TRELLIS CODE DESIGN FOR CORRELATED FADING AND ACHIEVABLE RATES FOR TOMLINSON–HARASHIMA PRECODING

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Richard Dale Wesel
August 1996

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

John M.Cioffi
(Principal adviser)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

John T. Gill

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Donald C. Cox

Approved for the University Committee on Graduate Studies:

Abstract

The increased data rates and reliability required to support emerging multimedia applications require new communications technology. We present results regarding two techniques used in high data rate transmission — trellis coding and Tomlinson-Harashima precoding.

New trellis code design metrics, *periodic effective length* and *periodic product distance*, identify robust codes for correlated fading permuted with periodic interleaving. These codes are useful for a variety of communication scenarios including broadcast and point-to-point multicarrier modulation over frequency selective channels, single carrier modulation over channels with Rayleigh fading, and frequency hopped single carrier transmissions.

Simulations of a 16-QAM rate-1/4, 64-state trellis code designed using the new metrics demonstrate very robust performance. The new code achieves a bit error rate of 10^{-6} at an equivalent AWGN SNR of 4.75 dB on a severely frequency selective channel. Codes designed for independent Rayleigh fading or AWGN fail completely on this channel at this SNR.

We introduce techniques useful in trellis code searches based on common design metrics. One technique identifies trellis codes that have the same metric without computing the metrics. This technique can reduce search complexity significantly. Also, we propose a new labeling strategy, which is to maximize the constellation *edge length profile*. Labeled constellations with maximum edge length profile produce trellis codes with the largest possible values of Euclidean distance, product distance, or periodic product distance. Some Ungerboeck and Gray labelings maximize the edge length profile, but others do not.

We apply Ungerboeck's mutual information comparison technique to single carrier transmissions with Rayleigh fading and broadcast multicarrier transmissions with frequency selective fading. Two bits of redundancy are sufficient for uncorrelated Rayleigh fading. The number of bits of redundancy useful for broadcast multicarrier transmissions depends on how high the subchannel SNRs can be.

We derive expressions characterizing the achievable rates possible with Tomlinson-Harashima precoding (THP). Precoding techniques use nonlinear, channel-dependent signal processing in the transmitter to achieve the same SNR as a decision feedback equalizer, but without error propagation. We use our achievable rate expressions to compare THP performance with channel capacity and to demonstrate the importance of symbol rate optimization for good THP performance.

Acknowledgements

My advisor John Cioffi provided me with an environment in which I could thrive. His confidence in me from the moment I joined his group was an inspiring gift for which I am extremely grateful. My success at Stanford is in large measure due to his support for me.

Professor Tom Cover gave me financial support and a home in his research group during my first two years at Stanford. His ability to present concepts in information theory in their simplest and most beautiful form made learning from him a pleasure.

I was also fortunate to interact with Professors Bob Gray, Don Cox, and John Gill at various stages of my research. I am grateful for the support and insight which they provided. I pressed Professors Gill and Cox into service as dissertation readers and members of my orals committee. Thank you for the careful reviews you provided for this dissertation. Thanks to Mark Horowitz who served as chair for my orals committee.

Many of my fellow students added substantively to my research through helpful discussions and insightful suggestions. Some especially helpful colleagues are Phil Bednarz, Suhas Diggavi, John Fan, Garud Iyengar, Krista Jacobsen, Amos Lapidoth, Costis Maglaras, Karen Oehler, Erik Ordentlich, Greg Raleigh, Tim Schmidl, Jose Tellado, Mitch Trott, Katie Wilson, and Nick Zogakis. Many of these people read chapters of my dissertation during those hectic final months. My wife, Dr. Ellen Wesel, also edited every chapter of my dissertation, but more about her later...

Mike Grant provided invaluable LaTeXand Unix assistance, responding above and beyond the call of duty on several occasions.

It was a pleasure to be one of Cioffi's "kids". This extended family of pleasant and friendly people includes my officemates Phil, Zi-ning, Minnie, Jose, Joe, Krista, Susan, Cory, Greg, Acha, Katie, and Kok-Wei. To Acha, I will my apartment at

659 Channing Avenue. To Krista, who finished her dissertation yesterday, I wish fulfillment at the mother ship and success in her real estate ventures. Not to be neglected are Nick Zogakis and Naofal Al-Dhahir who resided in the dungeon of McCullough, but were Cioffi kids nonetheless. Two Cioffi alumni I enjoyed getting to know are Pete Chow and Jim Aslanis who allowed me, by special dispensation, to attend irregular meetings of the Amati chess club. I would like to welcome the incoming members of the group: Louise Hoo, Joe Lauer, Joon-Suk Lee, and Rohit Negi.

My good friends Erik, Elza, and Vittorio were officemates during my days as a resident of Durand 141. Vittorio convinced me to join him in renting the apartment at 659 Channing Avenue. Vittorio was a great roommate and is a wonderful friend. While Vittorio was my roommate I watched his romance with Susan Bree blossom. Vittorio and Susan Bree were married just two weeks ago, and I wish them a lifetime of happiness.

Our friendly and helpful administrative staff Joice DeBolt, Denise Cuevas, Charlotte Coe, and Mieko Parker were a great resource. Things at ISL would grind quickly to a halt without them.

As I write this, I am afraid that several important people are being accidentally omitted. I apologize if you deserve to be here, but are not. I hope that you can forgive the oversight.

Thanks to my good friend and bike riding partner Bill Bencze. You helped me through some of those dark valleys, as well as up some pretty steep hills.

My parents, my brothers Mike, Dave, and Joe, and my sisters Marilyn, Lisa, Cathi, and Anne are a constant source of love and support. You will always have my love.

Most of all, I thank my wife Ellen. Her love has given me happiness. The day that she walked into my world the lights switched on and everything was wonderful. We have been married just over a year now, and I hope that this past year was the first of many happy years together. To her I dedicate this dissertation.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Popular demand is increasing for multimedia applications such as videoconferencing and digital video. These services require robust, high data rate communication over fading and dispersive channels. One physical layer technique that can provide robust communication over fading channels is multicarrier modulation. However, to perform well in a fading environment, multicarrier modulation must be combined with a well designed channel code. Chapters 2–6 of this dissertation provide a complete design strategy that produces trellis codes with very robust performance in the presence of correlated fading or interference. While multicarrier modulation was the initial application of interest, the resulting trellis codes provide reliable performance wherever periodic interleaving is used to mitigate correlated fading or interference.

A separate question addressed in this dissertation involves high data rate single carrier transmission over a wireline channel (e.g. twisted pair or coaxial cable). Wireline single carrier systems use some form of equalization to compensate for the dispersion introduced by the wire. Decision feedback equalization (DFE) allows reliable transmission at rates approaching channel capacity, assuming error propagation in the DFE feedback loop can be neglected. However, error propagation cannot be neglected when combining a DFE with a trellis code. Precoding uses signal processing in both the transmitter and receiver to achieve the error variances of DFE without introducing error propagation. Chapter 7 examines a commonly used form of precoding, Tomlinson-Harashima precoding (THP). This chapter quantifies how closely THP can approach the channel capacity at any specified SNR on a given dispersive channel.

1.1 A Road Map

Chapter 2 begins the investigation of trellis code design with a discussion of the various design metrics used to select codes for various channel environments. Chapter 3 formally introduces trellis codes and their principal building block, convolutional codes. Chapter 4 investigates how a constellation should be labeled to maximize the metrics of Chapter 2. All formal definitions are listed in Appendix A. This list of definitions should be a useful reference while reading Chapters 2–4.

Chapter 5 reviews multicarrier modulation and relevant capacity calculations for additive white Gaussian noise, Rayleigh fading, and intersymbol interference channels. This chapter then draws on information theory to investigate the number of points a constellation should have for a given rate under specified communication scenarios. The scenarios considered are single carrier transmission over additive white Gaussian noise channels, single carrier transmission over Rayleigh fading channels, and multicarrier modulation over intersymbol interference channels.

Chapter 6 uses the ideas of Chapters 2–5 to design three trellis codes. Simulations of the three trellis codes in a multicarrier modulation system over five channels show the utility of the new metrics introduced in Chapter 2. The code designed with the new metrics achieves an error rate of 10^{-6} on a severely frequency selective channel on which the two codes designed to be best according to the commonly used metrics fail completely.

Chapter 7 presents an achievable rate characterization for Tomlinson-Harashima precoding (THP). An exact expression is derived for the zero-forcing THP. Upper and lower bounds are provided for the minimum mean square error THP. The achievable rates for the zero-forcing and minimum mean square error formulations of THP are compared to each other and to channel capacity for some example channels. The importance of correctly choosing the symbol rate is demonstrated.

Figure 1.1 shows the dependence among chapters. Readers may safely go straight to Chapter 7 if the achievable rate characterization of THP is of primary interest. Chapters 2–6 are best read in order. However, Chapter 5 — Capacity and Constellation Size — may be read independently.

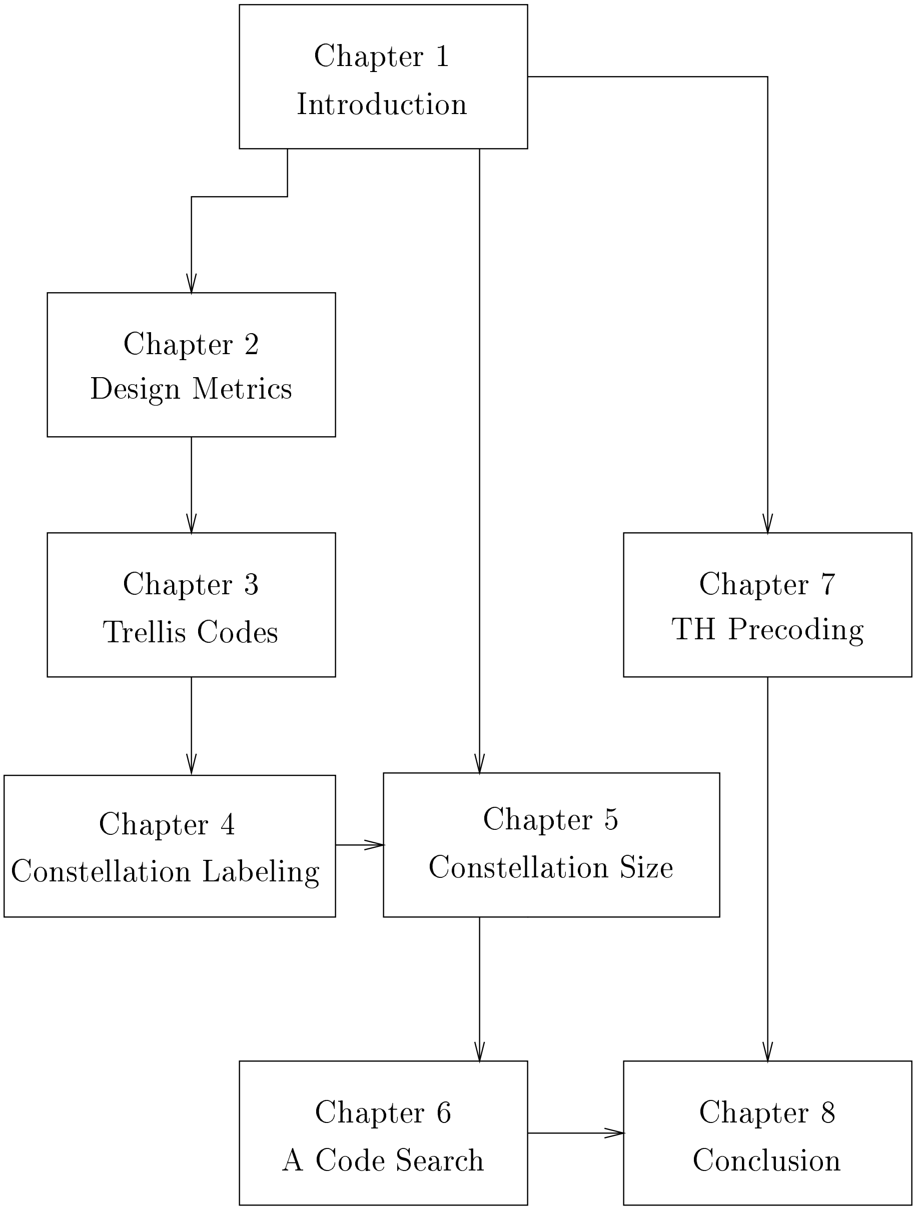


Figure 1.1: Dependence among chapters.

1.2 Contributions

The main contributions of this dissertation are a trellis code design strategy for correlated fading and/or interference and an achievable rate characterization of Tomlinson-Harashima precoding. The details of these contributions are listed below by chapter.

Chapter 2) Introduces the new metrics *periodic effective code length* and *code periodic product distance*. These metrics identify trellis codes that perform well over channels with correlated fading.

Chapter 3) Identifies a systematic method for identifying trellis codes that are equivalent according to the metrics of Chapter 2. This method reduces search complexity significantly since the metric is computed for only one code in a set of equivalent codes.

Chapter 4) Introduces the labeling strategy of maximizing the *edge length profile* of the labeled constellation. Also demonstrated is a close relationship between Ungerboeck labeling and Gray Coding.

