

ROBUST LOSSY SOURCE CODING FOR CORRELATED FADING CHANNELS

by

SHERVIN SHAHIDI

A thesis submitted to the
Department of Electrical and Computer Engineering
in conformity with the requirements for
the degree of Master of Applied Science

Queen's University
Kingston, Ontario, Canada

September 2011

Copyright © Shervin Shahidi, 2011

Abstract

Most of the conventional communication systems use channel interleaving as well as hard decision decoding in their designs, which lead to discarding channel memory and soft-decision information. This simplification is usually done since the complexity of handling the memory or soft-decision information is rather high.

In this work, we design two lossy joint source-channel coding (JSCC) schemes that do not use explicit algebraic channel coding for a recently introduced channel model, in order to take advantage of both channel memory and soft-decision information. The channel model, called the non-binary noise discrete channel with queue based noise (NBNDQ-QB), obtains closed form expressions for the channel transition distribution, correlation coefficient, and many other channel properties. The channel has binary input and 2^q -ary output and the noise is a 2^q -ary Markovian stationary ergodic process, based on a finite queue, where q is the output's soft-decision resolution.

We also numerically show that the NBNDQ-QB model can effectively approximate correlated Rayleigh fading channels without losing its analytical tractability. The first JSCC scheme is the so called channel optimized vector quantizer (COVQ) and the second scheme consists of a scalar quantizer, a proper index assignment, and a sequence maximum a posteriori (MAP) decoder, designed to harness the redundancy left in the quantizer's indices, the channel's soft-decision output, and noise

time correlation. We also find necessary and sufficient condition when the sequence MAP decoder is reduced to an instantaneous symbol-by-symbol decoder, i.e., a simple instantaneous mapping.

Acknowledgments

First of all, I would like to express my sincere gratitude to my supervisors, Professor Fady Alajaji and Professor Tamás Linder, for their constant support and precious comments. I will be always indebted to them for their guidance and motivation during this thesis, which will accompany me throughout my future life.

I would also like to thank my parents, my sister, and my brother-in-law who have always encouraged me; without their unconditional support and patience, I would have never been able to complete this thesis.

Last but not least, I am grateful of my dear friends for their help, kind support and understanding during this period. Specifically, I would like to thank my friend Saeed for all of our discussions and his useful comments during these two years, my close friends Behnam and Ali for all the fun we had together, and my friends Navid and Parisa who, although lived far away from me, were always there when I needed someone to talk to.

Table of Contents

Abstract	i
Acknowledgments	iii
Table of Contents	iv
List of Tables	vi
List of Figures	xi
List of Acronyms	xii
Chapter 1:	
Introduction	1
1.1 Literature Review	1
1.2 Thesis Contribution	2
1.3 Thesis Overview	4
Chapter 2:	
Preliminaries	5
2.1 Communication Channel Models	5
2.2 Source Coding and Quantization	17

Chapter 3:	
	Exploiting Channel Memory and Soft-Decision Information in COVQ Design 27
3.1	Channel Optimized Vector Quantization (COVQ) 28
3.2	Numerical Results 37
Chapter 4:	
	Joint Source-Channel MAP Decoding of the NBNDC-QB 55
4.1	Sequence MAP decoding 57
4.2	Numerical Results 80
Chapter 5:	
	Conclusions and Future Work 92
Bibliography 95

List of Tables

3.1	Simulated annealing parameters.	37
3.2	COVQ Training SDR results (in dB) for the memoryless NBNDQ-QB and the highly correlated NBNDQ-QB with parameters $\alpha = 1.0$, $M = 1$; memoryless Gaussian source.	42
3.3	COVQ Training SDR results (in dB) for the NBNDQ-QB with parameters $\alpha = 1.0$, $M = 1$ and channel correlations 0.5, 0.7 ; memoryless Gaussian source.	43
3.4	COVQ Training SDR results (in dB) for the memoryless NBNDQ-QB and the highly correlated NBNDQ-QB with parameters $\alpha = 1.0$, $M = 1$; Gauss-Markov source with correlation parameter $\phi = 0.9$	44
3.5	COVQ Training SDR results (in dB) for the memoryless NBNDQ-QB and the highly correlated NBNDQ-QB with parameters $\alpha = 1.0$, $M = 1$; memoryless Laplacian source.	45
3.6	COVQ Training SDR results (in dB) for the memoryless NBNDQ-QB and the highly correlated NBNDQ-QB with parameters $\alpha = 0.5$, $M = 5$; memoryless Gaussian source.	46
3.7	COVQ Training SDR results (in dB) for the memoryless NBNDQ-QB and the highly correlated NBNDQ-QB with parameters $\alpha = 1.0$, $M = 5$; memoryless Gaussian source.	47

3.8	COVQ Training SDR results (in dB) for the memoryless NBNDQ-QB and the highly correlated NBNDQ-QB with parameters $\alpha = 2.0$, $M = 5$; memoryless Gaussian source.	48
3.9	COVQ Training SDR results (in dB) for the memoryless NBNDQ-QB and the highly correlated NBNDQ-QB with parameters $\alpha = 1.0$, $M = 1$; memoryless Gaussian source- Decrease increase method (5 times).	49
3.10	NBNDQ parameters for fitting the Rayleigh DFC with $q = 2$	50
3.11	COVQ training results (for DFC-fitted NBNDQ-QB) and simulation results (for the Rayleigh DFC) in terms of SDR (dB); memoryless Laplacian source. $f_D T = 0.01$	51
3.12	COVQ training results (for DFC-fitted NBNDQ-QB) and simulation results (for the Rayleigh DFC) in terms of SDR (dB); memoryless Gaussian source. $f_D T = 0.005$	52
3.13	COVQ training results (for DFC-fitted NBNDQ-QB) and simulation results (for the Rayleigh DFC) in terms of SDR (dB); Gauss-Markov source with correlation factor $\phi = 0.9$. $f_D T = 0.005$	53
3.14	The ρ and δ values for the NBNDQ-QB for soft-decision resolutions $q = 1, 2, 3$	54
4.1	Symbol error rate (in%) for MAP decoding and instantaneous mapping θ^* for symmetric binary Markov sources with $p_{00} = 0.5$ (i.i.d. source). The channel model is the NBNDQ-QB, with $M = 1$, $Cor = 0, 0.9$, and $q = 1, 2, 3$. The values C are calculated from (4.20).	76

4.2	Symbol error rate (in%) for MAP decoding and instantaneous mapping θ^* for symmetric binary Markov sources with $p_{00} = 0.6$ (symmetric Markov source). The channel model is the NBNDC-QB, with $M = 1$, $Cor = 0, 0.9$, and $q = 1, 2, 3$. The values C are calculated from (4.20).	77
4.3	Symbol error rate (in%) for MAP decoding and instantaneous mapping θ^* for symmetric binary Markov sources with $p_{00} = 0.7$ (symmetric Markov source). The channel model is the NBNDC-QB, with $M = 1$, $Cor = 0, 0.9$, and $q = 1, 2, 3$. The values C are calculated from (4.20).	78
4.4	Symbol error rate (in%) for MAP decoding and instantaneous mapping θ^* for symmetric binary Markov sources with $p_{00} = 0.9$ (symmetric Markov source). The channel model is the NBNDC-QB, with $M = 1$, $Cor = 0, 0.9$, and $q = 1, 2, 3$. The values C are calculated from (4.20).	79
4.5	SQ-MAP Training SDR results (in dB) for the memoryless NBNDC-QB and the highly correlated NBNDC-QB with parameters $M = 1$, $\alpha = 1$; memoryless Gaussian source.	84
4.6	SQ-MAP Training SDR results (in dB) for the NBNDC-QB with parameters $M = 1$, $\alpha = 1$, and correlations 0.5, 0.7; memoryless Gaussian source.	84
4.7	SQ-MAP Training SDR results (in dB) for the memoryless NBNDC-QB and the highly correlated NBNDC-QB with parameters $M = 1$, $\alpha = 1$; Gauss-Markov source with correlation factor $\phi = 0.9$	85
4.8	SQ-MAP Training SDR results (in dB) for the memoryless NBNDC-QB and the highly correlated NBNDC-QB with parameters $M = 1$, $\alpha = 1$; memoryless Laplacian source.	85

4.9	SQ-MAP Training SDR results (in dB) for the memoryless NBND- QB and the highly correlated NBND- QB with parameters $M =$ 5 , $\alpha = 0.5$; memoryless Gaussian source.	86
4.10	SQ-MAP Training SDR results (in dB) for the memoryless NBND- QB and the highly correlated NBND- QB with parameters $M =$ 5 , $\alpha = 1$; memoryless Gaussian source.	86
4.11	SQ-MAP Training SDR results (in dB) for the memoryless NBND- QB and the highly correlated NBND- QB with parameters $M =$ 5 , $\alpha = 2$; memoryless Gaussian source.	87
4.12	SQ with instantaneous mapping- Training SDR results (in dB) for the memoryless NBND- QB and the highly correlated NBND- QB with parameters $M = 1$, $\alpha = 1$; memoryless Gaussian source.	87
4.13	SQ with instantaneous mapping- Training SDR results (in dB) for the memoryless NBND- QB and the highly correlated NBND- QB with parameters $M = 1$, $\alpha = 1$; Gauss-Markov source with correlation factor $\phi = 0.9$	88
4.14	SQ-MAP simulation SDR results (in dB) for the DFC-fitted NBND- QB and the DFC; memoryless Gaussian source, $q = 2$	88
4.15	SQ-MAP simulation SDR results (in dB) for the DFC-fitted NBND- QB and the DFC; Gauss-Markov source with correlation factor $\phi = 0.9$, $q = 2$	89
4.16	SQ-MAP simulation SDR results (in dB) for the DFC-fitted NBND- QB and the DFC; memoryless Laplacian source, $q = 2$	89

4.17	SQ-MAP simulation results in symbol error rate(%), for the DFC-fitted NBNDQ-QB and the DFC; memoryless Gaussian source, $q = 2$	90
4.18	SQ-MAP simulation results in symbol error rate(%), for the DFC-fitted NBNDQ-QB and the DFC; Gauss-Markov source with correlation fac- tor $\phi = 0.9$, $q = 2$	90
4.19	SQ-MAP simulation results in symbol error rate(%), for the DFC-fitted NBNDQ-QB and the DFC; memoryless Laplacian source, $q = 2$	91
4.20	$P(\mathbf{x})$ for different source distributions and SQ coding rates.	91

List of Figures

1.1	Block diagram of a separate (tandem) source and channel coding system and a joint source-channel coding system.	2
2.1	Block diagram of a digital communication system.	6
2.2	Binary symmetric channel.	8
2.3	The additive white Gaussian noise model.	9
2.4	The Gilbert-Elliot channel model.	10
2.5	A queue of length M	12
2.6	An urn of numbered balls with 2^q different numbers.	12
2.7	Rayleigh discrete fading channel, with BPSK modulation.	15
2.8	Block diagram of a vector quantizer in a communication system.	25
3.1	Block diagram of a COVQ system.	29
4.1	Block diagram of an SQ-MAP system.	57
4.2	FBC and NBC index assignments for an hypothetical source distribution and its quantization regions; MSB is the rightmost bit.	58

List of Acronyms

AWGN Additive White Gaussian Noise

BER Bit-Error Rate

BPSK Binary Phase-Shift Keying

BSC Binary Symmetric Channel

CC Centroid Condition

COVQ Channel Optimized Vector Quantizer

DFC Discrete Fading Channel

DMC Discrete Memoryless Channel

DMS Discrete Memoryless Source

FBC Folded Binary Code

GEC Gilbert-Elliott Channel

i.i.d. independent and identically distributed

LBG-VQ Linde-Buzo-Gray Vector Quantizer

MAP Maximum A Posteriori

MMSE Minimum Mean Square Error

MSE Mean Square Error

NBC Natural Binary Code

NBNDC Non-Binary Noise Discrete Channel

NBNDC-QB Non-Binary Noise Discrete Channel with Queue Based noise

NNC Nearest Neighbor Condition

QBC Queue Based Channel

SDR Signal-to-Distortion Ratio

SA Simulated Annealing

SNR Signal-to-Distortion Ratio

SQ Scalar Quantizer

VQ Vector Quantizer

Chapter 1

Introduction

Looking at the recent ever increasing demands for communication systems, it is obvious that in order to meet the present needs, modern communication systems are required to become faster, more reliable, and more suitable for wireless and mobile communications. As a result, the need arises for designing systems with less delay and complexity without sacrificing reliability in wireless mobile environments. In order to design such systems, some improvements can be proposed to enhance the performance of conventional systems.

1.1 Literature Review

It is well known that the separate treatment of source and channel coding, as in Shannon's source-channel coding theorem [31, 32], is not optimal in the presence of complexity and delay constraints. A variety of different joint-source channel coding schemes have been proposed to address this problem, such as [5, 20, 23, 26, 34] and

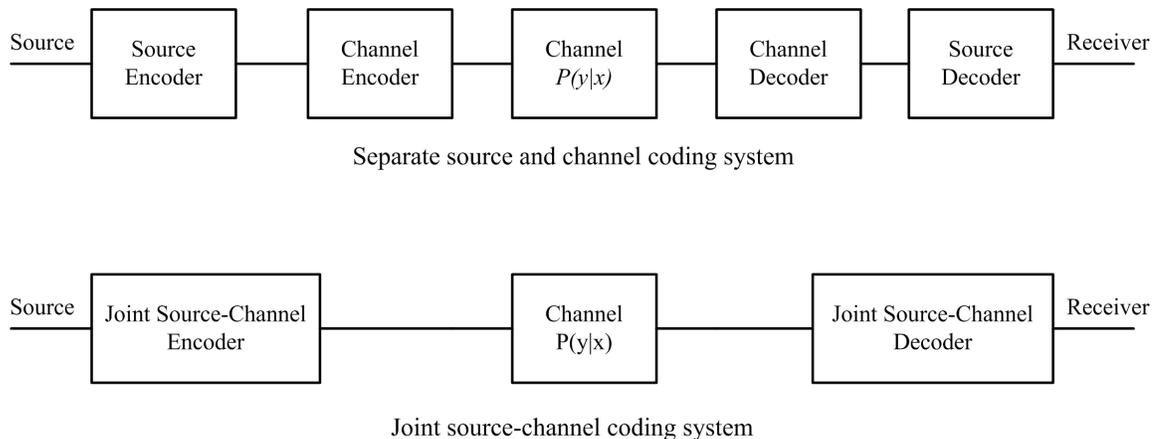


Figure 1.1: Block diagram of a separate (tandem) source and channel coding system and a joint source-channel coding system.

many others for the case of lossy coding. It is also known that the capacity of a well-behaved (ergodic) channel with memory is strictly greater than the capacity of its memoryless counterpart channel (with identical one-dimensional transition distribution) realized by ideal (infinite) interleaving [10, 37]. Consequently, given the memory statistics, a communication system can be designed to take advantage of the channel's memory and outperform systems that discard memory information via interleaving. Furthermore, it has been shown that the channel's soft-decision information can improve capacity and system performance over hard-decision decoded schemes (e.g., see [4, 6, 25, 33, 35]).

1.2 Thesis Contribution

In this thesis we design a joint source channel-coding scheme, known as channel optimized vector quantizer (COVQ), and a joint source-channel decoding scheme, which

consists of a scalar quantizer (SQ) with a maximum a posteriori (MAP) decoder. Both systems are source centric joint source-channel coding schemes (JSCC) and have less complexity than the conventional tandem source-channel coding systems in which the problem of source coding and channel coding are dealt with separately (Figure 1.1). There are other methods of JSCC such as channel centric approaches (that use explicit channel coding) or unequal error protection approaches.

Both system performances are studied for the non-binary noise discrete channel (NBND) and the correlated Rayleigh fading channel used with soft-decoding demodulation. The NBND is a channel model recently introduced in [28], where the channel has binary input and 2^q -ary output, making it possible to model soft-quantized output channels with memory. We use the NBND channel with the queue-based noise, introduced in [28], as the noise process in the NBND model to provide closed form expressions for the channel transition distribution, and then use the obtained model as an alternative representation of a correlated Rayleigh discrete fading channel (DFC). Note that in contrast to the NBND with queue-based noise model (which we refer to as NBND-QB), for the Rayleigh DFC no closed form transition distribution expression are known for block lengths of greater than 3, so the distribution must be determined via numerical methods. We test the system designed for the NBND model over the equivalent correlated Rayleigh DFC to simulate its performance in wireless communications.

We also obtain necessary and sufficient condition for the sequence MAP detector in the SQ-MAP system over the NBND-QB to reduce to an instantaneous symbol-by-symbol mapping with the same sequence error probability of decoding.

We show numerically that both systems can successfully exploit the channel's

memory and soft-decision information in order to gain better signal-to-distortion ratio (SDR) performance, in contrast with systems that use interleaving and hard quantization, hence disregarding the channel's memory and soft-decision information. We also validate numerically that the NBNDQ-QB can effectively approximate the Rayleigh DFC while in spite of the Rayleigh DFC, the NBNDQ-QB maintains its analytical tractability.

1.3 Thesis Overview

The organization of the thesis is as follows. In Chapter 2, we describe an overview of digital communication channel models and source coding theory. The design and study of the COVQ system is explained in Chapter 3 and the study of the SQ-MAP system is presented in Chapter 4. Chapter 5 is devoted to conclusions and future works.

Chapter 2

Preliminaries

2.1 Communication Channel Models

A communication channel whose input and output each have a finite alphabet is called a discrete channel. In a discrete (digital) channel model, the transmitted message is modeled as a digital signal. In contrast, in an analog channel model, the transmitted message is modeled as an analog (continuous) signal.

The physical transmission of signals over a transmission medium always takes place via analog signals. As a result, in order to transmit a digital signal on a physical medium, a digital to analog conversion must take place. Similarly, an analog to digital conversion is done at the receiver to retrieve the transmitted digital data. Consider Figure 2.1 as a generic point-to-point digital communication system. As can be seen, the data symbols are discrete before they are sent through the modulator, as well as the data symbols after the demodulator. Therefore, the whole modulation process, analog channel model, and demodulation process can be considered as a black-box, performing as a digital input, digital output channel. Hence, a discrete channel can

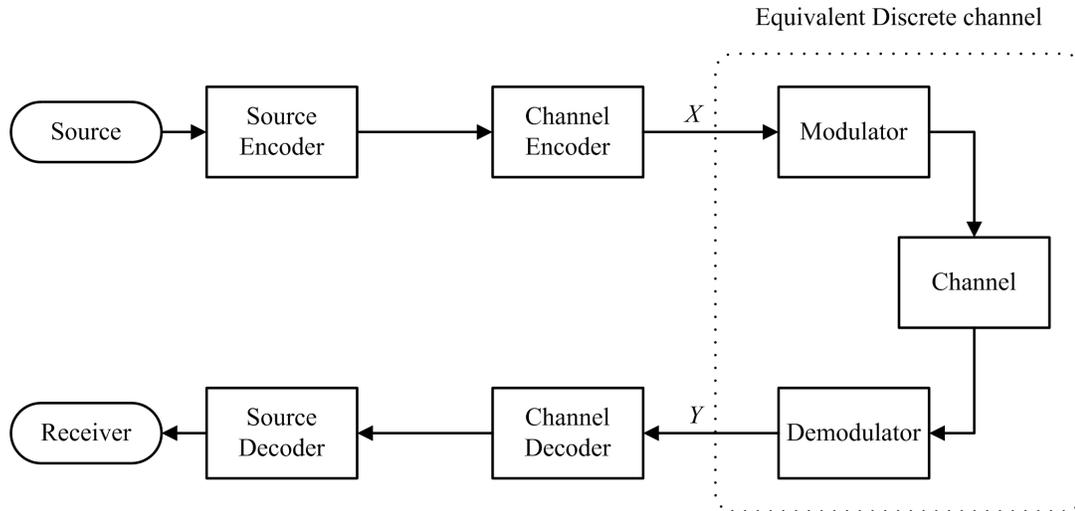


Figure 2.1: Block diagram of a digital communication system.

model an equivalent analog channel model, which uses a specific modulation scheme.

From another point of view, different channel models can be classified into two large classes: memoryless channels and dynamic channels (channels with memory). A channel is said to be memoryless if the probability distribution of the output depends only on the input at that time and is conditionally independent of previous and future channel inputs or outputs [9]. For a channel which is not memoryless, we consider the case where the statistical properties of the output signal at time t depend only on the present and past transmitted signal.

In the following, we review some of the well-known channel models, as well as some recently introduced channel models which are relevant to this thesis.

2.1.1 Memoryless channels

Discrete Memoryless Channels

The discrete memoryless channel (DMC) is a discrete channel with finite input alphabet \mathcal{X} and finite output alphabet \mathcal{Y} . Its transition probabilities, $P_{Y^n|X^n}(y^n|x^n)$, for any input n -tuple $x^n = (x_1, x_2, \dots, x_n)$ and received n -tuple $y^n = (y_1, y_2, \dots, y_n)$, satisfies

$$P_{Y^n|X^n}(y^n|x^n) = \prod_{i=1}^n P_{Y_i|X_i}(y_i|x_i), \quad (2.1)$$

for every $n = 1, 2, \dots$. A DMC is uniquely determined by the channel transition matrix $\mathbf{Q} = [P(y|x)]$. Condition (2.1) is actually equivalent to the following two sets of conditions:

$$\begin{cases} P_{Y_n|X^n, Y^{n-1}}(y_n|x^n, y^{n-1}) = P_{Y|X}(y_n|x_n), & \forall n = 1, 2, \dots, x^n, y^n; \\ P_{Y^{n-1}|X^n}(y^{n-1}|x^n) = P_{Y^{n-1}|X^{n-1}}(y^{n-1}|x^{n-1}), & \forall n = 1, 2, \dots, x^n, y^{n-1}. \end{cases}$$

$$\begin{cases} P_{Y_n|X^n, Y^{n-1}}(y_n|x^n, y^{n-1}) = P_{Y|X}(y_n|x_n), & \forall n = 1, 2, \dots, x^n, y^n; \\ P_{X_n|Y^{n-1}, X^{n-1}}(x_n|y^{n-1}, x^{n-1}) = P_{X_n|X^{n-1}}(x_n|x^{n-1}), & \forall n = 1, 2, \dots, x^n, y^{n-1}. \end{cases}$$

These conditions imply that the current output Y_n only depends on the current input X_n , but not on the past inputs X^{n-1} and outputs Y^{n-1} . Also, given the current input, the past outputs Y^{n-1} do not depend on the current input X_n . The current input X_n is also independent of past outputs Y^{n-1} given the past inputs X^{n-1} .

A simple example for a DMC is the binary symmetric channel (BSC) depicted in Figure 2.2, which has binary input and output alphabets, and is described by the

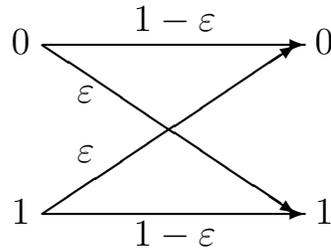


Figure 2.2: Binary symmetric channel.

transition matrix

$$\mathbf{Q} = \begin{bmatrix} 1 - \varepsilon & \varepsilon \\ \varepsilon & 1 - \varepsilon \end{bmatrix}.$$

It can be seen that the BSC can be described by a single parameter ε , referred to as the crossover probability.

Additive White Gaussian Noise

The additive white Gaussian noise (AWGN) channel model is a well-known example of analog memoryless channels. The model is discrete in time and continuous in amplitude, with input and output alphabets $\mathcal{X} = \mathcal{Y} = \mathbb{R}$. The output Y_t at sample time t is the sum of the input X_t and the noise Z_t (Figure 2.3)

$$Y_t = X_t + Z_t, \quad Z_t \sim \mathcal{N}(0, N_0/2)$$

where the noise is white (has a constant spectral density). Thus the noise random variables, Z_t , $t = 1, 2, \dots$, are independent and identically distributed (i.i.d.) each having a Gaussian distribution, which we assume to have zero-mean and variance $N_0/2$.

This channel provides tractable mathematical model which is very useful in some

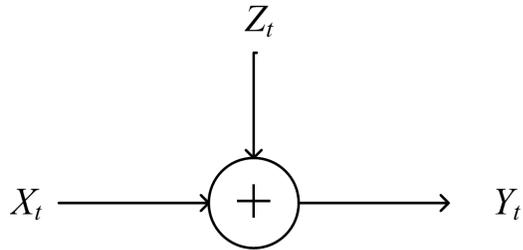


Figure 2.3: The additive white Gaussian noise model.

simplified real world applications, such as single user wireline communications, but cannot model phenomena such as fading, interference, and dispersion.

2.1.2 Channels With Memory

Discrete channels with memory are studied in detail in [16, p. 97-111]. A simple model, used extensively in the literature, is the Gilbert-Elliott Channel (GEC). The GEC is a varying binary symmetric channel, the crossover probabilities of which are determined by the current state of a discrete stationary binary Markov process (Figure 2.4) [24]. According to this model, the channel has two states known as the *good* state (with low crossover probability) and the *bad* state (with high crossover probability). Since the state of the channel cannot be determined from the observations, the channel is a so-called hidden Markov model.

