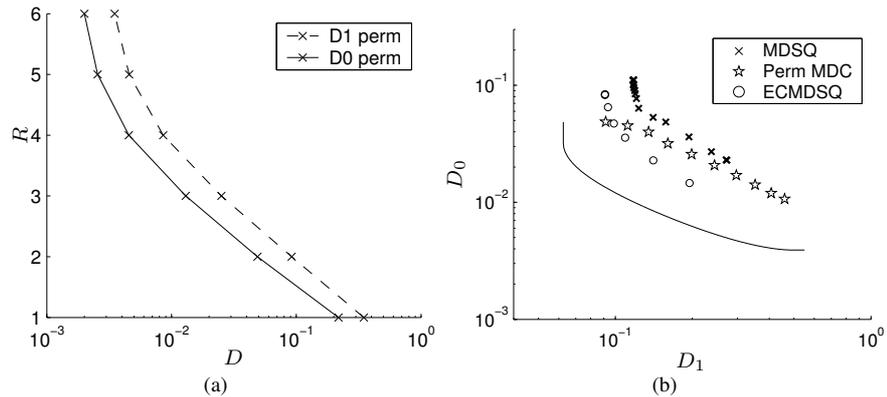


Figure 4: (a) Comparing D_0 (solid line) and D_1 (dashed line) for the suggested method. (b) Studying the relation between D_0 and D_1 for SMDQ (crosses), ECSMDQ (circles), MDC based on permutation codes (pentagrams) and the theoretical bound of [7] (solid line).

3.3 The Effect of the Generating Random Matrices

Since the matrices $\{\mathbf{U}_j\}_{j=1}^J$ are generated in a random fashion one might expect that different outcomes of this generation process will correspond to varying the performance of the system. However, our simulations indicate that when J and/or N grows large the performance converges to a fixed value. This is illustrated in Figure 3(b) where we analyze the performance of 10 different realizations of the system, hence we try 10 different sets of randomly generated matrices, when receiving all J descriptions. This is done for three cases, in the first (crosses) we use $J = 2, N = 5$. From the plot we see that for this case the performance of the different realizations become quite different meaning that some of the generated matrices are better than others. In the second case (asterisk) we increase J such that $J = 25, N = 5$. For this case it appears to be no greater difference in performance between the different realizations. This seems to be the case also for the third case (boxes) when we instead increase N such that $J = 2, N = 200$. We consider the proposed method to be most interesting when using large N and moderate values for J corresponding to the third case. Hence, the conclusion is that for our purposes generating the rotation matrices at random works satisfyingly.

4 Simulations

In the simulations we use i.i.d. zero-mean Gaussian source data with $\sigma^2 = 1, J = 2$ and for the permutation codes $N = 1000$. It should be noted that the rate, R , is defined *per channel*, that is the unit is bits/symbol/channel. In Figure 4(a) we study the usefulness of (11) by comparing the central distortion D_0 , achieved when both descriptors are received, and the side distortion D_1 , achieved when only one descriptor is received, for different rates. It is clear that receiving an extra descriptor will improve the performance. We also see the effect

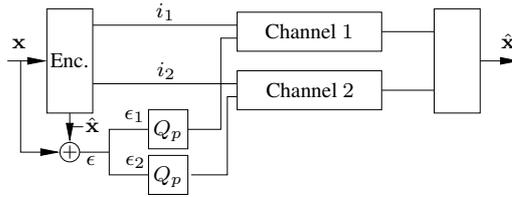


Figure 5: Proposed structure for improving the performance of D_0 .

that the permutation code stops to improve above a certain rate and in Figure 4(a) and this effect starts to become visible around $R = 4$. However, this 'saturation' level can be moved to a higher value of R by increasing N (see [10]).

We have also compared the proposed method with two well-known MDC methods: multiple description scalar quantization (MDSQ) [1] and entropy-constrained MDSQ (ECMDSQ) [2]. The comparison with MDSQ may seem a bit unfair since MDSQ use a small block size. However, using a much higher dimension in our scheme based on permutation coding will not necessarily give better performance and allowing for straightforward implementation of high-dimensional MDC is one of the main strengths of the suggested method. ECMDSQ is essentially a high-dimensional block coding scheme, since it (implicitly) assumes perfect entropy coding on top of scalar quantization. As shown in [11] the rate-distortion characteristics of permutation coding is equivalent to that of entropy coded scalar quantization, as $N \rightarrow \infty$. Hence, permutation coding is in this sense equivalent to entropy coded scalar quantization which makes the comparison of the suggested method to ECMDSQ relevant.

4.1 Introducing More Design Parameters

It is a well known fact that when performing two channel MDC there is a tradeoff between the central distortion D_0 and the side distortion D_1 such that if D_0 is decreased D_1 must be increased and vice versa (that is, assuming D_0 is close to the best theoretical performance for a given D_1 ; see e.g. [7, 8]). Although the previous simulation indicated good performance for the proposed system one of its drawbacks, in its present form, is that it cannot easily be adjusted to trace out the tradeoff between D_0 and D_1 , while this can be done when using MDSQ or ECMDSQ by adjusting some design parameters. Although we consider the task of developing such flexibility for our method to be a topic for future work, we here introduce a quite straightforward method to approach the tradeoff between central and side distortion. The proposed structure is shown in Figure 5 where the encoder works in the same fashion as in Section 3 with the additional feature that it also decodes the coded data to get $\hat{\mathbf{x}}$ based on i_1 and i_2 , that is the estimate produced when none of the descriptors are lost. This estimate is now compared to the original vector \mathbf{x} to create the error $\epsilon = \mathbf{x} - \hat{\mathbf{x}}$. The $N/2$ first samples of this error vector create a sequence ϵ_1 which is block quantized using a permutation code. This second permutation code will be designed using ϵ as training data. In the case that both descriptors are received we get $\hat{\mathbf{x}} = 0.5E[\mathbf{x}|i_1] + 0.5E[\mathbf{x}|i_2] + \hat{\epsilon}$. If only one descriptor is received, for instance i_1 , we use $\hat{\mathbf{x}} = E[\mathbf{x}|i_1] + \hat{\epsilon}_1$. Using the rate R_ϵ to quantize ϵ leaves the rate $R_{\hat{\mathbf{x}}} = R - R_\epsilon$ for quantizing \mathbf{x} . By increasing the value of

R_ϵ the performance in terms of D_0 will be improved. In fact, setting $R_\epsilon = R$ will divide \mathbf{x} into two blocks which are permutation coded and transmitted over one channel each.

In Figure 4(b) the result of simulating the modified system is shown when $R = 2$ bits/symbol/channel. The proposed method (pentagrams) outperforms MDSQ but does not beat ECMDSQ, except at the leftmost pentagram which corresponds to the case $R_\epsilon = 0$, that is, when only the approach of Section 3 is used. We consider this to indicate that the basic approach proposed in this paper is sound, while there may be further improvements possible upon the modified system in Figure 5 to introduce the desired flexibility. It is therefore our intention for further work to investigate alternative approaches to achieve efficient tradeoff between side and central distortion.

5 Conclusions

A new method for multiple description coding has been introduced and analyzed. The new approach was compared to two other well known techniques, and simulations indicated that for certain regions in the tradeoff between central distortion and side distortion the performance of the new scheme is comparable to the best of the benchmarks methods. However, in other regions the performance degrades slightly and as a topic for future work we intend to improve the new method in these regions.