Chapter 5) Applies Ungerboeck's mutual information comparison technique to examine the potential of various constellations on fading channels.

Chapter 6) Designs and simulates codes using standard metrics and the new design metrics presented in Chapter 2. Simulations on five example channels demonstrate the robustness of codes designed according to the new metrics.

Chapter 7) Derives new expressions characterizing the achievable rates possible with Tomlinson-Harashima precoding (THP). These expressions are used to compare THP performance with the channel capacity and to demonstrate the importance of symbol rate optimization for good THP performance.

Chapter 2

Design Metrics

The search for a good trellis code requires the ability to compare several codes to determine which is best. The ultimate measure of a channel code is its ability to support an intended application. Since errors affect different applications in different ways, error rates are computed differently for different applications. Depending on the application, the error rate may be measured in bits, bytes, blocks, or packets.

Computing the error rate of interest for each candidate trellis code is prohibitively complex. This chapter introduces metrics that are closely related to the error rates of interest, but which can be computed with relatively low complexity. These metrics might be the sole basis for selection of a code, or they might be used to identify a handful of candidate codes, with the final selection based on simulations.

2.1 A General Encoder and Channel Model

Three different sets of metrics are derived, each appropriate for a different communication scenario. Each of these scenarios is a special case of the communication system shown in Figure 2.1. This figure shows a binary information sequence $\mathbf{b} = \{b_i\}$ encoded as a sequence $\mathbf{x} = \{x_i\}$ of real or complex values. Each x_i is scaled by the gain a_i to produce the sequence $\mathbf{y} = \{y_i\}$. The scaled values y_i are then distorted by additive white Gaussian noise (AWGN) terms n_i to produce the sequence $\mathbf{z} = \{z_i\}$ that is available to the receiver.

In the discussion that follows, x_i, a_i, y_i, n_i, and z_i are assumed to be complex numbers. However, the same concepts and metrics apply when they are real. The

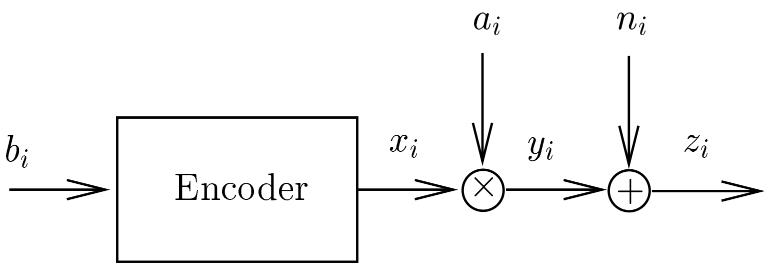


Figure 2.1: Encoder and channel model.

variance of the real and imaginary parts of the AWGN are assumed to be constant and both equal to σ^2 so that the noise power is

$$E|n_i|^2 = 2\sigma^2. \tag{2.1}$$

Actual variations in the noise power can be reflected in a_i.

The average power of each transmitted symbol is assumed to be the same so that the signal power is

$$E|x_i|^2 = E|x|^2. \tag{2.2}$$

Practical transmitters must obey a power constraint, which is reflected in the constraint that

$$E|x|^2 \leq \mathcal{E}_x. \tag{2.3}$$

In general, increasing $E|x|^2$ lowers all of the error rates mentioned above. Increasing $E|x|^2$ certainly improves each of the three sets of metrics presented below. As a result, it is assumed in the following discussion that the power constraint in (2.3) is met with equality:

$$E|x|^2 = \mathcal{E}_x. \tag{2.4}$$

An important measure of channel quality is the signal to noise ratio (SNR) defined to be

$$\mathrm{SNR} = \frac{E|x|^2}{E|n|^2} = \frac{\mathcal{E}_x}{2\sigma^2}. \tag{2.5}$$

The following discussion assumes that the receiver has perfect knowledge of σ^2 and the sequence of scale factors $\{a_i\}$. In practice, these values must be estimated by the receiver. The details of channel estimation are beyond the scope of this dissertation. However, in many situations, the values change slowly enough that low complexity estimation techniques provide the receiver with accurate estimates. For more information about channel estimation techniques, see [1]. For an example of channel estimation specific to multicarrier modulation, see [2].

The communications scenarios considered below are distinguished from each other by the behavior of the sequence $\{a_i\}$. For the AWGN channel considered in Section 2.2, the a_i values are all equal to the same constant. For the Rayleigh Fading channel considered in Section 2.3, the magnitude of each a_i has a Rayleigh probability density function (PDF), and

$$a_i \perp a_j \qquad \text{for } i \neq j \tag{2.6}$$

where $\perp$ indicates independence. For the arbitrary independent fading considered in Section 2.4, (2.6) still applies, but no particular PDF is considered. For the correlated fading channel considered in Section 2.5, adjacent values of a_i are strongly correlated so (2.6) does not apply.

In each of these cases, the receiver must estimate the input information sequence $\mathbf{b}$ based on its observation of the sequence $\mathbf{z}$ of received values. In this dissertation, the receiver chooses the sequence $\mathbf{b}$ most likely to have produced the observed sequence $\mathbf{z}$. This method of reception, commonly referred to as maximum likelihood sequence detection, minimizes the probability of a sequence error, but does not necessarily minimize the bit error rate (BER). Maximum likelihood sequence detection for trellis codes is performed with the Viterbi algorithm [3]. For an example of an algorithm that minimizes BER, see [4].

Consider transmission of a length L sequence (row vector) $\mathbf{x}$ corresponding to an information sequence $\mathbf{b}$. For a given sequence of scale factors $\mathbf{a}$, the sequences of scaled outputs $\mathbf{y}$ is computed as

$$\mathbf{y} = \mathbf{x}\ \mathrm{Diag}(\mathbf{a}). \tag{2.7}$$

The length L complex sequence (or row vector) $\mathbf{y}$ is a point in $2L$-dimensional space. For the channel described in Figure 2.1, the most likely information sequence $\hat{\mathbf{b}}$ is the one whose scaled output sequence $\hat{\mathbf{y}}$ is closest in Euclidean distance to the received sequence $\mathbf{z}$. This is the sequence that would be identified by the Viterbi algorithm.

Figure 2.2 shows the set of possible scaled output sequences $\mathbf{y}$ as points in $2L$-dimensional Euclidean space. As an example of an error, the figure shows the point $\mathbf{y}$ corresponding to the actual $\mathbf{b}$ and a different point $\hat{\mathbf{y}}$ that is closer than $\mathbf{y}$ to the received point $\mathbf{z} = \mathbf{y} + \mathbf{n}$.

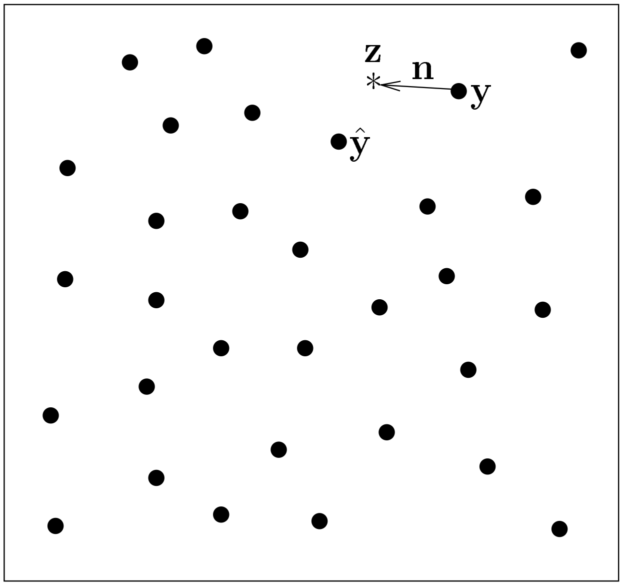


Figure 2.2: The possible values of $\mathbf{y}$ are shown as •'s. When the received value $\mathbf{z}$ produced by a noise vector $\mathbf{n}$ added to the transmitted $\mathbf{y}$ is closer to $\hat{\mathbf{y}}$ that the transmitted $\mathbf{y}$, an error occurs.

In general, the probability of a sequence error such as the one demonstrated in Figure 2.2 depends on the position of all the possible points $\hat{\mathbf{y}}$. However, as σ^2 decreases, the probability of a sequence error is closely approximated by considering only the points $\hat{\mathbf{y}}$ nearest to the point $\mathbf{y}$ corresponding to the transmitted $\mathbf{x}$.

The pairwise probability of error associated with the pair of points $\{\mathbf{y}, \hat{\mathbf{y}}\}$ is the probability that $\mathbf{z} = \mathbf{y} + \mathbf{n}$ is closer to $\hat{\mathbf{y}}$ than $\mathbf{y}$. The region in which such a pairwise error occurs is always the halfspace containing $\hat{\mathbf{y}}$ bounded by the hyperplane that bisects the line segment connecting $\mathbf{y}$ and $\hat{\mathbf{y}}$. This halfspace is the shaded region in Figure 2.3.

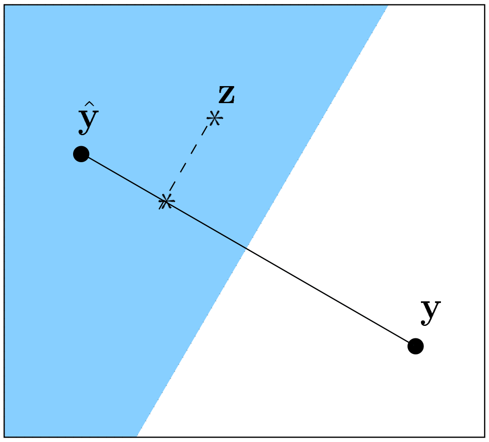


Figure 2.3: Decision Regions for $\mathbf{y}$ and $\hat{\mathbf{y}}$. In the shaded region, $\mathbf{z} = \mathbf{y} + \mathbf{n}$ is closer to $\hat{\mathbf{y}}$ than $\mathbf{y}$.

Equivalently, this pairwise error occurs when the projection of $\mathbf{z}$ onto the line containing $\mathbf{y}$ and $\hat{\mathbf{y}}$ lies on the same side of the bisecting hyperplane as $\hat{\mathbf{y}}$. Figure 2.3 shows the projection of such a $\mathbf{z}$ onto the line.

The one dimensional noise n along the line containing $\mathbf{y}$ and $\hat{\mathbf{y}}$ resulting from this projection has variance σ^2 (by symmetry since the noise variance in every dimension is σ^2). Thus the probability of making the pairwise error between $\mathbf{b}$ and $\hat{\mathbf{b}}$ conditioned on $\mathbf{a}$ is

$$\begin{aligned} P(\mathbf{b} \to \hat{\mathbf{b}} \mid \mathbf{a}) &= P(\mathbf{y} \to \hat{\mathbf{y}}) && (2.8) \\ &= Q\left(\frac{\|\mathbf{y} \perp \hat{\mathbf{y}}\|_2}{2\sigma}\right), && (2.9) \end{aligned}$$

where $\| \cdot \|_2$ is Euclidean distance (the 2-norm) and $Q(t)$ is the probability in the tail of a zero mean unit variance Gaussian PDF:

$$Q(t) = \frac{1}{\sqrt{2\pi}} \int_t^{\infty} e^{-u^2/2} du. \tag{2.10}$$

All of the metrics presented below are derived by considering the possible pairs $(\mathbf{y}, \hat{\mathbf{y}})$ or $(\mathbf{x}, \hat{\mathbf{x}})$ to find the pairs that are most likely to be mistaken for each other.

2.2 The AWGN Channel

The simplest scenario considered is the AWGN channel, which has $a_i = 1$ for all i. This implies that $\mathbf{x} = \mathbf{y}$ in Figure 2.1. The pairwise probability of error for this scenario is

$$\begin{aligned} P(\mathbf{b} \to \hat{\mathbf{b}}) &= P(\mathbf{x} \to \hat{\mathbf{x}}) \qquad (2.11) \\ &= Q\left(\frac{\|\mathbf{x} \perp \hat{\mathbf{x}}\|_2}{2\sigma}\right). \qquad (2.12) \end{aligned}$$

The ordered pair $(\mathbf{b}, \hat{\mathbf{b}})$ that has the largest pairwise probability of error is exactly the pair that has the smallest value of $\|\mathbf{x} \perp \hat{\mathbf{x}}\|_2$. For this reason, the primary metric used to evaluate a code for use on an AWGN channel is the smallest value of $\|\mathbf{x} \perp \hat{\mathbf{x}}\|_2$ for two valid output sequences of the code. This metric is usually referred to as the *code Euclidean distance* or *free Euclidean distance.*

Definition 1 *The (normalized) squared Euclidean distance* ED^2 *of the error sequence* $\mathbf{b} \to \hat{\mathbf{b}}$ *is*

$$\text{ED}^2 = \frac{\|\mathbf{x} \perp \hat{\mathbf{x}}\|_2^2}{\mathcal{E}_x} \tag{2.13}$$

Definition 2 *The code Euclidean distance* CED *of a code is the smallest* ED *of an error sequence* $\mathbf{b} \to \hat{\mathbf{b}}$ *associated with that code.*

Ungerboeck identified trellis codes with large Euclidean distances in [5]. A secondary metric for the AWGN channel is the number of *Euclidean nearest neighbors.*

Definition 3 *The number of Euclidean nearest neighbors* N_{CED} *is the number of sequences* $\hat{\mathbf{x}}$ *that are* CED *away from a transmitted sequence* $\mathbf{x}$.