A more explicit channel with memory is the binary Markov channel, in which the channel memory is represented by an additive Markovian noise process. The output Y_j at time j is affected via the additive noise Z_j , which is assumed to be independent

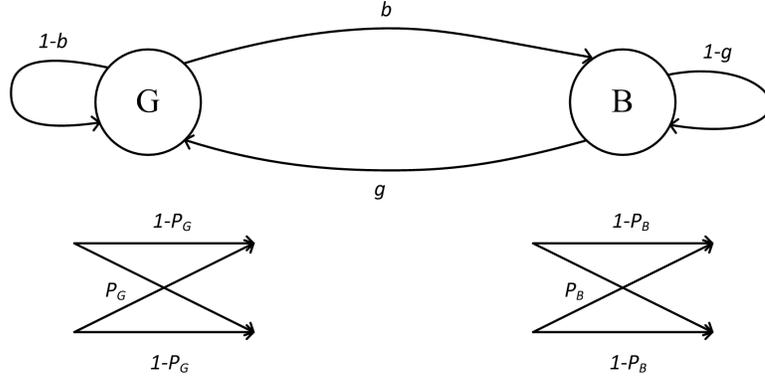


Figure 2.4: The Gilbert-Elliot channel model.

of the input of the channel X_j ,

$$Y_j = X_j \oplus Z_j, \quad j = 1, 2, 3, \dots,$$

where the \oplus represents modulo 2 addition. The noise process $\{Z_j\}_{j=1}^{\infty}$ is a binary stationary ergodic Markov process with memory order M , generated by the finite-memory contagion urn model described in [1]. More precisely, according to this model, the noise transition probabilities are given by

$$\begin{aligned} Pr\{Z_j = 1 | Z_{j-1} = z_{j-1}, \dots, Z_1 = z_1\} &= Pr\left\{Z_j = 1 | Z_{j-1} = z_{j-1}, \dots, Z_{j-M} = z_{j-M}\right\} \\ &= Pr\left\{Z_j = 1 | \sum_{i=j-M}^{j-1} Z_i = \sum_{i=j-M}^{j-1} z_i\right\} \\ &= \frac{\epsilon + \left(\sum_{i=j-M}^{j-1} z_i\right) \delta}{1 + M\delta}, \end{aligned}$$

where $z_i \in \{0, 1\}$, ϵ is the channel bit-error rate (BER), and δ determines the noise correlation via $Cor = \delta/(\delta+1)$. It can be seen that letting $\delta = 0$, the channel is equivalent to a BSC with crossover probability ϵ . This channel can be fully characterized by the parameters δ , ϵ , and memory order M .

Non-Binary Noise Discrete Channel

The Non-Binary Noise Discrete Channel (NBND) [28] is a binary-input and 2^q -ary-output channel model. The input data bits X_j are affected by noise Z_j via the relation

$$Y_j = (2^q - 1)X_j + (-1)^{X_j} Z_j, \quad (2.2)$$

where $Y_j, Z_j \in \{0, 1, \dots, 2^q - 1\}$ for $j = 1, 2, \dots$, with $\{Y_j\}$ denoting the channel output process, and $\{X_j\}$ denoting the channel input binary process. Also, $\{Z_j\}$ is the noise process assumed to be independent of $\{X_j\}$. According to (2.2), Z_j can also be written in terms of input and output symbols,

$$Z_j = \frac{Y_j - (2^q - 1)X_j}{(-1)^{X_j}}, \quad j = 1, 2, \dots \quad (2.3)$$

We will extensively use this channel model in this thesis, since we are interested in taking advantage of the soft-decision information for improving the system's performance, which is feasible using the NBND because it has a non-binary output. Furthermore, the number of the NBND's parameters, and hence the model's complexity, is independent of the channel noise memory order M , so that noise models with arbitrarily large memory orders can be implemented with no extra complexity. On the other hand, the complexity is exponentially proportional to the soft-decision quantization resolution q , although typical values for q are as low as 2 or 3.

The noise process $\{Z_j\}_{j=1}^{\infty}$ can in general be any stochastic process. For example, if $\{Z_j\}$ is a binary stationary memoryless process (hence $q = 1$), the NBND reduces to the BSC. The noise process applied in [28] (and also in this thesis), is a non-binary generalization of the queue-based (QB) noise [37]. We refer to this channel model as NBND-QB. The noise is modeled via a 2^q -ary M^{th} -order Markovian stationary

ergodic process with $2^q + 2$ independent parameters. According to this model, at each sample time j , the noise is generated via one of the two following packages:

- A queue with M cells, shown in Figure 2.5, that contains a ball in each cell. The balls are numbered, with integers ranging from 0 to $2^q - 1$. Each number represents an error symbol.
- An urn that contains a large set of numbered balls; the same numbered balls as in the queue(Figure 2.6).



Figure 2.5: A queue of length M .

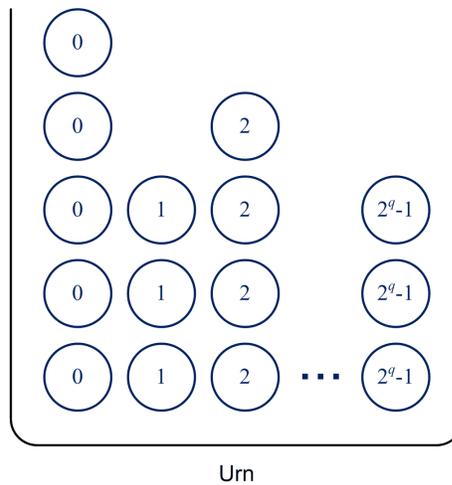


Figure 2.6: An urn of numbered balls with 2^q different numbers.

At each sample time j , a biased coin is flipped to choose one of the packages. Assume that the queue is selected with probability ε . If the queue is selected, the noise symbol is determined by selecting one of the M cells and taking the number of the ball it contains as the generated noise symbol. The probability of selecting the k th cell of the queue is chosen as

$$\begin{cases} \frac{1}{M-1+\alpha}, & \text{if } k = 1, 2, \dots, M-1; \\ \frac{\alpha}{M-1+\alpha}, & \text{if } k = M. \end{cases}, \quad \alpha \geq 0$$

If the urn is selected, one of the balls will be taken out of the urn, and the ball number will determine the noise symbol. The numbered balls in the urn satisfy the probability distribution $(\rho_0, \rho_1, \dots, \rho_{2^q-1})$, i.e., a ball with number i on it is taken out of the urn with probability ρ_i . After the noise symbol is generated, no matter which package was selected, a ball with the same number as the generated noise symbol will be pushed in front of the queue, pushing out the ball in M th cell of the queue.

The resulting QB noise process is a stationary M th order Markov source and has only $2^q + 2$ independent parameters: the size of the queue M , the probability distribution of the balls in the urn, and correlation parameters $0 \leq \varepsilon < 1$ and $\alpha \geq 0$.

The state process of the queue based noise $\{\mathbf{S}_n\}_{-\infty}^{\infty}$, which is defined by $\mathbf{S}_n \triangleq (Z_n, Z_{n-1}, \dots, Z_{n-M+1}) \in \{0, 1, \dots, 2^q - 1\}^M$, is a homogeneous, first-order Markov process. Define the noise state transition probability by

$$Q(\mathbf{s}_n | \mathbf{s}_{n-1}) \triangleq Pr\{\mathbf{S}_n = \mathbf{s}_n | \mathbf{S}_{n-1} = \mathbf{s}_{n-1}\},$$

where $\mathbf{s}_n = (z_n, z_{n-1}, \dots, z_{n-M+1})$, $z_n \in \{0, 1, 2, \dots, 2^q - 1\}$. According to [28],

$$Q(\mathbf{s}_n | \mathbf{s}_{n-1}) = \left(\sum_{\ell=1}^{M-1} \delta_{z_n, z_{n-\ell}} + \alpha \delta_{z_n, z_{n-M}} \right) \frac{\varepsilon}{M-1+\alpha} + (1-\varepsilon)\rho_{z_n}, \quad (2.4)$$

where,

$$\delta_{i,j} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j. \end{cases}$$

Since the noise process is independent of the input, we have

$$Pr\{Y^m = y^m \mid X^m = x^m\} = Pr\{Z^m = z^m\}. \quad (2.5)$$

Hence, the m -fold channel transition probability $Pr\{Z^m = z^m\} \triangleq P_{\text{NBND C}}^{(m)}(z^m) = Pr_{\text{NBND C-QB}}\{Z_1 = z_1, Z_2 = z_2, \dots, Z_m = z_m\}$ is given by [28]:

- For $m > M$

$$P_{\text{NBND C-QB}}^{(m)}(z^m) = \prod_{i=M+1}^m \left[\left(\sum_{\ell=i-M+1}^{i-1} \delta_{z_i, z_\ell} + \alpha \delta_{z_i, z_{i-M}} \right) \times \frac{\varepsilon}{M-1+\alpha} + (1-\varepsilon)\rho_{z_i} \right] \pi_{(z_1, z_2, \dots, z_M)}, \quad (2.6)$$

where

$$\pi_{(z_1, z_2, \dots, z_M)} = \frac{\prod_{\ell=0}^{2^q-1} \prod_{j=0}^{\xi_\ell} \left((1-\varepsilon)\rho_\ell + j \frac{\varepsilon}{M-1+\alpha} \right)}{\prod_{k=0}^{M-1} \left((1-\varepsilon) + k \frac{\varepsilon}{M-1+\alpha} \right)}, \quad (2.7)$$

where $\xi_\ell = \sum_{k=1}^M \delta_{z_k, \ell}$,

- For $m \leq M$

$$P_{\text{NBND C-QB}}^{(m)}(z^m) = \frac{\prod_{\ell=0}^{2^q-1} \prod_{j=0}^{\xi'_\ell} \left((1-\varepsilon)\rho_\ell + j \frac{\varepsilon}{M-1+\alpha} \right)}{\prod_{k=0}^{m-1} \left((1-\varepsilon) + k \frac{\varepsilon}{M-1+\alpha} \right)}, \quad (2.8)$$

where $\xi_\ell' = \sum_{k=1}^m \delta_{z_k, \ell}$. Note from (2.8), for $m = 1$ we have $P_{\text{NBND C-QB}}^{(1)}(z_1) = \rho_{z_1}$ for $z_1 \in \{0, 1, \dots, 2^q - 1\}$. Finally the channel noise correlation is given by

$$\text{Cor} = \frac{\mathbf{E}[Z_k Z_{k+1}] - \mathbf{E}[Z_k]^2}{\text{Var}(Z_k)} = \frac{\frac{\epsilon}{M-1+\alpha}}{1 - (M-2+\alpha)\frac{\epsilon}{M-1+\alpha}}.$$

Rayleigh Fading Channel

The Rayleigh fading channel is a continuous alphabet, discrete time example for channel with memory. The channel noise affects the input data $X_k, k = 1, 2, 3, \dots$ with both multiplicative and additive noise. The multiplicative noise $A_k, k = 1, 2, 3, \dots$ is a sequence of Rayleigh distributed random variables (correlated in general), known as the fading coefficients, which cause an attenuation of the received signal. The additive noise is an AWGN with zero-mean and variance $N_0/2$. As a result, the output Y_k is given by

$$Y_k = A_k X_k + N_k, \quad k = 1, 2, 3, \dots$$

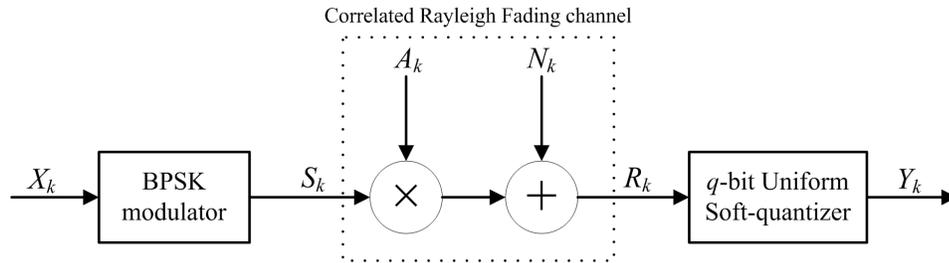


Figure 2.7: Rayleigh discrete fading channel, with BPSK modulation.

The channel setup we are using in this thesis is a discrete fading channel model (DFC) consisting of a binary phase-shift keying (BPSK) modulator, a time-correlated flat Rayleigh fading channel with additive white Gaussian noise (AWGN), and a q -bit

soft-quantized demodulator, shown in Figure 2.7. Let the input and output alphabet be $\mathcal{X} = \{0, 1\}$ and $\mathcal{Y} = \{0, 1, \dots, 2^q - 1\}$, respectively. Denoting the DFC binary input by $\{X_k\}$, $k = 1, 2, \dots$, the received channel symbols are given by

$$R_k = \sqrt{E_s} A_k S_k + N_k, \quad k = 1, 2, \dots$$

where E_s is the energy of signal sent over the channel, $S_k = 2X_k - 1 \in \{-1, 1\}$ is the BPSK modulated signal and $\{N_k\}$ is an additive white noise process, represented by a sequence of independent and identically distributed (i.i.d.) Gaussian random variables of variance $N_0/2$. $\{A_k\}$ is the channel's fading process with $A_k = |G_k|$, where $\{G_k\}$ is a time-correlated complex wide-sense stationary Rayleigh process with autocorrelation function given by $R[k] = J_0(2\pi f_D T |k|)$ from Clarke's model [8], where $f_D T$ is the normalized maximum doppler frequency and $J_0(\cdot)$ is the zeroth-order Bessel function of the first kind. Therefore, A_k is Rayleigh distributed, with unit second moment. The fading process $\{A_k\}$ is assumed to be independent of the noise and input processes. The channel signal-to-noise ratio (SNR) is given by $\text{SNR} = E_s/N_0$.

In the last part of the DFC model, a soft-decision demodulator consisting of a uniform quantizer with a resolution of q bits, takes the output R_k to produce the discrete channel output:

$$Y_k = j, \quad \text{if } R_k \in (T'_{j-1}, T'_j],$$

where the T'_j are uniformly spaced thresholds with step-size Δ , such that

$$T'_j = \begin{cases} -\infty, & \text{if } j = -1 \\ (j + 1 - 2^{q-1})\Delta, & \text{if } j = 0, 1, \dots, 2^q - 2 \\ \infty, & \text{if } j = 2^q - 1. \end{cases}$$

Let $\delta \triangleq \Delta/\sqrt{E_s}$ and $T_j \triangleq T'_j/\sqrt{E_s}$. The m -fold transition probability for the DFC

can be calculated via [28]

$$\begin{aligned} P_{\text{DFC}}^{(m)}(y^m | x^m) &\triangleq \Pr\{Y^m = y^m | X^m = x^m\} \\ &= \mathbf{E}_{A_1 \dots A_m} \left[\prod_{k=1}^m q_{x_k, y_k}(A_k) \right], \end{aligned} \quad (2.9)$$

where $y^m = (y_1, y_2, \dots, y_m)$, $x^m = (x_1, x_2, \dots, x_m)$, $q_{i,j}(a_k) \triangleq \Pr\{Y_k = j | X_k = i, A_k = a_k\}$, and $\mathbf{E}_X[\cdot]$ denotes expectation with respect to the random variable X . For $m = 1$, there is closed form expression for $P_{\text{DFC}}^{(1)}(y|x) = P_{\text{DFC}}^{(1)}(j)$, $y \in \mathcal{Y}$, $x \in \mathcal{X}$, which is given by

$$P_{\text{DFC}}^{(1)}(j) = n(-T_{j-1}) - n(-T_j), \quad (2.10)$$

where $j = \frac{y-(2^q-1)x}{(-1)^x} \in \mathcal{Y}$, and

$$n(T_j) = 1 - Q(T_j \sqrt{2\text{SNR}}) - \frac{\left[1 - Q\left(\frac{T_j \sqrt{2}}{\sqrt{\frac{1}{\text{SNR}} + 1}}\right) \right] e^{-\frac{T_j^2}{\left(\frac{1}{\text{SNR}} + 1\right)}}}{\sqrt{\frac{1}{\text{SNR}} + 1}},$$

where $Q(\cdot)$ is the Gaussian Q-function. In general, for $m \leq 3$, $P_{\text{DFC}}^{(m)}(y^m | x^m)$ can be calculated in closed form. For $m > 3$, since the joint probability density function of arbitrarily correlated Rayleigh and Rician random variables is not known in closed form, it can only be determined via numerical methods. It can be shown that the DFC is actually an NBNDC as given by (2.2) with a stationary ergodic noise process [28].

2.2 Source Coding and Quantization

Consider Figure 2.1 again. In a general digital communication system, in order to send the information from the source to receiver, at first some processing on the source data takes place (encoding). The data is expressed using the symbols of a

given alphabet and sent over a channel with a certain statistical characteristics. As well, the receiver needs to do some processing on the received data, as a means to retrieve the source information (decoding). This encoding/decoding process is often performed in two separate steps. The first step is referred to as source coding and the second step is called channel coding.

2.2.1 Source Coding

In a separate source and channel coding system, also known as a tandem coding system, the source coding process aims to represent the source information as compactly as possible. To do this, source *redundancy* should be removed. The redundancy of a source can be measured quantitatively, via the concept of source *entropy*. According to Shannon [30], entropy is a measure of the uncertainty of a random variable. Assume the source is a stationary discrete memoryless source (DMS). Hence it can be modeled as a random variable X with alphabet $\mathcal{X} = \{0, 1, \dots, N - 1\}$. The entropy $H(X)$ of such source is defined as:

$$H(X) = - \sum_{x \in \mathcal{X}} p(x) \log p(x) = -\mathbf{E}_X[\log p(X)],$$

where $p(x) = Pr\{X = x\}$. For a source with memory, the source can be modeled as a stochastic process $\{X_i\}$, $i = 1, 2, \dots$. The amount of randomness or uncertainty for such sources is measured via the *entropy rate*, $H_\infty(\mathcal{X})$, defined as

$$H_\infty(\mathcal{X}) = \lim_{n \rightarrow \infty} \frac{1}{n} \left(-\mathbf{E}_{X_1, X_2, \dots, X_n}[\log p(X_1, X_2, \dots, X_n)] \right),$$

(if the limit exists,) where $p(x_1, x_2, \dots, x_n) = Pr\{X_1 = x_1, X_2 = x_2, \dots, X_n = x_n\}$.

Note that for a DMS, the entropy and entropy rate are equal. Setting the base

of the logarithms to 2, the unit of these measures is expressed in *bits*. It can be shown [9, p. 29-30], that for any stochastic process $\{X_i\}$, $i = 1, 2, \dots$,

$$H_\infty(\mathcal{X}) \leq \log_2 N,$$

where N is the size of the source alphabet, i.e. $|\mathcal{X}| = N$.

For a discrete source $\{X_i\}$, $i = 1, 2, \dots$ with alphabet size N , the redundancy is defined as

$$\rho_T = \log_2 N - H_\infty(\mathcal{X}).$$

The redundancy of a source can be due to non-uniformity of its marginal probability distribution or its memory. As a result, the total redundancy ρ_T can be decomposed into two parts:

$$\begin{aligned} \rho_T &= \rho_D + \rho_M, \\ \rho_D &= \log_2 N - H(X_1), \\ \rho_M &= H(X_1) - H_\infty(\mathcal{X}). \end{aligned}$$

Here ρ_D is the redundancy due to non-uniformity and ρ_M is due to the memory.

In general, we distinguish between two kinds of source coding; lossless and lossy source coding. In lossless source coding, the aim is to remove all the source's redundancy by compressing the source, while the data is still fully retrievable; i.e., the retrieved data after decompression is identical to the source data. In [30], Shannon showed that for a lossless fixed-to-variable length source coding scheme and a stationary ergodic source $\{X_i\}_{i=1}^\infty$, the coding rate R can be arbitrarily close to source entropy rate $H_\infty(\mathcal{X})$. Conversely, he showed that for a source code with coding rate R less than the entropy rate $H_\infty(\mathcal{X})$, the probability of decoding error is arbitrarily close to one, for sufficiently large source blocks k .

In a lossy source coding scheme, the minimum coding rate is a function of the amount of allowed distortion. The *rate distortion function* $R(D)$ is the infimum of rates R such that (R, D) is in the *rate distortion region* of the source for a given distortion D , where the rate distortion region for a source is the closure of the achievable rate distortion pairs (R, D) [9, p. 306]. In the rate distortion theorem, it is proved that for an independent and identically distributed (i.i.d.) source X with distribution $p(x)$ and bounded distortion function $d(x, \hat{x})$, the rate distortion function satisfies

$$R(D) = \min_{p(\hat{x}|x): \sum_{x, \hat{x}} p(x)p(\hat{x}|x)d(x, \hat{x}) \leq D} I(X; \hat{X}),$$

where $I(X; \hat{X})$ is the mutual information of the two random variables $X \in \mathcal{X}$ and $\hat{X} \in \hat{\mathcal{X}}$ and is defined as

$$I(X; \hat{X}) = \sum_{x \in \mathcal{X}} \sum_{\hat{x} \in \hat{\mathcal{X}}} p(x, \hat{x}) \log \frac{p(x, \hat{x})}{p(x)p(\hat{x})},$$

where $p(x, \hat{x})$ is the joint probability distribution of the two random variables.

2.2.2 Quantization

Since the conventional entropy function $H_\infty(\mathcal{X})$ is undefined for continuous sources, the information content of such sources is theoretically infinite. Thus it is practically impossible to present a continuous source using a source code with finite rate. As a result, using lossy source coding rather than lossless coding is inevitable. This is a process in which the analog source is converted to a discrete (digital) source at the cost of some distortion with respect to the original source. This task is done via quantization.

A quantizer in general, partitions the continuous domain \mathcal{R} of the source into a finite number of regions and assigns a value to each region, which represents all the

members of that region. The representing values are called *output levels* or *reconstruction points*. More specifically, consider a source X with a continuous alphabet \mathcal{R} . An N -level quantizer Q is a mapping from the domain \mathcal{R} to a set of output levels, known as *codebook* $\mathcal{C} = \{c_i : i \in \mathcal{I}\}$, with \mathcal{I} being the index set of the quantizer, such that

$$\hat{x} \triangleq Q(x) = c_i \quad \text{if and only if} \quad x \in \mathcal{R}_i.$$

Equivalently, $\mathcal{R}_i = \{x \in \mathcal{R} : Q(x) = c_i\}$.

To measure the quality of a quantizer, a distortion measure $d(X, \hat{X})$ is defined, where X is the continuous random variable to be quantized and \hat{X} is the quantized value. Different distortion measures may be chosen to quantify the quality of a quantizer. When \mathcal{R} is the real line, a popular choice is the r th power of the magnitude error $d(X, \hat{X}) = |x - \hat{x}|^r$, $r > 0$. A very common distortion measure which is exclusively used in this thesis is the square error distortion $d(X, \hat{X}) = (x - \hat{x})^2$. For a random variable X , often called a random source X , the expected distortion is considered as measure of quality for a quantizer Q

$$D(Q) = E[d(X, Q(X))]. \quad (2.11)$$

For the square error distortion, $E[d(X, Q(X))]$ is called mean square error (MSE) and is calculated via $D(Q) = E[d(X, Q(X))] = E[(X - Q(X))^2]$.

There are two necessary conditions for a quantizer to be optimal. One of the conditions is about the quantization regions and the other is about output levels [17, p. 176-185], [9, p. 303]:

- *The Nearest neighbor condition (NNC)*: Given a codebook \mathcal{C} , the distortion is minimized by mapping a random source X to the output level c_i that has

minimum distortion with respect to it. The collection of quantization regions in \mathcal{R} defined by this mapping is called a *Voronoi* or *Dirichlet partition* indexed by the output levels. These are given by

$$\mathcal{R}_i \subset \{x : d(x, \hat{x}_i) \leq d(x, \hat{x}_j), j \in \mathcal{I}\} \quad i \in \mathcal{I} \quad (2.12)$$

- *The centroid condition (CC)*; The reconstruction points should minimize the conditional expected distortion over their respective assignment regions. Hence, the output level is simply the *centroid* of the part of source that lies in the region \mathcal{R}_i , i.e.,

$$\hat{x}_i = \arg \min_{\hat{x} \in \mathcal{R}} E[d(X, \hat{x}) | X \in \mathcal{R}_i], \quad i \in \mathcal{I}. \quad (2.13)$$

For the MSE distortion, it can be shown that the centroid of a region is its *center of mass*, given by

$$\hat{x}_i = E[X | X \in \mathcal{R}_i], \quad i \in \mathcal{I}. \quad (2.14)$$

As can be seen, according to the above optimality conditions, the quantizer regions can be obtained from the codebook and vice versa. Consequently, in order to describe a quantizer, it is enough to specify its codebook.

Applying the NNC and CC conditions to design a quantizer, if the quantizer only takes one source symbol at a time and outputs L q -ary symbols, it is called a scalar quantizer (SQ) with rate $R = (L \log_2 q)/1$ bits/sample. If the output alphabet is q -ary, then the quantizer's codebook will have $N = 2^R$ output levels. In this thesis we always consider output of the quantizer to be binary. The most common algorithm for SQ design is the *Lloyd algorithm* [22]. Furthermore, a quantizer that takes k source symbols at a time and outputs L quantized symbols, is called a vector quantizer

(VQ) with rate $R = (L \log_2 q)/k$ bits/sample and $N = 2^R$ output levels. The design algorithm for VQ is the generalized Lloyd algorithm, also known as *Linde-Buzo-Gray vector quantizer algorithm* (LBG-VQ) [21].

The Lloyd algorithm for SQ design is an iterative algorithm. Its idea comes directly from the CC and NNC conditions: For a fixed codebook, optimize the partition set $\{\mathcal{R}_i\}$. For the resulting partition, optimize the codebook \mathcal{C} . Iterate. Algorithm 2.1 depicts the Lloyd-Max algorithm in more detail. A quantizer that is designed using the Lloyd algorithm, is called a *Lloyd-Max quantizer*. When using the Lloyd algorithm, the distortion in each iteration is either reduced or will be leaved unchanged. As a result, since the distortion is nonnegative, the algorithm is guaranteed to converge.

In order to start the algorithm, an initial codebook must be given. A well-known technique for initial codebook selection is the splitting algorithm, [12, 21] which has experimentally been shown to yield well-designed final codebooks. The splitting algorithm is described in Section 3.1.2.