Appendix

We will here derive an upper and lower bound on r^* for the example illustrated in Figure 2, hence $N = 2, J = 2, n_1 = n_2 = 1$. Finally, we will also discuss the lower and upper bounds for the general case. When doing so we will use hyperspherical coordinates such that

$$\begin{aligned} x_k &= r \cos \theta_k \prod_{l=1}^{k-1} \sin \theta_l \quad \forall k = 1, \dots, N-1 \\ x_N &= r \prod_{l=1}^{N-1} \sin \theta_l \end{aligned} \quad (12)$$

and the Jacobian determinant for this transformation is

$$J = r^{N-1} \prod_{p=1}^{N-1} (\sin \theta_p)^{N-p-1}. \quad (13)$$

As can be seen in Figure 2 the resulting intersection created by the received descriptions i_1 and i_2 can be described as $\mathcal{V}_\cap = \mathcal{V}_1 \cap \mathcal{V}_2 = \{\mathbf{x} : 0 \leq r, \alpha_1 \leq \theta \leq \alpha_2\}$, when using polar coordinates. When deriving r^* we can instead consider the area $\mathcal{V}'_\cap = \{\mathbf{x} : 0 \leq r, 0 \leq \theta \leq \alpha\}$, with $\alpha = \alpha_2 - \alpha_1$, since $f(\mathbf{x})$ is rotation invariant. Hence,

$$E[x_1 | \mathbf{x} \in \mathcal{V}'_\cap] = \frac{\int_{\mathcal{V}'_\cap} x_1 f(\mathbf{x}) d\mathbf{x}}{\int_{\mathcal{V}'_\cap} f(\mathbf{x}) d\mathbf{x}} = \frac{\int_0^\infty \int_0^\alpha r \cos \theta e^{-\frac{r^2}{2\sigma^2}} r d\theta dr}{\int_0^\infty \int_0^\alpha e^{-\frac{r^2}{2\sigma^2}} r d\theta dr} = \frac{\sin \alpha \sqrt{2}\sigma \Gamma(\frac{3}{2})}{\alpha \Gamma(\frac{2}{2})} \quad (14)$$

and similar equations gives

$$E[x_2 | \mathbf{x} \in \mathcal{V}'_\alpha] = \frac{1 - \cos \alpha}{\alpha} \frac{\sqrt{2}\sigma\Gamma(\frac{3}{2})}{\Gamma(\frac{2}{2})} \quad (15)$$

where $\Gamma(\cdot)$ is the Gamma function. From this we get r^* as

$$\|E[\mathbf{x} | \mathbf{x} \in \mathcal{V}'_\alpha]\| = \frac{\sqrt{2 - 2\cos \alpha}}{\alpha} \frac{\sqrt{2}\sigma\Gamma(\frac{3}{2})}{\Gamma(\frac{2}{2})} \quad (16)$$

which is a strictly decreasing function for $\alpha \in (0, \pi]$. Hence, the smaller intersection, the larger value of r^* . The upper bound on r^* is therefore given in the limit when $\alpha \rightarrow 0$. The lower bound is given from the case when the intersection is as large as possible, that is when $\mathcal{V}_1 = \mathcal{V}_2$, corresponding to the case when only one description is received. This gives

$$\sqrt{\mu_1^2 + \mu_2^2} \leq r^* \leq \frac{\sqrt{2}\sigma\Gamma(\frac{3}{2})}{\Gamma(\frac{2}{2})}. \quad (17)$$

In general we have shown that the upper bound on r^* is created by the smallest possible intersection, $\mathcal{V}_\epsilon = \{\mathbf{x} : 0 \leq r, 0 \leq \theta_i \leq \epsilon \forall i = 1..N-1\}$ when $\epsilon \rightarrow 0$. Using (12)–(13) this length is calculated below:

$$\begin{aligned} \lim_{\epsilon \rightarrow 0} \|E[\mathbf{x} | \mathbf{x} \in \mathcal{V}_\epsilon]\| &= \\ &= \frac{\int_0^\infty \int_0^\epsilon \dots \int_0^\epsilon r \cos \theta_1 e^{-\frac{r^2}{2\sigma^2}} r^{N-1} \prod_{p=1}^{N-1} (\sin \theta_p)^{N-p-1} d\theta_1 \dots d\theta_{N-1} dr}{\int_0^\infty \int_0^\epsilon \dots \int_0^\epsilon e^{-\frac{r^2}{2\sigma^2}} r^{N-1} \prod_{p=1}^{N-1} (\sin \theta_p)^{N-p-1} d\theta_1 \dots d\theta_{N-1} dr} = \\ &= \lim_{\epsilon \rightarrow 0} \frac{\int_0^\infty r^N e^{-\frac{r^2}{2\sigma^2}} dr}{\int_0^\infty r^{N-1} e^{-\frac{r^2}{2\sigma^2}} dr} \frac{\int_0^\epsilon \cos \theta_1 (\sin \theta_1)^{N-2} d\theta_1}{\int_0^\epsilon (\sin \theta_1)^{N-2} d\theta_1} = \\ &= \frac{\sqrt{2}\sigma\Gamma(\frac{N+1}{2})}{\Gamma(\frac{N}{2})} \end{aligned} \quad (18)$$

where l'Hospital's rule is used in the final step on the second part of the expression. For the lower bound we conject that it corresponds to the vector length when only one description is received, hence

$$\left(\sum_{i=1}^K n_i \mu_i^2 \right)^{\frac{1}{2}} \leq r^*. \quad (19)$$

(18)–(19) gives (9).

Acknowledgments

The authors wish to thank J. Jaldén and K. Werner for interesting and helpful discussions.

References

- [1] V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 821–834, 1993.
- [2] V. A. Vaishampayan and J. Domaszewicz, "Design of entropy-constrained multiple-description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 40, pp. 245–250, January 1994.
- [3] S. D. Servetto, V. A. Vaishampayan, and N. J. A. Sloane, "Multi description lattice vector quantization," in *Proceedings IEEE Data Compression Conference*, Snowbird, UT, March 1999, pp. 13–22.
- [4] V. K. Goyal, J. Kovacević, and M. Vetterli, "Multiple description transform coding: Robustness to erasures using tight frame expansion," in *Proceedings IEEE Int. Symp. Information Theory*, Cambridge, MA, August 1998, p. 408.
- [5] —, "Quantized frame expansions as source-channel codes for erasure channels," in *Proceedings IEEE Data Compression Conference*, Snowbird, UT, March 1999, pp. 326–335.
- [6] V. Goyal, M. Vetterli, and N. Thao, "Quantized overcomplete expansions in \mathbb{R}^n : analysis, synthesis, and algorithms," *IEEE Transactions on Information Theory*, vol. 44, pp. 16–31, January 1998.
- [7] L. Ozarow, "On a source coding problem with two channels and three receivers," *Bell Syst. Tech. J.*, vol. 59, no. 10, pp. 1909–1921, December 1980.
- [8] A. E. Gamal and T. Cover, "Achievable rates for multiple descriptions," *IEEE Transactions on Information Theory*, vol. 28, no. 6, pp. 851–857, November 1982.
- [9] D. Slepian, "Permutation modulation," *Proc. IEEE*, vol. 53, pp. 228–236, March 1965.
- [10] T. Berger, F. Jelinek, and J. Wolf, "Permutation codes for sources," *IEEE Transactions on Information Theory*, vol. 18, no. 1, pp. 160–169, January 1972.
- [11] T. Berger, "Optimum quantizers and permutation codes," *IEEE Transactions on Information Theory*, vol. 18, no. 6, pp. 759–765, November 1972.

Paper F

Improved Quantization in Multiple Description Coding by Correlating Transforms

Niklas Wernersson, Tomas Skölleremo and Mikael Skoglund

Published in *Proceedings IEEE Workshop on Multimedia Signal Processing 2004*

©2004 IEEE
The layout has been revised

Improved Quantization in Multiple Description Coding by Correlating Transforms

Niklas Wernersson, Tomas Skölleremo and Mikael Skoglund

Abstract

The objective with Multiple Description Coding (MDC) is to code one source of data into multiple bitstreams. The coding is done in such a way that multiple levels of quality is achieved. This means that even if one or a few of the bitstreams are lost, the received bits should make it possible to get an approximated version of the original data. One way to do this is to use pairwise correlating transforms which will introduce correlation between the bitstreams. This correlation can be used in order to get an estimate of a lost stream. In this paper a new approach for MDC using pairwise correlating transforms is presented. In this approach, contrary to previous work, quantization of the source data is performed after the data has been transformed. This makes it possible to improve the shape of the quantization cells and to tailor these to the employed transform. We demonstrate that this offers a substantial performance gain compared with previous approaches to MDC using pairwise correlating transforms.

1 Introduction

Packet networks have gained in importance in recent years, for instance by the wide-spread use of the Internet. By using these networks large amounts of data can be transmitted. When transmitting for instance an image a current network system typically uses the TCP protocol to control the transmission as well as the retransmission of lost packages. Unfortunately, packet losses can in general not be neglected and this problem therefore has to be considered when constructing a communication system. The compression algorithms in conventional systems quite often put quite a lot of faith into the delivery system which gives rise to some unwanted effects.