2.3 Independent Rayleigh Fading

In wireless communications, scatterers and reflectors often cause the transmitted signal to arrive at the receiver from many different directions, with each direction having a different associated complex gain. By the central limit theorem, the scale factors $\{a_i\}$ resulting from the sum of the complex gains along these many paths are approximately complex Gaussian random variables, each having covariance matrix

$$\begin{bmatrix} E[a_{\mathrm{R}}a_{\mathrm{R}}] & E[a_{\mathrm{R}}a_{\mathrm{I}}] \\ E[a_{\mathrm{I}}a_{\mathrm{R}}] & E[a_{\mathrm{I}}a_{\mathrm{I}}] \end{bmatrix} = \begin{bmatrix} \gamma^2 & 0 \\ 0 & \gamma^2 \end{bmatrix}. \tag{2.14}$$

The random variable $q = a_{\mathrm{R}}^2 + a_{\mathrm{I}}^2$ has an exponential PDF:

$$p(q) = \frac{e^{-q/2\gamma^2}}{2\gamma^2} \qquad q \geq 0. \tag{2.15}$$

The random variable $r = \sqrt{q}$ has a Rayleigh PDF:

$$p(r) = \frac{r\,e^{-r^2/2\gamma^2}}{\gamma^2} \qquad r \geq 0. \tag{2.16}$$

Because the PDF of r is Rayleigh, channels with fading sequences $\mathbf{a}$ of this type are said to exhibit *Rayleigh fading.* In this section, the metrics for Rayleigh fading assume $\mathbf{a}$ is a sequence of IID complex Gaussian random variables. Often, adjacent values a_i and a_{i+1} are in fact strongly correlated. However, it is assumed that interleaving at the transmitter and de-interleaving at the receiver disperse correlated fading enough for adjacent a_i's to be uncorrelated at the decoder.

Dispersing the correlated symbols improves performance in complexity-constrained systems. However, ignoring the correlation in $\mathbf{a}$ leads to suboptimal performance even after this dispersal. Section 2.5 derives metrics that account for the correlation after interleaving.

The pairwise sequence error probability (2.9) for the pair $(\mathbf{b}, \hat{\mathbf{b}})$ conditioned on the sequence $\mathbf{a}$ of scale factors can be written in terms of $\mathbf{x}$ and $\mathbf{a}$ as

$$P(\mathbf{b} \rightarrow \hat{\mathbf{b}} \mid \mathbf{a}) = Q\left(\sqrt{\frac{\sum_{i=1}^{L} |a_i|^2 \; \|x_i \perp \hat{x}_i\|_2^2}{4\sigma^2}}\right). \tag{2.17}$$

When **a** is a sequence of IID random variables, the marginal pairwise sequence error can be computed by integrating (2.17) over the PDF for each a_i. When this PDF is the complex Gaussian with covariance matrix of (2.14), integrating (2.17) over the Rayleigh PDF for each $r_i = |a_i|$ produces the marginal pairwise sequence error probability

$$P(\mathbf{b} \to \hat{\mathbf{b}}) = \int_{r_1=0}^{\infty} \cdots \int_{r_L=0}^{\infty} Q\left(\sqrt{\frac{\sum_{i=1}^{L} r_i^2 \|x_i \perp \hat{x}_i\|_2^2}{4\sigma^2}}\right) \frac{r_1\, e^{-r_1^2/2\gamma^2}}{\gamma^2} dr_1 \ldots \frac{r_L\, e^{-r_L^2/2\gamma^2}}{\gamma^2} dr_L. \tag{2.18}$$

Unfortunately, (2.18) does not have a closed form solution. As a result, it does not lead to a useful design metric. Wilson & Leung [6] and Divsalar & Simon [7, 8] obtained a closed form expression that upper bounds (2.18). This upper bound is derived again here because of the importance of the metrics that result. (See also [9] for an exhaustive review of the literature on fading channel code design metrics.)

To obtain a closed form expression, a Chernoff bound (see page 318 of [10]) is applied to the zero mean unit variance Gaussian distribution in order to bound $Q(t)$:

$$\begin{aligned} Q(t) &\leq e^{-st} e^{s^2/2} \quad \text{for any } s && (2.19)\\ &= e^{-t^2/2} \qquad \text{for } s = t && (2.20) \end{aligned}$$

Applying (2.20) to (2.18) yields

$$\begin{aligned} P(\mathbf{b} \to \hat{\mathbf{b}}) &\leq \prod_{i=1}^{L} \int_{r_i=0}^{\infty} e^{-(r_i^2 \|x_i - \hat{x}_i\|_2^2 / 8\sigma^2)} \; \frac{r_i\, e^{-r_i^2/2\gamma^2}}{\gamma^2} \; dr_i && (2.21)\\ &= \prod_{i=1}^{L} \int_{r_i=0}^{\infty} \frac{r_i}{\gamma^2} \; e^{-r_i^2/2\gamma^2 \left(1 + \frac{\gamma^2 \|x_i - \hat{x}_i\|_2^2}{4\sigma^2}\right)} dr_i && (2.22)\\ &= \prod_{i=1}^{L} \left(1 + \frac{\gamma^2 \|x_i \perp \hat{x}_i\|_2^2}{4\sigma^2}\right)^{-1} && (2.23)\\ &= \prod_{i \in A} \left(1 + \frac{\gamma^2 \|x_i \perp \hat{x}_i\|_2^2}{4\sigma^2}\right)^{-1} \quad \text{where } A = \{i \mid x_i \neq \hat{x}_i\} && (2.24)\\ &\approx \prod_{i \in A} \left(\frac{\gamma^2 \|x_i \perp \hat{x}_i\|_2^2}{4\sigma^2}\right)^{-1} \quad \text{when } \frac{\gamma^2 \|x_i \perp \hat{x}_i\|_2^2}{4\sigma^2} \gg 1. && (2.25) \end{aligned}$$

For practical codes with the values of x_i drawn from a finite alphabet, there is a minimum nonzero value for $\|x_i \perp \hat{x}_i\|_2^2$. For these codes, the approximation of (2.25) is valid regardless of the specific sequences $\mathbf{x}$ in the code as long as γ^2/σ^2 is large enough.

The approximation in (2.25) is equivalent to (7) in [7] and (9.22) in [8]. Also, (3.5) in [6] is a sum of pairwise error probabilities equivalent to (2.25). In [7] Divsalar and Simon based two metrics on (2.25), *effective code length* and *code product distance.*

Definition 4 *The effective length* EL *of the error sequence* $\mathbf{b} \to \hat{\mathbf{b}}$ *is the cardinality of the set* $\{i \mid x_i \perp \hat{x}_i \neq 0\}$.

Definition 5 *The effective code length* ECL *of a code is the smallest* EL *associated with that code.*

Definition 6 *The product distance* PD *of the error sequence* $\mathbf{b} \to \hat{\mathbf{b}}$ *is defined to be*

$$\text{PD} = \prod_{i \in A} \frac{\|x_i \perp \hat{x}_i\|_2^2}{\mathcal{E}_x} \qquad \textit{where } A = \{i \mid x_i \neq \hat{x}_i\}. \tag{2.26}$$

Definition 7 *The code product distance of order i of a code,* CPD$_i$*, is the smallest product distance of an error sequence having* EL$= i$ *associated with that code.*

Rewriting (2.25) in terms of these metrics yields

$$P(\mathbf{b} \to \hat{\mathbf{b}}) \leq \left(\frac{\gamma^2 \mathcal{E}_x}{4\sigma^2}\right)^{-\text{EL}} \frac{1}{\text{PD}}. \tag{2.27}$$

The marginal pairwise sequence error probability decreases exponentially with EL and is inversely proportional to PD. Du & Vucetic have designed numerous rate $k/k+1$ trellis codes using the metrics of ECL and CPD [11, 12, 13, 14]. As with Euclidean distance for the AWGN channel, the number of EL nearest neighbors and PD nearest neighbors are secondary considerations in the code selection process.

2.4 Arbitrary Independent Fading

The previous section demonstrated that the code product distance and the effective code length are informative metrics in the context of independent Rayleigh fading.

However, the sequence of scale factors $\mathbf{a}$ do not always obey a Rayleigh PDF, as demonstrated with the following two examples.

Signal power from interfering transmitters at time index i lowers the signal to noise ratio for x_i, which is equivalent to lowering a_i for the purposes of computing code design metrics. The effective values of a_i due to interference often do not follow a Rayleigh distribution. For example, narrow band frequency hopping where some hops have severe interference and others have no interference produces a bimodal distribution of a_i values very different from Rayleigh.

With multicarrier modulation, blocks in the sequence $\mathbf{a}$ correspond to the discrete Fourier transform (DFT) of a frequency selective channel. When neither transmitter nor receiver is moving (an HDTV transmission, for example) the frequency response is fixed for an extended period of time. The scale factors associated with this frequency response need not have a Rayleigh distribution.

In these cases of interference and fading, effective code length and code product distance are still informative metrics, even though the derivation of the previous section does not apply. Effective code length is essentially a measure of the diversity provided by the code [9, 15]. Whenever transmission is over a channel with independent scale factors, increasing diversity (and thus effective code length) improves performance.

Two motivations for the usefulness of product distance outside the context of Rayleigh fading are given below. Both of these arguments apply Lagrange multipliers to solve relevant constrained minimization problems. The code product distance essentially measures how evenly Euclidean distance is distributed to the "diversity branches" of the code. To see this, consider the following primal maximization problem:

$$\begin{aligned} &\text{Maximize} \quad \prod_{i=1}^{N} d_i \\ &\text{Subject to} \quad \sum_{i=1}^{N} d_i \perp s \leq 0 \end{aligned}$$

where s is a constant. The Lagrangian dual problem [16] is

$$\text{Minimize} \quad \max_{\mathbf{d}} \left(\prod_{i=1}^{N} d_i \perp \lambda \left(\sum_{i=1}^{N} d_i \perp s \right) \right)$$

$$\text{Subject to} \quad \lambda > 0$$

The solution to the dual problem (and thus the primal as well) must satisfy

$$\sum_{i=1}^{N} d_i = s \tag{2.28}$$

and

$$d_i = \frac{\prod_{i=1}^{N} d_i}{\lambda} \qquad \text{for all } i. \tag{2.29}$$

The only choice of $(d_1, \ldots, d_N)$ and λ that satisfies these conditions is

$$d_i = \frac{s}{N} \qquad \text{for } i = 1, 2, \ldots, N \tag{2.30}$$

and

$$\lambda = \left(\frac{s}{N} \right)^{N-1}. \tag{2.31}$$

In other words, an even distribution of distance maximizes the product. As the distribution of distance becomes more skewed, the product decreases.

Ordentlich [17] provides a second argument motivating the product distance. This argument fixes the pair $(\mathbf{b}, \hat{\mathbf{b}})$ and seeks the fading sequence $\mathbf{a}$ that maximizes the pairwise sequence error probability

$$P(\mathbf{b} \to \hat{\mathbf{b}} \mid \mathbf{a}) = Q\left(\sqrt{\frac{\sum_{i=1}^{L} |a_i|^2 \; \|x_i \perp \hat{x}_i\|_2^2}{4\sigma^2}} \right). \tag{2.32}$$

If no constraints are placed on $\mathbf{a}$, this leads to the uninteresting result that $\mathbf{a}$ is the all zeros vector and the probability of error is the same as when the transmitter sends nothing. To make the analysis meaningful, the fading sequences are constrained to support at least some minimum amount of information transmission. That is, all the fading sequences considered must induce at least some specified *mutual information* [10] on the channel. The total mutual information I for a specified fading sequence $\mathbf{a}$ is

$$\begin{aligned} I &= \sum_{i=1}^{L} \log_2 \left(1 + \frac{|a_i|^2 E|x_i|^2}{E|n_i|^2} \right) && (2.33) \\ &= \sum_{i=1}^{L} \log_2 \left(1 + \frac{|a_i|^2 \mathcal{E}_x}{2\sigma^2} \right) \qquad \text{by (2.1) and (2.4).} && (2.34) \end{aligned}$$

Considering only the values of a_i that affect the pairwise error probability (2.32), the constraint on $\mathbf{a}$ is

$$\sum_{i \in A} \log_2 \left(1 + \frac{|a_i|^2 \mathcal{E}_x}{2\sigma^2} \right) \geq R\,\mathrm{EL} \tag{2.35}$$

where $A = \{i \mid x_i \neq \hat{x}_i\}$ and R is the desired per symbol information rate. Neglecting the 1 in (2.35) produces the approximate constraint

$$\sum_{i \in A} \log_2 \left(\frac{|a_i|^2 \mathcal{E}_x}{2\sigma^2} \right) \geq R\,\mathrm{EL} \qquad \text{for high SNR.} \tag{2.36}$$

The above approximation is not as accurate as the approximation that produced (2.25) since even at large values of $\mathcal{E}_x/\sigma^2$, there are some fading sequences that easily satisfy (2.35) but not (2.36). These sequences are incorrectly removed from the set over which minimization is performed. Nonetheless, this is an informative exercise as long as the limitations of its interpretation are made clear.