Algorithm 2.1 The Lloyd algorithm for SQ design (for MSE distortion)

Input: pdf $f(x)$, initial codebook $\mathcal{C}_1 = \{\hat{x}_1^{(1)}, \hat{x}_2^{(1)}, \dots, \hat{x}_N^{(1)}\}$, threshold ϵ .

$M \leftarrow 1$

$D_0 \leftarrow \infty$

$D_1 \leftarrow E[d(X, Q^{(1)}(X))]$

while $\frac{D_{m-1} - D_m}{D_{m-1}} > \epsilon$ **do**

$m \leftarrow m + 1$

$\mathcal{R}_i^{(m)} \leftarrow \{x : d(x, \hat{x}_i^{(m)}) \leq d(x, \hat{x}_j^{(m)}), \quad j = 1, 2, \dots, N\} \quad i = 1, 2, \dots, N$

$\hat{x}_i^{(m)} \leftarrow \arg \min_{\hat{x} \in \mathcal{R}} E[d(X, \hat{x}) | X \in \mathcal{R}_i^{(m)}], \quad i = 1, 2, \dots, N$

$\mathcal{C}_m \leftarrow \{\hat{x}_1^{(m)}, \hat{x}_2^{(m)}, \dots, \hat{x}_N^{(m)}\}$

$D_m \leftarrow E[d(X, Q^{(m)}(X))]$

end while

Output: \mathcal{C}_m

The generalized Lloyd algorithm for vector quantizer design also follows the same idea as the Lloyd algorithm. The only difference is that since the input is a vector $\mathbf{x} = (x_1, \dots, x_k)$, the distortion measure is extended to k -dimensional vectors. For instance the square error distortion is defined as $d(\mathbf{x}, Q(\mathbf{x})) = \sum_{i=1}^k (x_i - \hat{x}_i)^2$, where $\hat{\mathbf{x}} \triangleq Q(\mathbf{x})$. Note that (2.12), (2.13), and (2.14) also change into vector forms

$$\mathcal{R}_i \subset \{\mathbf{x} : d(\mathbf{x}, \hat{\mathbf{x}}_i) \leq d(\mathbf{x}, \hat{\mathbf{x}}_j), j \in \mathcal{I}\} \quad i \in \mathcal{I}, \quad (2.15)$$

$$\hat{\mathbf{x}}_i = \arg \min_{\hat{\mathbf{x}} \in \mathcal{R}} E[d(\mathbf{X}, \hat{\mathbf{x}}) | \mathbf{X} \in \mathcal{R}_i], \quad i \in \mathcal{I}, \quad (2.16)$$

and

$$\hat{\mathbf{x}}_i = E[\mathbf{X} | \mathbf{X} \in \mathcal{R}_i], \quad i \in \mathcal{I}. \quad (2.17)$$

The per-sample MSE distortion for a k -dimensional vector quantizer is calculated via

$$\begin{aligned} D(Q) &= \frac{1}{k} E[d(\mathbf{X}, Q(\mathbf{X}))] \\ &= \frac{1}{k} \sum_{i \in \mathcal{I}} E[\|\mathbf{X} - \hat{\mathbf{x}}_i\|^2 | \mathbf{X} \in \mathcal{R}_i] P(\mathbf{X} \in \mathcal{R}_i) \\ &= \frac{1}{k} \sum_{i \in \mathcal{I}} \int_{\mathcal{R}_i} p(\mathbf{x}) \|\mathbf{x} - \hat{\mathbf{x}}_i\|^2 d\mathbf{x}, \end{aligned} \quad (2.18)$$

where $\|\mathbf{a} - \mathbf{b}\|^2 \triangleq \sum_{i=1}^k (a_i - b_i)^2$, for k -dimensional vectors \mathbf{a}, \mathbf{b} .

Note that in order to send the quantized values (output levels) over a channel, or store them in a storage medium, it is not necessary to send the actual quantized values. In practice, assuming that the receiver already knows the quantizer codebook \mathcal{C} , it is enough to send the index $i \in \mathcal{I}$ of an output level c_i to retrieve it in the receiver. The index i is coded to be sent over the channel. As a result, each output level c_i is in correspondence with a *codeword*. For a binary input channel, the codewords can be the binary form of the indices $i \in \mathcal{I}$.

A quantizer can also be viewed from another perspective. Consider a digital communication system with a continuous valued stochastic process $\{V_i\}$ as source (Figure 2.8). The VQ (or in general any quantizer) can be considered as an encoder/decoder pair, where the source is encoded via the mapping γ , and decoded via the mapping β

$$\begin{aligned}\gamma(\mathbf{v}) &= i \quad \text{if } \mathbf{v} \in \mathcal{R}_i, \\ \beta(j) &= \mathbf{c}_j, \quad \mathbf{c}_j \in \mathcal{C},\end{aligned}$$

where \mathbf{c}_j is the output level corresponding to index $j \in \mathcal{J}$ and \mathcal{J} is the output index set. Note that we assume here that the quantizer codewords are sent through a noiseless channel. As a result, for a quantizer model $j = i$, but for a general channel this is not the case. Since the channel is noiseless, the end-to-end distortion can be calculated via (2.18). It is interesting to note that in this case, assigning the indices to the output levels can be done arbitrarily (as long as the same assignment is used for quantization regions), with no effect on the system's end-to-end distortion. That is because a different index assignment will only change the order of integrals over the quantization regions in (2.18). Since the sum of elements is independent of the order of summation, the VQ's performance is independent of index assignment. Note that SQ is a specific case of VQ and hence the statement holds for SQ as well.

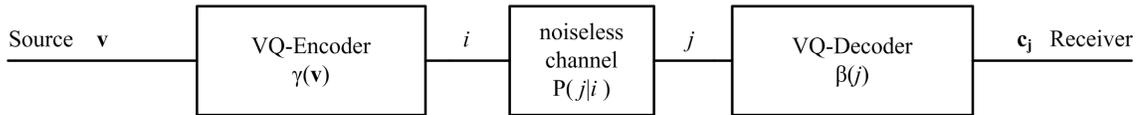


Figure 2.8: Block diagram of a vector quantizer in a communication system.

In Chapter 3, we survey the case where the channel is noisy and discuss that in that case, in contrast with the noiseless channel case, the end-to-end distortion is not

only due to quantization, but also the distortion due to channel noise and hence the index-assignment has a vital effect on the end-to-end distortion.

Chapter 3

Exploiting Channel Memory and Soft-Decision Information in COVQ Design

As mentioned earlier, where applicable, the performance of a communication system can be potentially improved by using joint source-channel coding instead of tandem coding, as well as using the channel memory, rather than discarding it via interleaving, and using soft-decision decoding, in contrast to hard decoding. In this chapter we review a well-known joint source-channel coding scheme, channel optimized vector quantization (COVQ), and present the design procedure of the COVQ for the non-binary noise discrete channel with queue based noise (NBNDQ-QB), described in Chapter 2. We numerically illustrate that such a system can successfully exploit the channel memory and soft decision information, in order to improve the performance over systems designed for ideally interleaved (hence memoryless) channels and hard-quantized output channels. We end this chapter by numerically confirming that the

NBNDC-QB is an effective model for approximating the Rayleigh DFC, described in Chapter 2, for COVQ design. The confirmation is done via fitting the NBNDC-QB to the Rayleigh DFC, designing a COVQ for the NBNDC-QB, testing the COVQ over the Rayleigh DFC, and comparing the training result of the COVQ for the NBNDC-QB with the simulation results for the Rayleigh DFC.

3.1 Channel Optimized Vector Quantization (COVQ)

In Chapter 2 we reviewed vector quantization (VQ) and its design procedure. Since VQ is a source coding method, it disregards the channel's statistics. As a result, since the channel's noise plays a vital role on the overall performance of a communication system, it is not beneficial to send the output codewords of the VQ encoder directly over the channel. Therefore, in a tandem coding system, a channel coding module is added to the system, whose aim is to make the codewords robust to channel errors at the cost of adding extra bits (and thus increasing the coding rate) to the VQ codewords. This also adds extra online computational complexity and delay to the system. However, in COVQ, as a joint source-channel coding scheme, the VQ is designed by incorporating the channel's statistics into the design algorithm. As a result, the COVQ is expected to be more robust to noisy channels, in comparison with VQ, while adding no extra redundancy to the codes, and adding less computational complexity and delay to the encoder/decoder in comparison with channel coding schemes.

The COVQ approach was first introduced by Kumazawa et al. [20]. It was further

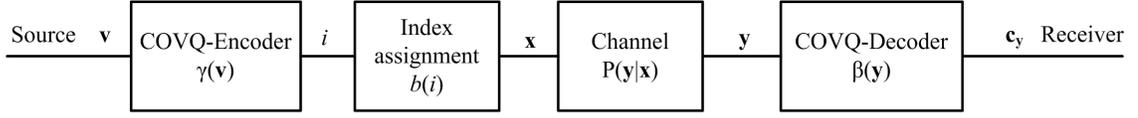


Figure 3.1: Block diagram of a COVQ system.

developed and studied by Farvardin and Vaishampayan in [12, 13]. Later in [14], it was shown that the complexity of the encoding of a COVQ is proportional to the number of non-empty encoding regions, and it was observed that as the channel gets noisier, the number of non-empty encoding regions decrease.

3.1.1 System setup

Consider the communication system depicted in Figure 3.1. The input source to the COVQ encoder is a real-valued stationary and ergodic process $\{V_i\}_{i=1}^{\infty}$. The COVQ encoder is a mapping γ that takes a vector of k source symbols $\mathbf{v} \in \mathbb{R}^k$ and outputs an index $i \in \{0, 1, \dots, 2^n - 1\}$, such that

$$\gamma(\mathbf{v}) = i \quad \text{if } \mathbf{v} \in \mathbf{S}_i,$$

where $\{\mathbf{S}_i : i \in \{0, 1, \dots, 2^n - 1\}\}$ is a partition of \mathbb{R}^k . Note that the encoder is similar to a VQ encoder, except that the quantization regions \mathbf{S}_i are chosen according to both the source and channel statistics.

The index i is then mapped to a binary vector $\mathbf{x} \in \{0, 1\}^n$ via an index assignment mapper $b(\cdot)$. The index assignment is a one-to-one mapping

$$b : \{0, 1, \dots, 2^n - 1\} \rightarrow \{0, 1\}^n, \quad b(i) = \mathbf{x}$$

where \mathbf{x} is binary n -tuple. Since the mapping is one-to-one, for a given index mapping b , we can denote the quantization regions by \mathbf{S}_x instead of \mathbf{S}_i , where $b(i) = \mathbf{x}$.

The binary n -tuples $\mathbf{x} \in \{0, 1\}^n$, each representing k source symbols, are then sent over the NBNDC-QB. As noted in Chapter 2, the channel output is 2^q -ary. Denote the channel output by \mathbf{y} . The decoder is a mapping β that maps the received n -tuple 2^q -ary \mathbf{y} to output levels of the quantizer codebook:

$$\beta(\mathbf{y}) = \mathbf{c}_y, \mathbf{c}_y \in \mathbb{R}^k, \mathbf{y} \in \{0, 1, \dots, 2^q - 1\}^n.$$

For a given k dimensional source symbol $\mathbf{v} = (v_1, \dots, v_k)$ and its corresponding retrieved output level $\mathbf{c}_y = (c_{y1}, \dots, c_{yk})$, the end-to-end per-sample distortion of the system is given by

$$d(\mathbf{v}, \mathbf{c}_y) = \frac{1}{k} \|\mathbf{v} - \mathbf{c}_y\|^2 = \frac{1}{k} \sum_{i=1}^k (v_i - c_{yi})^2.$$

3.1.2 COVQ Design

The COVQ training algorithm aims to select the codebook $\mathcal{C} = \{\mathbf{c}_i, \mathbf{i} \in \{0, 1, \dots, 2^q - 1\}^n\}$ and the partition set $\mathcal{P} = \{\mathbf{S}_i, \mathbf{i} \in \{0, 1\}^n\}$, as well as the index mapping b , to minimize the following average distortion-per-sample measure:

$$\begin{aligned} D(\mathcal{C}, \mathcal{P}, b) &= \frac{1}{k} \sum_{\mathbf{x}} \int_{\mathbf{S}_x} p(\mathbf{v}) \sum_{\mathbf{y}} P(\mathbf{y}|\mathbf{x}) d(\mathbf{v}, \mathbf{c}_y) d\mathbf{v} & (3.1) \\ &= \frac{1}{k} \sum_{\mathbf{x}} \int_{\mathbf{S}_x} p(\mathbf{v}) \sum_{\mathbf{y}} P(\mathbf{y}|\mathbf{x}) \|\mathbf{v} - \mathbf{c}_y\|^2 d\mathbf{v} \\ &= \frac{1}{k} \sum_{\mathbf{x}} \sum_{\mathbf{y}} P(\mathbf{y}|\mathbf{x}) \int_{\mathbf{S}_x} p(\mathbf{v}) \|\mathbf{v} - \mathbf{c}_y\|^2 d\mathbf{v}, \end{aligned}$$

where $p(\mathbf{v})$ is the source k -fold probability density function and $P(\mathbf{y} | \mathbf{x})$ is calculated via (2.6)-(2.8). The performance of this coding system is generally measured via

$D(\mathcal{C}, \mathcal{P}, b)$ and the encoding rate R , which is given by

$$R = \frac{1}{k} \log_2 |\mathcal{P}| = \frac{n}{k} \text{ bits/sample.}$$

It can be seen that the COVQ design problem is similar to the VQ design problem; the difference is that the codebook and partition index sets are not the same here, since the partition sets are indexed via binary n -tuples $\mathcal{I} = \{0, 1\}^n$ and the codebook is indexed via 2^q -ary n -tuples $\mathcal{J} = \{0, 1, \dots, 2^q - 1\}^n$, where q is the soft decision resolution. Another difference between the two design problems is the applied distortion measure. For a VQ the distortion is given by (2.18), while COVQ distortion is given in (3.1). Having these differences in mind, the LBG-VQ algorithm can be generalized to be used for COVQ design. In this algorithm, two necessary optimality conditions are satisfied iteratively. The conditions are called the generalized NNC and generalized CC conditions:

- *Generalized NNC* [14]: Let $\mathcal{P}^* = \{\mathbf{S}_x^*\}$ be the optimal partition set for a fixed codebook \mathcal{C} and a given index mapping b . The MSE distortion is minimized if the partitions satisfy

$$\mathbf{S}_x^* = \left\{ \mathbf{v} : \sum_{\mathbf{y} \in \mathcal{J}} P(\mathbf{y}|\mathbf{x}) \|\mathbf{v} - \mathbf{c}_y\|^2 \leq \sum_{\mathbf{y} \in \mathcal{J}} P(\mathbf{y}|\tilde{\mathbf{x}}) \|\mathbf{v} - \mathbf{c}_y\|^2, \quad \tilde{\mathbf{x}} \in \mathcal{I} \right\}, \quad \mathbf{x} \in \mathcal{I}. \quad (3.2)$$

As a result, the encoder function γ cascaded with the index mapping can be written as

$$b(\gamma(\mathbf{v})) = \arg \min_{x \in \mathcal{I}} \sum_{\mathbf{y} \in \mathcal{J}} P(\mathbf{y}|\mathbf{x}) \|\mathbf{v} - \mathbf{c}_y\|^2. \quad (3.3)$$

It can be seen that the computational complexity of encoding, using (3.3) is $\mathcal{O}(|\mathcal{I}| \times |\mathcal{J}| \times k) = \mathcal{O}(k2^{(q+1)n})$ which becomes intractable, even for moderate

values of q and n . In order to decrease the computational complexity, the following technique is proposed in [14]. For a fixed \mathcal{C} and b , the encoding region \mathbf{S}_x^* can be written as

$$\mathbf{S}_x^* = \bigcap_{x \neq t} \mathbf{S}_{xt}^*,$$

where \mathbf{S}_{xt}^* is described as follows:

$$\mathbf{S}_{xt}^* = \left\{ \mathbf{v} : 2 \sum_{y \in \mathcal{J}} [P(\mathbf{y}|\mathbf{x}) - P(\mathbf{y}|\mathbf{t})] \langle \mathbf{v}, \mathbf{c}_y \rangle \leq \sum_{y \in \mathcal{J}} [P(\mathbf{y}|\mathbf{x}) - P(\mathbf{y}|\mathbf{t})] \|\mathbf{c}_y\|^2 \right\}, \quad (3.4)$$

where $\langle \mathbf{v}, \mathbf{c}_y \rangle$ represents the inner product of \mathbf{v} and \mathbf{c}_y . According to (3.4), the region \mathbf{S}_{xt}^* is characterized by a hyperplane \mathbf{H}_{xt} , which separates the encoding regions \mathbf{S}_x^* and \mathbf{S}_t^* and is described by

$$\mathbf{H}_{xt} = \left\{ \mathbf{v} : 2 \sum_{y \in \mathcal{J}} [P(\mathbf{y}|\mathbf{x}) - P(\mathbf{y}|\mathbf{t})] \langle \mathbf{v}, \mathbf{c}_y \rangle = \sum_{y \in \mathcal{J}} [P(\mathbf{y}|\mathbf{x}) - P(\mathbf{y}|\mathbf{t})] \|\mathbf{c}_y\|^2 \right\}. \quad (3.5)$$

This fact can be used to reduce the encoding complexity in the following way.

Define for all nonempty encoding regions

$$\hat{\mathbf{v}}_x \triangleq E[\hat{\mathbf{V}}|\mathbf{X} \in \mathbf{S}_x] = \sum_{y \in \mathcal{J}} P(\mathbf{y}|\mathbf{x}) \mathbf{c}_y, \quad \mathbf{x} \in \mathcal{I}$$

and

$$\alpha_x \triangleq E[\|\hat{\mathbf{V}}\|^2|\mathbf{X} \in \mathbf{S}_x] = \sum_{y \in \mathcal{J}} P(\mathbf{y}|\mathbf{x}) \|\mathbf{c}_y\|^2, \quad \mathbf{x} \in \mathcal{I},$$

where $\hat{\mathbf{V}}$ denotes the random variable at the output of the decoder. Then, it can be shown that (3.5) can be written as

$$\mathbf{H}_{xt} = \{ \mathbf{v} : 2 \langle \mathbf{v}, \hat{\mathbf{v}}_t - \hat{\mathbf{v}}_x \rangle = \alpha_t - \alpha_x \}. \quad (3.6)$$

It can also be shown that the distance of \mathbf{v} to $\hat{\mathbf{v}}_x$, $d_x(\mathbf{v}) \triangleq \alpha_x - 2 \langle \mathbf{v}, \hat{\mathbf{v}}_x \rangle$ can be used to determine which side of \mathbf{H}_{xt} the point \mathbf{v} belongs to. In other words, to

determine the quantization region for a source symbol \mathbf{v} , the following relation can be used as the encoder function

$$b(\gamma(\mathbf{v})) = \arg \min_{x \in \mathcal{I}} d_{\mathbf{x}}(\mathbf{v}) = \arg \min_{x \in \mathcal{I}} (\alpha_{\mathbf{x}} - 2\langle \mathbf{v}, \hat{\mathbf{v}}_{\mathbf{x}} \rangle). \quad (3.7)$$

Using this technique, the parameters $\hat{\mathbf{v}}_{\mathbf{x}}$ and $\alpha_{\mathbf{x}}$ are calculated only once with complexities $\mathcal{O}(2^{nq})$ and $\mathcal{O}(k2^{nq})$ respectively, and for each source vector \mathbf{v} to be encoded, the complexity is $\mathcal{O}(k2^n)$. It can be seen that the second technique has a notable speedup over the technique that uses (3.3) for encoding, specially for higher values of q . Furthermore, it is important to note that we only need to calculate $d_{\mathbf{x}}(\mathbf{v})$ for the non empty regions, whose number increases as the channel gets noisier.

- *Generalized CC* [14]: Similar to the CC condition, the generalized CC condition states that for a fixed index mapping b and fixed partition set $\mathcal{P} = \{\mathbf{S}_{\mathbf{x}}\}, \mathbf{x} \in \mathcal{I}$, the optimal codebook $\mathcal{C}^* = \{\mathbf{c}_{\mathbf{y}}\}, \mathbf{y} \in \mathcal{J}$ satisfies

$$\mathbf{c}_{\mathbf{y}}^* = \arg \min_{\hat{\mathbf{v}} \in \mathbb{R}^k} E\{d(\mathbf{V}, \hat{\mathbf{v}}) | \mathbf{Y} = \mathbf{y}\}, \quad \mathbf{y} \in \mathcal{J}, \quad (3.8)$$

where \mathbf{Y} denotes the random vector at the channel output. For the MSE distortion, this relation is simplified to

$$\mathbf{c}_{\mathbf{y}}^* = \frac{\sum_{\mathbf{x}} P(\mathbf{y}|\mathbf{x}) \int_{\mathbf{S}_{\mathbf{x}}} \mathbf{v} f(\mathbf{v}) d\mathbf{v}}{\sum_{\mathbf{x}} P(\mathbf{y}|\mathbf{x}) \int_{\mathbf{S}_{\mathbf{x}}} f(\mathbf{v}) d\mathbf{v}}. \quad (3.9)$$

Similar to the Lloyd algorithm, a successive application of (3.2) and (3.9) will lead to a non-increasing sequence of distortions. Hence the algorithm has to converge. Therefore, we start with an initial point and iteratively satisfy one of the two aforementioned conditions, in succession.

To select the initial codebook \mathcal{C}^0 we have used the splitting algorithm [12,21]. This method, which is also used for initial codebook selection in VQ and SQ design, first designs a VQ (COVQ with noiseless channel) with only one output level $\mathcal{C}_1^0 \triangleq \{\mathbf{c}_0\}$ using the CC. Then, the number of output levels are doubled to 2 by letting $\mathcal{C}_2^0 = \{\mathbf{c}_0 + \mathbf{d}, \mathbf{c}_0 - \mathbf{d}\}$, where \mathbf{d} is a vector with very small values in each coordinate, and training the quantizer using the initial codebook \mathcal{C}_2^0 . After the algorithm converges to the locally optimum code $\mathcal{C}_2^* = \{\mathbf{c}_0, \mathbf{c}_1\}$, the number of output levels is doubled to 4, by letting $\mathcal{C}_4^0 = \{\mathbf{c}_0 + \mathbf{d}, \mathbf{c}_0 - \mathbf{d}, \mathbf{c}_1 + \mathbf{d}, \mathbf{c}_1 - \mathbf{d}\}$, and again training the quantizer for 4 output levels. This procedure is continued until an initial codebook with desired number of output levels (i.e., 2^n) is obtained.

To design the COVQ we also need to address the index assignment problem. In Chapter 2 we mentioned that for a VQ (over a noiseless channel) the order of assigning indices to code levels does not affect the performance of the system. This is not the case when the channel is noisy. According to the distortion expression (3.1) for a COVQ, it can be shown [12] that for the MSE distortion and for a COVQ satisfying the generalized CC, the average end-to-end distortion can be broken into two terms

$$D_{COVQ} = D_{VQ} + D_C,$$

where

$$D_{VQ} = \frac{1}{k} \sum_{\mathbf{x} \in \mathcal{I}} \int_{\mathbf{S}_{\mathbf{x}}} p(\mathbf{v}) \|\mathbf{v} - \mathbf{c}_{\mathbf{x}}\|^2 d\mathbf{v} \quad (3.10)$$

where $\mathbf{c}_{\mathbf{x}}$ represents the output level from the codebook \mathcal{C} , the output of the COVQ decoder, when the channel is noiseless and $\mathbf{v} \in \mathbf{S}_{\mathbf{x}}$. In other words, D_{VQ} is the expected distortion when the channel deterministically outputs $\mathbf{y} \in \mathcal{J}$ corresponding to each binary vector $\mathbf{x} \in \mathcal{I}$ sent over the channel. The other term D_C is the distortion due to the channel noise. Recalling that $\gamma(\mathbf{v}) = i$ if $\mathbf{v} \in \mathbf{S}_i$ and $\mathbf{x} = b(i)$, D_C is given

by

$$D_C = \frac{1}{k} \sum_{i=0}^{2^n-1} \sum_{\mathbf{y} \in \mathcal{J}} Pr\{\mathbf{v} \in \mathbf{S}_i\} P(\mathbf{y}|b(i)) \|\mathbf{c}_{b(i)} - \mathbf{c}_{\mathbf{y}}\|^2. \quad (3.11)$$

It can be seen that for a different index mapping $\mathbf{x}' = b'(i)$, the distortion caused by the channel can be different. As a result, it is important to choose the mapping b judiciously, in order to minimize the channel distortion D_C . We have used the simulated annealing algorithm, as suggested in [12], to perform this minimization.

