



Graduate Theses, Dissertations, and Problem Reports

2011

Constellation Shaping for Communication Channels with Quantized Outputs

Chandana Nannapaneni
West Virginia University

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Nannapaneni, Chandana, "Constellation Shaping for Communication Channels with Quantized Outputs" (2011). *Graduate Theses, Dissertations, and Problem Reports*. 3290.
<https://researchrepository.wvu.edu/etd/3290>

This Thesis is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Thesis has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

Constellation Shaping for Communication Channels with Quantized Outputs

by

Chandana Nannapaneni

Thesis submitted to the
College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Master of Science
in
Electrical Engineering

Arun A. Ross, Ph.D.
Brian D. Woerner, Ph.D.
Matthew C. Valenti, Ph.D., Chair

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2011

Keywords: Communications, Information Theory, Constellation Shaping, Quantization,
Capacity, Optimization

Copyright 2011 Chandana Nannapaneni

Abstract

Constellation Shaping for Communication Channels with Quantized Outputs

by

Chandana Nannapaneni
Master of Science in Electrical Engineering

West Virginia University

Matthew C. Valenti, Ph.D., Chair

Channel capacity is an important aspect of every digital communication system. Capacity can be defined as the highest rate of information that can be transmitted over the channel with low error probability. The purpose of this research is to study the effect of the input symbol distribution on the information rate when the signal is transmitted over an Additive White Gaussian Noise (AWGN) channel with a quantized output. The channel was analyzed by transforming it into a Discrete Memoryless Channel(DMC), which is a discrete-input and discrete-output channel. Given the quantizer resolution and Signal-to-Noise Ratio (SNR), this thesis proposes a strategy for achieving the capacity of a certain shaping technique previously proposed by Le Goff, et. al. Under the constraints of the modulation, the shaping technique, and the quantizer resolution, the capacity is found by jointly optimizing the input distribution and quantizer spacing. The optimization is implemented by exhaustively searching over all feasible input distributions and a finely-spaced set of candidate quantizer spacings. The constrained capacity for 16-QAM modulation is found using the proposed technique.

Acknowledgements

My graduate career has been an educating and joyful journey. I cherished every moment of my graduate study, which included interesting courses, exciting research work and thought-provoking internship. I take the pleasure of thanking those who made an impact on my graduate life.

First and foremost, I am heartily thankful to my advisor, Dr. Matthew C. Valenti, who has guided and supported me in every way possible throughout my graduate career. I admire his technical expertise, time management and the respect he shows for his students. The challenging tasks he assigned to me made me realize my potential. I feel great pride in saying that I am Dr. Valenti's student. It is an honor for me to work for him. I owe all the success in my graduate life to my advisor.

I thank my parents, Venkateswara Rao Nannapaneni and Anuradha Nannapaneni, for their confidence in me. I especially thank my father who always reminded me the importance of hard work and time management. I thank my brother, Saideep Nannapaneni, who supported me in everything I did and provided valuable suggestions. I thank my sisters, Rishitha and Vanela for their love and affection for me. I thank Umesh Shewalla for his encouragement and for the driving lessons.

I would like to thank my colleague, Xingyu Xiang, who was very helpful in writing my thesis. I appreciate his patience in making me understand some of the basic concepts in my thesis. I am thankful to my other colleagues, Terry R. Ferret and Mohammad Fanaei. I approached them for all the technical questions I had. I really enjoyed the miscellaneous discussions we had.

Thanks to John Svitek, my supervisor at ANSYS Inc., for helping me improve my programming skills. Thanks to all my friends for making my stay at WVU memorable.

Contents

Acknowledgements	iii
List of Figures	vi
List of Tables	vii
1 Introduction	1
1.1 Introduction	1
1.2 Thesis Outline	2
2 Digital Modulation	3
2.1 Communication Model	3
2.2 Modulation and Demodulation	4
2.2.1 Modulation	4
2.2.2 Demodulation	6
2.2.3 16-QAM	8
2.3 Quantization	9
2.3.1 Vector vs. Scalar Quantizers	10
2.3.2 Uniform vs. non-Uniform Quantizers	12
2.3.3 Discrete Memoryless Channels	12
2.4 Summary	15
3 Information Rate	16
3.1 Capacity and Mutual Information	16
3.2 Mutual Information Evaluation for Quantized Output	17
3.3 Mutual Information Evaluation under Continuous Output	18
3.4 Information Rate Results	20
3.5 Summary	22
4 Constellation Shaping	23
4.1 Constellation Normalization	23
4.2 Constellation Shaping	25
4.3 Controlling the constellation distribution and symbol locations	29
4.4 Summary	32

5	Optimization Results	33
5.1	Finding Optimum PDF	33
5.2	Finding the optimal quantizer spacing	34
5.3	Joint Optimization	36
5.4	Observations	36
5.4.1	For two quantization bits	38
5.4.2	For three quantization bits	40
5.4.3	For four quantization bits	42
5.4.4	Optimum parameters	44
5.5	Summary	45
6	Conclusions	46
A	Appendix	48
A.1	Algorithm for Information rate calculation for two dimensions	48
	References	50

List of Figures

2.1	System model for a digital communications system.	4
2.2	16-QAM constellation diagram	8
2.3	2-D scalar quantization	10
2.4	Quantization regions in scalar quantization	11
2.5	Quantization regions in vector quantization	11
2.6	Uniform quantizer	13
2.7	Discrete M input, L output channel	14
3.1	Information rate variation with SNR for 16-QAM under continuous output with uniformly-distributed inputs.	21
3.2	Variation of information rate with SNR for 16-QAM under optimum spaced quantizer and non-optimum spaced quantizer with uniformly-distributed inputs	21
4.1	16-QAM constellation diagram	24
4.2	Normalized 16-QAM Constellation Diagram	25
4.3	Bit positions	26
4.4	16 QAM Constellation representation	27
4.5	Shaping operation	29
4.6	Constellation distribution of 16-QAM	30
4.7	Constellation diagram of 16-QAM when $p_0=0.9$	31
4.8	Constellation distribution of 16-QAM when $p_0=0.1$	32
5.1	Variation of information rate with quantizer spacing of 16-QAM at SNR = 2dB with 2 quantization bits/dimension	34
5.2	Two bits of quantization	38
5.3	Shaping Gain and Quantization Loss curves for two bits of quantization . . .	39
5.4	Three bits of quantization	40
5.5	Shaping Gain and Quantization Loss curves for three bits of quantization . .	41
5.6	Four bits of quantization	42
5.7	Shaping Gain and Quantization Loss curves for four bits of quantization . .	43
5.8	Variation of optimum p_0 with SNR for 16-QAM at different number of quantization bits/dimension	44
5.9	Variation of optimum p_0 with SNR for 16-QAM at different number of quantization bits/dimension	44

List of Tables

4.1	(3,5) shaping encoder	28
4.2	16-QAM constellation points	31
4.3	Constellation when $p_0 = 0.9$	31
4.4	Constellation when $p_0 = 0.1$	32
6.1	Quantization Loss comparison with R	46
6.2	Statistics at zero shaping gain	47
6.3	Statistics at shaping gain $\leq 0.1dB$	47

Chapter 1

Introduction

1.1 Introduction

Shannon's paper in 1948 [1] defines *capacity* as the highest rate at which data could be transmitted with low error probability. The capacity is found by computing the average mutual information between the channel input and channel output maximized over all possible input distributions. The average mutual information of the system is also known as the *information rate*. Finding conditions under which capacity can be maximized is important because a higher information rate enables a faster rate of reliable transmission. Papers related to channel capacity generally find capacity (mostly by Monte Carlo simulation) under the assumption of uniformly-distributed input symbols. This thesis relaxes this assumption in order to gain insight into the relationship between the input distribution and the achievable information rate. Furthermore, the thesis studies the effect of quantization, which is a necessary operation by modern receivers, on the capacity.

There has been some papers related to this topic in the recent years. The paper by Stefan Krone and Gerhard Fettweis in 2010 [2], considers the effect of uniform quantizer design on capacity. Also, the authors study the impact of phase offset between transmitter and receiver on the achievable rate of transmission. The paper by Wu, Davis and Calderbank in 2009 [3], talks about optimizing the input distribution, in conjunction with a particular quantization scheme called 'modulo-quantization', to achieve capacity. They do not consider the quantizer optimization. The Blahut-Arimoto algorithm [4], which is described by U.

Madhoo in [5], provides a methodology for optimizing the input distribution. His paper in 2008 [6] with Jaspreet Singh and Onkar Dabeer refers to the cutting-plane algorithm that gives the bounds for the optimal input distribution along with quantizer optimization.

1.2 Thesis Outline

The thesis is divided into 6 chapters. Chapters 2, 3 and 4 covers the digital communication concepts in general with a focus on 16-QAM. Chapter 5 presents the results of the work. Chapter 6 gives the conclusions based on the results observed and provides the scope for further research. The detailed contents of each chapter is further explained below.

Chapter 2 discusses all the blocks in a digital communication system model. The chapter walks over the concepts of modulation and demodulation. An entire subsection is dedicated to quantization basics and design. The importance of transition probabilities, in information rate evaluation when a discrete channel is considered, is discussed.

Chapter 3 deals with the crux of this work, capacity. Computing the capacity for quantized output and continuous output is discussed. For continuous case, a numerical method, gauss-hermite quadratures is used. The effect of the quantizer spacing in information rate is examined.

Chapter 4 talks about the constellation shaping. Constellation shaping refers to picking up a particular set of symbols more often than the other. Shaping can improve system performance in terms in achieving a better capacity, conserving energy etc.

Chapter 5 covers the results of our work. Capacity, achieved using uniform constellation and shaped constellation with continuous and quantized output, is determined and plotted. Using these results, shaping gain obtained by shaping the constellation is determined for continuous and quantized output. Also, quantization loss due to quantizing the received symbol is estimated for uniform and shaped constellations. Also, optimum input distribution and quantizer spacing curves that maximizes the information rate are shown. All these curves give us an understanding about the circumstances under which shaping is advantageous.

Chapter 6 presents the conclusions inferred from the results. We suggest the ideas for future research.

Chapter 2

Digital Modulation

This chapter deals with concepts related to digital communication systems. Section 2.1 discusses the blocks of a generic communication system. Section 2.2 focusses on digital modulation and demodulation, and introduces 16 QAM. Section 2.3 deals with the quantization operation at the receiver and in particular describes the concept of uniform quantization. Section 2.4 gives a summary of the chapter.

2.1 Communication Model

The generalized communication model using a digital modulation scheme is shown in Fig. 2.1. The input to a digital communication system is always assumed to be a binary sequence. If an analog signal needs to be transmitted, the analog signal is converted to a binary sequence. This input, which in this thesis is called the *data sequence*, is sent to the source encoder. The purpose of the source encoder is to represent the binary sequence by as few bits as possible. Thus the source encoder reduces or sometimes eliminates the redundancy in the data sequence. The output of the source encoder, which in this thesis is called the *information sequence*, is passed to the channel encoder. The channel encoder adds extra bits to this information sequence, which may be used to detect and correct errors introduced by the channel. Therefore, this redundancy increases the reliability of transmission. The output of the channel encoder is called *codeword*.

The modulator simply maps codeword produced by the channel encoder to a signal.

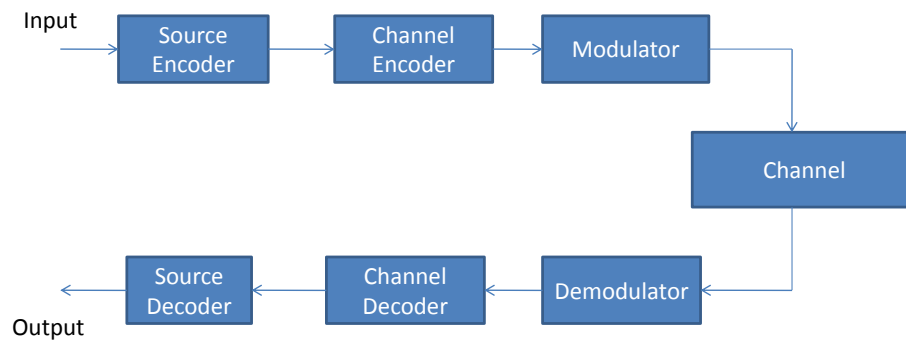


Figure 2.1: System model for a digital communications system.