The resulting constrained minimization problem can be expressed as

$$\begin{aligned} \text{Minimize} \quad & \sum_{i \in A} |a_i|^2 \, \|x_i \perp \hat{x}_i\|_2^2 \\ \text{Subject to} \quad & c \perp \sum_{i \in A} \log_2(|a_i|^2) \leq 0 \end{aligned}$$

where

$$c = \mathrm{EL} \cdot \left(R \perp \log_2 \left(\frac{\mathcal{E}_x}{2\sigma^2} \right) \right). \tag{2.37}$$

The Lagrangian dual problem [16] is

$$\begin{array}{ll} \text{Maximize} & \displaystyle\min_{\mathbf{a}} \left(\sum_{i \in A} |a_i|^2 \, \|x_i \perp \hat{x}_i\|_2^2 + \lambda \left(c \perp \sum_{i \in A} \log_2(|a_i|^2) \right) \right) \\ \text{Subject to} & \lambda > 0 \end{array}$$

The solution to the dual (and thus the primal) is

$$|a_i|^2 = \frac{\lambda}{\|x_i \perp \hat{x}_i\|_2^2} \tag{2.38}$$

$$\lambda = \left(2^c \prod_{i \in A} \|x_i \perp \hat{x}_i\|_2^2 \right)^{1/\mathrm{EL}} \tag{2.39}$$

The worst case fading sequence $\mathbf{a}_{\text{worst}}$ identified in (2.38) applied to (2.32) produces

$$P(\mathbf{b} \to \hat{\mathbf{b}} \mid \mathbf{a}_{\text{worst}}) = Q\left(\sqrt{\frac{\mathrm{EL}\,\lambda}{4\sigma^2}} \right). \tag{2.40}$$

Using (2.39) and (2.37) to substitute for lambda in (2.40) produces

$$P(\mathbf{b} \to \hat{\mathbf{b}} \mid \mathbf{a}_{\text{worst}}) = Q\left(\sqrt{2^{R-1}\ \mathrm{PD}^{1/\mathrm{EL}}\ \mathrm{EL}} \right). \tag{2.41}$$

Note that (2.41) is independent of SNR. This is because the mutual information constraint (2.36) induces a constant geometric mean SNR. If all the pairs $(\mathbf{b}, \hat{\mathbf{b}})$ with a given EL are considered, the pair (or pairs) with the smallest PD have the largest worst case error probability according to (2.41). Thus, increasing the code product distance improves this worst case performance.

Applying the Chernoff bound of (2.20) to (2.41) produces

$$P(\mathbf{b} \to \hat{\mathbf{b}} \mid \mathbf{a}_{\text{worst}}) \leq e^{2^{R-2}\,\mathrm{PD}^{1/\mathrm{EL}}\,\mathrm{EL}}. \tag{2.42}$$

Comparing (2.42) with (2.27) reveals that both expressions are made smaller by larger values of PD. However, (2.42) decreases exponentially in $\text{PD}^{1/\text{EL}}$ while (2.27) is only inversely proportional to PD. Thus increasing PD improves performance in both cases, but the improvement is more dramatic for the worst case situation. This makes sense because the even spread of distance is more crucial for the extreme variations found in the worst case fades than in the relatively gentle Rayleigh distribution.

2.5 Correlated Fading or Interference

The effective length EL and product distance PD are informative only when the scale factors a_i are uncorrelated. The arguments that justified EL and PD in the previous sections are not compelling when the scale factors are strongly correlated. In particular, if $a_i \approx a_j$ because they are strongly correlated, no diversity is achieved by spreading distance between the two corresponding symbols.

The contribution of these two symbols to the pairwise probability of error remains the same as long as

$$\|x_i \perp \hat{x}_i\|_2^2 + \|x_j \perp \hat{x}_j\|_2^2 \tag{2.43}$$

remains the same. This section introduces new metrics closely related to EL and PD that are informative in the context of correlated fading.

Define the autocorrelation function of the fading sequence to be

$$R_a(k) = E[a_i a_{i+k}^*] \tag{2.44}$$

where a_{i+k}^* denotes the complex conjugate of a_{i+k}. Often adjacent scale factors are strongly correlated, and the correlation $R_a(k)$ decreases as the separation k between scale factors increases.

The rate of decrease in correlation depends on the Doppler frequency for portable wireless transmissions and on the coherence bandwidth for multicarrier modulation (MCM) systems. MCM systems commonly have coherence bandwidths wide enough that large groups of adjacent scale factors are strongly correlated. Similarly, portable wireless systems often have Doppler frequencies low enough that several consecutive symbols are strongly correlated.

So far, this chapter has considered a general class of codes that map binary sequences to points in Euclidean space. In this dissertation, the primary interest is in a special case of such codes called *trellis codes*. Trellis codes are formally introduced in the following chapter, but an observation about them is required here.

A practical trellis code (one with a relatively small number of delay elements) has many error sequences for which all the nonzero values of $|x_i \perp \hat{x}_i|$ occur within a small number of adjacent symbols. For example, Theorem 1 in the next chapter states that

$$\textsc{ecl} \leq \left\lfloor \frac{\nu}{k} \right\rfloor + 1 \tag{2.45}$$

where $\lfloor \cdot \rfloor$ is the floor function, k is the number of input bits per symbol, and ν is the number of memory elements used by the encoder. The proof of this theorem demonstrates that there is always an error sequence for which the nonzero values of $|x_i \perp \hat{x}_i|$ occur within $\lfloor \nu/k \rfloor + 1$ adjacent symbols.

Consider applying such a trellis code directly to a fading channel for which any group of $f > \textsc{ecl}$ adjacent scale factors are strongly correlated. Whenever the nonzero values of $|x_i \perp \hat{x}_i|$ all occur within f, they are multiplied by scale factors a_i with similar values because of the strong correlation in the fading.

For practical trellis codes to have low error rates on channels with correlated fading, adjacent trellis encoder outputs must not be transmitted on adjacent symbols $x_i, x_{i+1}, \ldots, x_f$. An interleaver is used to change the ordering of the trellis encoder outputs before transmission, as illustrated in Figure 2.4. The receiver uses a deinterleaver to return the symbols to their original order before decoding.

There are several types of interleavers, but this dissertation restricts its attention to the class of interleavers known as *periodic interleavers*. A periodic interleaver is specified by two parameters, the block size B and the period P. Figure 2.5 shows a periodic interleaver with $B = 512$ and $P = 8$ implemented by writing the trellis encoder outputs c_i column-by-column into a matrix and reading the interleaver outputs x_i row-by-row out of this matrix. Figure 2.6 shows the associated deinterleaver.

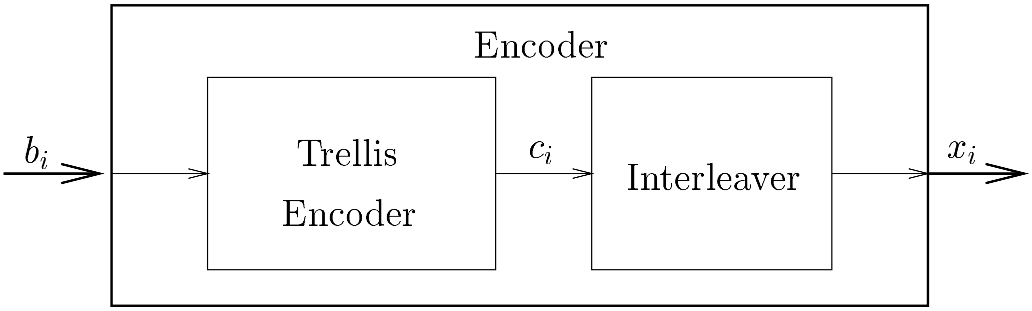
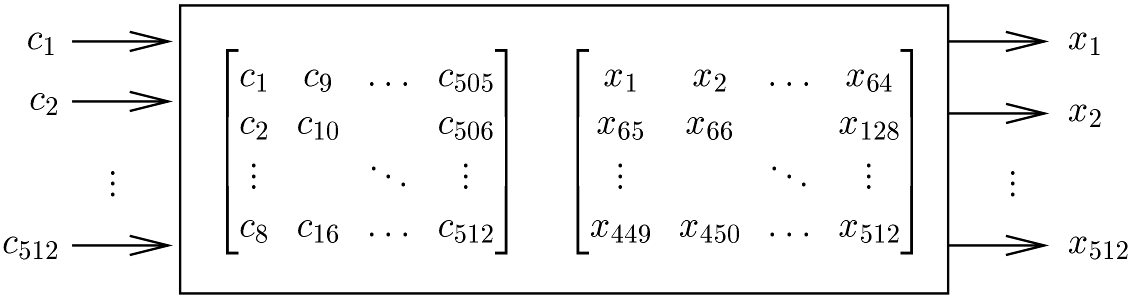
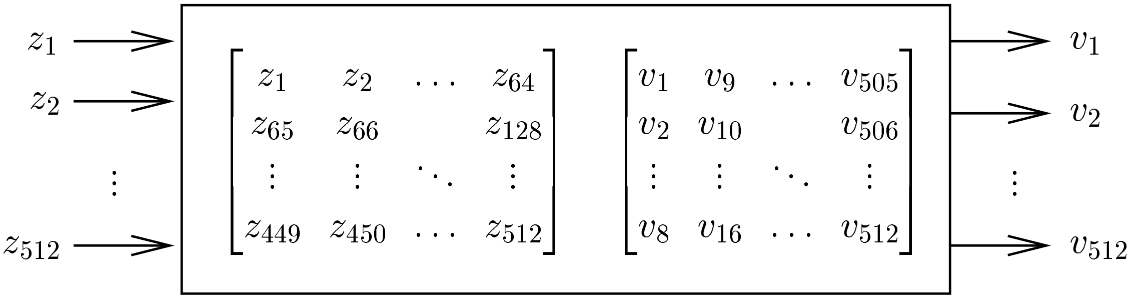


Figure 2.4: The encoder of Figure 2.1 can be composed of a trellis encoder followed by an interleaver.



$$\begin{bmatrix} c_1 & c_9 & \cdots & c_{505} \\ c_2 & c_{10} & & c_{506} \\ \vdots & & \ddots & \vdots \\ c_8 & c_{16} & \cdots & c_{512} \end{bmatrix} \quad \begin{bmatrix} x_1 & x_2 & \cdots & x_{64} \\ x_{65} & x_{66} & & x_{128} \\ \vdots & & \ddots & \vdots \\ x_{449} & x_{450} & \cdots & x_{512} \end{bmatrix}$$

Figure 2.5: Write-by-column, read-by-row implementation of a periodic interleaver with period $P = 8$ and block size $B = 512$. The left matrix shows write labeling, and the right matrix shows read labeling.



$$\begin{bmatrix} z_1 & z_2 & \cdots & z_{64} \\ z_{65} & z_{66} & & z_{128} \\ \vdots & \vdots & \ddots & \vdots \\ z_{449} & z_{450} & \cdots & z_{512} \end{bmatrix} \quad \begin{bmatrix} v_1 & v_9 & \cdots & v_{505} \\ v_2 & v_{10} & & v_{506} \\ \vdots & \vdots & \ddots & \vdots \\ v_8 & v_{16} & \cdots & v_{512} \end{bmatrix}$$

Figure 2.6: Write-by-row, read-by-column implementation of a deinterleaver for the interleaver shown above.

The periodic interleaver-deinterleaver pair shown in Figures 2.5 and 2.6 disperses any group of $P = 8$ consecutive trellis encoder outputs so that they are multiplied by scale factors a_i separated by at least $B/P = 512/8 = 64$ symbols. As an example, consider the eight consecutive deinterleaver outputs $v_1 \dots v_8$ produced by the interleaver, deinterleaver pair shown in Figures 2.5 and 2.6. The values of $v_1 \dots v_8$ are expressed in terms of $c \times a + n$ in (2.46). Note that no two scale factors a_i, a_j in these equations have indices separated by less than 64.

$$\begin{aligned} v_1 &= c_1 \times a_1 + n_1 \\ v_2 &= c_2 \times a_{65} + n_{65} \\ v_3 &= c_3 \times a_{129} + n_{129} \\ v_4 &= c_4 \times a_{193} + n_{193} \\ v_5 &= c_5 \times a_{257} + n_{257} \\ v_6 &= c_6 \times a_{321} + n_{321} \\ v_7 &= c_7 \times a_{385} + n_{385} \\ v_8 &= c_8 \times a_{449} + n_{449} \end{aligned} \tag{2.46}$$

Periodic interleaving lowers the probability of error sequences consisting of a small number of nearby terms by separating these terms so that they are multiplied by uncorrelated scale factors. However, there are other error sequences that must also be considered. Periodic interleaving does not *remove* the correlation in the fading sequence $\mathbf{a}$, it merely permutes the correlation pattern. There are exactly as many strongly correlated scale factors after interleaving as before.