The simulated annealing (SA) algorithm is a well-known probabilistic algorithm [7, 19] which is shown to yield a good approximation to a globally optimum solution in a large search space. At each step, the SA algorithm replaces the current solution by a random nearby solution, chosen with a probability that depends both on the difference between the corresponding function values and also on a global parameter called temperature T , which is gradually decreased during the process. At the final steps with T close to zero, the search is more local (downhill), while at the beginning (high temperatures) it is more likely to have random solutions selected from the state space to avoid being trapped in a local optima. Algorithm 3.1 generally describes how the SA algorithm is applied to the index assignment problem. In this algorithm, the system state is defined as a particular mapping $\mathbf{b} = (b(1), b(2), \dots, b(2^N))$, and the energy function is defined as the channel distortion D_C of the system, given in (3.11). Some minor modifications can be made to the general algorithm, such as terminating the algorithm if the number of trials exceeds an specific number, implying relative stability of the state. The SA algorithm parameters we have used in this thesis are the same as those used in [12] and are listed in Table 3.1

To select the initial codebook, at first we consider the error free channel and

Algorithm 3.1 Simulated annealing algorithm for index assignment

Input: random initial state \mathbf{b} , initial temperature T_0 , temperature decrease factor α , final temperature T_f , and number of failed trials allowed, before temperature decrease N_{cut}

$T \leftarrow T_0$

$count \leftarrow 0$

while $T > T_f$ **do**

while $count < N_{cut}$ **do**

 choose state \mathbf{b}' by randomly perturbing \mathbf{b}

$\Delta D_C \leftarrow D_C(\mathbf{b}') - D_C(\mathbf{b})$

if $\Delta D_C \leq 0$ **then**

$\mathbf{b} \leftarrow \mathbf{b}'$

$count \leftarrow 0$

else {with probability $e^{\Delta D_C/T}$, do}

$\mathbf{b} \leftarrow \mathbf{b}'$

$count \leftarrow 0$

end if

$count \leftarrow count + 1$

end while

$T \leftarrow \alpha T$

end while

Output: \mathbf{b}

train the COVQ with the initial codebook obtained from the splitting algorithm, followed by simulated annealing for a locally optimum index assignment. Then we use the resulting codebook as the initial state for a channel with high SNR. After the training for high SNR channel is done, we decrease the channel SNR slightly and train the COVQ again, setting the previously trained codebook (for higher SNR) as the initial state of the system with the new SNR. We continue this process until we eventually reach the desired channel SNR.

Table 3.1: Simulated annealing parameters.

T_0	10.0
T_f	2.5×10^{-4}
α	0.97
N_{cut}	200

3.2 Numerical Results

We now present the numerical results obtained using the training algorithm and the NBNDC-QB channel model described previously.

Several source distributions were tested, including independent and identically distributed (i.i.d.) Gaussian and Laplacian sources and correlated Gauss-Markov sources. The correlated source was modeled via a Markov process of first-order :

$$V_i = \phi V_{i-1} + U_i$$

where $\phi \in (-1, 1)$ is the correlation parameter and $\{U_i\}$ is a Gaussian i.i.d. process. All of the source models had zero mean and unit variance.

For each source model, the COVQ was trained using 500,000 source vectors.

3.2.1 Exploiting memory and soft-decision information

Tables 3.2 and 3.3 depict COVQ training results for the memoryless Gaussian source and NBNDC-QB with different channel noise correlation coefficients. The NBNDC-QB noise one-dimensional distribution ρ_j is expressed in terms of the SNR of the underlying DFC by setting $\rho_j = P_{DFC}^{(1)}(j)$, where $P_{DFC}^{(1)}(j)$ is given in (2.10) for $j \in \{0, 1, \dots, 2^q - 1\}$ and δ is chosen as in [28] to maximize the DFC capacity. The values of ρ_j and δ are shown in Table 3.14. In practice, the memoryless channel behavior

is realized by interleaving. It can be shown that by ideal (infinite) interleaving, a correlated channel is reduced to its memoryless counterpart. It can be seen that the highly correlated channel consistently outperforms the memoryless channel and gains of more than 5 dB are achievable (e.g., for $q = 1$, $R = 3$, $k = 3$, SNR=0 in Table 3.2), using the channel memory. Note that since the COVQ only makes use of intra-block memory, for rate $R = 1$ and low dimensions k , the block length is so small that there is not much channel memory to be harnessed. As a result, the performance is constant for different channel correlations. However, it is observed that in some cases, interleaving may give better COVQ performance over channels with lower noise correlations. For example, the SDR performance of the memoryless channel for $q = 3$, $R = 3$, $k = 2$ (see Tables 3.2 and 3.3,) is slightly better than the corresponding performance for noise correlations 0.5 and 0.7 at high SNRs. Since the capacity of the correlated channel is strictly higher than that of the memoryless channel, this degradation may be due to poor selection of the initial codebook for the vector quantizer. However, the results indicate a general trend of improved COVQ performance when the channel noise correlation is increased. Similar observations can be made for sources with Laplacian or Gauss-Markov distributions (see Tables 3.4 and 3.5).

Additionally, it is observed in Tables 3.2-3.5 that the system with soft-decision quantization ($q > 1$), considerably outperforms hard-quantization ($q = 1$), by as much as 2.3 dB (for $R = 3$, $k = 1$, Cor = 0, SNR=5) for a memoryless Gaussian source, when using only a 2-bit soft-decision quantizer ($q = 2$). For 3-bit quantization further gain is obtained, although the gain for $q = 3$ over $q = 2$ is less than the gain for $q = 2$ over $q = 1$.

Comparing the overall performance of a COVQ system that uses the combination of memory and soft-decision information to a system that uses interleaving to discard channel memory and hard-decision quantization in the output, it can be seen that gains more than 5.7 dB (at $R = 3$, $k = 3$, SNR=0) can be achieved for memoryless Gaussian sources. More gain is obtained for correlated Gauss-Markov sources. It is important to notice that in a uniform soft-quantized output system, the SDR performance also depends on the soft-quantizer step size δ . However, according to our results, the system's performance is not very sensitive to the choice of δ . In this work, we selected the δ values that maximize the capacity of the memoryless channel as the initial choice and then tried to improve the performance by applying a local search over the δ values.

In Tables 3.6-3.8, the training results for memoryless Gaussian source with different channel parameters α and M are depicted. Table 3.9 shows the results of using the same COVQ design technique, with a slight modification in the initial codebook selection. More precisely, in the modified version, after training the COVQ for the lowest SNR, the resulting codebook was again used as an initial codebook for designing a new COVQ with higher SNR and the training is continued until the COVQ for the highest SNR is trained. We did the same decrease-increase method [6] 5 times to obtain the final COVQ codebooks for desired SNRs. As can be seen, the results show a good improvement (up to 2.1 dB at $q = 1$, $R = 3$, $k = 1$, Cor=0.9, SNR=10) at the cost of extra complexity in the training algorithm.

3.2.2 Validating the NBNDC-QB model

As mentioned earlier, the m -fold probability distribution of correlated Rayleigh random variables is not known in closed form, for $m > 3$. As a result, the channel transition distribution $P_{DFC}^{(m)}(y^m|x^m)$ can only be calculated numerically. It is shown in [28] that the NBNDC-QB model can approximate the Rayleigh DFC in terms of channel capacity and noise autocorrelation function, while providing closed form expression for the channel transition distribution for arbitrary channel block length m and memory order M .

Approximating a given Rayleigh DFC (with fixed SNR and $f_D T$ and q) via the NBNDC-QB is done via the following steps:

- Matching the noise one-dimensional probability distributions (as in Section 3.2.1) by setting $\rho_j = P_{DFC}^{(1)}(j)$ for $j \in \mathcal{Y}$, where $P_{DFC}^{(1)}(j)$ is given by (2.10), in terms of δ , q , and SNR. The values of ρ_j are given in Table 3.14.
- Matching the noise correlation coefficients (so that parameter α is determined).
- The remaining QB parameters (M, ε) are estimated by minimizing the Kullback-Leibler divergence rate between the two (2^q -ary) noise processes.

It is important to notice that in general, two channel models can not be matched to have the exact same behavior. However, for the memoryless case (with $\text{Cor}=0$), the NBNDC-QB is statistically identical to the ideally interleaved DFC. We have used the values given in [29] in which the Kullback-Leibler divergence rate between the two channel (2^q -ary) noise processes is minimized over M and ε for $\text{SNR}_{(dB)} \in \{2.0, 5.0, 10.0, 15.0\}$, $f_D T \in \{0.005, 0.01\}$, and $q = 2$. The resulting values for $q = 2$, given in Table 3.10, are also used for case of $q = 1$.

Using the approximated NBNDQ-QB to fit a given Rayleigh DFC, we trained a COVQ using the previously described COVQ design algorithm. The resulting channel optimized quantizer's performance was then tested over the given Rayleigh DFC. For generating the fading coefficients, we used the modified Clarke's method introduced in [36]. Training and simulation results (over the NBNDQ-QB and Rayleigh DFC channels) in terms of SDR are shown in Tables 3.12 and 3.13 for memoryless and correlated Gaussian sources, and in Table 3.11 for a memoryless Laplacian source. The channel parameters used for training/simulation are given in Table 3.10. Comparing the training and simulation performance of the COVQ, we observe that there is a good conformity between the results of the two channel models, where the NBNDQ-QB is used for training and the Rayleigh DFC for testing. However, for higher rates, some degradation between the simulation and training results is observed.

Table 3.2: COVQ Training SDR results (in dB) for the memoryless NBNDC-QB and the highly correlated NBNDC-QB with parameters $\alpha = 1.0$, $M = 1$; memoryless Gaussian source.

q	R	k	Memoryless (Cor=0)					Cor=0.9				
			SNR (dB)					SNR (dB)				
			15	10	5	2	0	15	10	5	2	0
1	1	1	4.18	3.77	2.88	2.16	1.67	4.18	3.77	2.88	2.16	1.67
		2	4.16	3.75	2.87	2.15	1.66	4.16	3.75	2.87	2.15	1.66
		3	4.23	3.78	2.87	2.15	1.67	4.27	3.88	3.64	3.26	2.97
	2	1	8.16	6.58	4.23	2.85	2.06	8.35	7.05	5.24	5.70	5.09
		2	8.32	6.72	4.88	3.67	2.88	8.55	7.32	7.05	6.13	5.82
		3	8.57	7.12	5.10	3.78	2.95	8.81	8.37	7.49	6.91	6.47
	3	1	11.12	8.09	4.83	4.45	3.57	11.71	9.68	9.45	8.04	7.22
		2	11.64	9.30	6.60	4.90	3.88	12.09	11.63	9.98	8.89	8.24
			3	11.99	9.76	6.90	5.08	3.97	12.54	12.40	10.79	9.69
2	1	1	4.23	3.88	3.10	2.43	1.95	4.23	3.88	3.10	2.43	1.95
		2	4.21	3.87	3.09	2.42	1.94	4.21	3.87	3.09	2.42	1.94
		3	4.28	3.90	3.09	2.42	1.95	4.31	3.99	3.77	3.43	3.17
	2	1	8.37	7.00	4.75	3.35	2.51	8.54	7.40	5.62	6.13	5.52
		2	8.55	7.13	5.46	4.29	3.41	8.76	7.67	7.48	6.75	6.20
		3	8.78	7.59	5.76	4.42	3.54	9.02	7.79	7.50	6.94	6.77
	3	1	11.67	8.81	7.20	5.34	4.13	12.18	11.24	9.57	8.46	7.76
		2	12.42	10.37	7.73	5.88	4.61	12.51	11.74	10.27	9.52	8.87
			3	12.57	10.71	8.01	6.09	4.85	13.13	12.33	11.13	10.30
3	1	1	4.24	3.91	3.16	2.49	2.01	4.24	3.91	3.16	2.49	2.01
		2	4.22	3.89	3.15	2.48	2.00	4.22	3.89	3.15	2.48	2.00
		3	4.29	3.93	3.15	2.48	2.01	4.32	4.01	3.81	3.48	3.22
	2	1	8.42	7.08	4.89	3.46	2.61	8.58	7.48	5.73	6.24	5.64
		2	8.60	7.22	5.61	4.43	3.55	8.80	7.74	7.43	6.90	6.42
		3	8.83	7.69	5.91	4.57	3.68	9.07	7.86	7.50	6.83	6.48
	3	1	11.79	8.96	7.45	5.60	4.35	12.28	10.41	9.01	8.44	7.90
		2	12.57	10.61	7.96	6.15	4.85	12.63	11.58	10.35	9.46	9.10

Table 3.3: COVQ Training SDR results (in dB) for the NBNDQ-QB with parameters $\alpha = 1.0$, $M = 1$ and channel correlations 0.5, 0.7 ; memoryless Gaussian source.

q	R	k	Cor=0.5					Cor=0.7				
			SNR (dB)					SNR (dB)				
			15	10	5	2	0	15	10	5	2	0
1	1	1	4.18	3.77	2.88	2.16	1.67	4.18	3.77	2.88	2.16	1.67
		2	4.16	3.75	2.87	2.15	1.66	4.16	3.75	2.87	2.15	1.66
		3	4.25	3.82	2.92	2.20	1.72	4.26	3.85	2.99	2.69	2.31
	2	1	8.26	6.81	4.66	3.37	2.64	8.30	6.92	4.91	3.75	3.69
		2	8.44	6.99	4.91	3.68	2.91	8.49	7.13	5.29	4.59	4.07
		3	8.65	7.09	5.04	3.94	3.27	8.72	7.17	6.15	5.11	4.46
	3	1	11.34	8.56	5.56	4.01	3.20	11.50	8.96	6.51	5.26	4.41
		2	11.64	9.13	6.35	4.72	4.21	11.78	9.25	7.79	6.49	5.70
		3	11.79	9.48	6.88	5.41	4.47	11.88	10.48	8.19	6.81	6.00
2	1	1	4.23	3.88	3.10	2.43	1.95	4.23	3.88	3.10	2.43	1.95
		2	4.21	3.87	3.09	2.42	1.94	4.21	3.87	3.09	2.42	1.94
		3	4.30	3.94	3.14	2.47	2.00	4.30	3.97	3.20	2.93	2.57
	2	1	8.47	7.20	5.11	3.84	3.10	8.50	7.30	5.34	4.20	4.17
		2	8.66	7.34	5.48	4.30	3.75	8.71	7.45	5.84	5.08	4.45
		3	8.87	7.53	5.80	4.65	3.89	8.94	7.57	6.14	5.57	5.07
	3	1	11.93	9.21	6.30	4.75	3.90	12.04	9.53	7.28	6.05	5.27
		2	12.21	10.22	7.74	5.98	5.14	12.34	10.18	8.36	7.20	6.59
		3	12.51	10.67	8.24	6.50	5.47	12.62	10.97	9.05	7.78	7.03
3	1	1	4.24	3.91	3.16	2.49	2.01	4.24	3.91	3.16	2.49	2.01
		2	4.22	3.89	3.15	2.48	2.00	4.22	3.89	3.15	2.48	2.00
		3	4.31	3.97	3.21	2.54	2.07	4.31	3.99	3.26	3.00	2.66
	2	1	8.51	7.28	5.26	3.97	3.24	8.55	7.37	5.47	4.33	4.34
		2	8.71	7.42	5.71	4.49	4.02	8.76	7.52	6.02	5.26	4.68
		3	8.93	7.62	6.01	4.92	4.19	8.99	7.64	6.27	5.61	5.06
	3	1	12.05	9.36	6.59	5.04	4.24	12.16	9.65	7.57	6.42	5.61
		2	12.34	10.48	8.09	6.46	5.51	12.47	10.57	8.69	7.68	6.85

Table 3.4: COVQ Training SDR results (in dB) for the memoryless NBNDC-QB and the highly correlated NBNDC-QB with parameters $\alpha = 1.0$, $M = 1$; Gauss-Markov source with correlation parameter $\phi = 0.9$.

q	R	k	Memoryless (Cor=0)					Cor=0.9				
			SNR (dB)					SNR (dB)				
			15	10	5	2	0	15	10	5	2	0
1	1	1	4.17	3.76	2.87	2.14	1.66	4.17	3.76	2.87	2.14	1.66
		2	7.09	5.85	3.87	2.63	1.91	7.23	6.23	4.74	5.13	4.61
		3	8.05	6.63	5.00	3.69	3.10	8.26	7.30	7.22	6.38	5.85
	2	1	8.17	6.57	4.22	2.83	2.05	8.36	7.05	5.23	5.70	5.08
		2	10.66	8.79	6.61	5.01	3.93	11.46	11.25	9.40	8.27	7.59
		3	12.03	10.31	7.70	5.75	4.81	12.53	12.08	10.60	9.88	9.37
	3	1	11.14	8.10	4.82	4.44	3.56	11.73	9.70	9.53	8.21	7.44
		2	13.93	11.62	8.33	6.43	5.02	16.00	14.38	12.28	11.20	10.50
		3	15.53	13.10	9.81	7.71	6.33	17.25	15.82	13.80	12.63	11.96
2	1	1	4.22	3.87	3.09	2.42	1.94	4.22	3.87	3.09	2.42	1.94
		2	7.25	6.19	4.32	3.08	2.32	7.37	6.51	5.06	5.49	4.98
		3	8.30	7.40	5.84	4.48	3.56	8.48	7.59	7.48	6.77	6.30
	2	1	8.38	7.00	4.74	3.34	2.49	8.55	7.41	5.61	6.13	5.51
		2	11.32	9.90	7.55	6.00	4.83	11.71	11.45	10.35	9.46	8.90
		3	12.79	11.15	8.84	7.12	5.84	13.21	12.50	11.41	10.63	10.00
	3	1	11.69	8.82	7.19	5.33	4.12	12.21	11.12	9.82	8.74	8.04
		2	15.06	12.85	9.83	7.86	6.30	16.13	14.99	13.30	12.15	11.38
		3	16.26	14.30	11.41	9.18	7.64	17.32	16.27	14.75	13.69	13.12
3	1	1	4.23	3.90	3.15	2.48	2.00	4.23	3.90	3.15	2.48	2.00
		2	7.28	6.26	4.44	3.19	2.41	7.40	6.56	5.15	5.58	5.08
		3	8.35	7.51	6.00	4.66	3.74	8.53	7.64	7.60	6.94	6.49
	2	1	8.43	7.08	4.89	3.45	2.60	8.59	7.48	5.72	6.24	5.64
		2	11.42	10.15	7.51	5.94	5.09	11.78	10.80	10.03	9.13	8.71
		3	12.94	11.44	9.17	7.25	6.16	13.22	12.57	11.51	10.98	10.47
	3	1	11.81	8.97	7.45	5.59	4.34	12.31	10.40	10.01	8.98	8.31
		2	15.33	13.22	10.15	8.11	6.56	15.61	14.62	13.15	12.18	11.57

Table 3.5: COVQ Training SDR results (in dB) for the memoryless NBNDQ-QB and the highly correlated NBNDQ-QB with parameters $\alpha = 1.0$, $M = 1$; memoryless Laplacian source.

q	R	k	Memoryless (Cor=0)					Cor=0.9				
			SNR (dB)					SNR (dB)				
			15	10	5	2	0	15	10	5	2	0
1	1	1	2.87	2.63	2.07	1.58	1.25	2.87	2.63	2.07	1.58	1.25
		2	3.46	3.12	2.42	1.84	1.44	3.48	3.16	2.45	1.85	1.44
		3	4.06	3.46	2.44	1.90	1.48	4.26	3.93	3.38	2.91	2.60
	2	1	6.64	5.39	3.48	2.32	1.67	6.71	5.64	4.12	4.34	3.83
		2	7.67	6.18	4.18	3.12	2.39	7.76	7.08	6.44	5.62	5.12
		3	8.11	6.61	4.55	3.38	2.59	8.40	8.16	7.07	6.38	5.88
	3	1	9.70	7.08	4.21	3.54	2.66	9.84	9.15	8.01	6.78	6.07
		2	10.92	8.52	5.96	4.36	3.37	11.27	10.91	9.33	8.17	7.71
			3	11.40	9.25	6.41	4.65	3.59	12.26	11.97	10.23	9.10
2	1	1	2.90	2.70	2.21	1.77	1.44	2.90	2.70	2.21	1.77	1.44
		2	3.50	3.22	2.59	2.06	1.67	3.52	3.25	2.63	2.08	1.68
		3	4.14	3.63	2.66	2.03	1.72	4.31	4.04	3.46	3.10	2.81
	2	1	6.85	5.79	3.96	2.77	2.06	6.95	6.05	4.48	4.74	4.21
		2	7.91	6.59	4.88	3.73	2.91	8.06	7.01	6.69	5.98	5.46
		3	8.37	7.13	5.24	3.94	3.14	8.66	7.62	7.25	6.51	6.18
	3	1	10.32	7.84	5.97	4.41	3.38	10.63	8.46	8.45	7.31	6.61
		2	11.45	9.51	7.04	5.30	4.18	11.71	10.52	9.02	8.51	8.01
			3	12.08	10.13	7.52	5.62	4.44	12.53	11.58	10.36	9.65
3	1	1	2.90	2.71	2.25	1.81	1.48	2.90	2.71	2.25	1.81	1.48
		2	3.51	3.24	2.64	2.11	1.72	3.53	3.27	2.68	2.14	1.73
		3	4.16	3.66	2.73	2.08	1.77	4.31	4.06	3.51	3.15	2.86
	2	1	6.89	5.86	4.08	2.88	2.15	6.99	6.12	4.59	4.84	4.31
		2	7.95	6.67	5.02	3.86	3.03	8.11	7.01	6.72	6.23	5.81
		3	8.42	7.26	5.40	4.08	3.30	8.71	7.53	7.20	6.60	6.21
	3	1	10.43	7.99	6.18	4.62	3.56	10.77	8.59	7.57	7.07	6.60
		2	11.59	9.72	7.26	5.53	4.39	11.87	10.41	9.18	8.61	8.24

Table 3.6: COVQ Training SDR results (in dB) for the memoryless NBNDC-QB and the highly correlated NBNDC-QB with parameters $\alpha = 0.5$, $M = 5$; memoryless Gaussian source.

q	R	k	Memoryless (Cor=0)					Cor=0.9				
			SNR (dB)					SNR (dB)				
			15	10	5	2	0	15	10	5	2	0
1	1	1	4.18	3.77	2.88	2.16	1.67	4.18	3.77	2.88	2.16	1.67
		2	4.16	3.75	2.87	2.15	1.66	4.16	3.75	2.87	2.15	1.66
		3	4.23	3.78	2.87	2.15	1.67	4.27	3.88	3.69	3.30	3.03
	2	1	8.16	6.58	4.23	2.85	2.06	8.35	7.05	5.24	5.70	5.09
		2	8.32	6.72	4.88	3.67	2.88	8.56	7.35	7.40	6.53	6.01
		3	8.57	7.12	5.10	3.78	2.95	8.85	8.73	7.74	6.92	6.41
	3	1	11.12	8.09	4.83	4.45	3.57	11.71	9.72	9.31	8.30	7.51
		2	11.64	9.30	6.60	4.90	3.88	12.22	12.43	10.28	8.96	8.26
			3	11.98	9.76	6.90	5.08	3.97	13.75	12.65	10.55	9.35
2	1	1	4.23	3.88	3.10	2.43	1.95	4.23	3.88	3.10	2.43	1.95
		2	4.21	3.87	3.09	2.42	1.94	4.21	3.87	3.09	2.42	1.94
		3	4.28	3.90	3.09	2.42	1.95	4.31	3.99	3.31	3.47	3.21
	2	1	8.55	7.13	5.46	4.29	3.41	8.76	7.70	7.76	7.00	6.49
		2	8.37	7.00	4.75	3.35	2.51	8.54	7.40	5.62	6.13	5.52
		3	8.78	7.59	5.76	4.42	3.54	9.05	8.90	8.05	7.30	6.82
	3	1	11.67	8.81	7.20	5.34	4.13	12.18	12.02	10.06	8.84	8.09
		2	12.42	10.37	7.73	5.88	4.61	12.56	12.69	10.77	9.60	8.92
			3	12.57	10.71	8.01	6.09	4.85	13.93	12.91	11.19	10.14
3	1	1	4.24	3.91	3.16	2.49	2.01	4.24	3.91	3.16	2.49	2.01
		2	4.22	3.89	3.15	2.48	2.00	4.22	3.89	3.15	2.48	2.00
		3	4.29	3.93	3.15	2.48	2.01	4.32	4.02	3.36	3.51	3.26
	2	1	8.42	7.08	4.89	3.46	2.61	8.58	7.48	5.73	6.24	5.64
		2	8.60	7.22	5.61	4.43	3.55	8.80	7.77	7.86	7.12	6.63
		3	8.83	7.69	5.91	4.57	3.68	9.09	8.94	8.12	7.44	7.01
	3	1	11.79	8.96	7.45	5.60	4.35	12.29	11.45	9.66	8.85	8.21
		2	12.57	10.61	7.96	6.15	4.85	12.68	12.26	10.96	9.89	9.22

Table 3.7: COVQ Training SDR results (in dB) for the memoryless NBNDC-QB and the highly correlated NBNDC-QB with parameters $\alpha = 1.0$, $M = 5$; memoryless Gaussian source.

q	R	k	Memoryless (Cor=0)					Cor=0.9				
			SNR (dB)					SNR (dB)				
			15	10	5	2	0	15	10	5	2	0
1	1	1	4.18	3.77	2.88	2.16	1.67	4.18	3.77	2.88	2.16	1.67
		2	4.16	3.75	2.87	2.15	1.66	4.16	3.75	2.87	2.15	1.66
		3	4.23	3.78	2.87	2.15	1.67	4.27	3.88	3.69	3.30	3.03
	2	1	8.16	6.58	4.23	2.85	2.06	8.35	7.05	5.24	5.70	5.09
		2	8.32	6.72	4.88	3.67	2.88	8.56	7.35	7.40	6.53	6.01
		3	8.57	7.12	5.10	3.78	2.95	8.85	8.73	7.74	6.93	6.42
	3	1	11.12	8.09	4.83	4.45	3.57	11.71	9.72	9.31	8.30	7.51
		2	11.64	9.30	6.60	4.90	3.88	12.22	12.36	10.36	9.03	8.27
			3	11.98	9.76	6.90	5.08	3.97	13.73	12.70	10.57	9.35
2	1	1	4.23	3.88	3.10	2.43	1.95	4.23	3.88	3.10	2.43	1.95
		2	4.21	3.87	3.09	2.42	1.94	4.21	3.87	3.09	2.42	1.94
		3	4.28	3.90	3.09	2.42	1.95	4.31	3.99	3.31	3.47	3.21
	2	1	8.37	7.00	4.75	3.35	2.51	8.54	7.40	5.62	6.13	5.52
		2	8.55	7.13	5.46	4.29	3.41	8.76	7.70	7.76	7.00	6.49
		3	8.78	7.59	5.76	4.42	3.54	9.05	8.94	8.03	7.30	6.84
	3	1	11.67	8.81	7.20	5.34	4.13	12.18	12.02	10.06	8.84	8.09
		2	12.42	10.37	7.73	5.88	4.61	12.56	12.68	10.76	9.62	8.93
			3	12.57	10.71	8.01	6.09	4.85	13.99	12.94	11.20	10.16
3	1	1	4.24	3.91	3.16	2.49	2.01	4.24	3.91	3.16	2.49	2.01
		2	4.22	3.89	3.15	2.48	2.00	4.22	3.89	3.15	2.48	2.00
		3	4.29	3.93	3.15	2.48	2.01	4.32	4.02	3.36	3.51	3.26
	2	1	8.42	7.08	4.89	3.46	2.61	8.58	7.48	5.73	6.24	5.64
		2	8.60	7.22	5.61	4.43	3.55	8.80	7.77	7.86	7.12	6.63
		3	8.83	7.69	5.91	4.57	3.68	9.09	8.96	8.15	7.43	6.97
	3	1	11.79	8.96	7.45	5.60	4.35	12.29	11.45	9.66	8.85	8.21
		2	12.57	10.61	7.96	6.15	4.85	12.68	12.20	10.89	9.87	9.22