Suppose that N packets are used to transmit a compressed image and the receiver reconstructs the image as the packets arrive. A problem would arise if the receiver is dependent on receiving all the previous packets in order to reconstruct the data. For instance if packets $\{1, 3, 4, \dots, N\}$ are received it would be an undesirable property if only the information in packet 1 could be used until packet 2 eventually arrives. This would produce delays in the system and great dependency on the retransmission process. In the case of a real time system the use of the received packets may have been in vain because of a lost packet. One

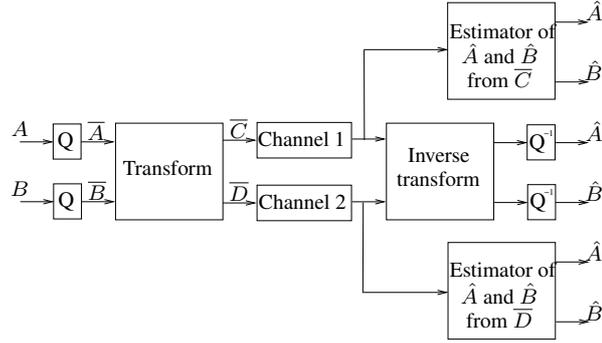


Figure 1: The basic structure of MDC using pairwise correlating transforms as presented in [1].

way to deal with this is to use Multiple Description Coding (MDC) where each received packet will increase the quality of the image no matter which other packets that have been received, see e.g. [1–8]. MDC will therefore enhance the reliability of the communication system.

In this paper a new approach for MDC using pairwise correlating transform is presented. In previous work, e.g. [1], the data is first quantized and then transformed. We suggest to reverse the order of these operations, leading to performance gains. The optimal cell shape of the transformed data relates to the optimal cell shape of the original data through some basic equations which makes it possible to perform quantization and designing the code-words after the data is transformed. Only the case with two descriptors will be considered but the theory can easily be extended to handle more descriptors. It is assumed that only one descriptor can be lost at a time (not both) and that the receiver knows when a descriptor is lost. The two channels are also assumed to have equal failure probability, p_{error} , and MSE is used as a distortion measure. The source signal is modelled as uncorrelated Gaussian distributed.

This paper is organized as follows. In Section 2 some preliminary theory of MDC using pairwise correlating transforms is discussed. In Section 3 the new approach for MDC using pairwise correlating transforms is presented. In Sections 4 and 5 some results and conclusions will be presented.

2 Preliminaries

Generally the objective with transform coding is to remove redundancy in the data in order to decrease the entropy. The goal of MDC is the opposite, namely to introduce redundancy in the data but in a controlled fashion. A quite natural approach for this is to first remove possible redundancy in the data by for instance using the Karhunen-Loeve transform. After this MDC is used in order to introduce redundancy again, but this time in selected amounts. In this paper it is assumed that the original data is uncorrelated Gaussian distributed so the

problem of removing initial redundancy will not be considered.

In Figure 1 the basic structure of the MDC described in [1] is shown. The data variables A and B are to be transmitted and are quantized into \bar{A} and \bar{B} . These values are then transformed using the transform

$$\begin{bmatrix} \bar{C} \\ \bar{D} \end{bmatrix} = \mathbf{T} \begin{bmatrix} \bar{A} \\ \bar{B} \end{bmatrix}, \quad (1)$$

where \mathbf{T} is a 2×2 matrix. This transform is invertible so that

$$\begin{bmatrix} \bar{A} \\ \bar{B} \end{bmatrix} = \mathbf{T}^{-1} \begin{bmatrix} \bar{C} \\ \bar{D} \end{bmatrix}. \quad (2)$$

Once the data have been transformed \bar{C} and \bar{D} are transmitted over two different channels. If both the descriptors are received the inverse transform from (2) is used in order to produce \hat{A} and \hat{B} . However, if one of the descriptors is lost, \hat{A} and \hat{B} can be estimated from the other descriptor. This comes from the fact the the transform matrix \mathbf{T} is nonorthogonal and introduces redundancy in the transmitted data. For instance, if the receiver receives only the descriptor \bar{C} , (\hat{A}, \hat{B}) is estimated to $E[(A, B)|\bar{C}]$.

For the two descriptors case the transform matrix \mathbf{T} , optimized according to [1], can be written as

$$\mathbf{T} = \begin{bmatrix} \cos \theta / \sqrt{\sin 2\theta} & \sin \theta / \sqrt{\sin 2\theta} \\ -\cos \theta / \sqrt{\sin 2\theta} & \sin \theta / \sqrt{\sin 2\theta} \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}. \quad (3)$$

where θ will control the amount of introduced redundancy.

The values \bar{C} and \bar{D} that are to be transmitted should be integers which is not necessarily the case in (1). Therefore the transform is implemented as follows (a, b, c and d are the values from (3) and $[\cdot]$ denotes rounding).

$$\bar{A} = \left\lfloor \frac{A}{A_{max}} q_A + 0.5 \right\rfloor, \quad \bar{B} = \left\lfloor \frac{B}{B_{max}} q_B + 0.5 \right\rfloor, \quad (4)$$

$$W = \bar{B} + \left\lfloor \frac{1+c}{d} \bar{A} \right\rfloor, \quad (5)$$

$$\bar{D} = [dW] - \bar{A}, \quad (6)$$

$$\bar{C} = W - \left\lfloor \frac{1-b}{d} \bar{D} \right\rfloor. \quad (7)$$

It is assumed that $A \in [0, A_{max}]$ and $B \in [0, B_{max}]$. q_A and q_B are integers deciding how many quantization levels there are for A and B respectively. It is also assumed, for the extremes, that $\left\lfloor \frac{0}{A_{max}} q_A + 0.5 \right\rfloor$ is rounded to 1 and $\left\lfloor \frac{A_{max}}{A_{max}} q_A + 0.5 \right\rfloor$ is rounded to q_A .

Assuming that both descriptors are received in the decoder the corresponding inverse transform is performed as

$$W = \bar{C} + \left\lfloor \frac{1-b}{d} \bar{D} \right\rfloor, \quad (8)$$

$$\bar{A} = [dW] - \bar{D}, \quad (9)$$

$$\bar{B} = W - \left\lfloor \frac{1+c}{d} \bar{A} \right\rfloor, \quad (10)$$

$$\hat{A} = \frac{(\bar{A} - 0.5)A_{max}}{q_A}, \quad \hat{B} = \frac{(\bar{B} - 0.5)B_{max}}{q_B}. \quad (11)$$

As mentioned before, if one of the descriptors is lost \hat{A} and \hat{B} are, depending on which descriptor that was lost, estimated to $E[(A, B)|\bar{C}]$ or $E[(A, B)|\bar{D}]$.

Note here that the number of quantization levels for \bar{A} and \bar{B} , q_A and q_B , will in general not equal the ones for \bar{C} and \bar{D} , q_C and q_D . (q_A, q_B) are however mapped to (q_C, q_D) by a function φ according to

$$\begin{aligned} \varphi : \mathbf{N}^2 &\longrightarrow \mathbf{N}^2, \\ \varphi(q_A, q_B) &= (q_C, q_D). \end{aligned} \quad (12)$$

Hence, if we want to transmit \bar{C} and \bar{D} using, e.g., 3 bits each we need to find q_A and q_B so that $\varphi(q_A, q_B) = (2^3, 2^3)$.

From (4) it is seen that the described MDC system in (4)–(11) uses uniform quantization. The system could easily be improved by introducing two nonuniform scalar quantizers, one for the A -values and one for the B -values. This improved system is what will be used and considered further on in this paper. This leads to modifications of (4) and hence also (11). Using the MSE as a distortion measure a codebook could be designed by using for instance the generalized Lloyd algorithm briefly explained in Section 3.

3 Improving the Quantization

In brief the algorithm in Section 2 can be summarized as

1. *Train encoder/decoder and quantize data.* The encoder uses two scalar quantizers in order to decrease the entropy of the data. This means that the data values are mapped onto a set of codevectors.
2. *Transform the quantized data.* Redundancy is introduced into the data by using (5)–(7).
3. *Transmit data.* The data is transmitted and packet or bit losses may occur, which means that some descriptors may be lost.
4. *Estimate lost data and do the inverse transform.* This is done using (8)–(10).

In this paper we suggest to do this algorithm in a different order. Changing the order of Steps 1 and 2 would mean that the transformation is done directly and training and quantization is done on the transformed values. Naturally, also the order in the receiver has to be reversed appropriately.