Suppose that any three adjacent scale factors a_i, a_{i+1}, a_{i+2} are strongly correlated. Then an interleaver with period 8 would cause c_1, c_9, and c_{17} to be multiplied by approximately equal scale factors as shown below:

$$v_1 = c_1 \times a_1 + n_1 \tag{2.47}$$

$$v_9 = c_9 \times a_2 + n_2 \tag{2.48}$$

$$v_{17} = c_{17} \times a_3 + n_3. \tag{2.49}$$

Thus EL and PD are not quite the right metrics even when periodic interleaving is applied to correlated fading channels.

Lapidoth examined the performance of convolutional codes combined with periodic interleaving in [18]. He used a periodic version of the block erasure channel model introduced by McEliece & Stark in [19]. Figure 2.7 illustrates the block erasure channel, showing a block of 512 encoder outputs divided into subblocks of length f. In the block erasure model, each subblock is either entirely erased (so that the receiver sees no information related to the subblock) or the subblock arrives without any distortion at the receiver (i.e., $z_i = x_i$ in unerased subblocks).

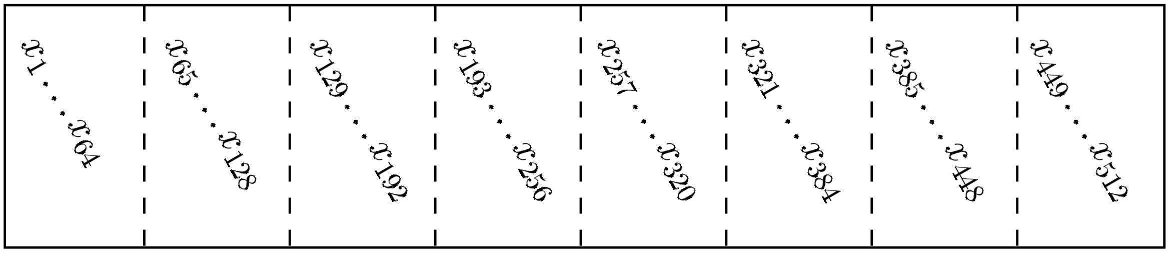


Figure 2.7: A block of 512 symbols is shown divided into subblocks of $f = 64$ symbols.

In the periodic block erasure channel, one erasure pattern describes which subblocks are erased for every block, and so the subblock erasures occur periodically. Lapidoth proposed searching for combinations of convolutional codes and interleavers that could perform without errors with the largest possible number of erased subblocks in the erasure pattern. An efficient algorithm for testing whether a code has this property is provided in Theorem 1 of [18].

Let U be the erasure pattern expressed as a sequence of bits where a 1 indicates a subblock erasure. The number of bits in U is the ratio M of the block size to the subblock size. For a periodically interleaved system, $M = P$. Define $|U|$ to be the number of erased subblocks per block (the Hamming weight of U). Lapidoth showed in [18] that error free communication of k bits per symbol with a rate k/n convolutional code is impossible if

$$|U| > \left\lfloor M \left(1 \perp \frac{k}{n} \right) \right\rfloor . \tag{2.50}$$

Simply put, when $|U|$ is large enough to satisfy (2.50) there are not enough unerased bits at the receiver to recover the entire information sequence.

The block erasure channel does not capture the full behavior of fading channels with additive noise. This dissertation develops a more detailed model — the block fading channel model — to design metrics for correlated fading. As in the previous model, the block fading model starts by dividing the transmitted sequence $\mathbf{x}$ into blocks of length f, where f depends on the Doppler frequency or coherence bandwidth of the channel. This model then includes the effects of fading and AWGN on every symbol. Once again n_i is complex AWGN. The fading scale factors now have the following behavior: $a_i = a_j$ if they are in the same block, and $a_i \perp a_j$ otherwise.

The block fading channel model captures two important characteristics common to channels with correlated fading:

1. Scale factors separated by more than f are completely uncorrelated.
2. Adjacent scale factors are strongly correlated.

The block fading channel model does not accurately reflect the gradual decrease in correlation that occurs in $R_a(k)$ as k increases. However, this effect can be safely neglected in the context of trellis codes with periodic interleaving, as explained below.

Periodic interleaving ensures that any two scale factors a_i, a_j multiplying symbols occurring even in a relatively long error sequence of $4f$ or $5f$ symbols are either within a few symbols of being adjacent or are separated by almost f symbols. Thus, the scale factors of interest for such an error sequence are either strongly correlated or almost independent as in the block fading channel model.

The $R_a(k)$ behavior neglected by the block fading channel model becomes important only when scale factors separated by, say, $f/2$ or $f/3$ symbols are involved in the same error sequence. This effect occurs only for error sequences so long that the pairwise errors are too small to warrant consideration in the design of a metric.

In effect, the block fading channel model has the independent symbol-by-symbol fading property described in Sections 2.3 and 2.4 for the supersymbols X_i defined below:

$$X_i = \begin{bmatrix} x_{t+1} & x_{t+2} & \ldots & x_{t+f} \end{bmatrix} \qquad \text{where } t = (i \perp 1)f. \tag{2.51}$$

Happily, the metrics introduced in those earlier sections have natural counterparts for the block fading channel. The EL for an error sequence on block fading channel is the cardinality of the set $A = \{i \mid X_i \neq \hat{X}_i\}$. The associated product distance is computed as

$$\text{PD} = \prod_{i \in A} \frac{\|X_i \perp \hat{X}_i\|_2^2}{\mathcal{E}_x}. \tag{2.52}$$

Recall that the block fading channel was motivated in the context of periodic interleaving, where the trellis code outputs are c_i rather than x_i. It is useful to express the metrics in terms of c_i using

$$\|X_i \perp \hat{X}_i\|_2^2 = \sum_{j=0}^{f-1} \|c_{i+jP} \perp \hat{c}_{i+jP}\|^2. \tag{2.53}$$

Note that there are P supersymbols in a block of $B = Pf$ symbols. For multicarrier modulation transmissions between a stationary transmitter and receiver, the frequency response and consequently the fading pattern $\mathbf{a}$ is the same for each block. Thus supersymbols with the same index modulo P have the same fading. In this case, the channel is modeled by periodic block fading, corresponding to the periodic block erasures studied by Lapidoth.

The effective length and product distance metrics for this case are defined in terms of the P element *periodic distance vector*

$$\begin{bmatrix} \tilde{d}_1^2 & \tilde{d}_2^2 & \ldots & \tilde{d}_P^2 \end{bmatrix}, \tag{2.54}$$

where

$$\tilde{d}_i = \mathcal{E}_x^{-1} \left(\|X_i \perp \hat{X}_i\|_2^2 + \|X_{i+P} \perp \hat{X}_{i+P}\|_2^2 + \cdots \right). \tag{2.55}$$

Using (2.53) to sum these terms yields

$$\tilde{d}_i^2 = \sum_{j=0}^{\infty} \frac{\|c_{i+jP} \perp \hat{c}_{i+jP}\|^2}{\mathcal{E}_x}. \tag{2.56}$$

Using the periodic distance vector, periodic versions of effective length and product distance are now defined.

Definition 8 *The periodic effective length* PEL *of an error sequence is the number of nonzero elements of the periodic distance vector.*

Definition 9 *The periodic effective code length* PECL *of a code is the smallest* PEL *for an error sequence associated with that code.*

Definition 10 *The periodic product distance* PPD *of an error sequence is the product of the nonzero elements of the periodic distance vector.*

Definition 11 *The code periodic product distance of order i of a code,* CPPD_i*, is the smallest* PPD *of an error sequence with* PEL$= i$.

Consider PECL in the context of the periodic block erasure channel. If an erasure pattern U causes errors, then it must hold that

$$\text{PECL} \leq |U|. \tag{2.57}$$

Otherwise, unerased nonzero elements would exist in every periodic distance vector, allowing error-free detection. From (2.50), error-free transmission is impossible when

$$|U| = \left\lfloor P\left(1 \perp \frac{k}{n}\right)\right\rfloor + 1. \tag{2.58}$$

The number of subblocks M per block in (2.50) is taken to be the interleaver period P. Using (2.58) to substitute for $|U|$ in 2.57 produces a useful bound on PECL:

$$\text{PECL} \leq \left\lfloor P\left(1 \perp \frac{k}{n}\right)\right\rfloor + 1. \tag{2.59}$$

Considering the periodic distance vector as an aliased version of the original sequence of distances, it follows that

$$\begin{align} \text{PECL} &\leq \text{ECL} \tag{2.60} \\ &\leq \left\lfloor \frac{\nu}{k} \right\rfloor + 1 \qquad \text{by (2.45).} \tag{2.61} \end{align}$$

When periodic interleaving is employed, blocks are often decoded independently. (The trellis is terminated at the end of each block.) For such block-based decoding, these periodic metrics are informative even when supersymbols X_i and X_{i+P} do not have the same fading.

2.6 Summary

This chapter began with a review of the metric of Euclidean distance for AWGN channels. For independent Rayleigh fading channels, the standard derivation of the effective length and product distance metrics was presented. The Rayleigh metrics were then motivated for use whenever fading is independent regardless of the PDF of the fading scale factors.

Channels with correlated fading were then examined. It was demonstrated that effective length and product distance are not appropriate metrics in the context of correlated fading. Instead, periodic effective length and periodic product distance are the metrics to be used to design trellis codes for periodically interleaved channels with correlated fading.

Chapter 3

Trellis Codes

Ungerboeck's famous 1982 paper [5] introduced trellis codes, which simplified the design of codes in Euclidean space, rather than in binary space. The name *trellis code* refers to the structure of the finite state machine encoder (also used for maximum likelihood sequence decoding of such a code via the Viterbi algorithm [20]).

The standard trellis code encoder, shown in Figure 3.1, comprises a convolutional encoder followed by a signal mapper. The rate k/n convolutional encoder maps sequences of k-bit symbols b_i to sequences of n-bit symbols. The signal mapper maps the n-bit symbols to points x_k from an alphabet of 2^n complex values. The set of 2^n points that constitute this alphabet is known as a *constellation*.

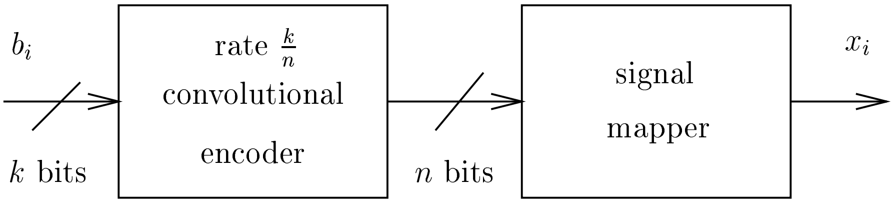


Figure 3.1: Encoder for a Trellis Code.

The design of a trellis code involves selecting a labeled constellation and convolutional encoder to maximize some subset of the metrics presented in Chapter 2. The labeling for the constellation describes how the signal mapper maps convolutional encoder outputs to constellation points.

This chapter presents the background material necessary to discuss these design choices in detail. An important tool in the search for trellis codes is the ability determine when different trellis codes have equivalent performance. This chapter concludes with an exploration of how different convolutional codes can lead to equivalent trellis codes.

Section 3.1 introduces convolutional codes. Section 3.2 explores convolutional codes in the context of the vector space of output symbols. This representation will be useful in Section 3.6 and in the next chapter for demonstrating how convolutional codes and constellations that look very different can be components of equivalent trellis codes.

Section 3.3 explores the error sequences produced by maximum likelihood sequence detection performed on a convolutional code. Section 3.4 gives an example of how a convolutional code error sequence maps to a symbol error sequence in a trellis code. This section also defines for trellis codes a “worst case” distance sequence corresponding to each error sequence of the underlying convolutional code.

Section 3.5 defines three ways that trellis codes can be considered equivalent. The weakest definition is *distance equivalence*, but even distance equivalent codes are identical in the metrics introduced in Chapter 4. Only one code in a set of distance equivalent codes needs to be evaluated in a code search program based on these metrics. Section 3.6 shows how certain changes of basis in the vector space introduced in Section 3.2 always produce distance equivalent codes.

3.1 Convolutional Codes

As shown in Figure 3.1, every trellis code has a convolutional code as a major component. As a result, the structure of trellis codes is closely related to that of convolutional codes. This study of trellis codes begins with an introduction to convolutional codes.

The central reference for convolutional codes used in this dissertation is the 1970 paper by Forney [21]. The following definition paraphrases the one in [21] and specializes to the binary field.

Definition 12 *A rate k/n convolutional encoder is a time invariant finite state machine with k-bit inputs and n-bit outputs.*

Without loss of generality, the following discussion is concerned only with convolutional encoders that do not employ feedback. The justification for this restriction is given in Section 3.5.

The finite state machine of a feedback free encoder is simply a feedback free shift register. Figure 3.2 shows a feedback free rate 1/4 convolutional code with 6 memory elements. Addition in the figure is modulo 2 (the exclusive-or operation). The input bits b_i pass through a shift register with 6 memory elements. The four output bit sequences $c_{i,1}$, $c_{i,2}$, $c_{i,3}$, and $c_{i,4}$ are modulo 2 sums of selected subsets of the current and 6 previous input bit values.