Table 3.8: COVQ Training SDR results (in dB) for the memoryless NBNDC-QB and the highly correlated NBNDC-QB with parameters $\alpha = 2.0$, $M = 5$; memoryless Gaussian source.

q	R	k	Memoryless (Cor=0)					Cor=0.9				
			SNR (dB)					SNR (dB)				
			15	10	5	2	0	15	10	5	2	0
1	1	1	4.18	3.77	2.88	2.16	1.67	4.18	3.77	2.88	2.16	1.67
		2	4.16	3.75	2.87	2.15	1.66	4.16	3.75	2.87	2.15	1.66
		3	4.23	3.78	2.87	2.15	1.67	4.27	3.88	3.69	3.30	3.03
	2	1	8.16	6.58	4.23	2.85	2.06	8.35	7.05	5.24	5.70	5.09
		2	8.32	6.72	4.88	3.67	2.88	8.56	7.35	7.40	6.53	6.01
		3	8.57	7.12	5.10	3.78	2.95	8.85	8.78	7.72	6.91	6.40
	3	1	11.12	8.09	4.83	4.45	3.57	11.71	9.72	9.31	8.30	7.51
		2	11.64	9.30	6.60	4.90	3.88	12.23	12.49	10.36	9.08	8.32
			3	11.98	9.76	6.90	5.08	3.97	13.76	12.74	10.63	9.40
2	1	1	4.23	3.88	3.10	2.43	1.95	4.23	3.88	3.10	2.43	1.95
		2	4.21	3.87	3.09	2.42	1.94	4.21	3.87	3.09	2.42	1.94
		3	4.28	3.90	3.09	2.42	1.95	4.31	3.99	3.31	3.47	3.21
	2	1	8.37	7.00	4.75	3.35	2.51	8.54	7.40	5.62	6.13	5.52
		2	8.55	7.13	5.46	4.29	3.41	8.76	7.70	7.76	7.00	6.49
		3	8.78	7.59	5.76	4.42	3.54	9.05	8.94	8.06	7.31	6.84
	3	1	11.67	8.81	7.20	5.34	4.13	12.18	12.02	10.06	8.84	8.09
		2	12.42	10.37	7.73	5.88	4.61	12.56	12.69	10.80	9.62	8.95
			3	12.57	10.71	8.01	6.09	4.85	14.05	12.97	11.25	10.23
3	1	1	4.24	3.91	3.16	2.49	2.01	4.24	3.91	3.16	2.49	2.01
		2	4.22	3.89	3.15	2.48	2.00	4.22	3.89	3.15	2.48	2.00
		3	4.29	3.93	3.15	2.48	2.01	4.32	4.02	3.36	3.51	3.26
	2	1	8.42	7.08	4.89	3.46	2.61	8.58	7.48	5.73	6.24	5.64
		2	8.60	7.22	5.61	4.43	3.55	8.80	7.77	7.86	7.12	6.63
		3	8.83	7.69	5.91	4.57	3.68	9.09	8.96	8.10	7.45	7.00
	3	1	11.79	8.96	7.45	5.60	4.35	12.29	11.45	9.66	8.85	8.21
		2	12.57	10.61	7.96	6.15	4.85	12.68	12.21	10.92	9.90	9.28

Table 3.9: COVQ Training SDR results (in dB) for the memoryless NBNDQ-QB and the highly correlated NBNDQ-QB with parameters $\alpha = 1.0$, $M = 1$; memoryless Gaussian source- Decrease increase method (5 times).

q	R	k	Memoryless (Cor=0)					Cor=0.9				
			SNR (dB)					SNR (dB)				
			15	10	5	2	0	15	10	5	2	0
1	1	1	4.18	3.77	2.88	2.16	1.67	4.18	3.77	2.88	2.16	1.67
		2	4.17	3.75	2.87	2.15	1.66	4.17	3.75	2.87	2.15	1.66
		3	4.23	3.78	2.88	2.15	1.67	4.37	4.16	3.68	3.27	2.98
	2	1	8.16	6.58	4.23	2.85	2.06	8.84	8.08	6.69	5.70	5.09
		2	8.42	6.98	4.95	3.67	2.88	9.20	8.54	7.29	6.40	5.84
		3	8.68	7.21	5.12	3.79	2.96	9.51	8.89	7.76	6.97	6.47
	3	1	11.12	8.48	5.80	4.45	3.57	13.41	11.80	9.45	8.04	7.23
		2	11.64	9.30	6.60	4.94	3.88	13.92	12.36	10.27	9.05	8.34
		3	12.16	9.78	6.92	5.11	3.99	14.40	12.94	10.95	9.76	9.06
2	1	1	4.23	3.88	3.10	2.43	1.95	4.23	3.88	3.10	2.43	1.95
		2	4.21	3.87	3.09	2.42	1.94	4.21	3.87	3.09	2.42	1.94
		3	4.28	3.91	3.10	2.43	1.95	4.39	4.22	3.81	3.44	3.17
	2	1	8.37	7.00	4.75	3.35	2.51	8.94	8.32	7.06	6.13	5.52
		2	8.55	7.43	5.65	4.32	3.43	9.34	8.77	7.60	6.75	6.20
		3	8.86	7.64	5.79	4.44	3.54	9.60	9.07	8.00	7.28	6.80
	3	1	11.67	9.43	7.21	5.35	4.13	13.69	12.34	10.10	8.72	7.91
		2	12.42	10.38	7.78	5.88	4.61	14.02	12.87	11.04	9.94	9.22
		3	12.80	10.84	8.07	6.12	4.86	14.51	13.35	11.65	10.60	9.94
3	1	1	4.24	3.91	3.16	2.49	2.01	4.24	3.91	3.16	2.49	2.01
		2	4.22	3.89	3.15	2.48	2.00	4.22	3.89	3.15	2.48	2.00
		3	4.29	3.93	3.15	2.49	2.01	4.40	4.23	3.85	3.49	3.23
	2	1	8.42	7.08	4.89	3.47	2.61	8.96	8.37	7.17	6.25	5.64
		2	8.60	7.52	5.75	4.46	3.56	9.32	8.80	7.75	6.95	6.45
		3	8.87	7.74	5.94	4.58	3.68	9.57	9.06	8.05	7.33	7.07
	3	1	11.79	9.65	7.47	5.60	4.35	13.74	12.44	10.32	8.95	8.16
		2	12.57	10.61	8.01	6.15	4.86	13.86	12.87	11.24	10.24	9.58

Table 3.10: NBNDC parameters for fitting the Rayleigh DFC with $q = 2$.

$\gamma(dB)$	$f_D T = 0.005$	$f_D T = 0.01$
2 ($\delta = 0.5$)	$M = 11$ $\varepsilon = 0.7537$ $\alpha = 0.6362$ Cor = 0.22	$M = 8$ $\varepsilon = 0.6846$ $\alpha = 0.5313$ Cor = 0.22
5 ($\delta = 0.4$)	$M = 10$ $\varepsilon = 0.7967$ $\alpha = 0.6318$ Cor = 0.29	$M = 7$ $\varepsilon = 0.7260$ $\alpha = 0.5286$ Cor = 0.29
10 ($\delta = 0.2$)	$M = 7$ $\varepsilon = 0.7563$ $\alpha = 0.5932$ Cor = 0.32	$M = 5$ $\varepsilon = 0.6765$ $\alpha = 0.4818$ Cor = 0.32
15 ($\delta = 0.12$)	$M = 5$ $\varepsilon = 0.7076$ $\alpha = 0.5511$ Cor = 0.35	$M = 4$ $\varepsilon = 0.6371$ $\alpha = 0.399$ Cor = 0.34

Table 3.11: COVQ training results (for DFC-fitted NBNDC-QB) and simulation results (for the Rayleigh DFC) in terms of SDR (dB); memoryless Laplacian source. $f_D T = 0.01$.

q	$R =$ n/k	k	Training				Simulation			
			SNR (dB)				SNR (dB)			
			15	10	5	2	15	10	5	2
1	1	1	2.87	2.63	2.07	1.59	2.87	2.63	2.07	1.59
		2	3.47	3.12	2.42	1.84	3.46	3.12	2.41	1.84
		3	4.15	3.63	2.64	1.88	4.06	3.45	2.39	1.70
	2	1	6.67	5.46	3.62	2.46	6.71	5.45	3.58	2.42
		2	7.69	6.22	4.14	2.90	7.71	6.20	4.10	2.89
		3	8.21	6.62	4.56	3.21	8.17	6.53	4.50	3.17
	3	1	9.65	7.14	4.40	2.91	9.73	7.15	4.30	2.83
		2	10.78	8.20	5.31	3.93	10.79	8.19	5.26	3.94
			3	11.16	8.69	5.92	4.24	10.95	8.60	5.89
2	1	1	2.90	2.70	2.21	1.77	2.90	2.70	2.21	1.77
		2	3.51	3.23	2.60	2.06	3.50	3.22	2.59	2.06
		3	4.21	3.78	2.86	2.13	4.15	3.62	2.62	1.94
	2	1	6.89	5.88	4.09	2.89	6.96	5.98	4.21	2.97
		2	7.97	6.69	4.75	3.57	8.01	6.76	4.76	3.53
		3	8.50	7.08	5.15	3.89	8.46	7.04	4.95	3.66
	3	1	10.45	8.00	5.01	4.19	10.64	8.28	5.18	4.03
		2	11.50	9.16	6.60	4.97	11.48	8.95	6.08	4.55
			3	11.95	9.87	7.25	5.41	11.49	8.87	6.18

Table 3.12: COVQ training results (for DFC-fitted NBNDQ-QB) and simulation results (for the Rayleigh DFC) in terms of SDR (dB); memoryless Gaussian source. $f_D T = 0.005$.

q	$R =$ n/k	k	Training				Simulation			
			SNR (dB)				SNR (dB)			
			15	10	5	2	15	10	5	2
1	1	1	4.17	3.76	2.87	2.14	4.17	3.76	2.86	2.13
		2	7.14	5.96	4.05	2.78	7.08	5.86	3.84	2.60
		3	8.11	6.75	4.52	3.10	8.07	6.54	4.23	3.03
	2	1	8.23	6.71	4.43	3.01	8.26	6.71	4.38	2.95
		2	10.87	8.27	5.46	4.21	10.84	8.16	5.45	4.31
		3	11.79	9.25	6.43	4.87	11.79	9.37	6.60	5.08
	3	1	11.29	8.36	5.16	3.41	11.29	8.32	5.02	3.31
		2	13.21	10.02	6.85	5.14	13.25	10.18	7.14	5.41
		3	14.25	10.93	7.54	5.78	14.25	11.11	7.93	6.19
2	1	1	4.22	3.87	3.09	2.42	4.21	3.87	3.09	2.42
		2	7.29	6.29	4.48	3.22	7.26	6.19	4.30	3.05
		3	8.37	7.02	5.53	4.29	8.30	6.91	5.89	4.49
	2	1	8.45	7.13	4.92	3.49	8.50	7.19	4.94	3.49
		2	11.30	9.26	6.79	5.28	11.38	9.02	6.36	4.90
		3	12.40	10.45	7.94	6.27	12.32	9.85	7.05	5.59
	3	1	11.89	9.07	6.19	5.03	12.01	9.23	6.03	4.83
		2	14.34	11.72	8.65	6.78	14.03	10.86	7.54	6.00
		3	15.62	13.05	9.94	7.77	14.27	11.29	8.15	6.54

Table 3.13: COVQ training results (for DFC-fitted NBNDQ-QB) and simulation results (for the Rayleigh DFC) in terms of SDR (dB); Gauss-Markov source with correlation factor $\phi = 0.9$. $f_D T = 0.005$.

q	$R =$ n/k	k	Training				Simulation			
			SNR (dB)				SNR (dB)			
			15	10	5	2	15	10	5	2
1	1	1	4.18	3.77	2.88	2.16	4.18	3.76	2.88	2.16
		2	4.16	3.75	2.87	2.15	4.16	3.75	2.87	2.14
		3	4.24	3.81	2.89	2.15	4.23	3.78	2.85	2.14
	2	1	8.23	6.71	4.44	3.02	8.23	6.71	4.36	2.95
		2	8.43	6.88	4.72	3.40	8.39	6.83	4.72	3.42
		3	8.67	7.08	4.94	3.59	8.65	7.05	4.96	3.61
	3	1	11.26	8.36	5.17	3.42	11.25	8.29	5.00	3.32
		2	11.66	8.85	5.98	4.42	11.62	8.88	6.04	4.53
		3	11.71	9.13	6.27	4.61	11.46	9.06	6.34	4.72
2	1	1	4.23	3.88	3.10	2.43	4.23	3.88	3.10	2.43
		2	4.21	3.87	3.09	2.42	4.21	3.87	3.08	2.42
		3	4.29	3.93	3.11	2.43	4.28	3.90	3.08	2.41
	2	1	8.44	7.12	4.93	3.50	8.47	7.18	4.93	3.50
		2	8.65	7.29	5.34	4.00	8.63	7.29	5.19	3.89
		3	8.90	7.49	5.54	4.27	8.87	7.48	5.34	4.07
	3	1	11.86	9.06	6.20	5.03	11.97	9.21	6.05	4.82
		2	12.25	9.94	7.20	5.54	12.25	9.65	6.58	5.06
		3	12.43	10.30	7.68	5.82	11.96	9.23	6.57	5.11

Table 3.14: The ρ and δ values for the NBNDQ-QB for soft-decision resolutions $q = 1, 2, 3$.

SNR (dB)	q	δ	ρ_0	ρ_1	ρ_2	ρ_3	ρ_4	ρ_5	ρ_6	ρ_7
15	1	-	0.992	0.008	-	-	-	-	-	-
	2	0.12	0.972	0.020	0.006	0.001	-	-	-	-
	3	0.60	0.955	0.018	0.012	0.008	0.004	0.002	0.001	0.000
10	1	-	0.977	0.023	-	-	-	-	-	-
	2	0.20	0.924	0.053	0.019	0.005	-	-	-	-
	3	0.11	0.865	0.051	0.037	0.024	0.013	0.006	0.003	0.001
5	1	-	0.936	0.064	-	-	-	-	-	-
	2	0.40	0.782	0.154	0.054	0.010	-	-	-	-
	3	0.18	0.703	0.100	0.078	0.055	0.034	0.018	0.008	0.005
2	1	-	0.892	0.108	-	-	-	-	-	-
	2	0.50	0.695	0.196	0.085	0.024	-	-	-	-
	3	0.25	0.563	0.132	0.112	0.084	0.054	0.031	0.015	0.009
0	1	-	0.854	0.146	-	-	-	-	-	-
	2	0.60	0.627	0.227	0.110	0.036	-	-	-	-
	3	0.31	0.472	0.145	0.132	0.104	0.071	0.041	0.021	0.013

Chapter 4

Joint Source-Channel MAP

Decoding of the NBNDC-QB

The COVQ system explained in Chapter 3 is only able to take advantage of the intra-block memory (the dependency within each codeword block) of the data sent over the channel. As a result, the inter-block memory of codewords (the dependency on the previous received blocks of the current block) are discarded during the system design. In this chapter, we describe a source-channel decoding approach which utilizes inter-block memory. This also extends the work in [26], where only binary output channels with Markovian noise are examined. In particular, we study the sequence maximum-a-posteriori (MAP) decoding problem of quantized sources over the NBNDC-QB and the correlated Rayleigh DFC described in Chapter 2. We consider a scalar quantizer (SQ) designed for a noiseless channel. The SQ output is passed through an index assignment mapping and then sent over the channel. The output is soft-demodulated with resolution q and delivered to a sequence MAP detector to combat channel errors. As in [26], we refer to such a coding scheme as SQ-MAP. We use scalar-quantization,

rather than vector-quantization (VQ), since although VQ achieves better signal-to-distortion (SDR) performance due to the space-filling gain, it retains less redundancy in the index codewords at the quantizer output; but using SQ, such redundancy is kept to be exploited by the MAP decoder together with the channel's characteristics, for error correction. As a result, it is likely for the system to perform better using an SQ.

It is important to mention that the SQ-MAP scheme is designed to minimize the sequence error probability, while we evaluate the performance of the system via the signal-to-distortion ratio (SDR) with the mean square error (MSE) distortion measure. Hence, the SQ-MAP is not necessarily optimal in terms of achieving minimum mean square error (MMSE). However, this system has tractably low complexity as well as good performance according to simulations results, which makes it an efficient joint source-channel coding scheme. MMSE optimal and suboptimal MAP decoding metrics are studied in [23, 34]. Furthermore, in this work, we study the use of residual source redundancy as well as noise correlation and soft-decision information on the NBNDC model, in order to achieve better SDR performance. Minimizing distortion via improved MAP metrics is not studied here.

Similar to Chapter 3, we show that the channel's memory and soft-decision information can be exploited to improve the SQ-MAP system performance. We also numerically show that the NBNDC-QB effectively models the Rayleigh DFC when measured in terms of SDR performance. Furthermore, we prove a theorem for a specific case of our system setup (SQ-MAP for the NBNDC-QB), in which we provide necessary and sufficient condition for a sequence MAP decoder to be replaceable with a simple instantaneous (symbol-by-symbol) decoding rule.

4.1 Sequence MAP decoding

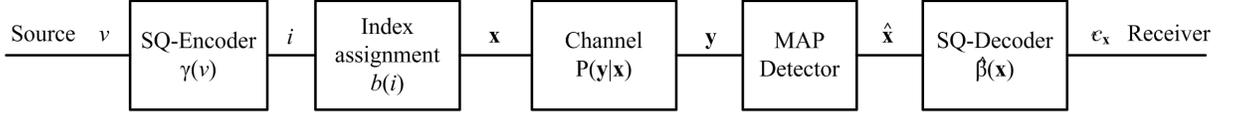


Figure 4.1: Block diagram of an SQ-MAP system.

4.1.1 System setup

Consider the communication system depicted in Figure 4.1. The analog source $\mathcal{V} = \{V_i\}_{i=1}^{\infty}$ is assumed to be a real-valued stationary ergodic process. The scalar quantizer (SQ) encoder is a mapping γ from the real domain of source symbols to the index set $\{0, 1, \dots, 2^n - 1\}$, such that

$$\gamma(v) = i \quad \text{if } v \in \mathbf{S}_i,$$

where $\{\mathbf{S}_i : i \in \{0, 1, \dots, 2^n - 1\}\}$ is a partition of \mathbb{R} . Hence the SQ rate is $R = n$. The partitions are chosen according to Lloyd-Max formulation given in Chapter 2, with the initial codebook selection obtained via the splitting algorithm, described in Section 3.1.2.

As in the COVQ system, the index assignment module is a one-to-one mapping, which maps each index i to a binary vector $\mathbf{x} \in \{0, 1\}^n$

$$b : \{0, 1, \dots, 2^n - 1\} \rightarrow \{0, 1\}^n, \quad b(i) = \mathbf{x}$$

where \mathbf{x} is represented in binary form. Since the mapping is one-to-one, for a given index mapping b , we can present the quantization regions by $\mathbf{S}_{\mathbf{x}}$ instead of \mathbf{S}_i , where

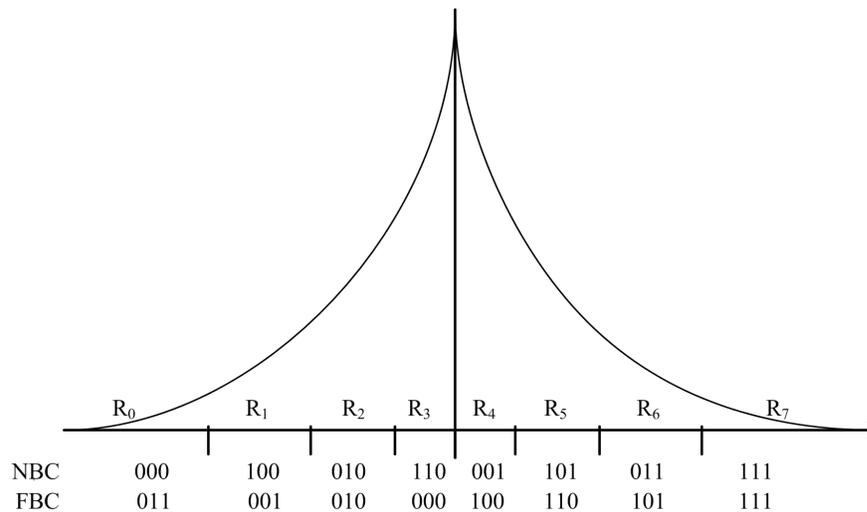


Figure 4.2: FBC and NBC index assignments for an hypothetical source distribution and its quantization regions; MSB is the rightmost bit.

$$b(i) = \mathbf{x}.$$

To assign a binary n -tuple codeword to each index, different index assignment methods such as the natural binary code (NBC), the folded binary code (FBC), simulated annealing, and some heuristic assignment methods were tested. The FBC was selected because of its simplicity and good performance. An example of the FBC and NBC index assignment (with the right-most bit as the most significant bit) is depicted in Figure 4.2, for a hypothetical source with the probability density shown in the figure. The horizontal line represents the real axis and the quantization regions are separated by the short vertical lines. Note that the FBC index assignment tends to set more zeros to the rightmost bits of the codewords corresponding to code-levels with moderate values. Since the source distributions used here are symmetric and denser around the origin, the rightmost bits are more likely to be zero. This non-uniformity of the distribution of zeros and ones in each codeword leads to more

redundancy in the codewords which helps to improve the performance of the sequence MAP decoder. The n -tuple codeword \mathbf{x} is then sent bit-by-bit over the NBNDQ-QB channel where it is affected by the error n -tuple z^n .

The channel output $\mathbf{y} \in \{0, 1, \dots, 2^q - 1\}^n$, corresponding to each binary n -tuple \mathbf{x} sent over the channel, is fed to a MAP decoder where the data redundancy is used for error correction. Finally, the SQ decoder β maps the decoder output $\hat{\mathbf{x}}$ into output levels of the quantizer codebook

$$\beta(\hat{\mathbf{x}}) = c_{\hat{\mathbf{x}}}, c_{\hat{\mathbf{x}}} \in \mathbb{R}, \hat{\mathbf{x}} \in \{0, 1\}^n.$$

4.1.2 MAP decoder design

The MAP decoder is designed to minimize the sequence error probability by exploiting the residual redundancy of the source and channel model statistics to combat channel errors. Since we are considering a general source, the redundancy ρ_T , in general, is due to a combination of non-uniformity of the distribution (ρ_D) and memory (ρ_M), such that $\rho_T = \rho_D + \rho_M$. Similar to [26], we first assume to have an i.i.d. source and then we modify the metric for sources with memory.

If \mathcal{V} is i.i.d., then the SQ encoder output process, $\mathcal{X} = \{X_i\}$, is also i.i.d. Hence $\rho_M = 0$ and the only remaining redundancy is due to non-uniformity of the source. Consequently, the MAP detector could be considered as a system observing a sequence of n -tuples

$$\mathbf{y}^N = (\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N) \in \{0, 1, \dots, 2^q - 1\}^{nN},$$

where q is the soft-decision quantization resolution, N denotes the number of source symbols to be transmitted over the channel, and n is the codeword length. Note that since the transmission over the channel is done bit-by-bit (and not n -tuple

by n -tuple), the observation of the MAP detector can also be considered as sequence of 2^q -ary symbols $y^{nN} = (y_1, y_2, \dots, y_{nN}) \in \{0, 1, \dots, 2^q - 1\}^{nN}$, where $\mathbf{y}_k = (y_{(k-1)n+1}, y_{(k-1)n+2}, \dots, y_{kn})$, $k = 1, 2, \dots, N$. \mathbf{y}^N is a noisy observation of the source sequence

$$\mathbf{x}^N = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N) \in \{0, 1\}^{nN},$$

which can be similarly represented bit-by-bit via $x^{nN} = (x_1, x_2, \dots, x_{nN}) \in \{0, 1\}^{nN}$, where $\mathbf{x}_k = (x_{(k-1)n+1}, x_{(k-1)n+2}, \dots, x_{kn})$, $k = 1, 2, \dots, N$. Therefore, the n -tuple by n -tuple and bit-by-bit notations can be interchangeably used for ease of explanation. The channel contaminates the source bits via 2^q -ary error symbols

$$z^{nN} = (z_1, z_2, \dots, z_{nN}) \in \{0, 1, \dots, 2^q - 1\}^{nN}.$$

The MAP decoder estimates \mathbf{x}^N by $\hat{\mathbf{x}}^N$ according to

$$\hat{\mathbf{x}}^N = \arg \max_{\mathbf{x}^N} Pr\{\mathbf{X}^N = \mathbf{x}^N \mid \mathbf{Y}^N = \mathbf{y}^N\}.$$

Similar to the reasoning in [2, 27], the above equation is equivalent to

$$\begin{aligned} \hat{\mathbf{x}}^N &= \arg \max_{x^{nN} \in \{0,1\}^{nN}} Pr\{Y^{nN} = y^{nN} \mid X^{nN} = x^{nN}\} Pr\{X^{nN} = x^{nN}\} \\ &= \arg \max_{x^{nN} \in \{0,1\}^{nN}} Pr\{Z^{nN} = z^{nN}\} Pr\{X^{nN} = x^{nN}\} \\ &= \arg \max_{x^{nN} \in \{0,1\}^{nN}} \left[Q(z_1^n) P(\mathbf{x}_1) \prod_{i=1}^{N-1} \left(Q(z_{ni+1}^{ni+n} | z_1^{ni}) P(\mathbf{x}_{i+1}) \right) \right] \end{aligned} \quad (4.1)$$

where

$$Q(z_{i+1}^{i+j} | z_{i-k}^i) \triangleq Pr\{Z_{i+1} = z_{i+1}, Z_{i+2} = z_{i+2}, \dots, Z_{i+j} = z_{i+j} \mid Z_i = z_i, \dots, Z_{i-k} = z_{i-k}\},$$

$$i, j, k \in \{1, 2, \dots, nN - 1\}, i + j \leq nN, i - k \geq 1,$$

and $P(\mathbf{x}_i) \triangleq Pr\{\mathbf{X}_i = \mathbf{x}_i\}$ is the probability mass function of the n -tuple codewords.