Using MSE as the distortion measure a point in the data is quantized to the K :th codevector according to

$$\begin{aligned} K &= \arg \min_k \left(\left[\begin{array}{c} A \\ B \end{array} \right] - \left[\begin{array}{c} \tilde{A}_k \\ \tilde{B}_k \end{array} \right] \right)^T \left(\left[\begin{array}{c} A \\ B \end{array} \right] - \left[\begin{array}{c} \tilde{A}_k \\ \tilde{B}_k \end{array} \right] \right) \\ &= \arg \min_k \left(\left[\begin{array}{c} \Delta A_k \\ \Delta B_k \end{array} \right] \right)^T \left[\begin{array}{c} \Delta A_k \\ \Delta B_k \end{array} \right] \right), \end{aligned} \quad (13)$$

where \tilde{A}_k and \tilde{B}_k are the coordinates of the different codewords. Using (2) this can also be written

$$\begin{aligned} K &= \arg \min_k \left(\mathbf{T}^{-1} \left(\begin{bmatrix} C \\ D \end{bmatrix} - \begin{bmatrix} \tilde{C}_k \\ \tilde{D}_k \end{bmatrix} \right) \right)^T \left(\mathbf{T}^{-1} \left(\begin{bmatrix} C \\ D \end{bmatrix} - \begin{bmatrix} \tilde{C}_k \\ \tilde{D}_k \end{bmatrix} \right) \right) \\ &= \arg \min_k \left(\mathbf{T}^{-1} \begin{bmatrix} \Delta C_k \\ \Delta D_k \end{bmatrix} \right)^T \left(\mathbf{T}^{-1} \begin{bmatrix} \Delta C_k \\ \Delta D_k \end{bmatrix} \right) \\ &= \arg \min_k \left(\begin{bmatrix} \Delta C_k \\ \Delta D_k \end{bmatrix}^T \mathbf{T}^{-1T} \mathbf{T}^{-1} \begin{bmatrix} \Delta C_k \\ \Delta D_k \end{bmatrix} \right). \end{aligned} \quad (14)$$

According to the discussion in Section 2 there should be q_C quantization levels for C and q_D quantization levels for D . Introducing this restriction in (14) and using (3) gives

$$(I, J) = \arg \min_{i,j} (\Delta C_i^2 + 2 \cos(2\theta) \Delta C_i \Delta D_j + \Delta D_j^2), \quad (15)$$

where $i \in \{1, 2, \dots, q_C\}$ and $j \in \{1, 2, \dots, q_D\}$. This equation will allow us to design a codebook for the transformed values instead of the original data. The generalized Lloyd algorithm can be used for this purpose. This algorithm is briefly summarized below.

1. Define initial codebook.
2. Quantize each data point to that codeword that minimizes the contribution to the distortion.
3. For each codeword (if it is possible), find a new optimal codeword for all the values that have been quantized to this particular codeword and update the codebook.
4. Until the algorithm converges go to Step 2.

For Step 2, (15) is used to quantize the data. In Step 3 we want to find an optimal codeword for those values that have been quantized to a particular codeword. Calculating the partial derivative of the total distortion as

$$\frac{\partial}{\partial \tilde{C}_I} \sum_{(C,D)} (\Delta C_i^2 + 2 \cos(2\theta) \Delta C_i \Delta D_j + \Delta D_j^2) \quad (16)$$

and minimizing by setting (16) equal to zero will give an equation for updating the codewectors, namely

$$\tilde{C}_I = \frac{1}{N_I} \sum_{\forall (C,D): Q(C,D)=(\tilde{C}_I, \tilde{D}_j)} (C + \cos(2\theta) \Delta D_j). \quad (17)$$

The sum is taken over all those points (C, D) which will be quantized to $(\tilde{C}_I, \tilde{D}_j)$ for a given I and an arbitrary j . N_I is the number of points within this set. In a similar manner we get

$$\tilde{D}_J = \frac{1}{N_J} \sum_{\forall (C,D): Q(C,D)=(\tilde{C}_i, \tilde{D}_J)} (D + \cos(2\theta) \Delta C_i) \quad (18)$$

and this is done for $I = 1, 2, \dots, q_C$ and $J = 1, 2, \dots, q_D$. Once the codebook has been generated the encoder and decoder are ready to use. The data to be transmitted is then transformed by the matrix \mathbf{T} , quantized and transmitted. In the decoder the reverse procedure is done. This is illustrated in Figure 2.

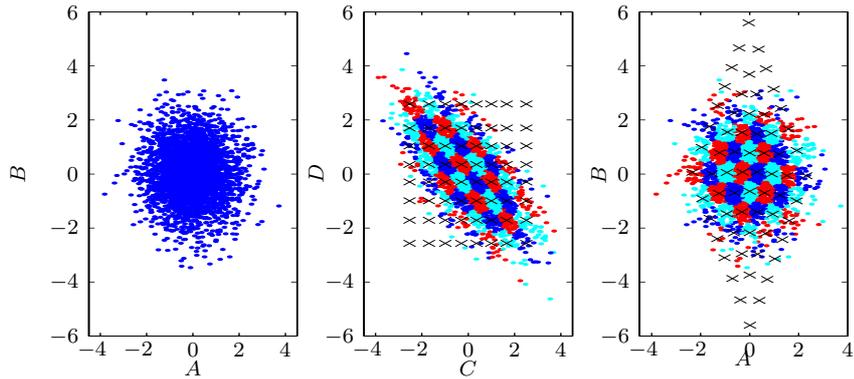


Figure 2: In the left plot the original set of data is shown. These values are first transformed and then quantized as shown in the middle plot. In the receiver the inverse transform is used as shown in the right plot. In this plot also the corresponding quantization cells are illustrated.

4 Simulation Results

In order to compare the system explained in Section 2 and [1] with the new system introduced in Section 3 these were implemented and simulated. Uncorrelated zero mean Gaussian data was generated and used to train the encoders/decoders and then to simulate the systems. In the simulations presented here the source data A and B have equal variances. Similar results have however been obtained also for the case of nonequal variances. As mentioned in Section 1 it is assumed that only one descriptor can be lost at a time and that the receiver knows when a descriptor is lost. The angle for the transform matrix \mathbf{T} used in the simulations was $\theta = \frac{\pi}{5}$. The result is presented in Figure 3. p_{error} shows the probability that one of the descriptors is lost and the y -axis shows the signal-to-distortion ratio, defined as $10 \log \frac{E[x^2]}{E[(x-\hat{x})^2]}$, where x is the data signal and \hat{x} is the reconstructed signal. In Figure 3(a) both \bar{C} and \bar{D} were transmitted using 3 bits each which gives $q_C = q_D = 2^3$. In order to accomplish this (q_A, q_B) had to be identified so that $\varphi(q_A, q_B) = (2^3, 2^3)$. This was found to be true for $q_A = 5$ and $q_B = 7$. Similar results are shown in Figures 3(b) and 3(c) when using 4 and 8 bits.

As can be observed in Figure 3 the new system outperforms the original system for all investigated values of p_{error} . In the case of 3 bits per description, as shown in Figure 3(a), the advantage of the new scheme is more noticeable at low packet loss rates. In particular we see that as $p_{error} \rightarrow 0$ the new system outperforms the original scheme by about 2 dB. When using 4 bits per description, as in Figure 3(b), we notice that the gain of the new approach is more-or-less constant over the range of different packet loss rates. Finally, studying Figure 3(c), we can observe that in the case of 8 bits per description the situation has changed and the gain is now more prominent at high packet error rates. In summary we see that in all cases considered there is a constant gain at medium to high packet loss rates and this gain increases with the transmission rate of the system, while at