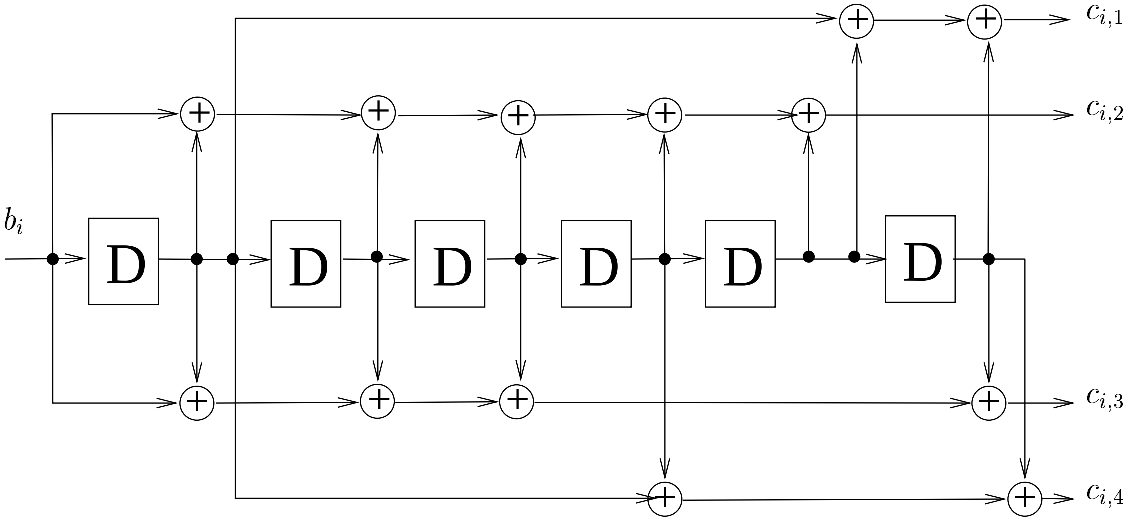


Figure 3.2: A feedback free rate 1/4 convolutional encoder.

Convolutional codes such as the one whose encoder is shown in Figure 3.2 are often described by the bit patterns that identify the subsets of b_i values to be summed. The four patterns for Figure 3.2 are the rows of the following matrix:

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}. \tag{3.1}$$

The D transform of the sequence $\{b_i\}$ is defined to be $b(D) = \sum b_i D^i$. Sequences are referred to in terms of their D transforms. The relationship between an input sequence $b(D)$ and output bit sequence $c_i(D)$ may be described in terms of the generator polynomial $g_i(D)$ by the equation:

$$c_i(D) = b(D)g_i(D). \tag{3.2}$$

The generator polynomial $g_i(D)$ for the rate 1/4 encoder shown in Figure 3.2 is the D transform of the i^{th} row of (3.1). The four generator polynomials for this encoder are shown below:

$$g_1(D) = D + D^5 + D^6 \tag{3.3}$$

$$g_2(D) = 1 + D + D^2 + D^3 + D^4 + D^5 \tag{3.4}$$

$$g_3(D) = 1 + D + D^2 + D^3 + D^6 \tag{3.5}$$

$$g_4(D) = D + D^4 + D^6. \tag{3.6}$$

A convolutional encoder is often described by the *encoder matrix* whose entries are the generator polynomials. The encoder matrix for the rate 1/4 encoder shown in Figure 3.2 is

$$\begin{bmatrix} g_1(D) & g_2(D) & g_3(D) & g_4(D) \end{bmatrix}^{\mathrm{T}}, \tag{3.7}$$

where the values of $g_1(D), \ldots, g_4(D)$ are those given in (3.3)–(3.6). A rate k/n encoder would have k rows in its encoder matrix.

Often, convolutional codes are identified by octal representations of the coefficients of the polynomials in the generator matrix. The rate 1/4 encoder shown in Figure 3.2 whose polynomial coefficients are the rows of (3.1) is identified by the four octal numbers

$$043 \quad 176 \quad 171 \quad 045. \tag{3.8}$$

For consistency with the notation commonly used in the literature, this method will be used throughout the dissertation to identify convolutional codes.

3.2 An Important Vector Space

For any rate k/n convolutional code, consider the n-dimensional vector space $\mathcal{S}$ composed of all the n-bit vectors. ($\mathcal{S}$ is also an additive group in Z_2^n.) The scalars of this vector space are bits. Addition in $\mathcal{S}$ is the componentwise exclusive-or of the vectors being added. Addition and subtraction are equivalent in $\mathcal{S}$ since each vector is its own additive inverse. Vectors in $\mathcal{S}$ will be denoted by boldface letters, eg. $\mathbf{v}_i$.

The convolutional encoder in Figure 3.2 is a mapping from input bit sequences $b(D)$ to output sequences $\mathbf{C}(D)$ whose elements are in $\mathcal{S}$. The four $g(D)$ polynomials can be combined into one polynomial $\mathbf{G}(D)$ whose coefficients are in $\mathcal{S}$. The rate $1/n$ encoder equation is then expressed concisely as follows:

$$\mathbf{C}(D) = \begin{bmatrix} c_1(D) \\ c_2(D) \\ c_3(D) \\ c_4(D) \end{bmatrix} = b(D)\mathbf{G}(D). \tag{3.9}$$

For the rate 1/4 encoder shown in Figure 3.2

$$\mathbf{G}(D) = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} D + \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} D^2 + \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} D^3 + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} D^4 + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} D^5 + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} D^6. \tag{3.10}$$

In this representation of a feedforward rate $1/n$ convolutional code, each output vector $\mathbf{C}_i$ is a linear combination of coefficients of $\mathbf{G}(D)$. For a general rate k/n convolutional code, each $\mathbf{C}_i$ is still a linear combination of vectors, but the vectors are now the coefficients of the k polynomials $\mathbf{G}_1(D)$, $\mathbf{G}_2(D)$, ... , $\mathbf{G}_k(D)$:

$$\mathbf{C}(D) = b_1(D)\mathbf{G}_1(D) + b_2(D)\mathbf{G}_2(D) + \ldots, +b_k(D)\mathbf{G}_k(D). \tag{3.11}$$

In this way, a rate k/n convolutional encoder is the sum (in $\mathcal{S}$) of the outputs of k rate $1/n$ encoders.

3.3 Error Sequences

In this dissertation, decoding is assumed to be maximum likelihood sequence detection. Consequently, the sequences chosen by the receiver are always valid output sequences (i.e., sequences that could have been produced by the encoder).

Suppose that a rate k/n convolutional encoder produces the output sequence $\mathbf{C}(D)$ from input sequences $b_1(D), \ldots, b_k(D)$, but the receiver mistakenly chooses $\hat{\mathbf{C}}(D)$, as though the input sequences were $\hat{b}_1(D), \ldots, \hat{b}_k(D)$. The output error sequence $\mathbf{E}(D) = \mathbf{C}(D) - \hat{\mathbf{C}}(D)$ is computed through the element by element vector subtraction (in $\mathcal{S}$) of $\mathbf{C}(D)$ and $\hat{\mathbf{C}}(D)$.

Each output error sequence $\mathbf{E}(D)$ corresponds to k input bit error sequences, $e_1(D), \ldots, e_k(D)$. For a particular pair of output sequences $(\mathbf{C}(D), \hat{\mathbf{C}}(D))$,

$$\begin{aligned}
\mathbf{E}(D) &= \mathbf{C}(D) - \hat{\mathbf{C}}(D) &\quad (3.12)\\
&= \sum_{j=1}^{k} b_j(D)\mathbf{G}_j(D) - \sum_{j=1}^{k} \hat{b}_j(D)\mathbf{G}_j(D) &\quad (3.13)\\
&= \sum_{j=1}^{k} (b_j(D) - \hat{b}_j(D))\mathbf{G}_j(D) &\quad (3.14)\\
&= \sum_{j=1}^{k} e_j(D)\mathbf{G}_j(D). &\quad (3.15)
\end{aligned}$$

Note that many combinations of $b_1(D), \ldots, b_k(D)$ and $\hat{b}_1(D), \ldots, \hat{b}_k(D)$ produce the same input error sequences $e_1(D), \ldots, e_k(D)$ and thus the same output error sequence $\mathbf{E}(D)$.

Every output error sequence is a valid encoder output sequence; it is the output sequence produced by the input bit sequences $e_1(D), \ldots, e_k(D)$. Conversely, every valid output sequence is also an error sequence. To see this, select any valid output sequence

$$\mathbf{C}(D) = \sum_{j=1}^{k} b_j(D)\mathbf{G}_j(D), \quad (3.16)$$

and take $\hat{b}_1(D), \ldots, \hat{b}_k(D)$ to be all zeros sequences. Then

$$\begin{aligned} \mathbf{C}(D) &= \sum_{j=1}^{k} b_j(D)\mathbf{G}_j(D) && (3.17) \\ &= \sum_{j=1}^{k} (b_j(D) - \hat{b}_j(D))\mathbf{G}_j(D) && (3.18) \\ &= \sum_{j=1}^{k} e_j(D)\mathbf{G}_j(D) && (3.19) \\ &= \mathbf{E}(D). && (3.20) \end{aligned}$$

3.4 Distance Sequences

When a convolutional code is used as part of a trellis code, each nonzero vector in the error sequence $\mathbf{E}(D)$ corresponds to the selection of an incorrect constellation point (i.e., a symbol error). Consider the following example of a symbol error sequence for a trellis code using the rate 1/4 encoder in Figure 3.2.

Let the correct input bit sequence $b(D)$ be all zeros, which produces the all zero output sequence $\mathbf{C}(D) = 0$. Suppose that $\hat{b}(D)$ has an incorrect first bit ($b_0 = 1$) but all zeros thereafter. Thus $\hat{b}(D) = 1$ and the incorrectly chosen output sequence is

$$\hat{\mathbf{C}}(D) = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} D + \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} D^2 + \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} D^3 + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} D^4 + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} D^5 + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} D^6. \tag{3.21}$$

With output vectors $\mathbf{C}_i$ mapped to constellation points using the labeled 16-QAM constellation shown in Figure 3.3, Figure 3.4 shows the first seven constellation points or symbols produced by $b(D)$ (top row) and $\hat{b}(D)$ (bottom row). All the other constellation points produced by $b(D)$ and $\hat{b}(D)$ are identical.

For any specified transmitted and received sequences $b(D)$ and $\hat{b}(D)$, the squared Euclidean distance between the correct and incorrect i^{th} symbol will be referred to as $d_i^2(b \to \hat{b})$. Table 3.1 shows the Euclidean distances for the example shown in Figure 3.4, assuming that nearest neighbors in the constellation are separated by a

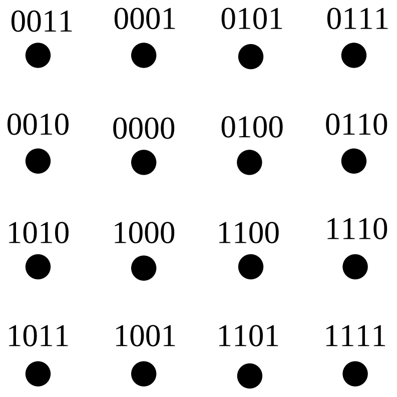


Figure 3.3: 16-QAM Constellation with Gray coding.

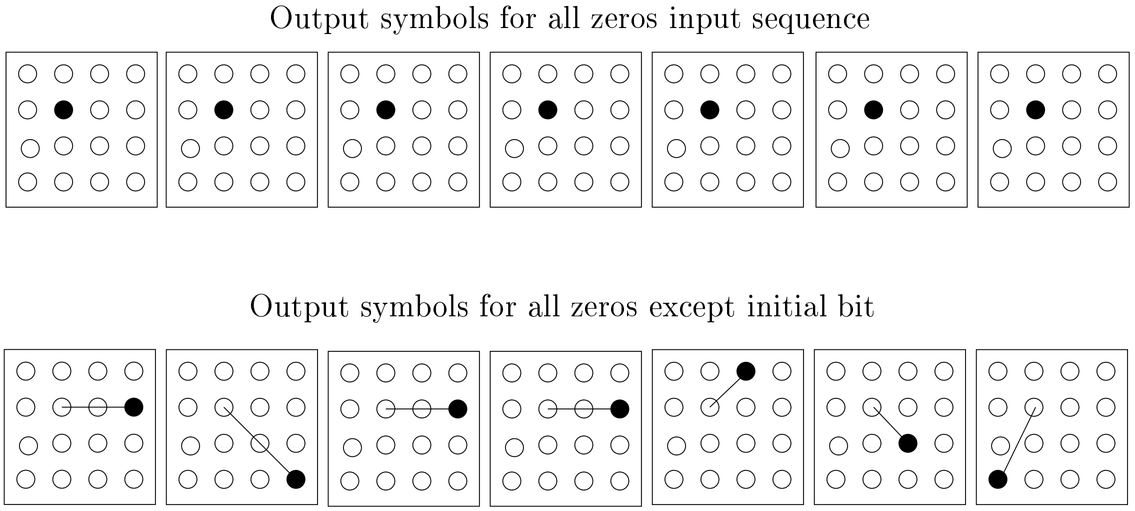

Output symbols for all zeros input sequence
Output symbols for all zeros except initial bit

Figure 3.4: A trellis code error event. Correct output symbols (top) and incorrect output symbols (bottom). The bottom sequence shows the symbol errors with respect to the top sequence as line segments.

i	0	1	2	3	4	5	6
$d_i^2(b \to \hat{b})$	4	8	4	4	2	2	5

Table 3.1: Euclidean distances for symbol errors in Figure 3.4.

distance of 1.