Suppose we want to send one bit at a time, then the modulator sends a signal $s_0(t)$ for bit ‘0’ and $s_1(t)$ for bit ‘1’. We may also map multiple bits to a single signal. For example, if we have M different signals, we may take $\log_2(M)$ bits and map it to a signal. This is called M -ary modulation. These signals are then sent over the channel. The channel is the physical medium used to send information from transmitter to receiver. The channel may be atmosphere, wire lines, optical fibre etc. The demodulator collects these signals and tries to map them back to codeword. This is just the inverse operation of a modulator. The channel decoder takes this codeword and reconstructs the information sequence. Optionally, the demodulator may pass soft or *quantized* observations of the codeword to the channel decoder. When this is done, the decoder is said to be performing *soft-decision* decoding. The source decoder takes the reconstructed information sequence and produces an estimate of the data sequence by using its knowledge of the source encoding method used at the source encoder.

2.2 Modulation and Demodulation

2.2.1 Modulation

As mentioned in the previous section, the primary goal of the modulator is to map its input bits to a corresponding signal. There are several digital modulation techniques

that may be used, including binary phase shift keying (BPSK), pulse amplitude modulation (PAM), phase shift keying (PSK), and quadrature amplitude modulation (QAM). In general, if we opt for higher-order modulation schemes like 16-QAM or 64-QAM, then the advantage is that we can convey more bits with a signal. On the other hand, when higher-order modulation is used, the receiver is more likely to mistake one signal for another, i.e if the transmitter sends $s_m(t)$, the receiver may decode the signal to be $s_n(t)$, where $n \neq m$. The reason for the increased likelihood of error is that as we move towards higher-order modulations, we have signals placed relatively closer. This means the channel noise can now easily mislead the receiver, thereby causing an erroneous detection. This results in having higher error rates. Higher error rates can be mitigated by using powerful error detection and correction codes. All these issues (and many more) have to be thought of before selection of the modulation.

M -ary modulation maps a group of ‘ n ’ bits to one of the ‘ M ’ available signals ¹, where $M = 2^n$. Each of the these M signals is represented by a weighted linear combination of orthogonal functions, called the *basis functions*. The cardinality, K , of the set of basis functions is also called *dimensionality* of the modulation. Let

$$K = |\Phi| \quad (2.1)$$

where

$$\Phi = \{\phi_0(t), \phi_1(t), \dots, \phi_{K-1}(t)\} \quad (2.2)$$

is the basis for the signal set. When $K = 1$, we have a one-dimensional modulations like BPSK and PAM. When $K = 2$, we have a two-dimensional modulation like QAM. K can be as large as M . Therefore each signal in the M -signal set, $s_m(t)$, can be represented as,

$$s_m(t) = \sum_{k=0}^{K-1} s_{m,k} \phi_k(t) \quad (2.3)$$

The signal can be represented by its set of weights $\{s_{m,0}, \dots, s_{m,K-1}\}$. The set of weights may be placed into a K -dimensional vector \mathbf{s}_m . The set of signals may be visualized by a set of M points in K -dimensional space. This is the idea of a constellation diagram.

¹By *signal*, we mean the continuous analog signal. By *symbol*, we mean the discrete representation of the signal i.e the set of weights.

2.2.2 Demodulation

The received signal $r(t)$ is the input signal $s(t)$ corrupted by noise $n(t)$

$$r(t) = s(t) + n(t). \quad (2.4)$$

The received signal is sent through a bank of K correlators. In the i -th correlator, the received signal is multiplied with i -th basis function and the result is integrated over the duration of the basis function, which is denoted T_s . Consider $r(t)$ to be the received signal and r_i be the output of the i -th correlator. Then r_i can be found as,

$$r_i = \int_0^{T_s} r(t)\phi_i(t)dt \quad (2.5)$$

From (2.4)

$$r_i = \int_0^{T_s} [s(t) + n(t)]\phi_i(t)dt \quad (2.6)$$

Assuming that the j -th signal from the M signal set is transmitted,

$$s(t) = s_j(t) = \sum_{k=0}^{K-1} s_{j,k}\phi_k(t) \quad (2.7)$$

Using (2.7) in (2.6),

$$\begin{aligned} r_i &= \int_0^{T_s} r(t)\phi_i(t)dt \\ &= \int_0^{T_s} \left[\sum_{k=0}^{K-1} s_{j,k}\phi_k(t) + n(t) \right] \phi_i(t)dt \\ &= \sum_{k=0}^{K-1} s_{j,k} \int_0^{T_s} \phi_k(t)\phi_i(t)dt + \int_0^{T_s} n(t)\phi_i(t)dt \\ &= s_{j,i} + n_i, \end{aligned} \quad (2.8)$$

where

$$n_i = \int_0^{T_s} n(t)\phi_i(t)dt \quad (2.9)$$

n_i is the correlation of the noise with the i -th basis function. Now the signal $r(t)$ is represented by a set of weights $\{r_0, r_1, \dots, r_{K-1}\}$. Let the vector \mathbf{r} contain these weights.

From this \mathbf{r} , we need to estimate the most likely transmitted signal $\hat{\mathbf{s}}$. For finding this $\hat{\mathbf{s}}$, we use the following rule

$$\hat{\mathbf{s}} = \arg \max_{\mathbf{s}_j \in \mathcal{S}} p(\mathbf{s}_j | \mathbf{r}). \quad (2.10)$$

where \mathcal{S} is the signal set containing the M signals, \mathbf{r} is the received vector.

When soft-decision decoding is used, the set of $p(\mathbf{s}_j | \mathbf{r})$ is passed to the decoder directly without first applying the argmax operation.

Since the probability of \mathbf{s}_j is not directly known, we may apply Bayes' Theorem. Eqn. (2.10) can be written using Bayes' Theorem as,

$$\hat{\mathbf{s}} = \arg \max_{\mathbf{s}_j \in \mathcal{S}} \frac{p(\mathbf{r} | \mathbf{s}_j) p(\mathbf{s}_j)}{p(\mathbf{r})} \quad (2.11)$$

where

$$\mathbf{r} | \mathbf{s}_j \sim \mathcal{N} \left(\mathbf{s}_j, \frac{N_0}{2} I_K \right) \quad (2.12)$$

The above expression is a shorthand notation indicating that \mathbf{r} conditioned on \mathbf{s}_j is Gaussian with a mean of \mathbf{s}_j and a variance of $(N_0/2)$ in each dimension, where N_0 is the one-sided noise spectral density. I_K is the identity matrix whose size depends on the dimensionality of the modulation considered.

$p(\mathbf{r})$ in (2.11) can be ignored as we are trying to maximize over the input symbol distribution. Finding the most likely input symbol now reduces to,

$$\hat{\mathbf{s}} = \arg \max_{\mathbf{s}_i \in \mathcal{S}} p(\mathbf{r} | \mathbf{s}_i) p(\mathbf{s}_i). \quad (2.13)$$

From (2.12) and the pdf of the a Gaussian vector, (2.13) can be rewritten as,

$$\hat{\mathbf{s}} = \arg \max_{\mathbf{s}_i \in \mathcal{S}} \left(\frac{1}{(\pi N_0)^{K/2}} \exp \left\{ -\frac{1}{N_0} \|\mathbf{r} - \mathbf{s}_i\|^2 \right\} p(\mathbf{s}_i) \right) \quad (2.14)$$

From this equation, we make the following observations:

- If we assume uniform input distribution, $p(\mathbf{s}_i)$ from (2.14) can be neglected. The $\hat{\mathbf{s}}$ will be the symbol in the signal set that is closest to the received vector \mathbf{r} . Euclidean distance is used for distance measurement.

- If we assume nonuniform input distribution, the probability of occurrence of a symbol along with the closeness of the symbol to the received vector \mathbf{r} determines $\hat{\mathbf{s}}$. A particular symbol in the set \mathcal{S} may be the closest to \mathbf{r} , but if the probability of the symbol is low then that symbol will not be picked as the most probable symbol.

2.2.3 16-QAM

Throughout our work we focus on 16-QAM modulation. QAM is a two dimensional modulation scheme. QAM conveys two digital bit streams using two basis functions. Using 16-QAM, 4 bits can be modulated at a time. 16-QAM constellation diagram is shown.

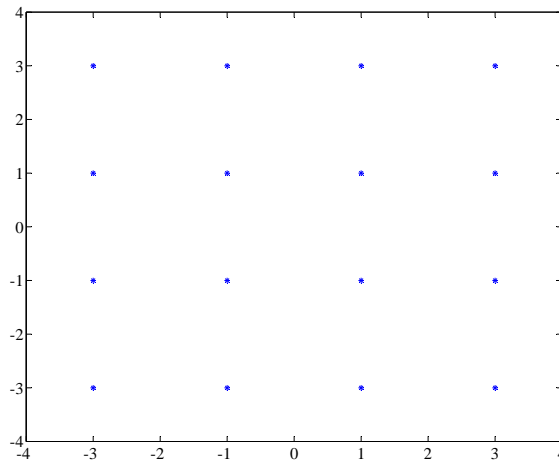


Figure 2.2: 16-QAM constellation diagram

In Fig. (2.2), where each axis represents a basis function, there are 4 equally spaced levels on each axis. The intersection of these levels forms symbols. Each symbol can be represented by the corresponding levels on both axes. For example, $(-1,-3)$ indicates a symbol.

2.3 Quantization

The received vector \mathbf{r} can be processed in two ways

- \mathbf{r} can be passed directly to the demodulator, which will operate on continuous-valued coefficients.
- \mathbf{r} can be quantized, and the quantized samples can be passed to the demodulator.

In practise, we always and should use the second way to process \mathbf{r} .

Quantization is a process of approximating the given value by a predefined set of levels, called quantization levels. As discussed in Section 2.2, r_i , the weights of the basis function can be found by passing the received signal $r(t)$ through a bank of correlators. Each of these weights is discrete in time but still continuous in amplitude. The purpose of the quantizer at the receiver is to make these weights discrete in amplitude. Suppose we have a one-dimensional quantizer with quantization levels $(-2, -1, 0, 1, 2)$. If a value 0.9 is given to the quantizer as input, the quantizer gives out the level closest to the input, i.e '1' in this case.

Every quantization level has a *quantization region* associated with it. If the input to the quantizer falls in the quantization region of quantization level q_i , the input is quantized to level q_i . One way of finding the quantization regions is by considering the midpoints of the quantization levels. Consider the one-dimensional quantizer stated in the above paragraph, the quantization levels are $(-2, -1, 0, 1, 2)$, the quantization regions can then be bounded by the midpoints $(-1.5, -0.5, 0.5, 1.5)$. For instance, any input that falls in $(-\infty -1.5)$ will be quantized to -2, i.e quantization region of level -2 is $(-\infty -1.5)$. Similarly the quantization region of level -1 is $(-1.5 -0.5)$, quantization region of level 0 is $(-0.5 0.5)$, quantization region of level 1 is $(0.5 1.5)$ and quantization region of level 2 is (1.5∞) . Number of quantization levels can be estimated by the number of bits of quantization. A 3-bit quantizer means we can use 3 bits to identify a quantization level. This indicates we can have 2^3 quantization levels.

In general, quantization results in a loss of precision. The system performance with quantization cannot be better than the system that works with direct continuous output. However, almost all practical systems have a quantizer in their receiver design. The analysis

with the continuous output is only to validate the results with quantization.

2.3.1 Vector vs. Scalar Quantizers

Scalar quantization and vector quantization are the different schemes of quantizing a given vector discussed in [7]. In scalar quantization, each element of the vector is quantized independently.