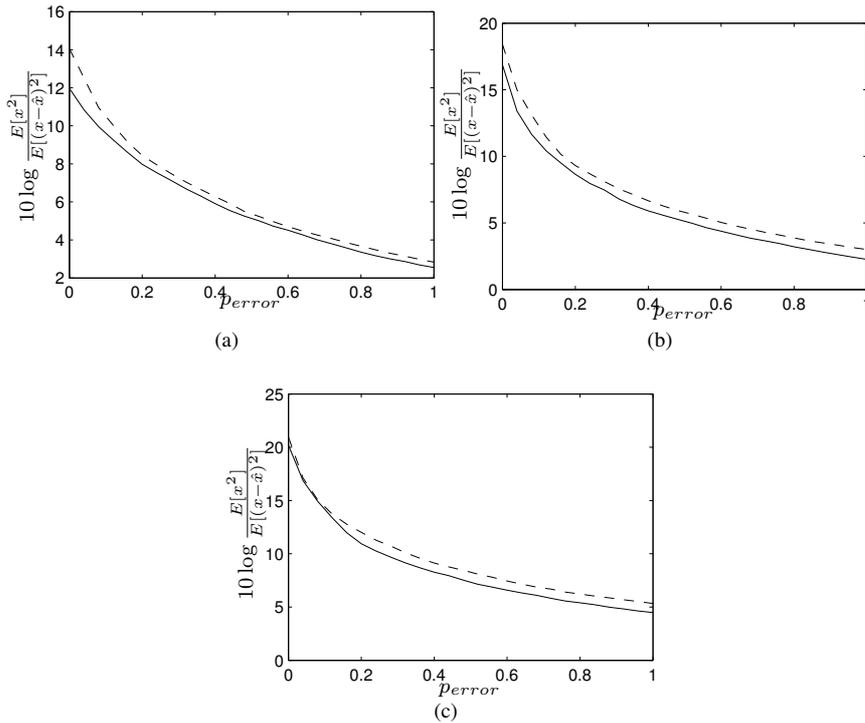


Figure 3: The solid line shows the performance of the original system [1] and the dashed line shows that of the new system, in terms of signal-to-distortion ratio versus packet loss rate, p_{error} . \bar{C} and \bar{D} are transmitted using (a) 3, (b) 4 and (c) 8 bits each and $\theta = \frac{\pi}{5}$.

low packet loss rates there is an additional gain at low rates (as in Figure 3(a)) and hardly no gain at high rates (as in Figure 3(c)). One possible explanation for this behavior is that the new approach in particular improves the performance at low transmission and packet loss rates due to the improved optimization of the individual quantizers. At high loss rates this gain is less pronounced, since when packet losses occur the redundancy introduced by the linear transform has an equal or higher influence on the total performance than has the performance of the individual quantizers.

5 Conclusions

A new MDC method has been introduced. The method is developed from an extended version of the MDC using pairwise correlating transforms described in [1]. Using the original method the data is quantized and then transformed by a matrix operator in order to

increase the redundancy between descriptors. In the new suggested method the data is first transformed and then quantized. In Section 3 it is shown that this transform leads to a modification of the distortion measure. Using the generalized Lloyd algorithm when designing the quantization codebook also leads to a new way to update the codevectors. In section 4 simulations were done that shows that the new method performs better than the original one when smaller amounts of redundancy are introduced into the transmitted data. For the simulations conducted in Section 4, using $\theta = \frac{\pi}{5}$, the new method gave 2 dB gain compared to the original system when no descriptors were lost. The gain decreased to about 0.5-1 dB when the probability of lost descriptors was increased.

References

- [1] Y. Wang, M. Orchard, V. Vaishampayan, and A. Reibman, "Multiple description coding using pairwise correlation transforms," *IEEE Transactions on Image Processing*, vol. 10, no. 3, March 2001.
- [2] V. K. Goyal and J. Kovacević, "Generalized multiple description coding with correlating transforms," *IEEE Transactions on Information Theory*, vol. 47, no. 6, pp. 2199–2224, sep 2001.
- [3] V. K. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Processing Magazine*, vol. 18, pp. 74–93, sep 2001.
- [4] V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 821–834, 1993.
- [5] S. Pradhan, R. Puri, and K. Ramchandran, " n -channel symmetric multiple descriptions-part i: (n, k) source-channel erasure codes," *IEEE Transactions on Information Theory*, vol. 50, no. 1, pp. 47–61, January 2004.
- [6] P. Koulgi, S. Regunathan, and K. Rose, "Multiple description quantization by deterministic annealing," *IEEE Transactions on Information Theory*, vol. 49, no. 8, pp. 2067 – 2075, August 2003.
- [7] R. Venkataramani, G. Kramer, and V. Goyal, "Multiple description coding with many channels," *IEEE Transactions on Information Theory*, vol. 49, no. 9, pp. 2106–2114, September 2003.
- [8] P. Dragotti, S. Servetto, and M. Vetterli, "Optimal filter banks for multiple description coding: analysis and synthesis," *IEEE Transactions on Information Theory*, vol. 48, no. 7, pp. 2036 – 2052, July 2002.

Paper G

On Source Decoding Based on Finite-Bandwidth Soft Information

Niklas Wernersson and Mikael Skoglund

Published in *Proceedings of IEEE International Symposium on Information Theory 2005*

©2005 IEEE
The layout has been revised

On Source Decoding Based on Finite-Bandwidth Soft Information

Niklas Wernersson and Mikael Skoglund

Abstract

Designing a communication system using joint source–channel coding in general makes it possible to achieve a better performance than when the source and channel codes are designed separately, especially under strict delay-constraints. The majority of work done in joint source-channel coding uses a discrete channel model, corresponding to an analog channel in conjunction with a hard decision modulation scheme. The performance of such a system can however be improved by using soft decision modulation. The main cost is a higher decoding complexity. An alternative is to quantize the soft information and store the pre-calculated soft decision values in a lookup table. In this paper we propose new methods for quantizing soft channel information, to be used in conjunction with soft-decision source decoding. We achieve a performance close to that of a system using unquantized soft information.

1 Introduction

Wireless distribution of multimedia has gained in importance in recent years. This has led to a number of different methods and standards, suitable for transmission, when representing audio, video, speech and images. A question of great interest is how to make the tradeoff between designing efficient source codes and designing channel codes robust toward transmission errors. This has made the area of joint source–channel coding of particular interest since it in general makes it possible to get a higher performance than if the source and channel codes are designed separately. Most of the work done in joint source-channel coding uses a discrete channel model, that is an analog channel in conjunction with a hard decision modulation scheme. Using soft decision modulation can however improve the performance of such a system.

There are several previous works that investigate source coding with decoding based on different kinds of soft decision modulation, e.g. [1–5]. While [1–3] assumed unquantized soft channel output-values, the work in [4, 5] was based on quantized soft information. More precisely, [4, 5] presented a scheme based on channel optimized vector quantization (COVQ), see e.g. [6–8], over a binary-input Gaussian channel and using uniform scalar quantization to quantize the soft channel outputs, resulting in a discrete memoryless channel

with two inputs and 2^q outputs. The main motivation in [4, 5] for working with quantized soft information was to reduce the decoding complexity compared e.g. with the decoder presented in [2], since over a discrete channel decoding can be based on a look-up table.

The present work is partly an extension of [4, 5] to using non-uniform quantization and vector quantization (VQ) to represent soft channel information. Our main motivation is however a scenario where the soft channel output information has to be bandlimited in order to be further conveyed to a node where it is used in soft source decoding. Consider, for example, applications where the transmitting node connects to a wired network via wireless access to a basestation, and where the receiving entity is connected to the wired network. The basestation will then have to limit the resolution of the received soft information in order to transport it to the receiving node. In practice the wired part of the network can often be assumed to be error-free, since it is implemented using high-reliability copper cables or optical fiber. Still, the receiving node may use soft information transported from the receiving basestation to counteract the effects of noise and signal fading in the wireless part of the connection.

Referring to a generic scenario similar to the one described above, we discuss two main issues: *a)* How to represent the soft information available at the basestation, the main alternatives being to re-quantize soft source estimates or to quantize the soft channel outputs directly, and; *b)* In the case of quantized soft channel values, different means of quantization, including non-uniform scalar quantization and vector quantization.

The paper is organized as follows. In Section 2 the problem is illustrated and explained. Three different approaches for transmitting bandlimited soft information is suggested. In Section 3 these three approaches are analyzed. An expression for scalar quantization of soft channel values is also derived. In Sections 4 and 5 some results and conclusions will be presented.

2 Problem formulation

We are considering the problem illustrated in Figure 1. A random vector $\mathbf{X} \in \mathbb{R}^d$ is vector quantized and transmitted over a wireless channel denoted Channel 1. The channel is modeled as a Gaussian channel and the transmission is done using BPSK. The m output bits from the encoder, represented by $\mathbf{i} \in \{-1, +1\}^m$, are corrupted by the noise $\mathbf{n} \in \mathbb{R}^m$. Hence, Receiver 1 measures $\mathbf{r} \in \mathbb{R}^m$ according to

$$\mathbf{r} = \mathbf{i} + \mathbf{n} \quad (1)$$

where \mathbf{n} is zero mean white Gaussian noise with covariance matrix $C_n = \sigma^2 I_n$, where I_n is the n -unity matrix.