Note that since the channel model we are considering is the NBNDQ-QB with Markovian memory order M , for $nN \geq M$ (which is always the considered case since N is

assumed to be large,) it can be shown that (4.1) is equivalent to

$$\hat{\mathbf{x}}^N = \arg \max_{\mathbf{x}^N} \{ \log[P_{\text{NBNDQ-QB}}^{(n)}(z_1^n) p(\mathbf{x}_1)] + \sum_{i=1}^{N-1} \log[Q(z_{in+1}^{(i+1)n} | z_{in-(M-1)}^{in}) p(\mathbf{x}_{i+1})] \}, \quad (4.2)$$

where

$$Q(z_{j+1}^{j+n} | z_{j-(M-1)}^j) = \prod_{i=j+1}^{j+n} \left[\left(\sum_{\ell=i-(M-1)}^{i-1} \delta_{z_i, z_\ell} + \alpha \delta_{z_i, z_{i-M}} \right) \times \frac{\varepsilon}{M-1+\alpha} + (1-\varepsilon) \rho_{z_i} \right], \quad (4.3)$$

which is obtained from (2.4), $z_i \triangleq 0$ if $i < 1$, $z_i^j = (z_i, z_{i+1}, \dots, z_j)$, $j \geq i$, $P_{\text{NBNDQ-QB}}^{(n)}(z_1^n) = Pr\{Z_1^n = z_1^n\}$ is given via (2.6)-(2.8), and z is related to its corresponding symbols x, y via (2.3).

According to (4.2) and (4.3), the MAP detection can be implemented using a modified version of the Viterbi algorithm [15]. We consider the state space to be the set of all possible n -tuple codewords. Therefore, the trellis has 2^n states, each having 2^n outgoing and entering states and the path metric at step i is

$$\log[Q(z_{in+1}^{(i+1)n} | z_{in-(M-1)}^{in}) P(\mathbf{x}_i)].$$

When the source has memory, we assume that it forms a discrete Markov chain of order 1 with state transition probability matrix $P(\mathbf{x}_i | \mathbf{x}_{i-1})$, and the path metric will be updated to

$$\log[Q(z_{in+1}^{(i+1)n} | z_{in-(M-1)}^{in}) P(\mathbf{x}_i | \mathbf{x}_{i-1})].$$

The pmf $P(\mathbf{x}_i)$ and state transition matrix $[P(\mathbf{x}_i | \mathbf{x}_{i-1})]$ of the source codewords are calculated from a training set of symbols (the same training set used for designing the SQ).

The SQ-MAP system has low complexity due to the Viterbi algorithm and the simplicity of SQ encoding/decoding. On the other hand, the sequence MAP detector imposes a significant amount of delay in the receiver since the decoder needs to receive all of the nN symbols to be able to minimize the sequence probability of error. Furthermore, the storage complexity of the system grows with asymptotical growth order $\mathcal{O}(nN2^n)$. Hence, the storage requirement becomes unmanageable for very large values of nN . Therefore, it is useful to know when it is possible to replace the MAP detector with an instantaneous (symbol-by-symbol) decoding rule, without sacrificing the system's optimality in terms of sequence probability of error.

4.1.3 A specific case of the SQ-MAP system

In this section, we consider the MAP decoder system with the SQ coding rate $R = n = 1$ for the NBNDC-QB with memory order $M = 1$. We investigate necessary and sufficient conditions for the MAP detector to be replaceable with a symbol-by-symbol instantaneous decoder (mapping) without losing system optimality. We call the MAP detector *useless*, when it can be replaced with the instantaneous mapping.

According to Figure 4.1, the output of the SQ is a binary n -tuple. Letting $R = n = 1$, the SQ output is binary. The binary sequence x_i , $i = 1, 2, \dots, N$ is transmitted across the NBNDC-QB. According to the queue noise model, as stated in Chapter 2, for $M = 1$, the noise process is a homogeneous first-order Markov process. Recall that

$$Q(z_n) \triangleq Pr\{Z_n = z_n\}$$

and

$$Q(z_n|z_{n-1}) \triangleq Pr\{Z_n = z_n|z_{n-1} = z_{n-1}\}, z_n \in \{0, 1, 2, \dots, 2^q - 1\}.$$

In this section, we always consider the special case $M = 1$. As a result, according to the channel noise model, $\alpha = 1$. Furthermore, it can be seen that the probability distribution of the noise is

$$Q(z_n) = \rho_{z_n}, \quad z_n \in \mathcal{Y}, \quad (4.4)$$

where ρ_{z_i} is known from the noise model.

The matrix $\mathbf{Q} = [Q(i|j)]$, $i, j \in \{0, 1, \dots, 2^q - 1\}$ can be calculated in closed form, for arbitrary q , via

$$Q(i|j) = [\varepsilon\delta_{i,j} + (1 - \varepsilon)\rho_{z_i}], \quad (4.5)$$

where

$$\delta_{i,j} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j. \end{cases}$$

For instance, letting $q = 2$ we have:

$$\begin{aligned} \mathbf{Q} &= \begin{bmatrix} Q(0|0) & Q(1|0) & Q(2|0) & Q(3|0) \\ Q(0|1) & Q(1|1) & Q(2|1) & Q(3|1) \\ Q(0|2) & Q(1|2) & Q(2|2) & Q(3|2) \\ Q(0|3) & Q(1|3) & Q(2|3) & Q(3|3) \end{bmatrix} \\ &= \begin{bmatrix} \varepsilon + (1 - \varepsilon)\rho_0 & (1 - \varepsilon)\rho_1 & (1 - \varepsilon)\rho_2 & (1 - \varepsilon)\rho_3 \\ (1 - \varepsilon)\rho_0 & \varepsilon + (1 - \varepsilon)\rho_1 & (1 - \varepsilon)\rho_2 & (1 - \varepsilon)\rho_3 \\ (1 - \varepsilon)\rho_0 & (1 - \varepsilon)\rho_1 & \varepsilon + (1 - \varepsilon)\rho_2 & (1 - \varepsilon)\rho_3 \\ (1 - \varepsilon)\rho_0 & (1 - \varepsilon)\rho_1 & (1 - \varepsilon)\rho_2 & \varepsilon + (1 - \varepsilon)\rho_3 \end{bmatrix}. \end{aligned}$$

The channel output is fed to the MAP decoder described in the previous section.

To figure out when the MAP detector is *useless*, we find necessary and sufficient conditions such that applying MAP detector does not yield a lower sequence error probability than applying a mapping θ , where θ is a mapping from the set of channel output alphabet to the binary input alphabet, i.e.

$$\theta(y_n) = \tilde{y}_n, \quad y_n \in \mathcal{Y} = \{0, 1, \dots, 2^q - 1\}, \tilde{y}_n \in \{0, 1\}, \quad (4.6)$$

with

$$\tilde{y}_n = \begin{cases} 0, & \text{if } y_n \in \mathcal{Y}_0; \\ 1, & \text{if } y_n \in \mathcal{Y}_1, \end{cases}$$

where, $\mathcal{Y}_1 = \mathcal{Y}_0^c \subseteq \mathcal{Y}$. To replace the MAP detector with the mapping we simply skip the MAP detector and set $\hat{x}_i = \tilde{y}_i$, $i = 1, 2, \dots, N$.

Lemma 4.1. *For the NBNDQ-QB with parameters satisfying the condition*

$$\rho_0 \geq \rho_1 \geq \rho_2 \geq \dots \geq \rho_{2^q-1}, \quad (4.7)$$

among all mappings $\theta : \{0, 1, \dots, 2^q - 1\} \rightarrow \{0, 1\}$, the following mapping θ^* yields the lowest symbol probability of error:

$$\theta^*(y_n) \triangleq \tilde{y}_n = \begin{cases} 0, & \text{if } y_n < k^*; \\ 1, & \text{otherwise.} \end{cases} \quad (4.8)$$

where $k^* \in \{0, 1, \dots, 2^q\}$ is the smallest value satisfying

$$\frac{\rho_{k^*}}{\rho_{2^q-k^*-1}} \leq \frac{P(1)}{P(0)}, \quad (4.9)$$

where $\rho_{-1} \triangleq \infty$, $\rho_{2^q} \triangleq 0$, and $P(x) \triangleq \Pr\{X = x\}$, $x \in \{0, 1\}$. In other words, assuming (4.7) holds, setting $\mathcal{Y}_0 = \{0, 1, \dots, k^* - 1\}$, $\mathcal{Y}_1 = \mathcal{Y}_0^c = \{k^*, k^* + 1, \dots, 2^q - 1\}$, with $\mathcal{Y}_0 = \emptyset$ for $k^* = 0$ and $\mathcal{Y}_1 = \emptyset$ for $k^* = 2^q$, yields the best symbol probability of error, among all instantaneous mapping rules.

Proof. We first show that any general mapping other than θ^* defined via (4.8), can be modified to have a less than or equal symbol error probability. To this end, we consider the mapping as a classification rule that classifies 2^q different output symbols from $\{0, 1, \dots, 2^q - 1\}$ into two classes \mathcal{Y}_0 and \mathcal{Y}_1 . According to (4.8), for θ^* we have $\mathcal{Y}_0^* = \{0, 1, \dots, k^* - 1\}$ and $\mathcal{Y}_1^* = \{k^*, k^* + 1, \dots, 2^q - 1\}$. Let P_e denote the symbol error probability under mapping θ

$$P_e \triangleq Pr\{\theta(Y) \neq X\}.$$

If $\theta \neq \theta^*$, at least one of the two following cases happen:

- i) There exists an element $a \in \mathcal{Y}_1$, such that $a < k^*$. Removing a from \mathcal{Y}_1 and adding it to \mathcal{Y}_0 yields

$$\begin{aligned} P_{e2} - P_{e1} &= Pr\{Y = a|X = 1\}Pr\{X = 1\} - Pr\{Y = a|X = 0\}Pr\{X = 0\} \\ &= Q(2^q - 1 - a)P(1) - Q(a)P(0) \\ &= P(1)\rho_{2^q-1-a} - P(0)\rho_a, \end{aligned}$$

where P_{e2} represents the error probability after a is placed in \mathcal{Y}_0 and P_{e1} is the error probability before this modification. According to (4.7), $\rho_a \geq \rho_{k^*-1}$ and $\rho_{2^q-k^*} \geq \rho_{2^q-1-a}$. Hence by (4.9),

$$\frac{P(1)}{P(0)} \leq \frac{\rho_{k^*-1}}{\rho_{2^q-k^*}} \leq \frac{\rho_a}{\rho_{2^q-1-a}}$$

and therefore,

$$P_{e2} - P_{e1} \leq 0.$$

Thus, putting a into \mathcal{Y}_0 will not increase the error probability.

ii) There exists an element $b \in \mathcal{Y}_0$, such that $b \geq k^*$. Similarly, it can be shown that

$$P_{e2} - P_{e1} = -P(1)\rho_{2^q-1-b} + P(0)\rho_b.$$

According to (4.7), $\rho_b \leq \rho_{k^*}$ and $\rho_{2^q-k^*-1} \leq \rho_{2^q-1-b}$. Hence by (4.9),

$$\frac{P(1)}{P(0)} \geq \frac{\rho_{k^*}}{\rho_{2^q-k^*-1}} \geq \frac{\rho_b}{\rho_{2^q-1-b}}$$

and therefore,

$$P_{e2} - P_{e1} \leq 0.$$

Thus, removing b from \mathcal{Y}_0 and adding it to \mathcal{Y}_1 does not increase error probability.

The proof is complete by observing that by repeatedly applying the above replacements, the mapping obtained will be identical to θ^* .

□

As mentioned in the previous section, the sequence $\{\mathbf{X}_i\}$ (which is the same as $\{X_i\}$ here, since $R = 1$) is modeled via a first order stationary Markov chain. Define

$$p_{x_n, x_{n-1}} \triangleq P(x_n | x_{n-1}) \triangleq Pr\{X_n = x_n | X_{n-1} = x_{n-1}\} \quad (4.10)$$

and

$$p_{x_n} \triangleq P(x_n) \triangleq Pr\{X_n = x_n\},$$

where $x_n \in \{0, 1\}$. For the special case of a symmetric binary Markov source i.e., $P(0)/P(1) = 1$, it can be seen from (4.9) that for $k^* = 2^{q-1}$

$$\begin{aligned} i) \quad & \frac{\rho_{k^*}}{\rho_{2^q-k^*-1}} = \frac{\rho_{2^{q-1}}}{\rho_{2^{q-1}-1}} \leq 1, \\ ii) \quad & \frac{\rho_{k^*-1}}{\rho_{2^q-k^*}} = \frac{\rho_{2^{q-1}-1}}{\rho_{2^{q-1}}} \geq 1. \end{aligned}$$

From now on in this subsection, we will only consider symmetric binary Markov sources (i.e., we assume that $p_{00} = p_{11}$).

Define the *auxiliary* binary noise symbol \tilde{z}_n which is related to its corresponding noise symbol z_n via

$$\tilde{z}_n = \begin{cases} 0, & \text{if } z_n < 2^{q-1}; \\ 1, & \text{otherwise.} \end{cases} \quad (4.11)$$

The sequence of auxiliary binary noise symbols represents the auxiliary noise process $\{\tilde{Z}_n\}_{n=1}^{\infty}$. Since the noise process $\{Z_n\}_{n=1}^{\infty}$ is independent of the input process $\{X_n\}_{n=1}^{\infty}$ and the auxiliary binary noise variable \tilde{Z}_n is only a function of Z_n , the auxiliary noise process $\{\tilde{Z}_n\}_{n=1}^{\infty}$ is also independent of the input process $\{X_n\}_{n=1}^{\infty}$. Correspondingly, we can define

$$\begin{aligned} \tilde{Q}(\tilde{z}_n) &\triangleq Pr\{\tilde{Z}_n = \tilde{z}_n\} \\ \tilde{Q}(\tilde{z}_n|\tilde{z}_{n-1}) &\triangleq Pr\{\tilde{Z}_n = \tilde{z}_n|\tilde{Z}_{n-1} = \tilde{z}_{n-1}\}. \end{aligned}$$

It can be seen that

$$\tilde{Q}(\tilde{z}_n) = \begin{cases} Pr\{Z_n \in \{0, 1, \dots, 2^{q-1} - 1\}\}, & \text{if } \tilde{z}_n = 0; \\ Pr\{Z_n \in \{2^{q-1}, 2^{q-1} + 1, \dots, 2^q - 1\}\}, & \text{if } \tilde{z}_n = 1. \end{cases} \quad (4.12)$$

Therefore, according to (4.4)

$$\tilde{Q}(0) = 1 - \tilde{Q}(1) = \rho_0 + \rho_1 + \rho_2 + \dots + \rho_{2^{q-1}-1}. \quad (4.13)$$

Lemma 4.2. *The auxiliary noise process $\{\tilde{Z}_n\}_{n=1}^{\infty}$ forms a first order Markov chain, if $\{Z_n\}_{n=1}^{\infty}$ is a first-order Markov chain.*

Proof. We use a theorem given in [11, p. 325] [18] which states that if $\{Z_n\}_{n=1}^\infty$ is a Markov process, then a sufficient condition for the process $\{\tilde{Z}_n\}_{n=1}^\infty$, where $\tilde{Z}_n = f(Z_n)$, to be a Markovian is that

$$Pr\{\tilde{Z}_{n+1} = \tilde{z}_{n+1} | Z_n = z_n\} = Pr\{\tilde{Z}_{n+1} = \tilde{z}_{n+1} | \tilde{Z}_n = f(z_n)\}, \quad (4.14)$$

for all \tilde{z}_{n+1} and z_n , where $f(\cdot)$ is a function mapping the state space $S_Z = \mathcal{Y}$ into the set $\{0, 1\}$. Let the function f be given by (4.11). Then using (4.5), we have

$$\begin{aligned} Pr\{\tilde{Z}_{n+1} = 0 | Z_n = z_n\} &= Pr\{Z_{n+1} \in \{0, 1, \dots, 2^{q-1} - 1\} | Z_n = z_n\} \\ &= Pr\{Z_{n+1} = 0 | Z_n = z_n\} + Pr\{Z_{n+1} = 1 | Z_n = z_n\} + \dots + \\ &\quad Pr\{Z_{n+1} = 2^{q-1} - 1 | Z_n = z_n\} \\ &= Q(0|z_n) + Q(1|z_n) + \dots + Q(2^{q-1} - 1|z_n) \\ &= \begin{cases} \varepsilon + \sum_{i=0}^{2^{q-1}-1} (1 - \varepsilon)\rho_i, & \text{if } z_n < 2^{q-1}; \\ \sum_{i=0}^{2^{q-1}-1} (1 - \varepsilon)\rho_i, & \text{if } z_n \geq 2^{q-1}. \end{cases} \end{aligned} \quad (4.15)$$

Also

$$\begin{aligned} Pr\{\tilde{Z}_{n+1} = 0 | \tilde{Z}_n = 0\} &= Pr\left\{\tilde{Z}_{n+1} = 0 | Z_n = 0\right\} \frac{Pr\{Z_n = 0\}}{Pr\{\tilde{Z}_n = 0\}} + \\ &\quad Pr\left\{\tilde{Z}_{n+1} = 0 | Z_n = 1\right\} \frac{Pr\{Z_n = 1\}}{Pr\{\tilde{Z}_n = 0\}} \\ &\quad + \dots + \\ &\quad Pr\left\{\tilde{Z}_{n+1} = 0 | Z_n = 2^{q-1} - 1\right\} \frac{Pr\{Z_n = 2^{q-1} - 1\}}{Pr\{\tilde{Z}_n = 0\}}. \end{aligned} \quad (4.16)$$

Note that according to (4.15),

$$\begin{aligned} Pr\{\tilde{Z}_{n+1} = 0 | Z_n = 0\} &= \{ \tilde{Z}_{n+1} = 0 | Z_n = 1\} = \dots \\ &= \{ \tilde{Z}_{n+1} = 0 | Z_n = 2^{q-1} - 1\} = \varepsilon + \sum_{i=0}^{2^{q-1}-1} (1 - \varepsilon)\rho_i. \end{aligned}$$

Hence, (4.16) is equal to

$$\begin{aligned}
& \left[\varepsilon + \sum_{i=0}^{2^{q-1}-1} (1-\varepsilon)\rho_i \right] \times \frac{Pr\{Z_n = 0\} + Pr\{Z_n = 1\} + \dots + Pr\{Z_n = 2^{q-1} - 1\}}{Pr\{\tilde{Z}_n = 0\}} \\
&= \left[\varepsilon + \sum_{i=0}^{2^{q-1}-1} (1-\varepsilon)\rho_i \right] \times \frac{Pr\{\tilde{Z}_n = 0\}}{Pr\{\tilde{Z}_n = 0\}} \\
&= \varepsilon + \sum_{i=0}^{2^{q-1}-1} (1-\varepsilon)\rho_i.
\end{aligned}$$

With the same reasoning, it can be shown that

$$Pr\{\tilde{Z}_{n+1} = 0 | \tilde{Z}_n = 1\} = 1 - Pr\{\tilde{Z}_{n+1} = 1 | \tilde{Z}_n = 1\} = \sum_{i=0}^{2^{q-1}-1} (1-\varepsilon)\rho_i. \quad (4.17)$$

Thus condition (4.14) is satisfied for the NBNDQ-B with memory order $M = 1$ and for the function $f(\cdot)$ given via (4.11). Consequently, the process $\{\tilde{Z}_n\}_{n=1}^{\infty}$ is a first order Markov chain. \square

Since $\{\tilde{Z}_n\}_{n=1}^{\infty}$ is Markov process, it can be shown that its state transition probability matrix \tilde{Q} is

$$\begin{aligned}
\tilde{Q} &= \begin{bmatrix} \tilde{Q}(0|0) & \tilde{Q}(1|0) \\ \tilde{Q}(0|1) & \tilde{Q}(1|1) \end{bmatrix} \\
&= \begin{bmatrix} \varepsilon + (1-\varepsilon)\tilde{Q}(0) & (1-\varepsilon)\tilde{Q}(1) \\ (1-\varepsilon)\tilde{Q}(0) & \varepsilon + (1-\varepsilon)\tilde{Q}(1) \end{bmatrix},
\end{aligned} \quad (4.18)$$

where $\tilde{Q}(\tilde{z}_n)$ is given in (4.12).

Note by the definition of the NBNDQ given via (2.2), the mapping θ^* described in (4.8), and the fact that for symmetric binary Markov sources $k^* = 2^{q-1}$, it can be seen that if $\tilde{y}_n = x_n$, then $z_n < 2^{q-1}$ (and hence $\tilde{z}_n = 0$) and vice versa. As a result,

the auxiliary binary noise symbol can also be defined in terms of the input x_n and \tilde{y}_n

$$\tilde{z}_n \triangleq \begin{cases} 0, & \text{if } \tilde{y}_n = x_n; \\ 1, & \text{if } \tilde{y}_n = x_n^c, \end{cases} \quad (4.19)$$

where x_n^c is the binary complement of x_n and $\tilde{y}_n = \theta^*(y_n)$.

We next seek the answer to question “for a given sequence MAP decoder system, when is it optimal to replace the N -sequence MAP detector with an application of an instantaneous mapping (such as θ^*) applied N times?”. In other words, “when is $\hat{X}^N = \tilde{Y}^N$ an optimal sequence (MAP) detection rule, in the sense of minimizing the sequence error probability?”.

The answer to this question is partly given in [3]. To be more specific, for $q = 1$ the NBNDQ model is identical to the queue based channel (QBC) model which is introduced in [37]. It is shown there that for $\alpha = 1$ (which is the case here since $M = 1$), the channel reduces to the binary Markov channel described in Chapter 2. Theorem 1 of [3] states necessary and sufficient conditions for the MAP decoder to be *useless* over a binary Markov channel and for binary Markov sources. In this case, a MAP decoder is defined to be useless when it decodes what it sees (i.e., $\hat{X}^N = Y^N$) and thus does not make any improvement in the channel bit error rate. As a result, they show that under certain conditions, it is optimal to skip the MAP decoder and believe in what is seen in the receiver. Note that skipping the decoder and believing in the output sequence can only be applied for $q = 1$ so that the output sequence is also binary. On the other hand, for $q \geq 2$, the received sequence is not binary; hence, we use the mapping θ^* in order to convert the 2^q -ary received sequence Y^N , into a binary sequence \tilde{Y}^N . Since θ^* is also (trivially) defined for $q = 1$ ($k^* = 1$ here and $\tilde{y} = y$), Theorem 1 of [3] yields necessary and sufficient conditions for the mapping

θ^* to be an optimal sequence detection rule for $q = 1$. The following theorem gives a necessary and sufficient condition for the mapping θ^* to be an optimal sequence detection rule for $q > 1$.

Theorem 4.1. *For a symmetric binary Markov source with $p_{00} = p_{11} \in [\frac{1}{2}, 1]$ and the NBND-CQB with correlation parameter $\varepsilon \geq 0$, memory order $M = 1$, $q > 1$, and satisfying (4.7), assume that sequence length $N \geq 3$, $X_1 = \tilde{Y}_1$, and $X_N = \tilde{Y}_N$. Then $\hat{X}^N = \tilde{Y}^N$ is an optimal sequence MAP detection rule if and only if*

$$\frac{\rho_{2^q-1-1}}{\rho_{2^q-1}} \times \left[\frac{1-p_{00}}{p_{00}} \right]^2 \geq 1, \quad (4.20)$$

where $\tilde{Y}^N = \theta^*(Y^N)$ is obtained via applying the mapping θ^* N -times component-wise to Y^N .