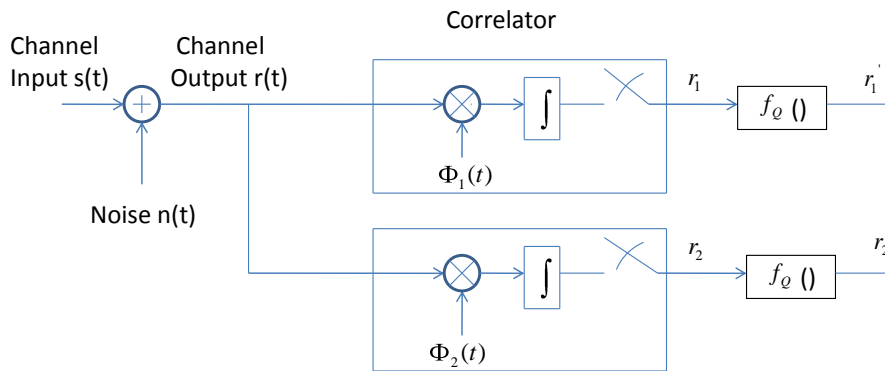


Figure 2.3: 2-D scalar quantization

Fig. 2.3 illustrates a receiver using two-dimensional scalar quantization. $s(t)$ is the channel input and $n(t)$ is the channel noise. The output of the channel $r(t) = s(t) + n(t)$. The $r(t)$ is then passed through the correlator to get the weights of the basis functions. \mathbf{r} is the vector containing all the weights. Since we have two basis functions, $\mathbf{r} = [r_1 \quad r_2]$. Each of the weights in the vector \mathbf{r} is quantized separately as shown. Another point to be noted is that the quantizers used for each dimension may not be symmetrical. If f_Q represents the quantization function. Then

$$r'_1 = f_Q(r_1)$$

$$r'_2 = f_Q(r_2)$$

In most cases, the quantization regions of a scalar quantizer are rectangles as shown in Fig. 2.4

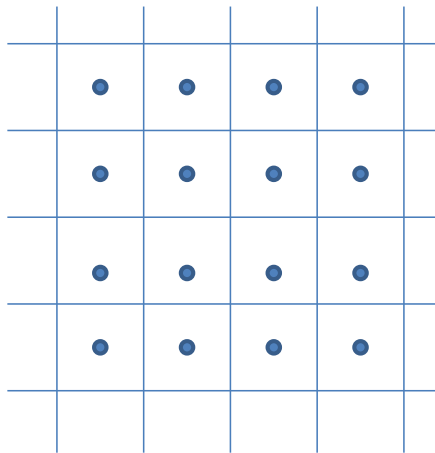


Figure 2.4: Quantization regions in scalar quantization

In vector quantization, quantization of the vector \mathbf{r} is done as a whole. We first divide the two-dimensional space into cells. All vectors that fall in a cell C_k are quantized to a vector q_k . The fundamental result from rate-distortion theory is that vector quantization gives better performance than the scalar quantization. Generally, vector quantization is more complex than scalar quantization. Taking the complexity of vector quantization into account, we focus our attention on scalar quantization.

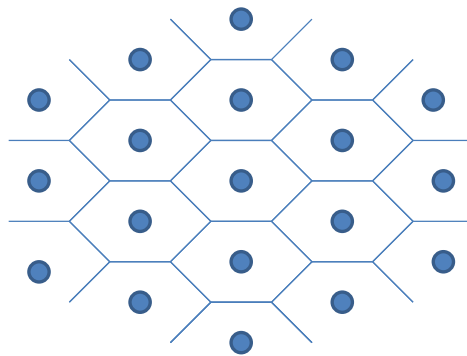


Figure 2.5: Quantization regions in vector quantization

The quantization regions of a vector quantizer may have any shape. Vector quantizer

with hexagonal quantization regions is shown in Fig. 2.5

2.3.2 Uniform vs. non-Uniform Quantizers

Quantizers may also be classified according to the spacing between the quantization levels. With a *uniform* quantizer, the quantization levels are equally spaced. *Quantization spacing* δ is defined as the spacing between the quantization levels. When the input distribution is symmetric, a uniform quantizer can be solely defined by the number of quantization bits and the quantization spacing.

With a *nonuniform quantizer*, the spacing between the quantization levels is arbitrary. An example of 3-bit non-uniform quantizer can have the levels (-8, -5, -4, -1, 1, 4, 5, 8). As seen, the spacing between levels is not constant. For a nonuniform quantizer, it is not sufficient to give the spacing between a single pair of levels. Instead, the entire set of levels must be specified.

Building a uniform quantizer is relatively easy than the non-uniform quantizer. Uniform quantizer requires only one parameter to change. So, optimization with a uniform quantizer is simple.

An example of a uniform quantizer is a quantizer having quantization levels (-6, -2, 2, 6). The quantizer spacing is 4. This quantizer can be defined as a *one-dimensional 2-bit quantizer* with $\delta = 4$.

Given the quantizer spacing and number of quantization levels, the following equation gives the quantization levels.

$$q_i = \left((i - 1) - \left(\frac{L - 1}{2} \right) \right) \delta, \text{ for } i = \{0, \dots, (L - 1)\} \quad (2.15)$$

L is the number of quantization levels. The number of quantization bits are then $\log_2 L$.

2.3.3 Discrete Memoryless Channels

A *discrete memoryless channel (DMC)* is a discrete-input, discrete-output channel. The channel output at a time 't' depends only on the channel input at time 't' and not on previous inputs and output, hence the term 'memoryless'. A digital communication system may be

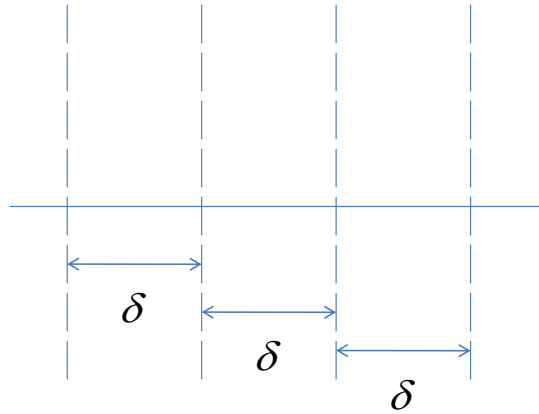


Figure 2.6: Uniform quantizer

modeled as a DMC when the output of the bank of correlators is quantized. In such a case, the DMC has M inputs and L outputs, where L is the number of quantization levels at the output of the bank of correlators. DMC can be completely described by the *transition probabilities*. Transition probability $P(y_i|s_j)$ is the probability of observing output y_i given input s_j .

$$P(Y = y_i|S = s_j) = P(y_i|s_j) \quad (2.16)$$

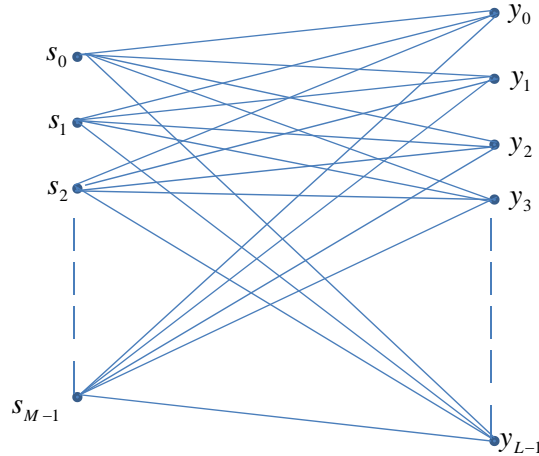
Suppose the channel input is $\mathcal{S} = (s_0, s_1, s_2, \dots, s_{M-1})$ and quantized channel output is $\mathcal{Y} = (y_0, y_1, y_2, \dots, y_{L-1})$, then there exists a total of ' ML ' transition probabilities. The channel output y_i is the quantized output of channel output \mathbf{r} .

Transition Probabilities

We assume that a digitally modulated signal is being transmitted over an AWGN channel and the output vector \mathbf{r} is quantized before estimating the transmitted symbol. For one dimensional(1-D) case: The transition or crossover probability $P(y_i|s_j)$ is given by,

$$P(y_i|s_j) = \int_{b_i}^{b_{i+1}} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y - s_j)^2}{2\sigma^2}\right) dy \quad (2.17)$$

where b_i, b_{i+1} are the boundaries of the quantization region associated with level y_i , σ is the variance of the Gaussian distribution of noise on the channel and s_j is the transmitted signal.

Figure 2.7: Discrete M input, L output channel

Suppose we have ‘ L ’ quantization levels denoted by $Y=(y_0, y_1, \dots, y_{L-1})$. The boundaries will be the midpoints of these quantization levels. The set of boundaries is $\mathcal{B} = -\infty, (y_0 + y_1)/2, \dots, (y_{L-2} + y_{L-1})/2, \infty$. This can be explained as follows: If we need to consider the transition probability of output y_i , we need to consider all the instances when the channel output will be quantized to level y_i . Therefore we use the boundaries b_i and b_{i+1} since all the symbols that fall within these limits will be quantized to y_i .

For two dimensional (2-D) case: Since the noise in one dimension is independent of the noise in the other dimension. The transition or cross over probability $P(y_i|s_j)$ is given by,

$$P(y_i|s_j) = \left(\int_{b_{ia}}^{b_{(i+1)a}} \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-(y_a - s_{j_a})^2}{2\sigma^2} dy_a \right) \left(\int_{b_{ic}}^{b_{(i+1)c}} \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-(y_c - x_{j_c})^2}{2\sigma^2} dy_c \right) \quad (2.18)$$

where

b_i, b_{i+1} are two dimensional points.

$b_{ia}, b_{(i+1)a}$ represents the coordinates of b_i, b_{i+1} on x-axis

$b_{ic}, b_{(i+1)c}$ represents the coordinates of b_i, b_{i+1} on y-axis

Similarly, y_a, x_{j_a} and y_c, y_{j_c} represents the projection of the points on x and y axis respectively.

The calculation of these crossover probabilities becomes easy when Q-functions are used.

Q-function can be defined as,

$$Q(x) = \frac{1}{2\pi} \int_x^\infty \exp\left(\frac{-u^2}{2}\right) du \quad (2.19)$$

Using (2.19) in (2.17),

$$P(y_i|s_j) = Q\left(\frac{b_i - s_j}{\sigma}\right) - Q\left(\frac{b_{i+1} - s_j}{\sigma}\right) \quad (2.20)$$

Also, using (2.19) in (2.18),

$$P(y_i|s_j) = \left(Q\left(\frac{b_{i_a} - s_j}{\sigma}\right) - Q\left(\frac{b_{(i+1)_a} - s_j}{\sigma}\right) \right) \left(Q\left(\frac{b_{i_c} - s_j}{\sigma}\right) - Q\left(\frac{b_{(i+1)_c} - s_j}{\sigma}\right) \right) \quad (2.21)$$

Finding cross over probabilities reduces to just computing the Q-function values. Matlab has an predefined Q-function which makes it much more easier. These crossover probabilities are crucial while finding capacity which will be discussed in chapter 3.

2.4 Summary

An overview of the general communication model is presented. The operations of a modulator and demodulator are explained by transforming the signals to symbols using the concept of basis functions. The concept of quantization is discussed, along with the different ways to classify quantizers. The complexity factor made us choose the scalar quantization. The uniform quantizer is more practical and preferable than the non-uniform quantizer. Thus we have chosen to focus on a scalar uniform quantizer. 16-QAM is very briefly introduced. Channel is analyzed by transforming it into a DMC. The concept of transition probability, important for mutual information calculation, is explained in detail.

Chapter 3

Information Rate

The chapter discusses how to compute the capacity of a communication channel with a modulation-constrained input. Both continuous outputs and quantized outputs are considered. Section 3.1 talks about mutual information and its relation to capacity. The term *input symbol distribution* is introduced in this section. Section 3.2 discusses the mutual information calculation with quantized output. The section also explains the usage of crossover probabilities, discussed in chapter 2, in the computation of mutual information. Section 3.3 deals with the capacity calculation with continuous output, using a numerical method called *Gauss Hermite Quadratures*. Section 3.4 presents the information results under continuous and quantized output with uniform input symbol distribution. Section 3.5 gives a summary of the chapter.

3.1 Capacity and Mutual Information

The *mutual information* between two random variables X and Y is the mutual dependence of one random variable on the other. High mutual information suggests that we can, with high certainty, predict Y with knowledge of X . Low mutual information indicates that there is less information shared between the random variables and therefore having knowledge of one random variable helps us guess the other random variable with very low certainty. If the mutual information between X and Y is zero, it implies that the two random variables are independent. In digital communications, mutual information can be defined as the certainty

of predicting the input symbol given the output symbol and the input symbol distribution.

Mathematically, mutual information between channel input X and channel output Y , is defined as,

$$I(X; Y) = E \left[\log \left(\frac{p(Y|X)}{p(Y)} \right) \right] \quad (3.1)$$

The numerical value of (3.1) depends on the type of channel, the type of receiver and the distribution of the input symbols. The *input distribution*¹, denoted by $p(x)$, is the probability distribution of the input symbols. We have discrete input symbols and each input symbol occurs with some probability depending on the encoders used before the modulator. The input distribution is the probability mass function (PMF) of these input symbols.