Receiver 1 wishes to pass on the data represented by \mathbf{r} to an other unit, Receiver 2, and this is assumed done over an error-free connection, e.g. via optical fiber. The link from Receiver 1 to 2 is assumed to be of high but finite bandwidth, and the transmission needs therefore to be limited to qm bits per received \mathbf{r} . This means that a vector $\mathbf{j} \in \{0, 1\}^{qm}$ will be transmitted and finally received by Receiver 2. In Receiver 2 \mathbf{j} is used in order to get $\tilde{\mathbf{X}} = \tilde{\mathbf{X}}_{\mathbf{j}}$ which is an estimate of \mathbf{X} .

The encoder can be designed using source optimized vector quantization (SOVQ) where the mean-square error (MSE) between the source data \mathbf{X} and the quantized source data represented by \mathbf{i} is minimized, based e.g. on the generalized Lloyd algorithm [9]. However,

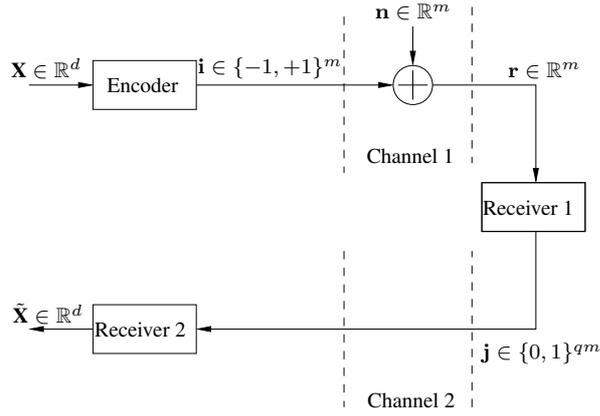


Figure 1: \mathbf{X} is transmitted to Receiver 1 using m bits. The bits are corrupted by the noise \mathbf{n} . In Receiver 1 the soft information contained in \mathbf{r} has to be bandlimited to qm bits, represented by \mathbf{j} , which are transmitted to Receiver 2. Finally $\tilde{\mathbf{X}}$ is estimated from \mathbf{j} .

since the transmission will constitute a random Markov-chain, the overall performance can be improved using COVQ. In the COVQ approach, we know from e.g. [6–8] that for a fixed encoder, represented by the encoder regions $\{\mathcal{S}_i\}$, such that $\mathbf{X} \in \mathcal{S}_i \implies$ transmit index \mathbf{i} , the optimal reconstruction value when \mathbf{j} is received is

$$\tilde{\mathbf{X}}_{\mathbf{j}} = E[\mathbf{X}|\mathbf{j}]. \quad (2)$$

That is, for a fixed encoder these vectors should be used for decoding. Furthermore, when the decoder is known and fixed the encoder regions that minimize the overall distortion are

$$\mathcal{S}_i = \{\mathbf{x} : \sum_{\mathbf{j}} P(\mathbf{j}|\mathbf{i}) \|\mathbf{x} - \tilde{\mathbf{X}}_{\mathbf{j}}\|^2 \leq \sum_{\mathbf{j}} P(\mathbf{j}|\mathbf{i}') \|\mathbf{x} - \tilde{\mathbf{X}}_{\mathbf{j}}\|^2, \forall \mathbf{i}'\}. \quad (3)$$

The expressions (2) and (3) together specify local optimality of a COVQ and by training the system in an iterative fashion using these equations the encoder and decoder can be designed [6–8]. Both SOVQ and COVQ will be investigated for the system described in Figure 1.

Our main interest in this paper concerns how to best make use of the soft information contained in \mathbf{r} , under the constraint that Receiver 1 has to compress \mathbf{r} into the finite-bandwidth representation \mathbf{j} . Three different approaches are suggested and explained in the following subsections.

2.1 Re-Quantization of a Soft Source Estimate

From estimation theory it is a well known fact that the optimal decoder that minimizes the MSE distortion between the decoded value $\tilde{\mathbf{X}}$, estimated from the unquantized soft

information in \mathbf{r} , and the original data \mathbf{X} is given as

$$\hat{\mathbf{X}} = E[\mathbf{X}|\mathbf{r}] = \int \mathbf{x} f_{\mathbf{x}|\mathbf{r}}(\mathbf{x}|\mathbf{r}) d\mathbf{x}. \quad (4)$$

Based on this, one way to implement the system in Figure 1 is as shown in Figure 2.

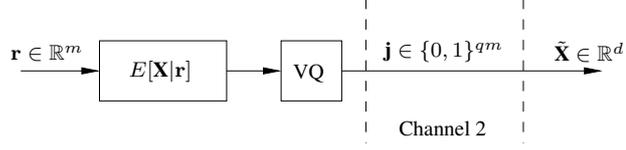


Figure 2: Structure of the first approach. All the soft information is used when calculating $E[\mathbf{X}|\mathbf{r}]$. This value is then vector quantized in order to meet with the condition of limited bandwidth at Channel 2.

In this approach all the soft information in \mathbf{r} is used in order to calculate the soft value $E[\mathbf{X}|\mathbf{r}]$. This value is then vector quantized, in order to meet with the condition of a limited bandwidth, in such a way that the distortion of the reconstructed value is minimized. Additional distortion will hence be introduced in the final reconstructed value $\tilde{\mathbf{X}}$, compared with $\hat{\mathbf{X}}$.

2.2 Vector Quantization of the Soft Channel Values

Here we propose to quantize the “soft bits” represented by the components of \mathbf{r} , in such a way that the distortion between \mathbf{X} and $\tilde{\mathbf{X}}$ is minimized. The basic structure is shown in Figure 3. This leads to that \mathbf{r} is quantized into one of 2^{qm} quantization cells. The index of this quantization cell is transmitted over Channel 2 and in Receiver 2 the reconstruction value $\tilde{\mathbf{X}}$ is computed.

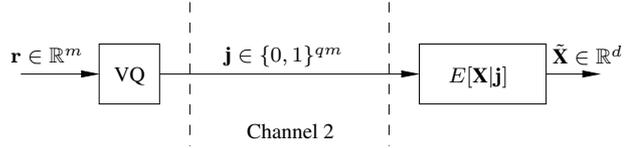


Figure 3: Structure of the second approach. The soft information in \mathbf{r} is vector quantized and transmitted over Channel 2. In the final receiver $\tilde{\mathbf{X}}$ is calculated as $\tilde{\mathbf{X}} = E[\mathbf{X}|\mathbf{j}]$.

2.3 Scalar Quantization of Soft Channel Values

Finally, a less complex approach than the one in Section 2.2 is suggested and illustrated in Figure 4. This is basically a simplified version of the previous method with the difference being the use of m scalar quantizers instead of an m -dimensional vector quantizer. We can expect this structure to have a lower performance than the previous one, since scalar quantization gives a lower freedom than vector quantization in designing the quantization cells. However, as it turns out, the complexity is substantially lower. Scalar quantization of soft channel outputs has previously been proposed in [4, 5].

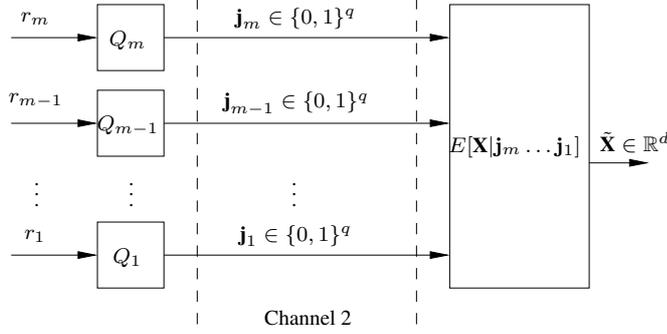


Figure 4: Structure of the third approach. The soft information in \mathbf{r} is quantized using m scalar quantizers. In the final receiver $\tilde{\mathbf{X}}$ is calculated as $\tilde{\mathbf{X}} = E[\mathbf{X} | \mathbf{j}_m \dots \mathbf{j}_1]$.