Proof. For θ^* to be the optimal detection rule, then $\forall x^N \in \{0, 1\}^N$ and $\forall y^N \in \{0, 1, \dots, 2^q - 1\}^N$, we should have

$$\gamma \triangleq \frac{Pr\{X^N = \tilde{y}^N | Y^N = y^N\}}{Pr\{X^N = x^N | Y^N = y^N\}} \geq 1.$$

γ can be written as

$$\gamma = \frac{Pr\{Y^N = y^N | X^N = \tilde{y}^N\} Pr\{X^N = \tilde{y}^N\}}{Pr\{Y^N = y^N | X^N = x^N\} Pr\{X^N = x^N\}}.$$

Note that by (2.3) and (2.5), we have $Pr\{Y^N = y^N | X^N = x^N\} = Pr\{Z^N = z^N\}$, where $z_i \triangleq \frac{y_i - (-1)^{x_i}}{(-1)^{x_i}}$, $i = 1, 2, \dots, N$. Also note that by definition of θ^* (with $k^* = 2^{q-1}$), we have that $a_i \triangleq \frac{y_i - (-1)^{\tilde{y}_i}}{(-1)^{\tilde{y}_i}} \in \{0, 1, \dots, 2^{q-1} - 1\}$, where $\tilde{y}_i = \theta^*(y_i)$, $i =$

1, 2, \dots, N. Define

$$\begin{aligned} A &\triangleq Pr\{Z^N = a^N\}Pr\{X^N = \tilde{y}^N\}, \quad a^N \in \{0, 1, \dots, 2^{q-1} - 1\}^N, \\ B &\triangleq Pr\{Z^N = z^N\}Pr\{X^N = x^N\}, \quad z^N \in \{0, 1, \dots, 2^q - 1\}^N, \end{aligned}$$

where $a^N = (a_1, a_2, \dots, a_N)$ and $z^N = (z_1, z_2, \dots, z_N)$ are defined above. Hence,

$$\begin{aligned} \gamma &= \frac{A}{B} = \frac{Pr\{Z^N = a^N\}Pr\{X^N = \tilde{y}^N\}}{Pr\{Z^N = z^N\}Pr\{X^N = x^N\}} \\ &= \left[\frac{Pr\{Z_1 = a_1\}P(\tilde{y}_1)}{Pr\{Z_1 = z_1\}P(x_1)} \right] \left[\prod_{k=2}^N \frac{Q(a_k|a_{k-1})P(\tilde{y}_k|\tilde{y}_{k-1})}{Q(z_k|z_{k-1})P(x_k|x_{k-1})} \right] \\ &= \prod_{k=2}^N \frac{Q(a_k|a_{k-1})P(\tilde{y}_k|\tilde{y}_{k-1})}{Q(z_k|z_{k-1})P(x_k|x_{k-1})}, \end{aligned}$$

where the last equality follows from $Pr\{Z_1 = a_1\}P(\tilde{y}_1) = Pr\{Z_1 = z_1\}P(x_1)$ since $X_1 = \tilde{Y}_1$ according to the hypothesis.

We partition the index set as follows: $\mathcal{K} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3 \cup \mathcal{A}_4$, where

$$\begin{aligned} \mathcal{K} &\triangleq \{2, 3, \dots, N\}, \\ \mathcal{A}_1 &\triangleq \{k \in \mathcal{K} : x_k = \tilde{y}_k, x_{k-1} = \tilde{y}_{k-1}\}, \\ \mathcal{A}_2 &\triangleq \{k \in \mathcal{K} : x_k \neq \tilde{y}_k, x_{k-1} \neq \tilde{y}_{k-1}\}, \\ \mathcal{A}_3 &\triangleq \{k \in \mathcal{K} : x_k \neq \tilde{y}_k, x_{k-1} = \tilde{y}_{k-1}\}, \\ \mathcal{A}_4 &\triangleq \{k \in \mathcal{K} : x_k = \tilde{y}_k, x_{k-1} \neq \tilde{y}_{k-1}\}. \end{aligned}$$

Hence,

$$\gamma = \prod_{i=1}^4 \prod_{k \in \mathcal{A}_i} \frac{Q(a_k|a_{k-1})P(\tilde{y}_k|\tilde{y}_{k-1})}{Q(z_k|z_{k-1})P(x_k|x_{k-1})}.$$

In set \mathcal{A}_1 , since $x_k = \tilde{y}_k$ and $x_{k-1} = \tilde{y}_{k-1}$ we see that $z_k = a_k$ and $z_{k-1} = a_{k-1}$. Thus,

$$\prod_{k \in \mathcal{A}_1} \frac{Q(a_k|a_{k-1})P(\tilde{y}_k|\tilde{y}_{k-1})}{Q(z_k|z_{k-1})P(x_k|x_{k-1})} = \prod_{k \in \mathcal{A}_1} \frac{Q(a_k|a_{k-1})P(x_k|x_{k-1})}{Q(a_k|a_{k-1})P(x_k|x_{k-1})} = 1,$$

In \mathcal{A}_2 , $x_k \neq \tilde{y}_k$ and $x_{k-1} \neq \tilde{y}_{k-1}$ imply that $x_k = 1 - \tilde{y}_k$ and $x_{k-1} = 1 - \tilde{y}_{k-1}$. Also, if $x_k \neq \tilde{y}_k$ and $x_{k-1} \neq \tilde{y}_{k-1}$, then by (4.11) and (4.19) $z_k, z_{k-1} \geq 2^{q-1}$. Now since the Markov source is symmetric ($p_{00} = p_{11}$), we obtain that $P(\tilde{y}_k|\tilde{y}_{k-1}) = P(x_k|x_{k-1})$. Noting $a_k, a_{k-1} < 2^{q-1}$ and according to (4.5) and (4.7)

$$\prod_{k \in \mathcal{A}_2} \frac{Q(a_k|a_{k-1})P(\tilde{y}_k|\tilde{y}_{k-1})}{Q(z_k|z_{k-1})P(x_k|x_{k-1})} = \prod_{k \in \mathcal{A}_2} \frac{Q(a_k|a_{k-1})}{Q(z_k|z_{k-1})} \geq 1.$$

We next note that since $X_1 = \tilde{Y}_1$ and $X_N = \tilde{Y}_N$, for each $k \in \mathcal{A}_3$ there should exist a corresponding index $\ell \in \mathcal{A}_4$ and vice versa. As a result, $|\mathcal{A}_3| = |\mathcal{A}_4|$ where $|\mathcal{A}_3|$ denotes the number of elements in set \mathcal{A}_3 . Furthermore, in \mathcal{A}_3 we have $x_k \neq \tilde{y}_k$ implying $x_k = 1 - \tilde{y}_k$. Therefore, according to the definitions of z_k and a_k and using (2.3), it can be seen that $z_k = (2^q - 1) - a_k$ and $z_{k-1} = a_{k-1}$. Similarly, in set \mathcal{A}_4 we have $z_k = a_k$ and $z_{k-1} = (2^q - 1) - a_{k-1}$. Besides, due to the source symmetry and noting that $p_{00} \in [\frac{1}{2}, 1]$, we have

$$\min_{k \in \mathcal{A}_3} \frac{P(\tilde{y}_k|\tilde{y}_{k-1})}{P(x_k|x_{k-1})} = \min_{j \in \mathcal{A}_4} \frac{P(\tilde{y}_j|\tilde{y}_{j-1})}{P(x_j|x_{j-1})} = \frac{p_{10}}{p_{00}} = \frac{1 - p_{00}}{p_{00}}.$$

Now according to (4.5) and (4.7), we have

$$\begin{aligned} & \prod_{k \in \mathcal{A}_3} \frac{Q(a_k|a_{k-1})P(\tilde{y}_k|\tilde{y}_{k-1})}{Q(z_k|z_{k-1})P(x_k|x_{k-1})} \times \prod_{k \in \mathcal{A}_4} \frac{Q(a_k|a_{k-1})P(\tilde{y}_k|\tilde{y}_{k-1})}{Q(z_k|z_{k-1})P(x_k|x_{k-1})} \\ & \geq \prod_{k \in \mathcal{A}_3} \frac{Q(a_k|a_{k-1})(1 - p_{00})}{Q(2^q - 1 - a_k|a_{k-1})p_{00}} \times \prod_{k \in \mathcal{A}_4} \underbrace{\frac{Q(a_k|a_{k-1})(1 - p_{00})}{Q(a_k|2^q - 1 - a_{k-1})p_{00}}}_{(\alpha)} \\ & \geq \prod_{k \in \mathcal{A}_3} \frac{(1 - \varepsilon)\rho_{2^{q-1}-1}(1 - p_{00})}{(1 - \varepsilon)\rho_{2^q-1}p_{00}} \times \prod_{k \in \mathcal{A}_4} \frac{(1 - p_{00})}{p_{00}} \\ & = \prod_{i=1}^{|\mathcal{A}_3|} \frac{\rho_{2^{q-1}-1}}{\rho_{2^q-1}} \times \left[\frac{1 - p_{00}}{p_{00}} \right]^2, \end{aligned}$$

where (α) follows by noting that $\frac{Q(a_k|a_{k-1})}{Q(a_k|2^q-1-a_{k-1})} \geq 1, \forall k \in \mathcal{A}_4$. Evidently, if (4.20)

holds, then

$$\prod_{i=1}^{|\mathcal{A}_3|} \frac{\rho_{2^{q-1}-1}}{\rho_{2^q-1}} \times \left[\frac{1-p_{00}}{p_{00}} \right]^2 \geq 1.$$

Thus $\gamma \geq 1$ and the mapping θ^* is an optimal MAP decoding rule.

To prove the converse, assume that (4.20) does not hold; i.e.,

$$\frac{\rho_{2^{q-1}-1}}{\rho_{2^q-1}} \times \left[\frac{1-p_{00}}{p_{00}} \right]^2 < 1.$$

Now for $X^N = (0, 0, \dots, 0)$ and $Y^N = (0, 0, \dots, 0, 2^{q-1}, 0, \dots, 0)$ where the only non-zero component of Y^N is in any arbitrary position $i \in \{2, \dots, N-1\}$, we have

$$\gamma = 1 \times 1 \times \dots \times \left(\frac{\rho_{2^{q-1}-1}}{\rho_{2^q-1}} \times \left[\frac{1-p_{00}}{p_{00}} \right]^2 \right) \times 1 \times \dots \times 1 < 1.$$

Hence, if (4.20) does not hold, there exists some X^N and Y^N such that the mapping θ^* does not decode optimally.

□

For the binary symmetric first-order Markov sources with $p_{00} = p_{11} \in [0, \frac{1}{2})$, with the same approach, a similar theorem can be proved with the following condition

$$\frac{\rho_{2^{q-1}-1}}{\rho_{2^q-1}} \times \left[\frac{p_{00}}{1-p_{00}} \right]^2 \geq 1.$$

Tables 4.1-4.4 depict the simulation results for three binary symmetric sources, in terms of symbol error rate over the NBNDQ-QB with noise correlations $Cor = 0.0, 0.9$, soft-decision resolutions $q = 1, 2, 3$, and different SNRs. The calculated values for condition (4.20), denoted by C , are also given in the tables. For $q = 1$, the values C are calculated via the condition in Theorem 1 of [3]. The ρ values for the simulations can be seen in Table 3.14. It is observed from Tables 4.1-4.4 that the results verify Theorem 4.1, since for all the cases where (4.20) holds the MAP decoder and the

instantaneous mapping perform equally (they both actually output the same decoded sequence). It is also observed that for the cases where (4.20) does not hold, the mapping θ^* does not perform as well as the MAP decoder, so that it is not optimal in these cases. An other interesting observation is that for $q = 1$, the optimality of the mapping θ^* is dependent on the channel noise correlation (as it is observed numerically and analytically in [3]), while for $q > 1$, this is not the case.

Table 4.1: Symbol error rate (in%) for MAP decoding and instantaneous mapping θ^* for symmetric binary Markov sources with $p_{00} = 0.5$ (i.i.d. source). The channel model is the NBND-C-QB, with $M = 1$, $Cor = 0, 0.9$, and $q = 1, 2, 3$. The values C are calculated from (4.20).

p_{00}	Cor	q	SNR											
			15		10		5		2		0			
			MAP	θ^*	MAP	θ^*	MAP	θ^*	MAP	θ^*	MAP	θ^*		
0.5	0	1	$C = 128.48 > 1$	$C = 41.98 > 1$	$C = 14.58 > 1$	$C = 8.22 > 1$	$C = 5.83 > 1$							
			0.76	0.76	2.30	2.30	6.43	6.43	10.85	10.85	14.65	14.65		
		2	$C = 3.12 > 1$	$C = 2.82 > 1$	$C = 2.85 > 1$	$C = 2.31 > 1$	$C = 2.06 > 1$							
	0.9	3	0.76	0.76	2.30	2.30	6.43	6.43	10.85	10.85	14.65	14.65		
			$C = 1.79 > 1$	$C = 1.79 > 1$	$C = 1.63 > 1$	$C = 1.54 > 1$	$C = 1.47 > 1$							
		1	0.76	0.76	2.30	2.30	6.43	6.43	10.85	10.85	14.65	14.65		
0.5	1	1	$C = 13028.96 > 1$	$C = 4379.55 > 1$	$C = 1643.61 > 1$	$C = 1011.65 > 1$	$C = 776.74 > 1$							
			0.70	0.70	2.21	2.21	6.12	6.12	10.69	10.69	14.56	14.56		
		2	$C = 3.12 > 1$	$C = 2.82 > 1$	$C = 2.85 > 1$	$C = 2.31 > 1$	$C = 2.06 > 1$							
	0.9	3	0.70	0.70	2.21	2.21	6.12	6.12	10.69	10.69	14.56	14.56		
			$C = 1.79 > 1$	$C = 1.79 > 1$	$C = 1.63 > 1$	$C = 1.54 > 1$	$C = 1.47 > 1$							
		1	0.70	0.70	2.21	2.21	6.12	6.12	10.69	10.69	14.56	14.56		

Table 4.2: Symbol error rate (in%) for MAP decoding and instantaneous mapping θ^* for symmetric binary Markov sources with $p_{00} = 0.6$ (symmetric Markov source). The channel model is the NBND-C-QB, with $M = 1$, $Cor = 0, 0.9$, and $q = 1, 2, 3$. The values C are calculated from (4.20).

p_{00}	Cor	q	SNR											
			15		10		5		2		0			
			MAP	θ^*	MAP	θ^*	MAP	θ^*	MAP	θ^*	MAP	θ^*		
0.6	0	1	$C = 57.10 > 1$	$C = 18.66 > 1$	$C = 6.48 > 1$	$C = 3.65 > 1$	$C = 2.59 > 1$							
			0.76	2.30	6.43	10.85	14.65							
		2	$C = 1.39 > 1$	$C = 1.25 > 1$	$C = 1.27 > 1$	$C = 1.03 > 1$	$C = 0.91 < 1$							
	0	3	0.76	2.30	6.43	10.85	15.03							
			$C = 0.80 < 1$	$C = 0.80 < 1$	$C = 0.73 < 1$	$C = 0.68 < 1$	$C = 0.65 < 1$							
		1	0.73	2.22	6.23	10.55	14.35							
0.9	0	1	$C = 5790.65 > 1$	$C = 1946.47 > 1$	$C = 730.49 > 1$	$C = 449.62 > 1$	$C = 345.22 > 1$							
			0.70	2.21	6.12	10.69	14.56							
		2	$C = 1.39 > 1$	$C = 1.25 > 1$	$C = 1.27 > 1$	$C = 1.03 > 1$	$C = 0.91 < 1$							
	0		0.70	2.21	6.12	10.69	14.58							
		3	$C = 0.80 < 1$	$C = 0.80 < 1$	$C = 0.73 < 1$	$C = 0.68 < 1$	$C = 0.65 < 1$							
			0.65	2.15	6.04	10.30	14.24							

Table 4.3: Symbol error rate (in%) for MAP decoding and instantaneous mapping θ^* for symmetric binary Markov sources with $p_{00} = 0.7$ (symmetric Markov source). The channel model is the NBNDQ-QB, with $M = 1$, $Cor = 0, 0.9$, and $q = 1, 2, 3$. The values C are calculated from (4.20).

p_{00}	Cor	q	SNR											
			15		10		5		2		0			
			MAP	θ^*	MAP	θ^*	MAP	θ^*	MAP	θ^*	MAP	θ^*		
0.7	0	1	$C = 23.60 > 1$	$C = 7.71 > 1$	$C = 2.68 > 1$	$C = 1.51 > 1$	$C = 1.07 > 1$							
			0.76	0.76	2.30	2.30	6.43	6.43	10.85	10.85	14.65	14.65		
		2	$C = 0.57 < 1$	$C = 0.52 < 1$	$C = 0.52 < 1$	$C = 0.42 < 1$	$C = 0.38 < 1$							
	0.9	3	0.64	0.76	1.91	2.30	5.55	6.43	9.50	10.85	12.99	14.65		
			$C = 0.33 < 1$	$C = 0.33 < 1$	$C = 0.30 < 1$	$C = 0.28 < 1$	$C = 0.27 < 1$							
		1	0.62	0.76	1.90	2.30	5.37	6.43	9.27	10.85	12.79	14.65		
0.9	1	$C = 2393.07 > 1$	$C = 804.41 > 1$	$C = 301.89 > 1$	$C = 185.81 > 1$	$C = 142.67 > 1$								
		0.70	0.70	2.21	2.21	6.12	6.12	10.69	10.69	14.56	14.56			
	2	$C = 0.57 < 1$	$C = 0.52 < 1$	$C = 0.52 < 1$	$C = 0.42 < 1$	$C = 0.38 < 1$								
0.9	3	0.62	0.70	1.87	2.21	5.61	6.12	9.86	10.69	13.47	14.56			
		$C = 0.33 < 1$	$C = 0.33 < 1$	$C = 0.30 < 1$	$C = 0.28 < 1$	$C = 0.27 < 1$								
	1	0.55	0.70	1.84	2.21	5.30	6.12	9.31	10.69	13.13	14.56			

Table 4.4: Symbol error rate (in%) for MAP decoding and instantaneous mapping θ^* for symmetric binary Markov sources with $p_{00} = 0.9$ (symmetric Markov source). The channel model is the NBND-C-QB, with $M = 1$, $Cor = 0, 0.9$, and $q = 1, 2, 3$. The values C are calculated from (4.20).

p_{00}	Cor	q	SNR												
			15		10		5		2		0				
			MAP	θ^*	MAP	θ^*	MAP	θ^*	MAP	θ^*	MAP	θ^*			
0.9	0	1	$C = 1.59 > 1$	$C = 0.52 < 1$	$C = 0.18 < 1$	$C = 0.10 < 1$	$C = 0.07 < 1$								
			0.76	0.76	1.74	2.30	3.65	6.43	6.56	10.85	8.68	14.65			
		2	$C = 0.04 < 1$	$C = 0.03 < 1$	$C = 0.04 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$			
	0.9	3		0.27	0.76	0.89	2.30	2.66	6.43	4.59	10.85	6.54	14.65		
			1	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$		
				0.22	0.76	0.69	2.30	2.16	6.43	4.17	10.85	6.00	14.65		
	0.9	1		$C = 160.80 > 1$	$C = 54.07 > 1$	$C = 20.29 > 1$	$C = 12.49 > 1$	$C = 9.59 > 1$							
				0.70	0.70	2.21	2.21	6.12	6.12	10.69	10.69	14.56	14.56		
			2	$C = 0.04 < 1$	$C = 0.03 < 1$	$C = 0.04 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$	$C = 0.03 < 1$		
0.9		3		0.29	0.70	0.96	2.21	3.48	6.12	7.03	10.69	10.73	14.56		
			1	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$	$C = 0.02 < 1$		
				0.23	0.70	0.76	2.21	2.64	6.12	5.34	10.69	8.50	14.56		

4.2 Numerical Results

We now present numerical results for the described communication system, over both the NBNDC-QB model and the Rayleigh DFC model.

Similar to Chapter 3, several source distributions are tested, including memoryless (i.i.d.) Gaussian and Laplacian sources and Gauss-Markov sources. All of the source models have zero mean and unit variance.

For each simulation, the SQ training and statistics collection is done over a set of 1,000,000 source symbols. Afterwards, 100,000 source symbols are transmitted for simulation and the SDR with the mean square error distortion is measured. We ran each simulation 10 times and took average for more consistent results.

4.2.1 Exploiting memory and soft-decision quantization

Tables 4.5-4.11 depict simulation results (in dB) for different sources over the NBNDC-QB model with several parameters of SNR, SQ codeword length n , noise correlation Cor, channel memory order M , and soft-decision resolution q .

Memoryless sources

As can be seen from the tables, the performance of a system with high noise correlation (or even moderate correlation as in Table 4.6,) can be significantly better than a system working over a fully-interleaved (Cor = 0) channel. For example, more than 2.5 dB of SDR gain is obtained for memoryless Gaussian sources at $q = 3$, $n = 3$, SNR = 2 (see Table 4.5). Furthermore, for $n = 1$ since the quantized codewords form a symmetric i.i.d. source, the results illustrate Theorem 1 of [3] and Theorem

4.1 (compare the results of Tables 4.5 and 4.12 for $n = 1$). Considerable gains (up to 2.25 dB) are also obtained by increasing the quantizer resolution to $q = 2$ (at $n = 3$, SNR = 5, Cor = 0.9 for Laplacian sources). More gain is obtained for a 3-bit quantizer.

Gauss-Markov sources

For Gauss-Markov sources, we have up to 3.8 dB SDR gain (at $q = 3$, $n = 3$, SNR = 0), by exploiting the noise correlation instead of interleaving the channel. As can be seen, in general better performance is observed when channel is highly correlated.

At low rates, especially at $n = 1$, the SDR performance for the correlated channel is worse than that for the uncorrelated channel. This behavior was expected for $n = 1$ and $q = 1$ using Corollary 3 of [3]. According to this corollary and the numerical results, for the correlated channel, the source memory has a mismatch with the channel memory. As a result, increasing the channel noise correlation will also increase the mismatch between the source and channel memory information. This makes the SQ-MAP perform worse on correlated channels than over uncorrelated channels. However, this mismatch does not occur for higher rates ($n > 2$) and the SDR performance of the system significantly improves with increasing channel noise correlation.

The mismatch between the source and channel memory is investigated for the binary Markov channel in [3], in which to handle this mismatch a convolutional coder is proposed in order to convert the source memory redundancy ρ_M to redundancy ρ_D due to non-uniformity.

It is also observed that system gains up to 2.8 dB (at $n = 3$, $\text{SNR} = 2$, $\text{Cor} = 0.9$ for Gauss-Markov sources) using only a 2-bit soft-decision quantizer in the receiver over a hard-decision quantizer ($q = 1$). More gain is obtained by further increasing of q .

4.2.2 Validating the NBNDC-QB model

We next assess how well the NBNDC-QB model can approximate the correlated Rayleigh DFC in terms of SDR performance of the SQ-MAP system.

To validate the NBNDC-QB model as a good representation of the Rayleigh DFC for SQ-MAP systems, we design the MAP detector using the path metric obtained for an NBNDC-QB (with proper parameters to match a specific Rayleigh DFC) and run simulation using both the NBNDC-QB and the Rayleigh DFC to compare their performance. For a given DFC (with fixed SNR and $f_D T$) and a given q , we have used the same technique described in Chapter 3, to fit the NBNDC-QB to the Rayleigh DFC. The ρ_j values and other channel parameters used for simulation are given in Tables 3.14 and 3.10 respectively.

To simulate the Rayleigh DFC, we generate the fading coefficients using the modified Clarke's method introduced in [36]. Simulation results (over the NBNDC-QB and Rayleigh DFC channels) in terms of SDR are shown in Tables 4.14-4.16 for memoryless i.i.d. Gaussian sources, Gauss-Markov sources with correlation factor $\phi = 0.9$, and memoryless i.i.d. Laplacian sources, respectively. The results in symbol error rate are depicted in Tables 4.17-4.19.

Comparing the performance of the system for the two channels, we observe that for lower rates (codeword lengths $n = 1$ and 2 for the memoryless Gaussian source),

there is a good conformity between the results for the two channel models. This agreement in SDR performance can be heuristically explained by noting that for low rates ($n = 1$ and 2), the SQ output sent to the channel input is nearly i.i.d. uniform. But the NBND-C-QB and DFC channels were matched by minimizing the divergence rate between their noise processes. Hence, when both channels are driven by the same capacity-achieving input (which must be i.i.d. uniform as both channels are symmetric), they will then have a similar probability of error performance in addition to nearly identical capacities. The same agreement in SDR performance is also observed for memoryless Laplacian and Gauss-Markov sources for $n = 1$. We finally note that for $n \geq 3$, some disagreement in SDR performance is observed between the two systems (in this case the SQ output is not i.i.d. uniform). The distribution $P(\mathbf{x})$ of the quantized symbols is depicted in Table 4.20.