Shannon defined capacity in [1] as the amount of information per unit time that can be transmitted over the channel with low error probability. Given the channel and the receiver, capacity is defined as

$$C = \max_{p(x)} I(X; Y) \quad (3.2)$$

Capacity is found by maximizing (3.1) over all possible input distributions. Therefore for estimating capacity we need to have:

- A tool for computing the mutual information given the type of channel, receiver and input distribution.
- A methodology for determining the input distribution that maximizes the mutual information.

3.2 Mutual Information Evaluation for Quantized Output

For the DMC and one-dimensional modulation, (3.1) can be rewritten as,

$$I(X; Y) = \sum_{j=0}^{M-1} \sum_{i=0}^{L-1} p(x_j) p(y_i|x_j) \log_2 \left(\frac{p(y_i|x_j)}{p(y_i)} \right) \quad (3.3)$$

where,

¹Input distribution, also termed as input symbol distribution, is the PMF of the input symbols.

- ‘M’ is the number of input symbols.
- ‘L’ is the number of output symbols.
- $p(x_j)$ is the probability of selecting input symbol x_j by the modulator. This is given by the input distribution.
- $p(y_i|x_j)$ is the transition probability of observing output y_i when input x_j is transmitted. The calculation of the transition probability is discussed in the previous chapter.
- $p(y_i)$ is the probability of observing output y_i . For finding $p(y_i)$, we use,

$$p(y_i) = \sum_{j=0}^{M-1} p(y_i|x_j)p(x_j). \quad (3.4)$$

For N -dimensional modulation, x_j and y_i would be vectors \mathbf{x}_j and \mathbf{y}_i of length N respectively. The value of ‘L’ in (3.3) is the number of quantization levels. ‘L’ depends on the bits of quantization considered. Implementing these equations will give the mutual information between the input and output of the DMC for a given input distribution. The next step is to find the input distribution that will maximize the mutual information. An approach for optimizing the input distribution will be discussed in chapter 5.

3.3 Mutual Information Evaluation under Continuous Output

The capacity with a modulation-constrained channel input and continuous channel output is required to validate our quantized output results. Due to the data processing theorem, a channel with a quantized output should never perform better than with a continuous output.

From (3.1) we have, the mutual information between output Y and input set X is

$$I(Y; X) = \sum_{j=0}^{M-1} p(\mathbf{x}_j) \int_{R^D} \log_2 \left(\frac{p(\mathbf{y}|\mathbf{x}_j)}{p(\mathbf{y})} \right) d\mathbf{y} \quad (3.5)$$

For solving the complex integration in (3.5), mathematical techniques discussed in [8] are used. In particular, Gauss-Hermite quadratures help us in solving (3.5) as follows:

$$I(Y; X) = \sum_{j=0}^{M-1} (p(\mathbf{x}_j))(H_j) \quad (3.6)$$

where R^D is the D-dimensional real space.

H_j in (3.6) is defined by (3.7).

$$H_j = \frac{1}{\pi^{\frac{D}{2}} \ln(2)} \int_{R^D} \exp(-\|\mathbf{y}\|^2) g_j(\mathbf{y}) d\mathbf{y} \quad (3.7)$$

g_j can be found by

$$g_j(y) = -\log\left(\sum_{n=0}^{M-1} \exp(-\beta_{j,n}(\mathbf{y}))\right) \quad (3.8)$$

and

$$\beta_{j,n}(\mathbf{y}) = -\log(p(\mathbf{x}_j)) + \sqrt{\frac{E_s}{N_0}} (\mathbf{x}_j - \mathbf{x}_n)^t (2\mathbf{y} + \sqrt{\frac{E_s}{N_0}} (\mathbf{x}_j - \mathbf{x}_n)) \quad (3.9)$$

where

- $\frac{E_s}{N_0}$ is the signal to noise ratio (SNR).
- $\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_M$ are the input symbols.
- \mathbf{y} is an instance of the output random variable Y .
- $p(\mathbf{x}_j)$ is the probability of selecting signal \mathbf{x}_j .

Some definitions used for solving (3.8) and (3.9) are,

$$\min^*(x_1, x_2, \dots, x_M) = -\log(\exp(-x_1) + \exp(-x_2) + \dots + \exp(-x_M)) \quad (3.10)$$

$$\min^*(x, y, z) = \min^*(\min^*(x, y), z) \quad (3.11)$$

Using (3.10) in (3.8) gives,

$$g_m(y) = \min_n^* \{\beta_{m,n}(y)\} \quad (3.12)$$

To evaluate (3.6), we use a numerical algorithm called *Gauss-Hermite Quadratures* defined as follows:

For 1-D integrals, $D=1$,

$$\int_{-\infty}^{\infty} \exp(-|z|^2) f(z) dz \approx \sum_{i=1}^I w_i f(z_i) \quad (3.13)$$

where ‘I’ is the number of sample points to use for approximating the integral on the left side of (3.13). The higher the value of ‘I’, the better the approximation.

z_i are the roots of the Hermite polynomial which is given by,

$$H_I(x) = (-1)^I \exp\left(\frac{x^2}{2}\right) \frac{d^I}{dx^I} \left[\exp\left(-\frac{x^2}{2}\right) \right] \quad (3.14)$$

w_i can be found using,

$$w_i = \frac{2^{I-1} I! \sqrt{\pi}}{I^2 [H_{I-1}(z_i)]^2} \quad (3.15)$$

For multidimensional integrals, $D>1$,

$$\int_{R^D} \exp(-|\mathbf{z}|^2) f(\mathbf{z}) d\mathbf{z} \approx \sum_{i_1, i_2, \dots, i_D}^I w_{i_1} w_{i_2} \dots w_{i_D} f(z_{i_1}, z_{i_2}, \dots, z_{i_D}) \quad (3.16)$$

where each of the indices i_1, i_2, \dots, i_D are made to vary from 1 to I.

An algorithmic implementation of the above approach is provided in the Appendix. The algorithm is suitable for implementation in Matlab.

3.4 Information Rate Results

Fig. 3.1 shows the variation of information rate with SNR for 16-QAM modulation under uniform input distribution and continuous output, i.e each symbol occurs with a probability of $(1/M)$ and under infinite output quantization levels. This curve represents the upper limit of the information that can be achieved at each SNR point. Information rate is calculated using (3.5). The results generated were verified with the curves in [9] and [8].

Fig. 3.2 shows the plots using 2 bits of quantization per dimension for 16-QAM modulation. The information rate values are obtained by solving (3.3). The figure shows the significance of having optimum spaced quantizer. The non-optimum quantizer curve is drawn

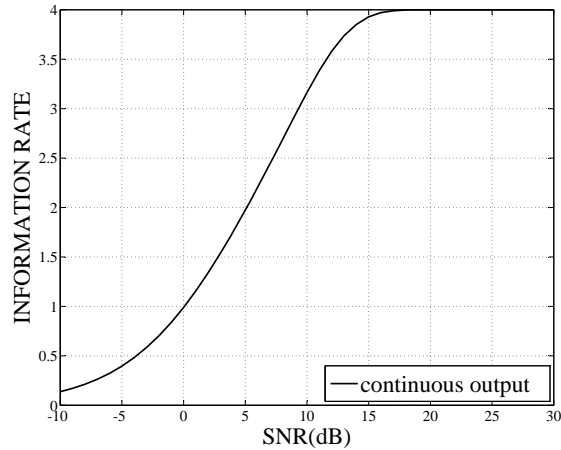


Figure 3.1: Information rate variation with SNR for 16-QAM under continuous output with uniformly-distributed inputs.

considering a quantizer spacing of ‘2’ at all SNR points. Observe that at all SNR points the information rate achieved using an optimum quantizer is higher than the information rate achieved by a non-optimum spaced quantizer. Therefore, it is completely worth investing time and effort in choosing the optimum spacing between the quantization levels. The strategy used for selecting the optimum spacing is discussed in chapter 5.

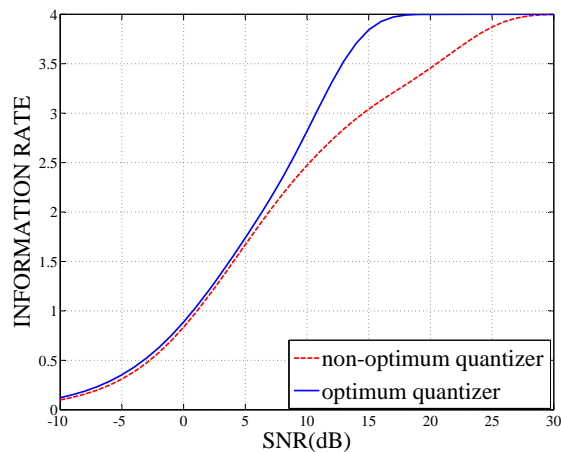


Figure 3.2: Variation of information rate with SNR for 16-QAM under optimum spaced quantizer and non-optimum spaced quantizer with uniformly-distributed inputs

3.5 Summary

This chapter gives an introduction to the concept of mutual information. It is then shown that capacity is the mutual information optimized over the possible input distributions. Methodologies for computing mutual information are given for both the quantized-output and continuous-output channels. For the quantized-output channel, the mutual information is evaluated using a summation that involves the crossover probabilities of the corresponding DMC. For the continuous-output channel, the mutual information must be evaluated using an integral. However, the integral lends itself to efficient numerical computation through the use of Gauss Hermite quadratures.

Chapter 4

Constellation Shaping

The term ‘constellation’ is defined in chapter 2. This chapter deals with the constellation normalization and constellation shaping. Section 4.1 talks about constellation normalization. Section 4.2 discusses how the encoder can shape the constellation. Section 4.3 explains the parameters involved in controlling the symbol distribution. Section 4.4 summarizes the chapter. All the concepts are discussed taking 16-QAM constellation as an example.

4.1 Constellation Normalization

Constellation normalization means modifying the constellation to maintain unit average energy. Normalizing the constellation ensures that the comparison of the system performance with different modulation schemes is fair. The goal is to maintain the average of the constellation, i.e E_s in (4.1), at unity. Let ‘M’ represent the number of input symbols. The average energy of the constellation is,

$$E_s = \sum_{i=0}^{M-1} p_i \mathcal{E}_i \quad (4.1)$$

where

- \mathcal{E}_i , the energy of a symbol ‘ s_i ’, is simply the square of distance of the point ‘ s_i ’ from origin.
- p_i is the probability of choosing symbol s_i .

Normalization is explained in detail taking 16-QAM as an example. For 16-QAM, we consider the symbols to be the points of intersection of lines:

$x=-3, x=-1, x=1, x=3$ and

$y=-3, y=-1, y=1, y=3$.

Therefore we have 16 intersection points. The constellation is shown in Fig. 4.1

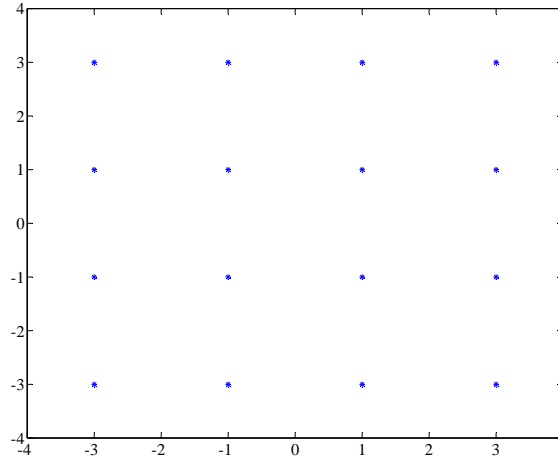


Figure 4.1: 16-QAM constellation diagram

In order to make the constellation satisfy the unit average energy constraint,

- find average energy of existing constellation.
- divide each symbol by the square root of the average energy calculated in above step.

From the constellation in Fig. 4.1, it is clear that there are four symbols with energy ‘2’, eight symbols with energy ‘10’ and four symbols with energy ‘18’. Assuming uniform input distribution, i.e probability of every symbol is $(\frac{1}{16})$, the average energy E_s is,

$$E_s = \frac{1}{16}((4 * 2) + (8 * 10) + (4 * 18)) = \frac{1}{16}(160) = 10 \quad (4.2)$$

Therefore to normalize the 16-QAM constellation, we have to divide the symbols by $\sqrt{10}$. The constellation points will now be the points of intersection of lines:

$x = \frac{-3}{\sqrt{10}}, x = \frac{-1}{\sqrt{10}}, x = \frac{1}{\sqrt{10}}, x = \frac{3}{\sqrt{10}}$ and

$y = \frac{-3}{\sqrt{10}}, y = \frac{-1}{\sqrt{10}}, y = \frac{1}{\sqrt{10}}, y = \frac{3}{\sqrt{10}}$.