3 Implementation of the Different Approaches

In order to optimize the performance of the system introduced in Figure 1 we want to minimize the MSE between \mathbf{X} and $\tilde{\mathbf{X}}$. By also using $\hat{\mathbf{X}}$ from (4) we get

$$\begin{aligned}
 \min_{\tilde{\mathbf{X}}} E \|\mathbf{X} - \tilde{\mathbf{X}}\|^2 &= \min_{\tilde{\mathbf{X}}} E \|\mathbf{X} - \hat{\mathbf{X}} + \hat{\mathbf{X}} - \tilde{\mathbf{X}}\|^2 \\
 &= \min_{\tilde{\mathbf{X}}} (E \|\mathbf{X} - \hat{\mathbf{X}}\|^2 + 2E[(\mathbf{X} - \hat{\mathbf{X}})^T (\hat{\mathbf{X}} - \tilde{\mathbf{X}})] + E \|\hat{\mathbf{X}} - \tilde{\mathbf{X}}\|^2) \\
 &= \min_{\tilde{\mathbf{X}}} E \|\hat{\mathbf{X}} - \tilde{\mathbf{X}}\|^2
 \end{aligned} \tag{5}$$

where the minimum is over all parameters in the system that determine how $\tilde{\mathbf{X}}$ depends on \mathbf{X} . In the final step in (5), the first term disappeared since it does not depend on $\tilde{\mathbf{X}}$. The

second term equals zero since

$$\begin{aligned} E[(\mathbf{X} - \hat{\mathbf{X}})^T (\hat{\mathbf{X}} - \tilde{\mathbf{X}})] &= E[E[(\mathbf{X} - \hat{\mathbf{X}})^T (\hat{\mathbf{X}} - \tilde{\mathbf{X}}) | \mathbf{r}]] \\ &= E[(\hat{\mathbf{X}} - \tilde{\mathbf{X}})^T E[(\mathbf{X} - \hat{\mathbf{X}}) | \mathbf{r}]] \\ &= E[(\hat{\mathbf{X}} - \tilde{\mathbf{X}})^T \mathbf{0}] = \mathbf{0} \end{aligned} \quad (6)$$

Note that (5) implies that minimizing the MSE between \mathbf{X} and $\hat{\mathbf{X}}$ is actually equivalent to minimizing the MSE between $\hat{\mathbf{X}}$ and $\tilde{\mathbf{X}}$.

3.1 Re-Quantization of a Soft Source Estimate

Based on (5) we see that the quantizer in Figure 2 should be constructed in such a way that $E[\|\hat{\mathbf{X}} - \tilde{\mathbf{X}}\|^2]$ is minimized. This means that the quantizer can be trained just as in regular VQ training where $E[\|\hat{\mathbf{X}} - \tilde{\mathbf{X}}\|^2]$, with $\hat{\mathbf{X}} = \hat{\mathbf{X}}(\mathbf{r}) = E[\mathbf{X} | \mathbf{r}]$, is the distortion measure and $\tilde{\mathbf{X}}$ is encoded as

$$J = \arg \min_j (E[\|\hat{\mathbf{X}} - \tilde{\mathbf{X}}_j\|^2 | \mathbf{r}]) \quad (7)$$

where J denotes the (random) integer representation of the binary index \mathbf{j} . When implementing (7) it is assumed that a decoder $\tilde{\mathbf{X}} = \tilde{\mathbf{X}}_j$, when $J = j$, is fixed. For a fixed encoder, represented by (7), optimal decoding is done according to

$$\tilde{\mathbf{X}}_j = E[\hat{\mathbf{X}} | J = j]. \quad (8)$$

From these equations it is seen that any given distribution of \mathbf{X} together with a given σ^2 will generate a distribution of $\hat{\mathbf{X}}$ which will decide how the quantizer should be designed.

3.2 Vector Quantization of the Soft Channel Values

In this approach we want to vector quantize the received \mathbf{r} into one of 2^{qm} different representations \mathbf{r}_j . After the quantization the index J is transmitted over the channel and the reconstruction value $\hat{\mathbf{X}}(\mathbf{r}_j) = E[\mathbf{X} | \mathbf{r}_j]$ is calculated. The problem is to find the different \mathbf{r}_j 's such that $E[\|\hat{\mathbf{X}} - \tilde{\mathbf{X}}\|^2]$, with $\tilde{\mathbf{X}} = \tilde{\mathbf{X}}_j = \hat{\mathbf{X}}(\mathbf{r}_j)$, is minimized. For an observed value of \mathbf{r} , the encoding should hence be done according to

$$J = \arg \min_j E[\|\hat{\mathbf{X}}(\mathbf{r}) - \hat{\mathbf{X}}(\mathbf{r}_j)\|^2 | \mathbf{r}] = \arg \min_j E[\|\hat{\mathbf{X}}(\mathbf{r}) - \tilde{\mathbf{X}}_j\|^2 | \mathbf{r}]. \quad (9)$$

This is the same encoder as in (7), which means that the two approaches, in this sense, are equivalent. That is, it does not matter whether we consider encoding \mathbf{r} into j as a problem in the " $\tilde{\mathbf{X}}$ domain" or in the " \mathbf{r} domain." This further means that the third approach, defined in Figure 4, can not have a better performance than the system in Figure 2.

The structure of the quantization cells obtained based on (9) is illustrated in Figure 5 for a simple example with $m = 2$, $q = 2$ and $d = 1$. For demonstration purposes, the example assumes that X is discrete and uniformly distributed over $\{1, 2, 3, 4\}$, and these four values are transmitted as $\mathbf{i} \in \{(\pm 1, \pm 1)\}$. The noise variance is $\sigma^2 = 0.5$. As can be seen in this figure the shape of the quantization cells is quite complex. It is interesting to note that different \mathbf{r} 's can be quantized to the same cell although they are quite far from each other. For instance we see that $(r_1, r_2) = (0, -1.5)$ and $(r_1, r_2) = (-1, 1)$ will be quantized to the same cell. This effect comes from the soft decoding. Since $(r_1, r_2) = (0, -1.5)$ lies just between two possible codewords the reconstruction value is essentially chosen as the mean of these.

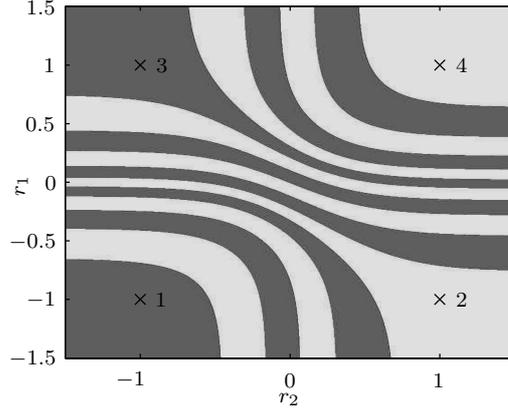


Figure 5: The shape of quantization cells obtained based on (9) when $m = 2$, $q = 2$ and $d = 1$. The different X -values are $\{1, 2, 3, 4\}$, with equal probabilities, and the noise variance $\sigma^2 = 0.5$.

3.3 Scalar Quantization of Soft Channel Values

As previously stated this method will have a lower performance than the ones suggested in Sections 2.1 and 2.2. However, the complexity is substantially lower which still makes the method interesting. In order to minimize $E[\|\hat{\mathbf{X}} - \tilde{\mathbf{X}}\|^2]$, in Figure 4, when quantizing r_k to j_k the encoding should be done according to

$$J_k = \arg \min_{j_k} E[\|\hat{\mathbf{X}}(r_m, \dots, r_1) - \tilde{\mathbf{X}}(j_m, \dots, j_1)\|^2 | r_k, j_k] \quad (11)$$

Also, for fixed encoders the optimal reconstructed value $\tilde{\mathbf{X}}(j_m, \dots, j_1)$ is obtained as

$$\tilde{\mathbf{X}}(j_m, \dots, j_1) = E[\mathbf{X} | j_m, \dots, j_1]. \quad (12)$$

The expressions in (11)–(12) make it possible to train the quantizers in an iterative fashion. It is readily seen that implementing (11) requires knowledge of the pdf for $\hat{\mathbf{X}}$. Otherwise the expected value needs to be estimated from a training set of data. This problem is also encountered when calculating (12).