Table 4.5: SQ-MAP Training SDR results (in dB) for the memoryless NBND-C-QB and the highly correlated NBND-C-QB with parameters $M = 1$, $\alpha = 1$; memoryless Gaussian source.

q	n	Memoryless (Cor=0)					Cor=0.9				
		SNR (dB)					SNR (dB)				
		15	10	5	2	0	15	10	5	2	0
1	1	4.17	3.75	2.78	1.94	1.33	4.19	3.77	2.85	1.97	1.35
	2	8.15	6.49	3.85	2.14	1.10	8.37	6.89	4.47	2.84	1.82
	3	11.05	7.80	4.02	1.93	0.70	11.58	8.43	4.76	2.76	1.62
2	1	4.17	3.75	2.78	1.94	1.33	4.19	3.77	2.85	1.97	1.35
	2	8.15	6.49	3.85	2.14	1.10	8.69	7.68	5.61	4.03	3.05
	3	11.10	7.94	4.33	2.53	1.45	12.61	10.15	6.64	4.51	3.30
3	1	4.17	3.75	2.78	1.94	1.33	4.19	3.77	2.85	1.97	1.35
	2	8.17	6.54	4.02	2.41	1.44	8.76	7.77	5.91	4.48	3.56
	3	11.15	7.98	4.38	2.57	1.51	12.86	10.52	7.16	5.12	4.00

Table 4.6: SQ-MAP Training SDR results (in dB) for the NBND-C-QB with parameters $M = 1$, $\alpha = 1$, and correlations 0.5, 0.7; memoryless Gaussian source.

q	n	Cor=0.5					Cor=0.7				
		SNR (dB)					SNR (dB)				
		15	10	5	2	0	15	10	5	2	0
1	1	4.17	3.74	2.78	1.95	1.32	4.18	3.75	2.79	1.94	1.34
	2	8.26	6.67	4.13	2.44	1.38	8.28	6.75	4.23	2.57	1.52
	3	11.18	7.90	4.16	2.06	0.84	11.30	8.06	4.30	2.22	1.00
2	1	4.17	3.74	2.78	1.95	1.32	4.18	3.75	2.79	1.94	1.34
	2	8.31	6.77	4.27	2.64	1.61	8.43	7.01	4.59	3.01	2.00
	3	11.40	8.31	4.61	2.66	1.51	11.76	8.76	5.08	3.08	1.92
3	1	4.17	3.74	2.80	1.95	1.33	4.17	3.73	2.79	1.95	1.35
	2	8.33	6.85	4.44	2.83	1.86	8.43	7.06	4.78	3.21	2.25
	3	11.49	8.43	4.82	2.87	1.76	11.87	8.92	5.36	3.38	2.22

Table 4.7: SQ-MAP Training SDR results (in dB) for the memoryless NBND-CQB and the highly correlated NBND-CQB with parameters $M = 1$, $\alpha = 1$; Gauss-Markov source with correlation factor $\phi = 0.9$.

q	n	Memoryless (Cor=0)					Cor=0.9				
		SNR (dB)					SNR (dB)				
		15	10	5	2	0	15	10	5	2	0
1	1	4.21	3.78	3.74	3.22	3.09	4.23	3.81	2.89	2.01	1.39
	2	8.95	8.29	6.97	6.34	5.41	8.94	8.24	6.88	5.71	4.90
	3	13.38	11.84	9.46	7.52	6.18	13.89	12.69	10.43	8.69	7.51
2	1	4.38	4.23	3.89	3.61	3.22	4.35	4.14	3.44	2.65	2.02
	2	9.16	8.81	8.00	6.98	6.08	9.24	8.97	8.26	7.36	6.59
	3	13.98	12.87	10.72	8.91	7.56	14.47	14.02	12.88	11.51	10.34
3	1	4.39	4.30	4.01	3.65	3.39	4.37	4.20	3.62	2.97	2.39
	2	9.18	8.87	8.09	7.24	6.43	9.29	9.18	8.64	8.00	7.38
	3	14.07	13.06	10.96	9.16	7.81	14.57	14.34	13.47	12.58	11.58

Table 4.8: SQ-MAP Training SDR results (in dB) for the memoryless NBND-CQB and the highly correlated NBND-CQB with parameters $M = 1$, $\alpha = 1$; memoryless Laplacian source.

q	n	Memoryless (Cor=0)					Cor=0.9				
		SNR (dB)					SNR (dB)				
		15	10	5	2	0	15	10	5	2	0
1	1	2.87	2.62	2.00	1.44	1.00	2.88	2.63	2.05	1.45	1.01
	2	6.65	5.27	2.91	1.30	0.27	6.89	5.88	4.28	3.21	2.56
	3	9.59	6.49	2.72	0.58	0.82	10.14	7.64	4.88	3.34	2.51
2	1	2.87	2.62	2.00	1.44	1.00	2.88	2.63	2.05	1.45	1.01
	2	6.69	5.42	3.32	2.01	1.18	7.26	6.72	5.53	4.47	3.79
	3	9.90	7.09	3.81	2.06	1.06	11.59	9.86	7.14	5.35	4.30
3	1	2.87	2.62	2.00	1.44	1.00	2.88	2.63	2.05	1.45	1.01
	2	6.72	5.45	3.27	2.01	1.20	7.32	6.91	5.88	4.97	4.36
	3	9.96	7.12	3.75	1.99	1.27	11.85	10.47	7.94	6.23	5.27

Table 4.9: SQ-MAP Training SDR results (in dB) for the memoryless NBND-CQB and the highly correlated NBND-CQB with parameters $M = 5$, $\alpha = 0.5$; memoryless Gaussian source.

q	n	Memoryless (Cor=0)					Cor=0.9				
		SNR (dB)					SNR (dB)				
		15	10	5	2	0	15	10	5	2	0
1	1	4.17	3.75	2.78	1.94	1.33	4.22	3.86	2.89	2.03	1.45
	2	8.15	6.49	3.85	2.14	1.10	8.37	7.06	4.61	2.91	1.80
	3	11.05	7.80	4.02	1.93	0.70	13.11	11.11	8.24	6.69	5.81
2	1	4.17	3.75	2.78	1.94	1.33	4.22	3.86	2.89	2.03	1.45
	2	8.15	6.49	3.85	2.14	1.10	9.03	8.32	6.77	5.34	4.35
	3	11.10	7.94	4.33	2.53	1.45	14.09	13.03	10.87	9.21	8.09
3	1	4.17	3.75	2.78	1.94	1.33	4.22	3.86	2.89	2.03	1.45
	2	8.17	6.54	4.02	2.41	1.44	9.14	8.75	7.42	6.39	5.69
	3	11.15	7.98	4.38	2.57	1.51	14.29	13.65	11.74	10.34	9.38

Table 4.10: SQ-MAP Training SDR results (in dB) for the memoryless NBND-CQB and the highly correlated NBND-CQB with parameters $M = 5$, $\alpha = 1$; memoryless Gaussian source.

q	n	Memoryless (Cor=0)					Cor=0.9				
		SNR (dB)					SNR (dB)				
		15	10	5	2	0	15	10	5	2	0
1	1	4.17	3.75	2.78	1.94	1.33	4.26	3.91	2.90	2.09	1.44
	2	8.15	6.49	3.85	2.14	1.10	8.40	7.07	4.61	2.95	1.85
	3	11.05	7.80	4.02	1.93	0.70	13.18	11.27	8.50	6.90	6.07
2	1	4.17	3.75	2.78	1.94	1.33	4.26	3.91	2.90	2.09	1.44
	2	8.15	6.49	3.85	2.14	1.10	9.09	8.35	6.57	5.25	4.20
	3	11.10	7.94	4.33	2.53	1.45	14.33	13.02	11.03	9.32	8.36
3	1	4.17	3.75	2.78	1.94	1.33	4.26	3.91	2.90	2.09	1.44
	2	8.17	6.54	4.02	2.41	1.44	9.12	8.72	7.46	6.38	5.46
	3	11.15	7.98	4.38	2.57	1.51	14.29	13.75	11.87	10.46	9.59

Table 4.11: SQ-MAP Training SDR results (in dB) for the memoryless NBND-C-QB and the highly correlated NBND-C-QB with parameters $M = 5$, $\alpha = 2$; memoryless Gaussian source.

q	n	Memoryless (Cor=0)					Cor=0.9				
		SNR (dB)					SNR (dB)				
		15	10	5	2	0	15	10	5	2	0
1	1	4.17	3.75	2.78	1.94	1.33	4.19	3.85	2.93	2.23	1.43
	2	8.15	6.49	3.85	2.14	1.10	8.41	7.09	4.52	2.86	1.68
	3	11.05	7.80	4.02	1.93	0.70	13.25	11.42	8.84	7.26	6.33
2	1	4.17	3.75	2.78	1.94	1.33	4.19	3.85	2.93	2.23	1.43
	2	8.15	6.49	3.85	2.14	1.10	9.03	8.00	6.12	4.56	3.57
	3	11.10	7.94	4.33	2.53	1.45	14.33	13.28	11.11	9.63	8.66
3	1	4.17	3.75	2.78	1.94	1.33	4.19	3.85	2.93	2.23	1.43
	2	8.17	6.54	4.02	2.41	1.44	9.07	8.54	7.06	5.74	4.84
	3	11.15	7.98	4.38	2.57	1.51	14.31	13.76	11.89	10.79	9.80

Table 4.12: SQ with instantaneous mapping- Training SDR results (in dB) for the memoryless NBND-C-QB and the highly correlated NBND-C-QB with parameters $M = 1$, $\alpha = 1$; memoryless Gaussian source.

q	n	Memoryless (Cor=0)					Cor=0.9				
		SNR (dB)					SNR (dB)				
		15	10	5	2	0	15	10	5	2	0
1	1	4.17	3.75	2.78	1.94	1.33	4.19	3.77	2.85	1.97	1.35
	2	8.15	6.49	3.85	2.14	1.10	8.37	6.89	4.42	2.74	1.66
	3	11.05	7.80	4.02	1.93	0.70	11.58	8.38	4.61	2.47	1.21
2	1	4.17	3.75	2.78	1.94	1.33	4.19	3.77	2.85	1.97	1.35
	2	8.15	6.49	3.85	2.14	1.10	8.37	6.89	4.42	2.74	1.66
	3	11.05	7.80	4.02	1.93	0.70	11.58	8.38	4.61	2.47	1.21
3	1	4.17	3.75	2.78	1.94	1.33	4.19	3.77	2.85	1.97	1.35
	2	8.15	6.49	3.85	2.14	1.10	8.37	6.89	4.42	2.74	1.66
	3	11.05	7.80	4.02	1.93	0.70	11.58	8.38	4.61	2.47	1.21

Table 4.13: SQ with instantaneous mapping- Training SDR results (in dB) for the memoryless NBND-C-QB and the highly correlated NBND-C-QB with parameters $M = 1$, $\alpha = 1$; Gauss-Markov source with correlation factor $\phi = 0.9$.

q	n	Memoryless (Cor=0)					Cor=0.9				
		SNR (dB)					SNR (dB)				
		15	10	5	2	0	15	10	5	2	0
1	1	4.21	3.78	2.82	1.97	1.36	4.23	3.81	2.89	2.01	1.39
	2	8.19	6.52	3.87	2.16	1.11	8.40	6.91	4.44	2.74	1.66
	3	11.13	7.81	4.03	1.93	0.71	11.59	8.38	4.64	2.51	1.24
2	1	4.21	3.78	2.82	1.97	1.36	4.23	3.81	2.89	2.01	1.39
	2	8.19	6.52	3.87	2.16	1.11	8.40	6.91	4.44	2.74	1.66
	3	11.13	7.81	4.03	1.93	0.71	11.59	8.38	4.64	2.51	1.24
3	1	4.21	3.78	2.82	1.97	1.36	4.23	3.81	2.89	2.01	1.39
	2	8.19	6.52	3.87	2.16	1.11	8.40	6.91	4.44	2.74	1.66
	3	11.13	7.81	4.03	1.93	0.71	11.59	8.38	4.64	2.51	1.24

Table 4.14: SQ-MAP simulation SDR results (in dB) for the DFC-fitted NBND-C-QB and the DFC; memoryless Gaussian source, $q = 2$.

Channel model	$f_D T$	n	SNR (dB)			
			15	10	5	2
			Cor=0.35	Cor=0.32	Cor=0.29	Cor=0.22
NBND-C-QB	0.005	1	4.18	3.76	2.77	1.94
		2	8.34	6.73	4.01	2.29
		3	11.75	8.62	4.48	2.28
	0.01	1	4.17	3.75	2.80	1.94
		2	8.34	6.75	4.07	2.28
		3	11.60	8.59	4.67	2.35
Rayleigh DFC	0.005	1	4.17	3.75	2.78	1.94
		2	8.15	6.52	3.91	2.18
		3	11.05	7.77	3.97	1.89
	0.01	1	4.18	3.75	2.79	1.95
		2	8.17	6.51	3.88	2.18
		3	11.13	7.77	3.91	1.90

Table 4.15: SQ-MAP simulation SDR results (in dB) for the DFC-fitted NBND-C-QB and the DFC; Gauss-Markov source with correlation factor $\phi = 0.9$, $q = 2$.

Channel model	$f_D T$	n	SNR (dB)			
			15 Cor=0.35	10 Cor=0.32	5 Cor=0.29	2 Cor=0.22
NBND-C-QB	0.005	1	4.35	4.15	3.73	3.19
		2	9.19	8.76	7.66	6.33
		3	14.25	13.07	10.48	8.02
	0.01	1	4.35	4.16	3.74	3.21
		2	9.20	8.80	7.71	6.41
		3	14.19	13.22	10.72	8.26
Rayleigh DFC	0.005	1	4.22	3.88	3.28	2.82
		2	8.82	7.99	6.39	5.05
		3	13.28	11.38	8.38	6.41
	0.01	1	4.24	3.93	3.39	2.85
		2	8.90	8.15	6.59	5.33
		3	13.35	11.52	8.60	6.66

Table 4.16: SQ-MAP simulation SDR results (in dB) for the DFC-fitted NBND-C-QB and the DFC; memoryless Laplacian source, $q = 2$.

Channel model	$f_D T$	n	SNR (dB)			
			15 Cor=0.35	10 Cor=0.32	5 Cor=0.29	2 Cor=0.22
NBND-C-QB	0.005	1	2.87	2.62	2.00	1.44
		2	6.92	5.77	3.26	1.66
		3	10.73	8.09	4.06	1.69
	0.01	1	2.87	2.62	2.02	1.44
		2	6.91	5.82	3.51	1.76
		3	10.62	8.03	4.21	1.80
Rayleigh DFC	0.005	1	2.87	2.61	2.01	1.43
		2	6.50	4.99	2.68	1.16
		3	9.50	6.42	2.64	0.63
	0.01	1	2.87	2.61	2.00	1.43
		2	6.57	5.09	2.67	1.20
		3	9.64	6.42	2.64	0.71

Table 4.17: SQ-MAP simulation results in symbol error rate(%), for the DFC-fitted NBND-CQB and the DFC; memoryless Gaussian source, $q = 2$.

Channel model	$f_D T$	n	SNR (dB)				
			15 Cor=0.35	10 Cor=0.32	5 Cor=0.29	2 Cor=0.22	
NBND-CQB	1	0.005	0.73	2.27	6.45	10.86	
			2	1.12	3.65	10.73	18.21
			3	1.33	4.31	13.10	22.77
	2	0.01	0.74	2.31	6.36	10.87	
			2	1.13	3.61	10.55	18.36
			3	1.42	4.35	12.75	22.94
Rayleigh DFC	1	0.005	0.75	2.31	6.41	10.85	
			2	1.46	4.29	11.61	19.42
			3	2.03	6.06	16.32	26.70
	2	0.01	0.73	2.30	6.39	10.76	
			2	1.42	4.34	11.78	19.53
			3	2.05	6.24	16.61	26.92

Table 4.18: SQ-MAP simulation results in symbol error rate(%), for the DFC-fitted NBND-CQB and the DFC; Gauss-Markov source with correlation factor $\phi = 0.9$, $q = 2$.

Channel model	$f_D T$	n	SNR (dB)				
			15 Cor=0.35	10 Cor=0.32	5 Cor=0.29	2 Cor=0.22	
NBND-CQB	1	0.005	0.36	1.20	3.30	6.27	
			2	0.36	1.36	4.55	9.36
			3	0.43	1.69	5.95	12.40
	2	0.01	0.37	1.21	3.44	6.31	
			2	0.38	1.39	4.67	9.65
			3	0.50	1.71	6.07	12.82
Rayleigh DFC	1	0.005	0.77	2.14	5.27	8.14	
			2	1.21	3.43	8.97	14.87
			3	1.57	4.61	12.39	20.32
	2	0.01	0.72	2.02	5.02	8.05	
			2	1.15	3.37	8.96	14.57
			3	1.59	4.72	12.66	20.46

Table 4.19: SQ-MAP simulation results in symbol error rate(%), for the DFC-fitted NBNDC-QB and the DFC; memoryless Laplacian source, $q = 2$.

Channel model	$f_D T$	n	SNR (dB)			
			15 Cor=0.35	10 Cor=0.32	5 Cor=0.29	2 Cor=0.22
NBNDC-QB	1	0.005	0.73	2.27	6.45	10.86
	2		0.89	2.95	9.76	16.75
	3		0.99	3.33	10.91	19.95
	1	0.01	0.74	2.31	6.36	10.87
	2		0.91	2.87	9.01	16.52
	3		1.08	3.43	10.71	20.29
Rayleigh DFC	1	0.005	0.77	2.34	6.44	10.88
	2		1.66	4.89	12.48	20.09
	3		2.06	6.14	16.42	26.85
	1	0.01	0.74	2.33	6.43	10.83
	2		1.59	4.78	12.59	20.09
	3		2.07	6.34	16.89	27.10

Table 4.20: $P(\mathbf{x})$ for different source distributions and SQ coding rates.

Distribution	R	$P(\mathbf{x})$							
<i>Gaussian</i>	1	0.499	0.501	-	-	-	-	-	-
	2	0.335	0.165	0.335	0.165	-	-	-	-
	3	0.186	0.111	0.160	0.043	0.186	0.111	0.160	0.043
<i>Laplace</i>	1	0.500	0.500	-	-	-	-	-	-
	2	0.397	0.104	0.396	0.103	-	-	-	-
	3	0.258	0.072	0.152	0.019	0.257	0.072	0.151	0.019
<i>Gauss-Markov</i>	1	0.497	0.503	-	-	-	-	-	-
	2	0.334	0.163	0.336	0.167	-	-	-	-
	3	0.185	0.110	0.160	0.043	0.185	0.111	0.161	0.045

Chapter 5

Conclusions and Future Work

We designed two joint source-channel coding schemes for the NBNDQ-QB channel in order to take advantage of channel memory and soft-decision information.

The COVQ performance results show that the COVQ system, as a joint source-channel coding technique, can successfully exploit the channel's memory and soft-decision information to combat channel errors while having the advantage of low encoding/decoding delay in comparison with tandem coding systems. Furthermore, the NBNDQ-QB model, which (unlike the Rayleigh DFC) is mathematically tractable by virtue of having closed-form expressions for its statistics, was experimentally shown to be a practical model for the Rayleigh DFC in terms of COVQ performance.

A scalar quantizer based MAP decoding system (SQ-MAP) was also designed over the recently introduced channel model NBNDQ-QB, which consists of a scalar quantizer, a proper index assignment and a sequence maximum a posteriori (MAP) decoder designed to harness the redundancy left in the quantizers indices, the channels soft-decision output, and noise time correlation. The system was tested for both correlated and uncorrelated source distributions and numerical results show that the

proposed system can successfully utilize memory and soft-decision information over the NBNDQ-QB channel model. It was also observed that for correlated sources, in some cases the system with interleaved channel (system with no noise correlation) performs better than the system with highly correlated channel. This observation was investigated and necessary and sufficient condition for this phenomenon to occur was obtained at rate $R = 1$. Finally, the channel model was compared to the Rayleigh DFC, in terms of SDR, and it was shown numerically that the NBNDQ-QB model can effectively approximate the Rayleigh discrete fading channels for low coding rates, while providing closed form expression for transition distribution. However, some degradation was observed for approximating the Rayleigh DFC for high rates.

This work can be extended in some directions. The COVQ system we presented here does not use the optimal quantizer step size δ for soft-decision decoding. Note that since our training algorithm uses the resulting codebook of each COVQ with a specific SNR and an initial codebook of an other COVQ with lower SNR, each δ value affects the whole collection of COVQ systems (for different SNRs). Consequently, the choice of proper δ value for a COVQ with specific SNR is rather complicated since each candidate δ value improves the performance for some cases and deteriorates it for some others. Defining a proper qualifying factor and further optimization of the systems performance by finding the optimal set of δ values could a subject of interest for future work.

For the SQ-MAP system, further investigations can be made in order to find the cases were the MAP-decoder is *useless* for higher rates or channel noise memory orders. Furthermore, the optimal index assignment is still an open problem, and we

observed that proper choice of index assignment can critically improve the performance of the SQ-MAP system.

Bibliography

- [1] F. Alajaji and T. Fuja. A communication channel modeled on contagion. *IEEE Trans. Inform. Theory*, 40:2035 –2041, Nov. 1994.
- [2] F. Alajaji, N. Phamdo, N. Farvardin, and T. E. Fuja. Detection of binary sources over discrete channels with additive Markov noise. *Institute for Systems Research Tech. Rep. TR 94-31, University of Maryland, College Park, MD 20742*, 1994.
- [3] F. Alajaji, N. Phamdo, N. Farvardin, and T. E. Fuja. Detection of binary Markov sources over channels with additive Markov noise. *IEEE Trans. Inform. Theory*, 42(1):230 –239, Jan. 1996.
- [4] F. Alajaji and N. C. Phamdo. Soft-decision covq for Rayleigh-fading channels. *IEEE Communications Letters*, 2(6):162 –164, Jun. 1998.
- [5] E. Ayanoglu and R. Gray. The design of joint source and channel trellis waveform coders. *IEEE Trans. Inform. Theory*, 33(6):855 – 865, Nov. 1987.
- [6] F. Behnamfar, F. Alajaji, and T. Linder. Channel-optimized quantization with soft-decision demodulation for space-time orthogonal block-coded channels. *IEEE Trans. Signal Proc.*, 54(10):3935 –3946, Oct. 2006.

- [7] V. Černý. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51, 1985.
- [8] R. H. Clarke. A statistical theory of mobile-radio reception. *Bell Syst. Tech. J.*, 47(6):957–1000, 1968.
- [9] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2006.
- [10] R. L. Dobrushin and M. S. Pinsker. Memory increases transmission capacity. *Problemy Peredachi Informatsii*, 5(1):94–95, 1969.
- [11] E.B. Dynkin. *Markov Processes. Vol. 1*. Springer-Verlag Berlin, 1965.
- [12] N. Farvardin. A study of vector quantization for noisy channels. *IEEE Trans. Inform. Theory*, 36(4):799 –809, Jul. 1990.
- [13] N. Farvardin and V. Vaishampayan. Optimal quantizer design for noisy channels: An approach to combined source - channel coding. *IEEE Trans. Inform. Theory*, 33(6):827 – 838, Nov. 1987.
- [14] N. Farvardin and V. Vaishampayan. On the performance and complexity of channel-optimized vector quantizers. *IEEE Trans. Inform. Theory*, 37(1):155 –160, Jan. 1991.
- [15] G. D. Forney Jr. The Viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.
- [16] R. G. Gallager. *Information Theory and Reliable Communication*. Wiley, 1968.

- [17] A. Gersho and R.M. Gray. *Vector quantization and signal compression*. Springer Netherlands, 1992.
- [18] F.P. Kelly. Markovian functions of a Markov chain. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 372–379, 1982.
- [19] S. Kirkpatrick, C. D. Gelatt, and M.P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671, 1983.
- [20] H. Kumazawa, M. Kasahara, and T. Namekawa. A construction of vector quantizers for noisy channels. *Electronics and Communications in Japan (Part I: Communications)*, 67(4):39–47, 1984.
- [21] Y. Linde, A. Buzo, and R. Gray. An algorithm for vector quantizer design. *IEEE Trans. Commun.*, 28(1):84 – 95, Jan. 1980.
- [22] S. Lloyd. Least squares quantization in pcm. *IEEE Trans. Inform. Theory*, 28(2):129 – 137, Mar. 1982.
- [23] D. J. Miller and M. Park. A sequence-based approximate mmse decoder for source coding over noisy channels using discrete hidden Markov models. *IEEE Trans. Commun.*, 46(2):222 –231, Feb. 1998.
- [24] M. Mushkin and I. Bar-David. Capacity and coding for the gilbert-elliott channels. *IEEE Trans. Inform. Theory*, 35(6):1277 –1290, Nov. 1989.
- [25] N. Phamdo and F. Alajaji. Soft-decision demodulation design for COVQ over white, colored, and ISI Gaussian channels. *IEEE Trans. Commun.*, 48(9):1499 –1506, Sep. 2000.

- [26] N. Phamdo, F. Alajaji, and N. Farvardin. Quantization of memoryless and Gauss-Markov sources over binary Markov channels. *IEEE Trans. Commun.*, 45(6):668–675, Jun. 1997.
- [27] N. Phamdo and N. Farvardin. Optimal detection of discrete Markov sources over discrete memoryless channels-applications to combined source-channel coding. *IEEE Trans. Inform. Theory*, 40(1):186–193, Jan. 1994.
- [28] C. Pimentel and F. Alajaji. A discrete channel model for capturing memory and soft-decision information: A capacity study. In *Proc. ICC '09*, pages 1–6, Jun. 2009.
- [29] C. Pimentel, F. Alajaji, and P. Melo. A discrete queue-based model for soft-decision demodulated correlated fading channels. in *Proc. VTC'11*, May 2011.
- [30] C. E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27:379–423, 623–656, 1948.
- [31] C. E. Shannon. Coding theorems for a discrete source with a fidelity criterion. *IRE Nat. Conv. Rec*, 4(142-163), 1959.
- [32] C. E. Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001.
- [33] J. Singh, O. Dabeer, and U. Madhow. On the limits of communication with low-precision analog-to-digital conversion at the receiver. *IEEE Trans. Commun.*, 57(12):3629–3639, Dec. 2009.
- [34] M. Skoglund. Soft decoding for vector quantization over noisy channels with memory. *IEEE Trans. Inform. Theory*, 45(4):1293–1307, May 1999.

- [35] N. Wernersson and M. Skoglund. On source decoding based on finite-bandwidth soft information. In *Information Theory, 2005. Proceedings. International Symposium on*, pages 87 –91, Sep. 2005.
- [36] Y.R. Zheng and C. Xiao. Improved models for the generation of multiple uncorrelated Rayleigh fading waveforms. *IEEE Communications Letters*, 6(6):256 –258, Jun. 2002.
- [37] Libo Zhong, F. Alajaji, and G. Takahara. A binary communication channel with memory based on a finite queue. *IEEE Trans. Inform. Theory*, 53(8):2815 –2840, Aug. 2007.