Thus the constellation now changes as shown in Fig. 4.2. Observe that normalizing the constellation changes the location of constellation points.

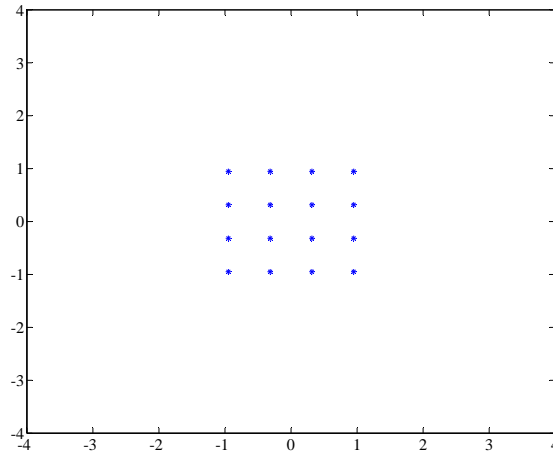


Figure 4.2: Normalized 16-QAM Constellation Diagram

4.2 Constellation Shaping

Our strategy for constellation shaping is from [10] and [11]. *Constellation shaping* means making the modulator to pick certain symbols more frequently than the other. In most modulations, there exists a particular non-uniform input PMF which performs better than the uniform PMF. The goal of constellation shaping is to find the PMF of the symbols that results in improved performance.

Desired constellation shaping can be obtained by careful design of the encoders employed before the modulator. We always want to have our constellation shaped such that the symbols closer to the origin are picked more frequently than the symbols far away, the reason being the symbols near the origin need less energy for transmission. This reduces the average energy per bit required.

Note: M -ary modulator maps a group $\log_2 M$ bits, to one of the M available symbols. Let these $\log_2 M$ bits be called a *bitword*.

The shaping encoder that we use should transmit more 0's than 1's making the 'probability of zero' higher than the 'probability of one'. This ' $p(0) > p(1)$ ' condition at the output of the encoder is necessary because at the modulator the less energy signals are mapped to a bitword that has more '0's at the higher bit positions. Also, bitword corresponding to higher energy signals has more '1's in the higher bit positions.

All these concepts are better understood by an example. Consider 16-QAM, the modulator takes a bitword, having four bits, from the encoder and maps the bitword to a symbol in the constellation. As said above, the modulator maps the least energy symbols, i.e. $(\frac{\pm 1}{\sqrt{5}}, \frac{\pm 1}{\sqrt{5}})$, to bitwords that has zeros in higher bit positions. The location of higher bit positions can be understood by Fig. 4.3

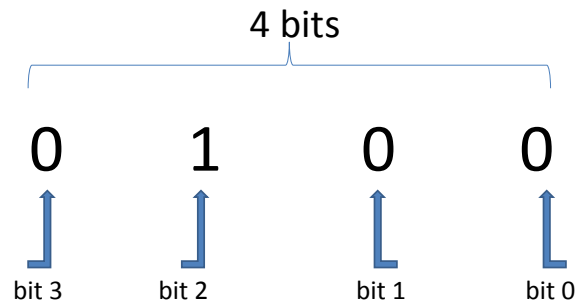


Figure 4.3: Bit positions

‘bit n ’ represents the n^{th} bit position. Therefore ‘bit 3 and bit 2’ are considered higher bit positions in this example. Less energy signals mapped to bitwords having more zeros in the higher bit positions and higher energy signals mapped to bitwords having more ones in the higher bit positions is illustrated in the Fig. 4.4.

From Fig. 4.4,

- four symbols represented by circles are closest to origin and have the least energy.
- eight symbols represented by squares are at an intermediate distance from origin and have the intermediate energy.
- four symbols represented by triangles are farthest to origin and have the highest energy.

Also, observe the bits in the higher bit positions (bit positions 2 and 3).

- bit 2 and bit 3 positions of the four bits associated with the least energy symbols is ‘00’
- bit 2 and bit 3 positions of the four bits corresponding to intermediate energy symbols are ‘10’ and ‘01’

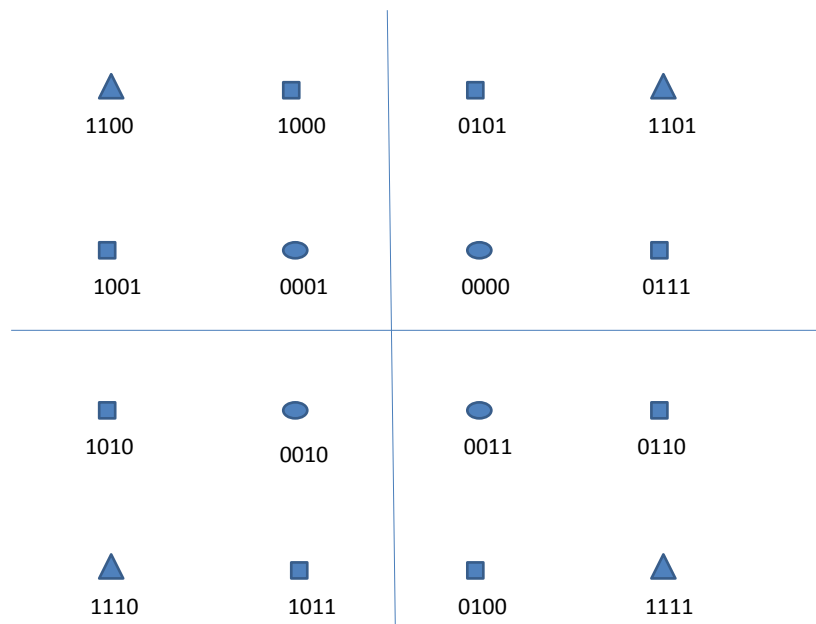


Figure 4.4: 16 QAM Constellation representation

- bit 2 and bit 3 positions of the four bits related to highest energy symbols is ‘11’

Everything should be clear by now that having 0’s at bit positions 2 and 3 of the bitword is preferred so that less energy is needed for transmitting the binary sequence. We take help of the shaping encoder to increase the probability of having 0’s at the higher bit positions. There are many shaping encoding schemes available.

Consider a (3,5) shaping code shown in table 4.1. The left column of the table has 3-bits, which is the input to the shaping encoder. The right side of the table contains the corresponding output of the shaping encoder. For example, if the encoder is given an input ‘000’, it gives out ‘00000’. Also, from the table it can be observed that, probability of ones and zeros at the output of the shaping encoder input are equal, whereas the output of the encoder has probability of zero ($\frac{31}{40}$) and probability of one ($\frac{9}{40}$). Therefore the chances of observing zero at the encoder output is very much higher than observing a one and thus the purpose of the shaping encoder is served. Fig. 4.5 clearly shows how an encoder is employed practically.

Table 4.1: (3,5) shaping encoder

3 input bits			5 output bits				
0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1
0	1	0	0	0	0	1	0
0	1	1	0	0	1	0	0
1	0	0	0	1	0	0	0
1	0	1	1	0	0	0	0
1	1	0	0	0	0	1	1
1	1	1	0	0	1	0	1

Fig. 4.5 shows the shaping operation in detail. ‘ENC’ represents the shaping encoder. The shaping encoder can be accommodated within the channel encoder or the modulator. The shaping operation is shown as separate blocks for better understanding. The length of bitword for 16 QAM is 4. Therefore, we need four streams of bits each giving out one bit at a time. Observe that the operation of the splitter is to split the incoming bits into 4 streams. The output of these streams at any instance ‘t’ will become a bitword. As said, we prefer having a bitword with more number of zeros in the 2^{nd} and 3^{rd} bit positions as this will enable us to transmit lower energy symbols. This is the reason behind choosing shaping encoder to be placed at the streams that gives out bits at the higher positions. Placing a shaping encoder at stream i , implies we are increasing the probability of the stream i to output bit ‘0’. The shaping encoder encodes data as per Table 4.1

The bitwords in the Fig. (4.5) are 0111, 1010, 0000, 0011, 0010. This means we have to transmit three least energy signals and two intermediate energy signals. Thus shaping helps us to choose the low energy signals more frequently than the high energy signals. We can achieve the required symbol distribution by using proper shaping encoding scheme.

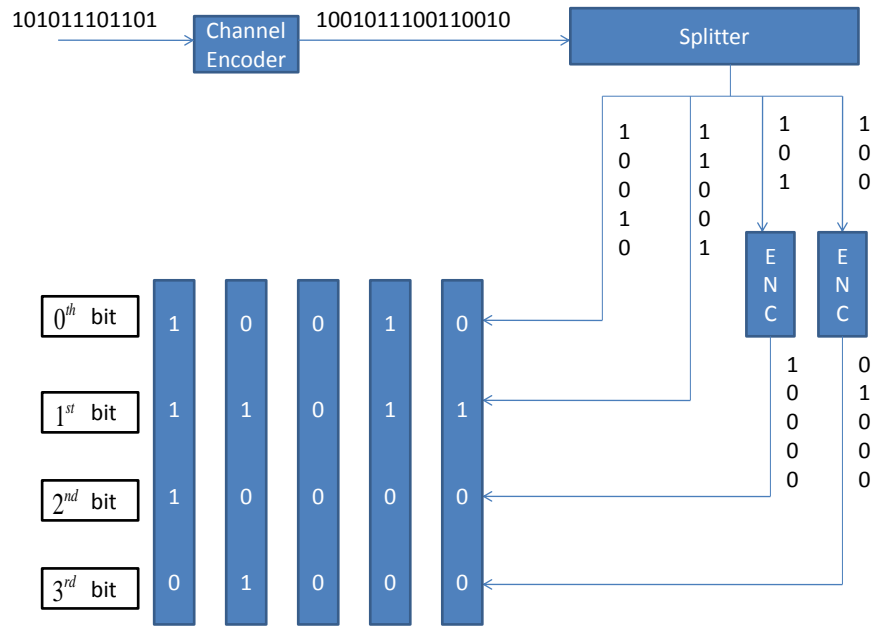


Figure 4.5: Shaping operation

4.3 Controlling the constellation distribution and symbol locations

Once we know the symbol distribution we want, we can then use the shaping code that will achieve that distribution. Choosing a different symbol distribution can be made easy by making the distribution to depend on a single parameter. The parameter p_0 , the probability of 0 at the output of the encoder, will describe the constellation distribution. Fig. 4.6 shows the probability distribution of symbols.

p_0 , p_1 represent the probability of observing zero and one respectively from the shaping encoder. This implies

$$p_0 + p_1 = 1 \quad (4.3)$$

$$(p_1 + p_0)^2 = 1 \quad (4.4)$$

From (4.3), $p_1 = 1 - p_0$. From (4.4),

$$p_0^2 + 2p_1p_0 + p_1^2 = 1 \quad (4.5)$$

Sum of the total probabilities of the constellation should be 1. Each energy level has a probability associated with it. In 16-QAM, we have symbols with three different energy levels. Therefore we need three probability values to represent each of these symbol energies. From (4.5), each of the terms on the right hand side can represent these three probabilities. Let p_0^2 represent the probability of selecting least energy symbols. We have four signals with least energy. Therefore, probability of each symbol will be $\left(\frac{p_0^2}{4}\right)$. $2p_1p_0$ represent the probability of intermediate energy symbols. We have eight symbols with intermediate energies. Therefore probability of each symbol will be $\left(\frac{p_1p_0}{4}\right)$. p_1^2 represents the probability of selecting highest energy symbols. We have four signals with highest energy. Thus each symbol is selected with a probability $\left(\frac{p_1^2}{4}\right)$. All these are represented in Fig. 4.6

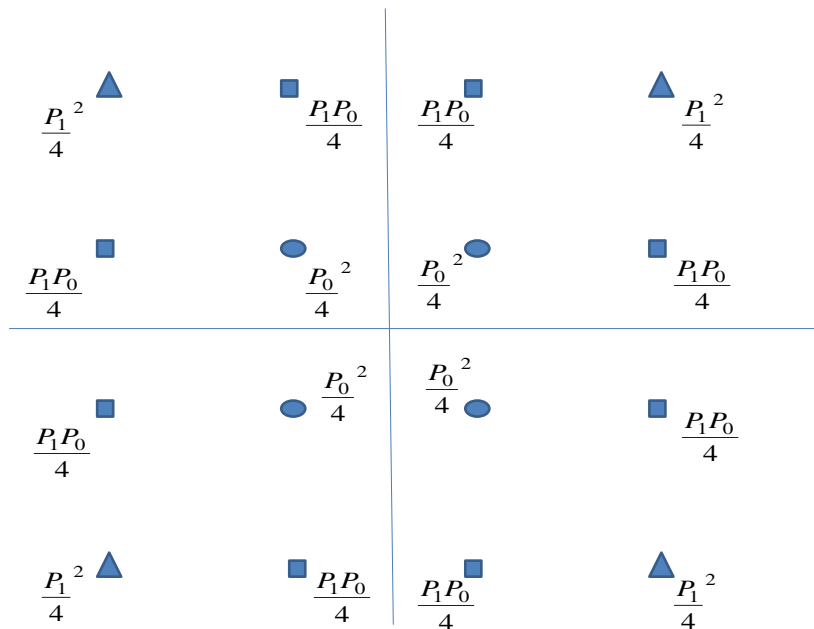


Figure 4.6: Constellation distribution of 16-QAM

Looking at the probabilities, the only parameter that can control the entire distribution is p_0 . As the total energy of the constellation depends on the symbol distribution, varying p_0 will change the total energy.