In the next section optimal design of the scalar quantizers will be compared to the following alternatives:

- *Hard decision.* This is the classical hard-decision source decoding approach, where a hard decision is made on all r_i 's such that a negative r_i gives -1 and vice versa.
- *The Lloyd approach.* A quite intuitive way to quantize the data is to minimize the MSE of the reconstructed \mathbf{r}_i . That is, to design \mathbf{r}_i such that $E[\|\mathbf{r} - \mathbf{r}_i\|^2]$ is minimized. However, when discussing Figure 5 it was concluded that the nearest neighbor criterion is not necessarily suitable, due to the complex shape of the cells based on (7). In the Lloyd approach, we base the decoding on (12), meaning that

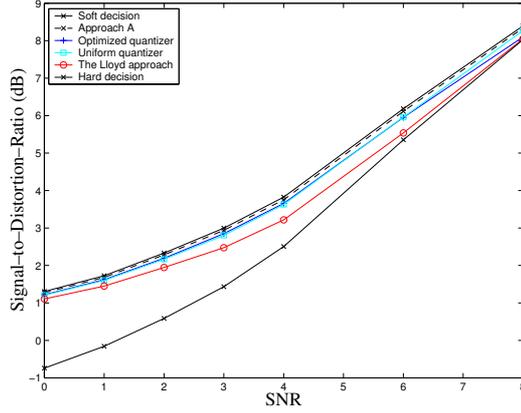


Figure 6: Results of the SOVQ system when \mathbf{X} is 2-dimensional Gaussian uncorrelated, $m = 4$ and $q = 2$. As can be seen the performance gain from using the optimal scalar quantizers over the uniform quantizers is minor.

knowledge about the pdf or, otherwise, a training set is required in order to estimate $\tilde{\mathbf{X}}(j_m \dots j_1)$.

- *Uniform quantizer.* Soft-decision source decoding based on uniform quantizers was proposed in [4, 5], and the idea is to construct uniform quantizers such that the mutual information between r_i and j_i is maximized. This is done for a few different channel models in [5] and some tables containing optimal stepsizes of uniform quantizers using this approach are presented for the AWGN case which will be used for comparison in our simulations. Then (12) is used for calculating $\tilde{\mathbf{X}}(j_m \dots j_1)$. Hence, also this approach requires knowledge about the pdf.

4 Simulation results

Different approaches for implementing the system in Figure 1 were introduced and analyzed in Sections 2–3. The system was implemented both for the SOVQ and the COVQ case in order to study the relation between channel SNR, defined as $E[\mathbf{i}^T \mathbf{i}] / E[\mathbf{n}^T \mathbf{n}] = 1/\sigma^2$, and signal-to-distortion-ratio (SDR) defined as $E\|\mathbf{X}\|^2 / E\|\mathbf{X} - \tilde{\mathbf{X}}\|^2$. The results of these simulations are presented below.

4.1 SOVQ

By using a training set of 2-dimensional Gaussian uncorrelated vectors, the encoder was designed such that \mathbf{X} is quantized using $m = 4$ bits. The soft channel values were quantized using $q = 2$ bits. In this case, the corresponding encoder output distribution is close

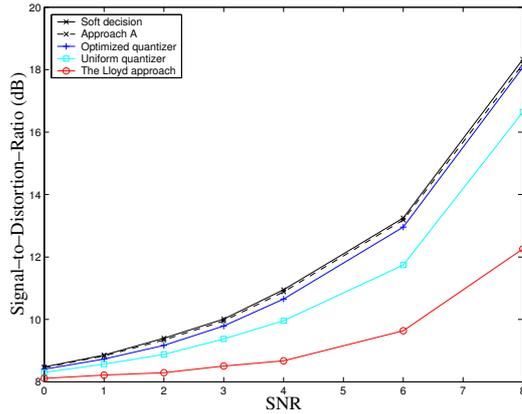


Figure 7: Results of the SOVQ system when \mathbf{X} is constructed as a 2-dimensional Gaussian Mixture model, $m = 4$ and $q = 2$. As can be seen the performance gain from using the optimal scalar quantizers over the uniform quantizers is now substantial, especially for high SNR's.

to uniform. Our simulations indicate that when \mathbf{i} is close-to uniformly distributed, the performance gain from using the optimal scalar quantizers over the uniform quantizers used in [5] is minor. Part of the explanation is that a uniform distribution will give a symmetric shape of the quantization cells resulting from (9).

Both these methods however have a significant performance gain compared to the Lloyd approach, and both methods have a quite small performance loss compared to 'Approach A' suggested in Section 2.1 (which was shown to be superior to all the other methods). These results are shown in Figure 6 where also the results for hard decisions and using the complete soft information ($q = \infty$) are shown.

In Figure 7 a similar simulation is conducted, however with the distribution of \mathbf{X} constructed as a Gaussian Mixture Model, that is the pdf is formed from a sum of different 2-dimensional Gaussian distributions. This results in a nonuniform distribution for \mathbf{i} which makes it more beneficial to use the optimal scalar quantizer design, which is clearly indicated in the plot. This comes from the fact that the unequal probabilities of \mathbf{i} now will create a less symmetric pdf of the received data \mathbf{r} . Hence, the choice of using a symmetric uniform quantizer centered around $r_i = 0$ is no longer as appropriate as in the previous case.

4.2 COVQ

We also wanted to see how the new methods perform when being used in a COVQ system matched to the present SNR. The authors suspected that due to COVQ's ability to insert extra error protection, especially at low SNR's, by creating a nonuniform distribution of \mathbf{i} would result in a gain for the optimized quantizers when compared to the uniform. However, our results so far indicate that there is no substantial difference in performance between the

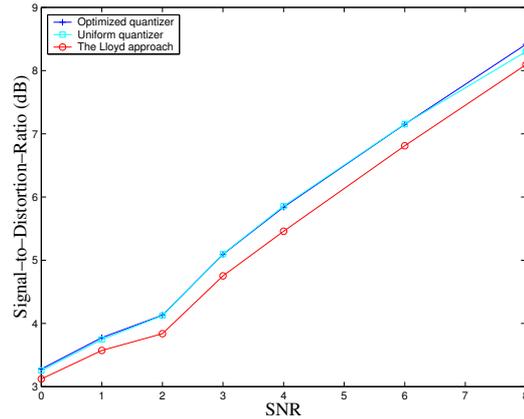


Figure 8: Results of the COVQ system when \mathbf{X} is 2-dimensional Gaussian uncorrelated vectors, $m = 4$ and $q = 2$. As can be seen the performance gain from using the optimal scalar quantizers over the uniform quantizers is insignificant.

optimal scalar quantizers and the uniform quantizers. This is demonstrated in Figure 8 where \mathbf{X} is 2-dimensional Gaussian uncorrelated vectors, using $m = 4$ and $q = 2$. One possible explanation to this result is that COVQ's property to remove codewords is most prominent at low SNR's. But in this region we see from Figures 6–7 that the gain of using the optimal quantizers is quite small. For higher SNR's the gain for the optimal quantizer is greater but at the same time the COVQ also makes the distribution of \mathbf{i} more uniform which removes this effect.

5 Conclusions

The issue on how to best construct finite-bandwidth representations of soft information has been studied. We studied three main approaches: 1) re-quantization of the soft source estimate; 2) vector quantization of the soft channel values, and; 3) scalar quantization of soft channel values. We showed analytically that 1) and 2) are essentially equivalent. Also, since 3) is a special case of 2) it can only yield similar or worse performance. However, we derived expressions that specify the optimal scalar quantizers, and when using designs based on these a performance close to that of approaches 1) and 2) was achieved. The gain of this suboptimality is a substantially lower complexity which makes the method interesting.

References

- [1] V. Vaishampayan and N. Farvardin, "Joint design of block source codes and modulation signal sets," *IEEE Transactions on Information Theory*, vol. 38, no. 4, pp. 1230–1248,

- July 1992.
- [2] M. Skoglund and P. Hedelin, "Hadamard-based soft decoding for vector quantization over noisy channels," *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 515–532, March 1999.
 - [3] M. Skoglund, "Soft decoding for vector quantization over noisy channels with memory," *IEEE Transactions on Information Theory*, vol. 45, no. 4, pp. 1293–1307, May 1999.
 - [4] F. Alajaji and N. Phamdo, "Soft-decision COVQ for Rayleigh fading channels," *IEEE Communications Letters*, vol. 2, no. 6, pp. 162–164, June 1998.
 - [5] N. Phamdo and F. Alajaji, "Soft-decision demodulation design for COVQ over white, colored and ISI Gaussian channels," *IEEE Transactions on Communications*, vol. 48, no. 9, pp. 1499–1506, September 2000.
 - [6] H. Kumazawa, M. Kasahara, and T. Namekawa, "A construction of vector quantizers for noisy channels," *Electronics and engineering in Japan*, vol. 67-B, pp. 39–47, January 1984.
 - [7] K. A. Zeger and A. Gersho, "Vector quantizer design for memoryless noisy channels," in *IEEE Internationell Conference on Communications*, Philadelphia, USA, 1988, pp. 1593–1597.
 - [8] N. Farvardin and V. Vaishampayan, "On the performance and complexity of channel-optimized vector quantizers," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 155–159, January 1991.
 - [9] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Dordrecht, The Netherlands: Kluwer academic publishers, 1992.