The terminology required to demonstrate the validity of (2.45) in the previous chapter has now been introduced. The result is restated below as a theorem.

Theorem 1 *The effective code length (see Definition 5 on page 13) of a trellis code T consisting of a feed-forward rate k/n convolutional encoder with ν memory elements must satisfy the following bound:*

$$\text{ECL} \leq \left\lfloor \frac{\nu}{k} \right\rfloor + 1. \tag{3.22}$$

Proof: The feed forward convolutional encoder employs a separate shift register for each input bit sequence $b_j(D)$ $(j = 1, \ldots, k)$. Identify an index $\check{j}$ corresponding to a shift register with the least number of memory elements. This shift register can have at most $\lfloor \nu/k \rfloor$ memory elements.

Consider a transmitted sequence in which each $b_i(D)$ is the all zeros sequence. For $j \neq \check{j}$ let $\hat{b}_j(D)$ also be the all zeros sequence Let the sequence $\hat{b}_{\check{j}}(D)$ be all zeros except an initial one. For this error sequence, $d_i^2(b \rightarrow \hat{b})$ is only nonzero for the initial symbol and the $\lfloor \nu/k \rfloor$ following symbols during which the initial 1 in $\hat{b}_{\check{j}}(D)$ passes through the shift register. Thus the effective code length cannot be larger than $\lfloor \nu/k \rfloor + 1$ and the theorem is proved. □

The sequence of distances depends on the encoder output sequence $\mathbf{C}(D)$ as well as the error sequence $\mathbf{E}(D)$. A particular error vector E can lead to different Euclidean distances depending on the constellation point that was transmitted. Consider the error vector $\mathbf{E}_4$ of the example discussed above (the coefficient of D^4 in (3.21)):

$$\mathbf{E}_4 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}. \tag{3.23}$$

The symbol errors associated with this error vector are shown in Figure 3.5. Half of the constellation points have $d^2 = 2$ as in the example. The other half have $d^2 = 10$.

The symbol errors will also be referred to as edges. The error vector that produces

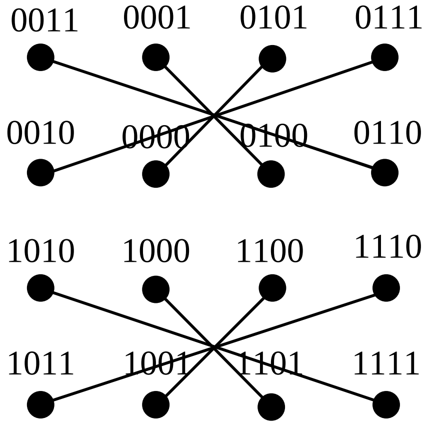


Figure 3.5: Symbol errors (edges) associated with the error vector (edge label) $[0101]^{\mathrm{T}}$.

a symbol error will be referred to as the label of that edge. Note that the mapping of an error vector to its associated symbol errors (or a label to its edges) depends intimately on the choice of constellation labeling. The next chapter explores how constellation labeling affects the mapping of labels to edges.

As demonstrated in Figure 3.5, a particular edge label can correspond to more than one distance. Thus an error sequence $\mathbf{E}(D)$ can correspond to several distinct distance sequences $\{d_i^2(b \to \hat{b})\}$ for different pairs $(b, \hat{b})$. The metrics presented in Chapter 2 are computed from the distance sequences, but only the worst distance sequence (the one producing the lowest metric value) for each error sequence is of interest. It is useful to define a worst case distance sequence for each $\mathbf{E}(D)$ by selecting the smallest possible distances for each error vector $\mathbf{E}_i$.

Definition 13 *For a specified labeled constellation, $d^2_{\mathrm{MIN}}(E)$ is the smallest squared Euclidean distance of an edge with label E.*

Note that d^2_{MIN} is a Euclidean distance, not a Hamming distance. It is the distance associated with one element of a sequence and not the entire sequence.

Definition 14 *For a specified labeled constellation and error sequence $\mathbf{E}(D)$,*

$$d_i^2(\mathbf{E}(D)) \triangleq d^2_{\mathrm{MIN}}(\mathbf{E}_i) \tag{3.24}$$

The term *distance sequence* always refers to $\{d_i^2(\mathbf{E}(D))\}$ as described in Definition 14 unless it is specifically stated to refer to $\{d_i^2(b \to \hat{b})\}$. (Note that Appendix A provides a list of all formal definitions.) The above definition allows code evaluation to restrict attention to the possible sequences $\mathbf{E}(D)$ rather than consider all possible combinations of $(b(D),\, \hat{b}(D))$. However, it is possible that this definition is too pessimistic. Perhaps the distance sequence $\{d_i^2(\mathbf{E}(D))\}$ is worse than any actual distance sequence $\{d_i^2(b \to \hat{b})\}$.

The question of whether or not $\{d_i^2(\mathbf{E}(D))\}$ is pessimistic is equivalent to asking whether for every $\mathbf{E}(D)$ there is a pair $(b(D),\, \hat{b}(D))$ such that

$$\mathbf{E}(D) = \sum_{j=1}^{k}(b_j(D) - \hat{b}_j(D))\mathbf{G}_j(D) \tag{3.25}$$

and

$$d_i^2(b \to \hat{b}) = d_i^2(\mathbf{E}(D)) \quad \text{for all } i. \tag{3.26}$$

For rate $k/(k+1)$ convolutional codes used with Ungerboeck labeling, Ungerboeck showed in [5] that such a a pair $(b(D),\hat{b}(D))$ exists for every $\mathbf{E}(D)$. However, Ungerboeck's proof is specific to rate $k/(k+1)$ codes and his labeling scheme. In general, such a pair may not exist. For certain code rates and labeled constellations there may be pathological sequences $\mathbf{E}(D)$ for which no pair $(b(D),\, \hat{b}(D))$ satisfies conditions (3.25) and (3.26). Such pathological cases do not exist for the rates and labeled constellations used in the code searches of Chapter 6.

When such pathological cases do exist, there is a risk associated with this pessimistic definition of distance sequence. A code search based on this definition might not find the best code available. A better code might exist whose performance is hidden by the pessimistic definition of distance sequences.

Ungerboeck's code searches used a still more pessimistic definition of distance sequence based on the structure of set partitioning. With this definition, there are sequences $\mathbf{E}(D)$ for which no pair $(b(D),\, \hat{b}(D))$ exists that satisfies conditions (3.25) and (3.26). Those searches produced excellent codes, but Ungerboeck's search procedure did not guarantee that there do not exist better codes at the same complexity.

3.5 Equivalent Codes

As mentioned in the previous section, convolutional codes and trellis codes are found through a computationally intensive exhaustive search. Papers listing the results of code searches often have a row or two marked with an asterisk indicating that the code listed was the best code found, but that the search was too computationally intensive to be completed.

These code searches are based on metrics computed entirely from the distance sequence defined in the previous section. Thus an efficient search should evaluate only one code from a group of codes that all have the same distance sequences. Below, three ways that codes can be considered equivalent are defined. The weakest form of equivalence corresponds to codes that have the same distance vectors.

The strongest form of equivalence is presented first. The definition applies both to convolutional codes and trellis codes.

Definition 15 *Two codes are strictly equivalent if they have the same mapping of input sequences to output sequences.*

Two strictly equivalent codes amount to different implementations of exactly the same encoder. Strictly equivalent codes always have the same probabilities of event error and bit error.

Note that two trellis codes can be strictly equivalent even if their respective convolutional codes are not, since differently labeled constellations can make different convolutional codes produce the same constellation points.

In [21], two convolutional codes are considered equivalent if they have the same range of possible output sequences. This form of equivalence is referred to as range equivalence in this dissertation.

Definition 16 *Two codes are range equivalent if they have the same set of possible output sequences.*

Definition 16 applies to trellis codes as well as convolutional codes. Range equivalence preserves the space of error events and thus the probability of an error event. However, two range equivalent codes may not have the same bit error probability since the mapping of information bit sequences to output sequences is not necessarily preserved.

Of most interest to the code searches in this thesis is distance equivalence. This form of equivalence is the weakest of the three.

Definition 17 *Two trellis codes are distance equivalent if they have the same set of distance sequences* $\{d_i^2(\mathbf{E}(D))\}$.

Two distance equivalent codes need not have the same bit error probability or even the same error event probabilities. However, as long as for every $\mathbf{E}(D)$ there is a pair $(b(D),\, \hat{b}(D))$ that satisfies conditions (3.25) and (3.26), distance equivalent codes have the same values for the metrics of Chapter 2. This fact makes distance equivalence a valuable tool for code searches based on those metrics.

Of course not all codes are distance equivalent. In certain cases the better code among two codes that are not distance equivalent is easily identified. In these cases the better code is called *distance superior* as defined below:

Definition 18 *Trellis code* T_1 *is distance superior to trellis code* T_2 *if the distances sequences of* T_1 *can be paired (one to one) with those of* T_2 *such that each term in every distance sequence of* T_1 *is greater than or equal to the corresponding term in the paired sequence of* T_2*. There must be at least one strict inequality, otherwise the trellis codes are distance equivalent.*

If T_1 is distance superior to T_2 it will have larger (better) values for all the code design metrics that will be introduced in Chapter 5.

For trellis codes, strict equivalence implies range equivalence, and range equivalence implies distance equivalence. This section concludes with three statements about range equivalent codes that justify the restriction to feedback free convolutional codes made in Section 3.1.

Theorem 2 *Two trellis codes are range equivalent if they use the same labeled constellation and have range equivalent convolutional codes.*

Proof: The equivalent sets of vector outputs $\{\mathbf{C}(D)\}$ must yield equivalent sequences of constellation points since the same labeled constellation is used. □

Theorem 3 *Every convolutional code is range equivalent to a feedback free code.*

This is a central result of [21], to which the reader is referred for the proof. Forney also shows in [21] that for any given convolutional code one can always find a range equivalent feedback free code that uses as few or fewer memory elements.

Corollary 1 *Every trellis code is range equivalent to a trellis code using the same constellation and a feedback free convolutional code.*

This corollary follows immediately from the two previous theorems. It explains why attention could be restricted to feedback free encoders in Section 3.1 without loss of generality.

3.6 Swapping Basis Vectors

For a fixed constellation labeling, certain operations on the convolutional encoder polynomials always produce a trellis code that is distance equivalent to the original. Identifying such operations and excluding distance equivalent codes appropriately can reduce the duration of a code search significantly.

Consider a trellis code with a labeled constellation and a rate k/n feedback free convolutional code. The edge labels and the coefficients of $\mathbf{G}_1(D)$, ..., $\mathbf{G}_k(D)$ are both elements of the n-dimensional vector space $\mathcal{S}$ discussed in Section 3.2.

This section examines what happens when the coefficients of $\mathbf{G}_1(D)$, ..., $\mathbf{G}_k(D)$ are replaced with a different set of coefficients through a bijective (one-to-one and onto) mapping $F : \mathcal{S} \to \mathcal{S}$. The goal is to identify which mappings F produce distance equivalent codes so that the distance equivalent codes can be eliminated from the code search.

F is assumed to be a bijective mapping $F : \mathcal{S} \to \mathcal{S}$ throughout this section. Mappings that implement a change of basis are considered first. Let $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ and $\{\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_n\}$ be bases for $\mathcal{S}$.