Let us consider the effect of p_0 in normalizing the constellation shown in table 4.2

Case 1: We choose $p_0 = 0.9$. Then the symbol distribution will be [0.0025 0.0225 0.0225

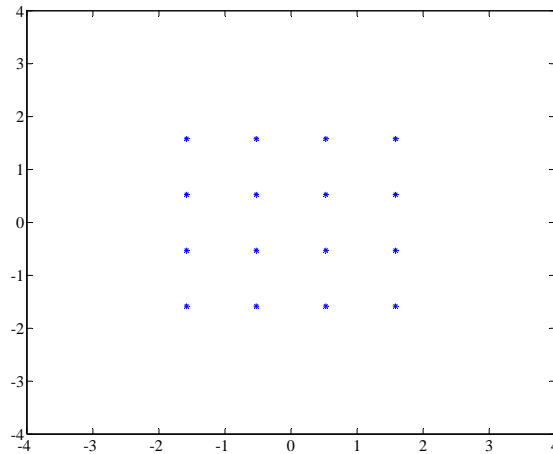
Table 4.2: 16-QAM constellation points

(-3 -3)	(-1 -3)	(1 -3)	(3 -3)
(-3 -1)	(-1 -1)	(1 -1)	(3 -1)
(-3 1)	(-1 1)	(1 1)	(3 1)
(-3 3)	(-1 3)	(1 3)	(3 3)

0.0025 0.0225 0.2025 0.2025 0.0225 0.0225 0.2025 0.2025 0.0225 0.0025 0.0225 0.0225 0.0025]. Total energy of the constellation will then be 3.6units. For normalizing the signal set, each symbol should be divided by $\sqrt{3.6}$. The constellation will then change according to table 4.3. The constellation diagram is shown in Fig. 4.7.

Table 4.3: Constellation when $p_0 = 0.9$

(-1.5811 -1.5811)	(-0.5270 -1.5811)	(0.5270 -1.5811)	(1.5811 -1.5811)
(-1.5811 -0.5270)	(-0.5270 -0.5270)	(0.5270 -0.5270)	(1.5811 -0.5270)
(-1.5811 0.5270)	(-0.5270 0.5270)	(0.5270 0.5270)	(1.5811 0.5270)
(-1.5811 1.5811)	(-0.5270 1.5811)	(0.5270 1.5811)	(1.5811 1.5811)

Figure 4.7: Constellation diagram of 16-QAM when $p_0=0.9$

Case 2: Suppose we choose $p_0 = 0.1$. Then probability distribution will be [0.2025 0.0225 0.0225 0.2025 0.0225 0.0025 0.0025 0.0225 0.0225 0.0025 0.0025 0.0225 0.2025 0.0225 0.0225 0.2025]. Total energy of the constellation will then be 16.4units. For normalizing the

signal set, each symbol should be divided by $\sqrt{16.4}$. The constellation will then change as shown in table 4.4 and corresponding constellation diagram is shown in Fig. 4.8.

Table 4.4: Constellation when $p_0 = 0.1$

(-0.7408 -0.7408)	(-0.2469 -0.7408)	(0.2469 -0.7408)	(0.7408 -0.7408)
(-0.7408 -0.2469)	(-0.2469 -0.2469)	(0.2469 -0.2469)	(0.7408 -0.2469)
(-0.7408 0.2469)	(-0.2469 0.2469)	(0.2469 0.2469)	(0.7408 0.2469)
(-0.7408 0.7408)	(-0.2469 0.7408)	(0.2469 0.7408)	(0.7408 0.7408)

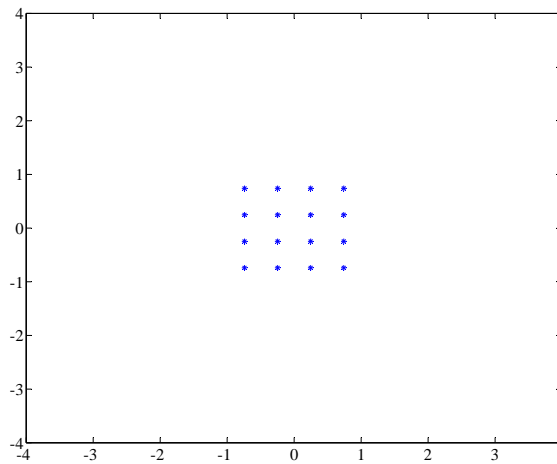


Figure 4.8: Constellation distribution of 16-QAM when $p_0=0.1$

Comparing figures 4.7 and 4.8, we can say that smaller p_0 brings the constellation closer to origin and larger p_0 makes the constellation to move farther from origin. Thus p_0 can alter the constellation.

4.4 Summary

This chapter explains normalizing a given constellation with an example. The role played by the shaping encoder in shaping the constellation is discussed. Further, the influence of the symbol distribution on normalizing the constellation is examined. The performance of the system with constellation shaping will be dealt in the next chapter.

Chapter 5

Optimization Results

This chapter discusses the results achieved by this work and the approach used to generate the results. Section 5.1 talks about the approach used to find optimum symbol PMF. Section 5.2 deals with the technique employed to find optimum spacing between the quantization levels given the constellation PMF. Section 5.3 explains the concept of joint optimization. The optimization is with respect to symbol PMF and the output quantizer. Section 5.4 presents the results. Section 5.5 summarizes the chapter.

5.1 Finding Optimum PDF

We now focus on finding the optimum distribution given the L and the SNR value. Chapter 4 explains that p_0 is the only parameter controlling the symbol distribution of the constellation. When optimising over two parameters, which in this case are δ and p_0 , we do not want to use searching strategies on both the parameters. We believe by doing this we may lose some combinations of (p_0, δ) which can actually be the optimum combination. Keeping this in mind, we want to use brute force technique on one parameter and use some searching strategy on the other parameter. Since p_0 can never be >1 , we have strict upper and lower bounds for p_0 . Also, there is no strict upper bound for δ . This encouraged use a brute force technique for finding the optimum distribution and use a search technique for finding optimum δ . We want p_0 to be at least 0.5. p_0 is varied in the range [0.5 0.99] with an increment of 0.01. The mutual information is calculated for all distributions and the

distribution that gives the highest mutual information is said to be the optimum distribution.

5.2 Finding the optimal quantizer spacing

As discussed in chapter 2, a uniform quantizer is utilized at the receiver section. The goal now is to find the optimal spacing between the quantization levels, denoted by δ . δ can be any value in the range $(0, \infty)$. A smoothly varying pattern was observed when trying to study the effect of different δ values on the mutual information, i.e mutual information increases as δ approaches the optimum δ and smoothly goes down as δ moves far away from optimum δ . Fig. 5.1 shows the variation of information rate with the quantizer spacing. The figure corresponds to 16-QAM modulation scheme under signal to noise ratio(SNR)=2dB with 4 quantization levels in each dimension.

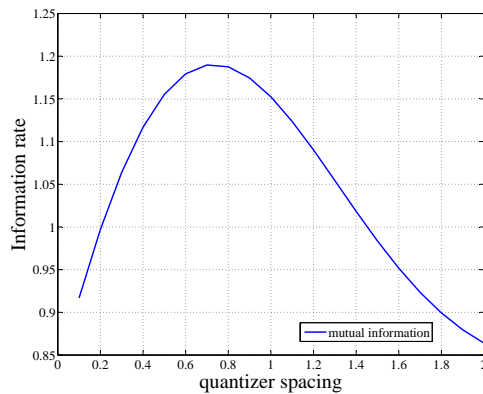


Figure 5.1: Variation of information rate with quantizer spacing of 16-QAM at SNR = 2dB with 2 quantization bits/dimension

We exploit the pattern shown in Fig. 5.1 by the technique ‘reduction in the search space’, i.e initially we search within a set of values which are widely spaced. Say we got a good system performance at $\delta=k$. This gives us an approximate idea of where optimum δ lies. Then we narrow down the search by looking around value k , i.e we now consider a set of values around k with smaller spacing. This way we keep narrowing down the search space, and search a small area with high resolution. We continue this process until we get a value up to the resolution we want or until the value converges. While implementing this technique,

the initial range we consider is $[1, 10]$ with a spacing of 'one'. We can pretty confidently say that optimum δ value will be within 5 because always a normalized signal set is considered while evaluating the system. Therefore the signal points will not be too far which means optimum spacing of the uniform quantizer will not be too large.

In our work we only considered δ till the fourth decimal. If optimum δ for a particular scenario is 3.7560, let us see how this algorithm lands to this value.

- range $[1, 10]$ with spacing '1' is searched. Then we find the information rate values, given the value L and signal constellation, for each value in the set. Optimum value of δ will now be around the value that gives the highest information rate. Value '3' in the set will give highest information rate.
- range $[2, 4]$ with spacing '0.1' is searched. Observe that we increased the resolution of our search. The value in this set that gives highest information rate is found. The value will be 3.7.
- range $[3.6, 3.8]$ with spacing '0.01' is searched. Observe that we further increased the resolution of our search. The value in the set that gives highest information rate is found. The value will now be 3.75.
- range $[3.74, 3.76]$ with spacing '0.001' is searched. Resolution of our search space is now '0.001'. The value in the set that gives highest information rate is found. The value will be 3.756.
- range $[3.755, 3.757]$ with spacing '0.0001' is searched. Resolution of our search space is now '0.0001'. The value in the set that gives highest information rate is found. The value will be 3.7560.
- We stop the search

Therefore, given the number of quantization levels and normalized constellation, this algorithm will find an optimum δ of the quantizer.

5.3 Joint Optimization

Sections 5.1 and 5.2 discussed optimizing the two parameters separately for maximizing the mutual information. One parameter *symbol distribution* is on the transmitter side and the other parameter *quantization spacing* is on the receiver side of the communication system. These two parameters have to be jointly optimized for achieving a better system performance. This calls for a need for joint optimization of both the parameters mentioned above. The approach is as follows:

- Specify the SNR value
- Provide the number of quantization levels
- Vary p_0 from 0.5 to 0.99 with increments of 0.01
- For each value of p_0 , find the optimum δ that maximizes mutual information for that particular distribution. Save the optimum δ and the maximum information rate achieved.
- Now we have two 2-dimensional arrays. *Array1* has p_0 and corresponding maximum mutual information that can be achieved and *Array2* containing p_0 and the corresponding optimum spacing of quantization levels.

From *Array1* pick the distribution that gives highest mutual information. Thus this distribution will be the optimum symbol distribution. From *Array2*, for the optimum distribution found from *Array1*, find the corresponding optimum spacing for the quantization levels. Now we have the optimum symbol distribution and quantizer spacing. We hope that these results when used in practical communication systems, will improve the performance of a communication system.

5.4 Observations

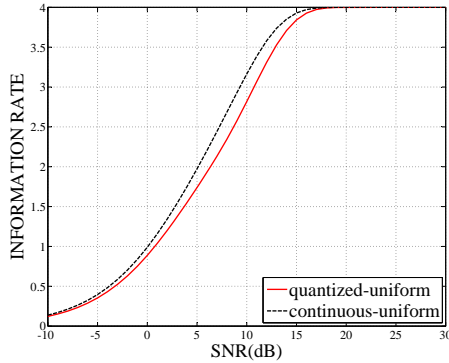
There are many results to be presented. For clarity, we define the following terms:

- Uniform Continuous - This means we are considering uniform symbol distribution and continuous output i.e infinite levels of quantization.
- Uniform Quantized - This means we are considering uniform symbol distribution and quantized output i.e finite levels of quantization with optimum spacing.
- Shaping Continuous - This means we are considering the optimum symbol distribution and continuous output i.e infinite levels of quantization.
- Shaping Quantized - This means we are considering the optimum symbol distribution and quantized output i.e finite levels of quantization with optimum spacing.

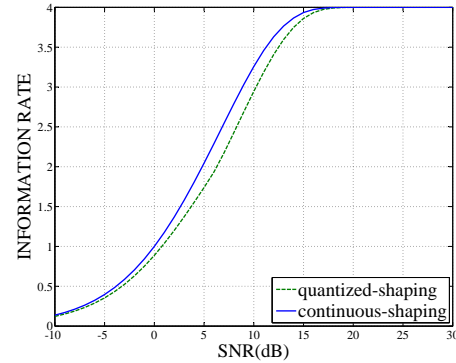
Note: We have uniform symbol distribution when $p_0 = 0.5$. Further, we consider 2,3 and 4 bits of quantization, i.e 4,8 and 16 number of quantization levels in each dimension. For these 3 cases, we present plots for all the four scenarios mentioned above.

5.4.1 For two quantization bits

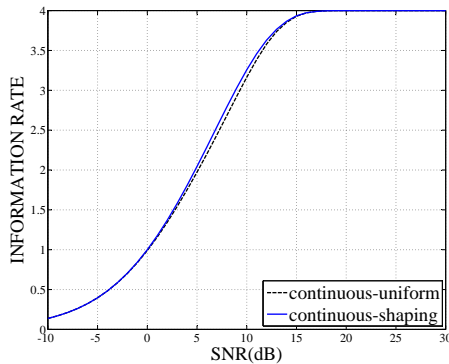
This means we can have 2^2 quantization levels in each dimension. Fig. 5.2 shows the variation of information rate with SNR(dB) for different scenarios.



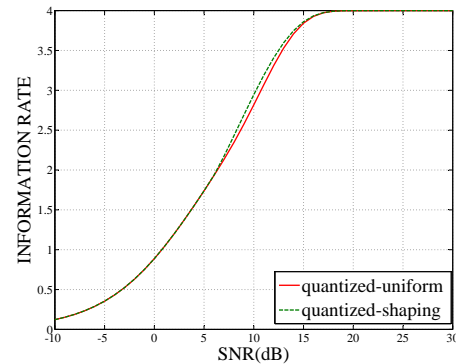
(a) Uniform Case



(b) Shaping case



(c) Continuous case



(d) Quantized case

Figure 5.2: Two bits of quantization

From Figs. 5.2(a) and 5.2(b), the continuous output can give an upper bound on the information rate that can be achieved at all SNRs under the uniform case and shaping case. But practically continuous output or output with infinite quantization levels is not possible. The curves are only to understand how much we lose by quantizing the output. Also from Figs. 5.2(c) and 5.2(d), shaping the input distribution will perform better than uniform input distribution at some SNR points(5dB - 15dB). At certain SNR points, trying to shape the input distribution to get better results is of no use as uniform input distribution itself is

optimal.

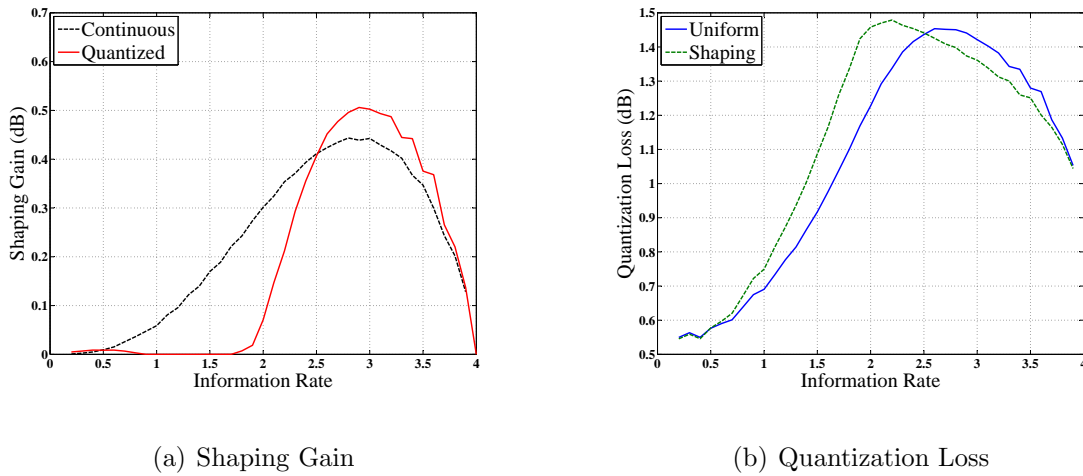


Figure 5.3: Shaping Gain and Quantization Loss curves for two bits of quantization

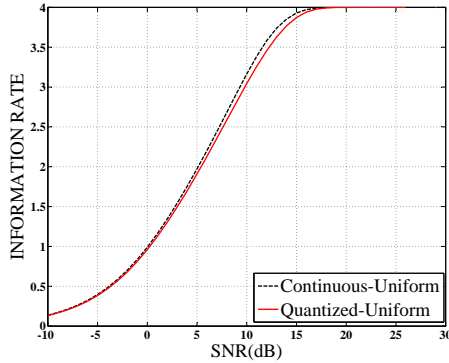
The *Shaping Gain* and *Quantization Loss* curves are shown in Figs. 5.3(a) and 5.3(b). From the Shaping Gain curve in Fig. 5.3(a), shaping is more beneficial under continuous output case as there is considerable gain at most of the SNR points. Under quantized output, shaping makes a difference at capacity values [1.7 4] which is equivalent to SNR range [7 17]. The benefits of using optimum symbol PMF at the input be explained in 2 ways

- If we need SNR snr_1 and snr_2 for uniform and shaped cases respectively for achieving capacity c , then $snr_2 \leq snr_1$. This means we can achieve the same capacity with smaller SNR when shaped input distribution is used.
- If c_1 and c_2 are the capacities achieved under uniform and shaped cases respectively at SNR snr , then $c_1 \leq c_2$. This means shaping gives us a better data rate at a particular SNR.

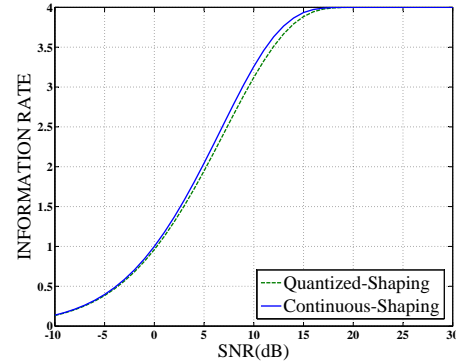
From the Quantization Loss curve in Fig. 5.3(b), it can be observed that irrespective of the SNR point and input distribution, quantization loss is always significant. One interesting point is that before a rate 2.5 bits/channel use, shaping the input results in higher quantization loss than the uniform input distribution and vice versa after an information rate 2.5 bits/channel use. The information rate value of 2.5 is equivalent to SNR (7-8)dB.

5.4.2 For three quantization bits

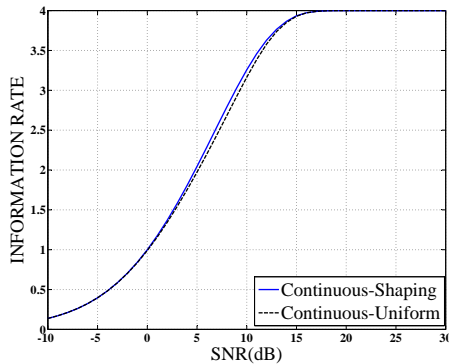
In this case, we can have 2^3 quantization levels in each dimension. Fig. 5.4 shows the variation of information rate with SNR(dB) for different scenarios.



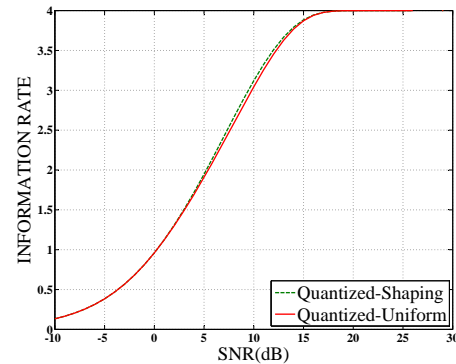
(a) Uniform Case



(b) Shaping case



(c) Continuous case

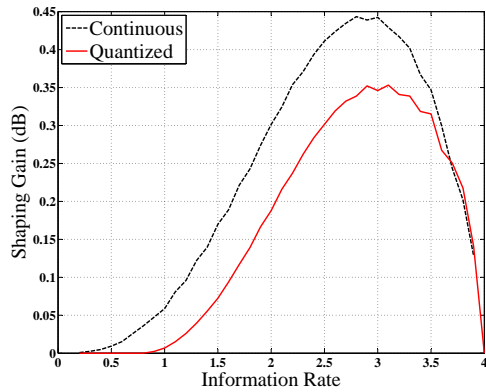


(d) Quantized case

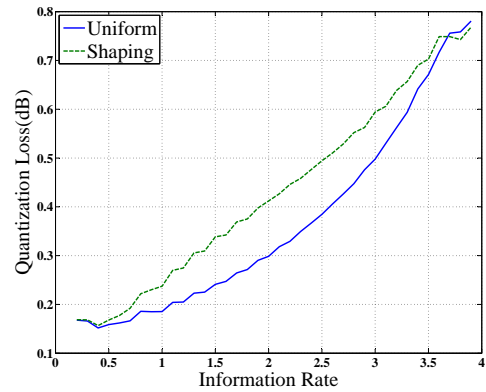
Figure 5.4: Three bits of quantization

Observe Figs. 5.2 and 5.4, the performance of the three bit quantizer is similar to that of a two bit quantizer.

The *Shaping Gain* and *Quantization Loss* curves are shown in subfigures 5.5(a) and 5.5(b) of Fig. 5.5. The Shaping Gain obtained using a three bit quantizer is different from a two bit quantizer. From Figs. 5.3(a) and 5.5(a), it can be observed that with a two bit quantizer shaping under continuous output gives better performance than shaping under quantized output at lower SNR points and vice versa at higher SNR points. But with a three bit



(a) Shaping Gain



(b) Quantization Loss

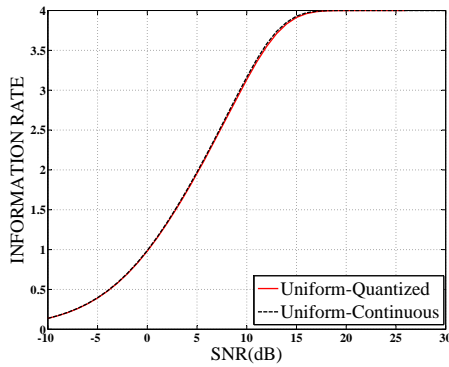
Figure 5.5: Shaping Gain and Quantization Loss curves for three bits of quantization

quantizer, shaping input under continuous output is always better at all SNR points.

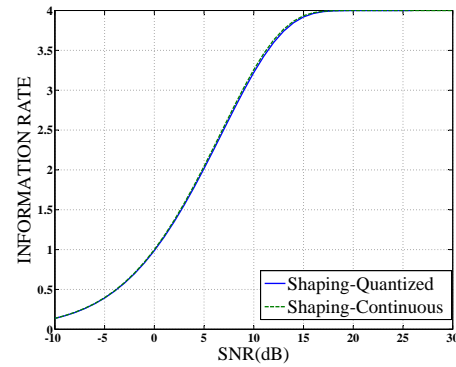
Also, quantization loss curve of a three bit quantizer differs from a two bit quantizer. From Fig. 5.5(b), observe higher quantization losses with shaped input distribution at all SNR points.

5.4.3 For four quantization bits

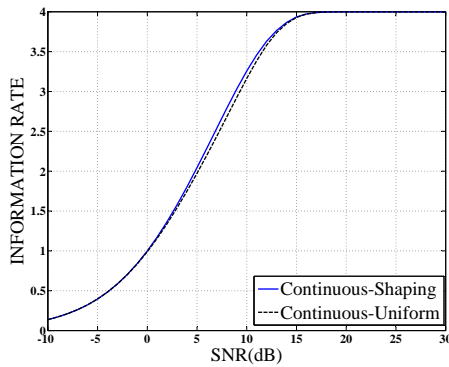
This means we can have 2^4 quantization levels in each dimension. Therefore the total quantization points are 256. The number of quantization levels are very much higher than the number of input symbols. It is expected that the performance with this quantizer should be far better than the two bit quantizer.



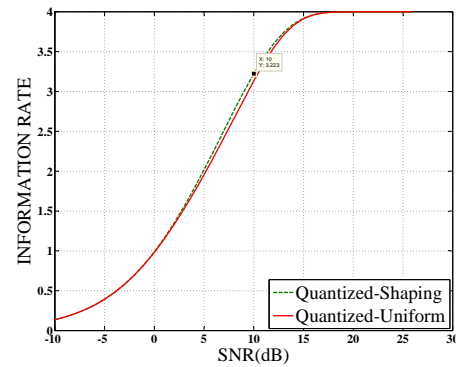
(a) Uniform Case



(b) Shaping case



(c) Continuous case

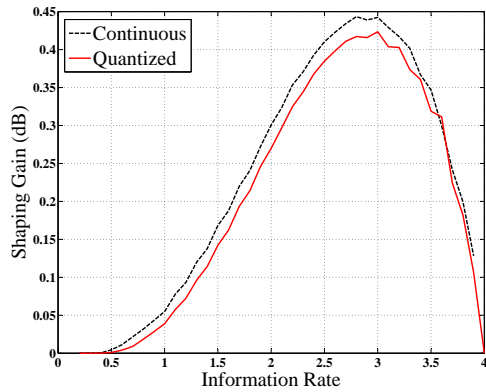


(d) Quantized case

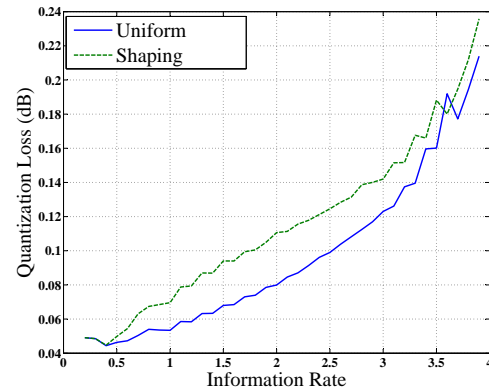
Figure 5.6: Four bits of quantization

Observe Figs. 5.6(a) and 5.6(b), the quantized output information rate is almost approaching the continuous output information rate. From Figs. 5.6(c) and 5.6(d), as expected, shaping the input distribution gives a better information rate.

The *Shaping Gain* and *Quantization Loss* curves are shown in subfigures 5.7(a) and 5.7(b) respectively. In the quantization loss curve, the quantization loss obtained under a four bit



(a) Shaping Gain



(b) Quantization Loss

Figure 5.7: Shaping Gain and Quantization Loss curves for four bits of quantization

quantizer is lower than the two and three bit quantizers. At a data rate of 3, the quantization loss was 0.59 and 0.5 under shaped input and uniform input distributions respectively with a three bit quantizer. The quantization loss is approximately 0.143 and 0.122 under shaped input and uniform input distributions respectively with a four bit quantizer. There is huge reduction in quantization loss when we use one bit more for quantization level representation.

5.4.4 Optimum parameters

Here the curves for optimum p_0 and optimum spacing are presented. Once we fix the number of quantization levels and the SNR, Fig. 5.8 gives us the optimum p_0 value at the transmitter side.

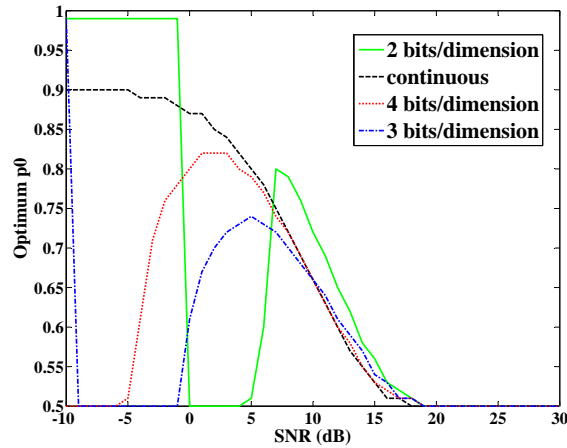


Figure 5.8: Variation of optimum p_0 with SNR for 16-QAM at different number of quantization bits/dimension

Similarly, Fig. 5.9 gives us the optimum spacing, between the quantization levels at the receiver. This plot helps us design the quantizer.

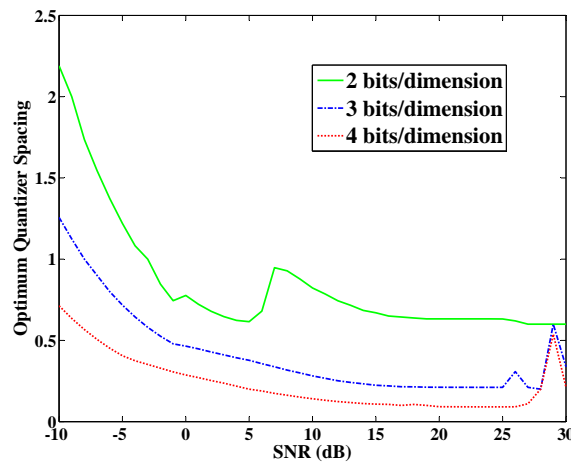


Figure 5.9: Variation of optimum p_0 with SNR for 16-QAM at different number of quantization bits/dimension

Figs. 5.8 and 5.9 together helps us select the PMF of the constellation and quantizer spacing. When these values are applied to the system, information rate is maximized.

5.5 Summary

Optimizing the quantizer spacing δ and input distribution individually for maximizing the information rate is discussed. Joint optimization of δ and p_0 under a given SNR is covered. The results of this joint optimization are presented and some observations made are discussed. The next chapter deals with the conclusions upon analyzing the results.

Chapter 6

Conclusions

All the results shown are for 16-QAM modulation scheme. The loss in information rate due to output quantization is observed. Quantization loss decreases significantly with increase the number of quantization levels.

Table 6.1: Quantization Loss comparison with R

L	QLoss(Shaped)	QLoss(Uniform)
2 bits	$\approx 1.45dB$	$\approx 1.22dB$
3 bits	$\approx 0.41dB$	$\approx 0.29dB$
4 bits	$\approx 0.11dB$	$\approx 0.07dB$

Table 6.1 shows the quantization loss at Information rate=2. At a glance, table 6.1 seems to tell us that shaping the symbols results in higher quantization loss. But, the columns ‘2’ and ‘3’ of each row cannot be compared because:

The value in column ‘2’ is the difference in SNR(dB) values between continuous shaping and quantized shaping curves whereas the value in column ‘3’ is the difference in SNR(dB) values between continuous uniform and quantized uniform curves.

The intent of table 6.1 is to show the reduction in quantization loss with increase in R . The complexity of the receiver increases with L .

It is expected that shaping the input will improve the information rate of the system. But it is noticed that, at certain SNR points, there is no improvement by shaping the input symbols. This means that at some SNR points the uniform distribution is the optimal

distribution. Table 6.2 shows the results

Table 6.2: Statistics at zero shaping gain

L	SNR(dB)	Shaping Gain(dB)
2 bits	$\approx 4.7dB$	0
3 bits	$\approx -1dB$	0
4 bits	$\approx -5dB$	0

At certain SNR points the shaping gain is so small ($\leq 0.1dB$) that putting effort in finding the optimum PMF is not worth. Table 6.3 shows the SNR points at which the shaping gain is insignificant.

Table 6.3: Statistics at shaping gain $\leq 0.1dB$

L	SNR(dB)	Shaping Gain(dB)
2 bits	$\approx \leq 6.5dB$ and $\geq 15dB$	0.1dB
3 bits	$\approx \leq -4dB$ and $\geq 15dB$	0.1dB
4 bits	$\approx \leq -2dB$ and $\geq 15dB$	0.1dB

By examining all the findings, it can be concluded that 2 bits of quantization results in very high quantization losses. Therefore using 3 bits of quantization is appropriate. Best results due to shaping can be observed at SNR points 6dB - 15dB. Therefore when the system is operated at SNR (6dB - 15dB), shaping can significantly improve the information rate.

Constellation shaping is discussed in the previous papers. But studying the constellation shaping in conjunction with quantization is important because almost every practical communication system uses a quantizer. However optimal our input symbol PMF may be, if we do not select an optimal quantizer, we are compromising on the information rate. Therefore it is very important that we use optimum symbol distribution along with optimal quantizer. The results of our work helps in choosing the optimal PMF and quantizer under given SNR and L.

This work can be further extended to a more-complex two-dimensional modulations like 16-APSK and to higher-dimensional modulations. Using a non-uniform vector quantizer can also be investigated.

Appendix A

Appendix

A.1 Algorithm for Information rate calculation for two dimensions

Assumptions:

$E_s|N_o$, the roots and weights of the Hermite polynomial, $w[i]$ and $z[i]$, are known. Signal set is normalized.

IR \leftarrow 0

for m \leftarrow 0 to (M-1)

$J_m \leftarrow$ 0

for $i_0 \leftarrow$ 0 to (I-1)

for $i_1 \leftarrow$ 0 to (I-1)

X \leftarrow ∞

for n \leftarrow 0 to (M-1)

$t_0 \leftarrow \sqrt{(E_s|N_0)}(x_m[0] - x_n[0])$

$t_1 \leftarrow \sqrt{(E_s|N_0)}(x_m[1] - x_n[1])$

$\beta_{m,n} \leftarrow -\ln(p_m) + t_0(t_0 + 2z[i_0]) + t_1(t_1 + 2z[i_1])$

X $\leftarrow \min^*(X, \beta_{m,n})$

end

$J_m = J_m + w[i_0]w[i_1]X$

end

end

$$\text{IR} \leftarrow \text{IR} + J_m * p_m$$

end

$$\text{IR} \leftarrow \text{IR} | (\pi \ln(2))$$

References

- [1] C. E. Shannon, “A mathematical theory of communication,” *Bell Syst. Tech. J.*, vol. 27, pp. 623–656, 1948.
- [2] S. Krone and G. Fettweis, “Achievable rate of single-carrier systems with optimal uniform quantization at the receiver,” Information Theory Workshop (ITW), 2010.
- [3] Y. Wu, L. M. Davis, and R. Calderbank, “On the capacity of the discrete-time channel with uniform output quantization,” Information Theory, ISIT, 2009.
- [4] S. Arimoto, “An algorithm for computing the capacity of arbitrary discrete memoryless channels,” *Information Theory, IEEE Transactions*, vol. 18, pp. 14–20, 1972.
- [5] U. Madhow, *Fundamentals of Digital Communications*. Cambridge University Press, 2008.
- [6] J. Singh, O. Dabeer, and U. Madhow, “Capacity of the discrete-time awgn channel under output quantization,” Information Theory, ISIT, 2008.
- [7] J. G. Proakis, *Digital Communications - fourth edition*. McGraw-Hill, 2001.
- [8] P. E. McIllree, *Channel capacity calculations for M-ary N-dimensional signal sets*. PhD thesis, The university of South Australia school of Electronic Engineering, 1994.
- [9] G. Ungerboeck, “Channel coding with multilevel/phase signals,” *Information Theory, IEEE Transactions*, vol. 28, pp. 55–67, 1982.
- [10] C. C. T. Stéphane Y. Le Goff, Boon Kien Khoo and B. S. Sharif, “Constellation shaping for bandwidth-efficient turbo-coded modulation with iterative receiver,” *Wireless Communications, IEEE Transactions*, vol. 6, 2007.
- [11] B. K. Khoo, S. Y. L. Goff, and B. S. Sharif, “Bit-interleaved coded modulation with iterative decoding using constellation shaping,” vol. 54, pp. 1517–1520, Sept. 2006.