Definition 19 *The mapping F implements a change of basis from $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ to $\{\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_n\}$ if for any $\mathbf{s} \in \mathcal{S}$ and any subset A of $\{1, 2, \ldots, n\}$:*

$$F(\mathbf{s}) = \sum_{i \in A} \hat{\mathbf{v}}_i \qquad \textit{when} \qquad \mathbf{s} = \sum_{i \in A} \mathbf{v}_i. \tag{3.27}$$

Lemma 1 *The mapping F implements a change of basis $\Longleftrightarrow$ F obeys the additive property for $\mathbf{s}_1, \mathbf{s}_2 \in \mathcal{S}$:*

$$F(\mathbf{s}_1 + \mathbf{s}_2) = F(\mathbf{s}_1) + F(\mathbf{s}_2). \tag{3.28}$$

Proof: Let

$$\mathbf{s}_1 = \sum_{i \in A_1} \mathbf{v}_i \qquad \text{and} \qquad \mathbf{s}_2 = \sum_{i \in A_2} \mathbf{v}_i \tag{3.29}$$

where A_1 and A_2 be subsets of $\{1, 2, \ldots, n\}$. Define

$$A_1 \oplus A_2 \triangleq (A_1 \bigcup A_2) - (A_1 \bigcap A_2). \tag{3.30}$$

If F implements a change of basis as defined above, then:

$$F(\mathbf{s}_1 + \mathbf{s}_2) = F\left(\sum_{i \in A_1} \mathbf{v}_i + \sum_{i \in A_2} \mathbf{v}_i\right) \tag{3.31}$$

$$= F\left(\sum_{i \in A_1 \oplus A_2} \mathbf{v}_i\right) \qquad \text{since } \mathbf{v}_i + \mathbf{v}_i = 0 \tag{3.32}$$

$$= \sum_{i \in A_1 \oplus A_2} \hat{\mathbf{v}}_i \qquad \text{by Definition 19} \tag{3.33}$$

$$= \sum_{i \in A_1} \hat{\mathbf{v}}_i + \sum_{i \in A_2} \hat{\mathbf{v}}_i \qquad \text{since } \hat{\mathbf{v}}_i + \hat{\mathbf{v}}_i = 0 \tag{3.34}$$

$$= F\left(\sum_{i \in A_1} \mathbf{v}_i\right) + F\left(\sum_{i \in A_2} \mathbf{v}_i\right) \qquad \text{by Definition 19} \tag{3.35}$$

$$= F(\mathbf{s}_1) + F(\mathbf{s}_2) \qquad \text{proving } \Longrightarrow . \tag{3.36}$$

Now suppose that F obeys the additive property (3.28). Choose any basis $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ and define $\hat{\mathbf{v}}_i = F(\mathbf{v}_i)$. Choose A to be any subset of $\{1, 2, \ldots, n\}$ and let

$$\mathbf{s} = \sum_{i \in A} \mathbf{v}_i. \tag{3.37}$$

$$\begin{aligned} F(s) &= F\left(\sum_{i \in A} \mathbf{v}_i\right) && && (3.38) \\ &= \sum_{i \in A} F(\mathbf{v}_i) && \text{by the additive property} && (3.39) \\ &= \sum_{i \in A} \hat{\mathbf{v}}_i && \text{by definition of } \hat{\mathbf{v}}_i && (3.40) \end{aligned}$$

This proves the $\Longleftarrow$ direction. □

Now consider using F to map one trellis code to another. The constellation labeling remains the same, but F is used to change the coefficients of the convolutional encoder polynomials. For a given labeled constellation, certain mappings always transform a trellis code into a different trellis code that is distance equivalent to the original. The next theorem shows that these special mappings must implement a change of basis.

Theorem 4 *If the bijective mapping $F : \mathcal{S} \to \mathcal{S}$ applied to the encoder polynomial coefficients of any trellis code with a given labeled constellation always produces a trellis code that is distance equivalent to the original, then F implements a change of basis.*

Proof: Suppose that F does not implement a change of basis. Then by Lemma 1 the additive property does not hold. Thus there are three distinct nonzero vectors $\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3 \in \mathcal{S}$ such that

$$\mathbf{s}_1 + \mathbf{s}_2 = \mathbf{s}_3 \tag{3.41}$$

$$F(\mathbf{s}_1) + F(\mathbf{s}_2) \neq F(\mathbf{s}_3). \tag{3.42}$$

$F(\mathbf{s}_1), F(\mathbf{s}_2),$ and $F(\mathbf{s}_3)$ must be distinct because F is bijective and must be nonzero or F trivially does not preserve distance sequences.

Consider the rate $1/n$ convolutional code $\mathbf{G}(D) = \mathbf{s}_3 D^2 + \mathbf{s}_2 D + \mathbf{s}_1$ and apply F to its coefficients to produce $\hat{\mathbf{G}}(D) = F(\mathbf{s}_3)D^2 + F(\mathbf{s}_2)D + F(\mathbf{s}_1)$. Now consider all error sequences $\mathbf{E}(D)$ for which $d_i^2(\mathbf{E}(D))$ is zero for i outside the index range $i = 0, \ldots, 4$ and is nonzero for the indices 0 and 4:

$$d_0^2 > 0 \tag{3.43}$$

$$d_4^2 > 0 \tag{3.44}$$

$$d_i^2 = 0 \text{ for } i > 4 \tag{3.45}$$

$$d_i^2 = 0 \text{ for } i < 0. \tag{3.46}$$

There are only two such sequences, shown below for the two codes $\mathbf{G}(D)$ and $\hat{\mathbf{G}}(D)$.

$$\begin{aligned}\mathbf{E}^{(1)}(D) &= (D^2 + D + 1)\mathbf{G}(D) &(3.47)\\ &= \mathbf{s}_3 D^4 + (\mathbf{s}_2 + \mathbf{s}_3)D^3 + (\mathbf{s}_1 + \mathbf{s}_2 + \mathbf{s}_3)D^2 + (\mathbf{s}_1 + \mathbf{s}_2)D + \mathbf{s}_1 &\\ \hat{\mathbf{E}}^{(1)}(D) &= (D^2 + D + 1)\hat{\mathbf{G}}(D) &(3.48)\\ &= F(\mathbf{s}_3)D^4 + (F(\mathbf{s}_2) + F(\mathbf{s}_3))\,D^3 + (F(\mathbf{s}_1) + F(\mathbf{s}_2) + F(\mathbf{s}_3))\,D^2 &\\ &\qquad + (F(\mathbf{s}_1) + F(\mathbf{s}_2))\,D + F(\mathbf{s}_1) &\end{aligned}$$

$$\begin{aligned}\mathbf{E}^{(2)}(D) &= (D^2 + 1)\mathbf{G}(D) &(3.49)\\ &= \mathbf{s}_3 D^4 + \mathbf{s}_2 D^3 + (\mathbf{s}_1 + \mathbf{s}_3)D^2 + \mathbf{s}_2 D + \mathbf{s}_1 &\\ \hat{\mathbf{E}}^{(2)}(D) &= (D^2 + 1)\hat{\mathbf{G}}(D) &(3.50)\\ &= F(\mathbf{s}_3)D^4 + F(\mathbf{s}_2)D^3 + (F(\mathbf{s}_1) + F(\mathbf{s}_3))\,D^2 + F(\mathbf{s}_2)D + F(\mathbf{s}_1) &\end{aligned}$$

The D^2 term of $\mathbf{E}^{(1)}(D)$ must be zero by (3.41). Thus $d_2^2(\mathbf{E}^{(1)}(D)) = 0$. However, the D^2 term of $\hat{\mathbf{E}}^{(1)}(D)$ must be nonzero by (3.42). Thus $d_2^2(\hat{\mathbf{E}}^{(1)}(D)) \neq 0$. The D^2 term of $\hat{\mathbf{E}}^{(2)}(D)$ is also nonzero since $F(\mathbf{s}_1)$ and $F(\mathbf{s}_3)$ are distinct. Thus $\mathbf{G}(D)$ has a distance sequence that satisfies (3.43)–(3.46) and has $d_2^2 = 0$, but $\hat{\mathbf{G}}(D)$ does not. Thus the codes are not distance equivalent, and the theorem is proved. □

So the only mappings that can always produce distance equivalent codes are those that implement a change of basis. Note that the choice of initial basis is not important. Since the choice of basis $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ in Lemma 1 was arbitrary, all mappings F that implement a change of basis are found by selecting one arbitrary initial basis and considering only changes of basis with respect to that initial basis.

The exercise of checking all possible changes of basis might appear tedious. However, when the edge labels and their corresponding minimum distances are considered, relatively few mappings F need to be considered. A procedure for efficiently identifying the mappings of interest is presented below, and the mappings are found for the example of the 16-QAM constellation in Figure 3.3 on page 34.

For QAM and PSK constellations, every point can be reached from every other point by traversing one or more minimum d^2_{MIN} edges. Thus a basis must exist among the labels of these edges. For example, all Gray coded constellations (including the constellation in Figure 3.3) have a basis formed by the Hamming weight one edge labels, which are the only minimum d^2_{MIN} edge labels for Gray coded constellations.

Choose n linearly independent edge labels corresponding to minimum d^2_{MIN} edges to be the basis $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$. The only vectors that can replace these vectors in a change of basis that produces distance equivalent codes must be labels of minimum d^2_{MIN} edges. Otherwise, $\mathbf{G}(D)$ with exactly one nonzero coefficient equal to an $\mathbf{v}_i$ mapping to a larger d^2_{MIN} is an example where the mapping does not yield a distance equivalent code.

The restriction to minimum distance edge labels greatly reduces the number of possible bases that must be considered. In fact, typical good labelings such as Gray coding and many Ungerboeck labelings will have exactly n minimum d^2_{MIN} labels, forcing F to be a permutation of $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$. Each candidate basis $\{\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_n\}$ is checked using the following theorem.

Theorem 5 *A bijective mapping $F : \mathcal{S} \to \mathcal{S}$ that changes bases from $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ to $\{\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_n\}$ always produces a distance equivalent code $\Longleftrightarrow$*

$$d^2_{\text{MIN}}\left(\sum_{i \in A} \mathbf{v}_i\right) = d^2_{\text{MIN}}\left(\sum_{i \in A} \hat{\mathbf{v}}_i\right) \tag{3.51}$$

for every subset A of $\{1, 2, \ldots, n\}$.

Proof: If (3.51) does not hold, there is a vector $s \in \mathcal{S}$ such that $d^2_{\text{MIN}}(s) \neq d^2_{\text{MIN}}(F(s))$. Then the rate $1/n$ code $\mathbf{G}(D) = s$ is not distance equivalent to $\mathbf{G}(D) = F(s)$, proving the $\Longrightarrow$ direction.

Because the mapping F is a change of basis, it obeys the additive property (3.28). Thus applying F to the coefficients of the convolutional code polynomials is equivalent to applying F to the outputs of the convolutional encoder.

Recall that the set of error sequences is the same as the set of valid encoder output sequences. Thus the error sequences for the code obtained by the application of F are found by applying F to the original error sequences. If (3.51) holds, then these new error sequences map to the same distance sequences as the original error sequences. Thus the codes are distance equivalent and the $\Longleftarrow$ direction of the theorem is proved. □

In summary, the following procedure identifies all mappings F that always produce distance equivalent codes when applied to a trellis code with a specified labeled constellation:

1. Identify the edge labels of the minimum d^2_{MIN} edges.
2. Choose a basis for $\mathcal{S}$ from among these labels.
3. Use Theorem 5 to check all possible bases comprising edge labels identified in step 1.

To clarify how this procedure works, it is applied to the labeled 16-QAM constellation in Figure 3.3. Because the constellation is Gray coded, the only minimum distance edge labels are those containing three 0s and one 1. There are exactly enough minimum distance edge labels to form a basis. Select the following basis:

$$\mathbf{v}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \qquad \mathbf{v}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \qquad \mathbf{v}_3 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \qquad \mathbf{v}_4 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{3.52}$$

Consider swapping $\mathbf{v}_2$ and $\mathbf{v}_3$. That is, F maps $\mathbf{v}_i$ to $\hat{\mathbf{v}}_i$ with

$$\hat{\mathbf{v}}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \qquad \hat{\mathbf{v}}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \qquad \hat{\mathbf{v}}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \qquad \hat{\mathbf{v}}_4 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{3.53}$$

The minimum squared distances associated with the edge labels expressed in terms of the two bases are given in Table 3.2. As shown in the table, the condition of Theorem 5 is satisfied and thus the mapping from the basis in (3.52) to the basis in (3.53) always produces distance equivalent codes when applied to the trellis code in Figure 3.3.

There are $4! = 24$ possible permutations of the four original basis vectors (including the identity). Of these mappings, eight always produce distance equivalent codes. These eight mappings consist of any combination of the following three operations:

1. Swap $\mathbf{v}_1$ for $\mathbf{v}_4$.

2. Swap $\mathbf{v}_2$ for $\mathbf{v}_3$.

3. Swap $\mathbf{v}_1$ for $\mathbf{v}_2$, and swap $\mathbf{v}_3$ for $\mathbf{v}_4$.

Let's apply these operations to the encoder of Figure 3.2. These operations are all equivalent to permuting rows of (3.1). Thus the eight distance equivalent codes are simply eight of the 24 permutations of the the octal words in (3.8). These eight permutations are shown in Table 3.3.

Note that the 16 remaining permutations consist of two groups of eight distance equivalent codes. A rate 1/4 code search using the Gray coded 16-QAM constellation can restrict its attention to three of the 24 permutations of any set of four code polynomials $g_1(D), \ldots, g_2(D)$. Thus code search complexity is reduced by a factor of 8.

The Ungerboeck labeled 16-QAM constellation in [5] also has eight mappings that maintain distance equivalence. However, they cannot be expressed as simple permutations of code polynomials $g_1(D), \ldots, g_2(D)$. This is because the vectors having exactly one bit cannot be swapped since they have different values of d_{MIN}. The four minimum distance edge labels (the swappable basis vectors) are given below:

$$\mathbf{v}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \qquad \mathbf{v}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \qquad \mathbf{v}_3 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \qquad \mathbf{v}_4 = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}. \tag{3.54}$$

E	Original Basis	$d^2_{\text{MIN}}(E)$	$d^2_{\text{MIN}}(\hat{\mathbf{E}})$	New Basis	$\hat{\mathbf{E}}$
0001	$\mathbf{v}_1$	1	1	$\hat{\mathbf{v}}_1$	0001
0010	$\mathbf{v}_2$	1	1	$\hat{\mathbf{v}}_2$	0100
0011	$\mathbf{v}_2 + \mathbf{v}_1$	2	2	$\hat{\mathbf{v}}_2 + \hat{\mathbf{v}}_1$	0101
0100	$\mathbf{v}_3$	1	1	$\hat{\mathbf{v}}_3$	0010
0101	$\mathbf{v}_3 + \mathbf{v}_1$	2	2	$\hat{\mathbf{v}}_3 + \hat{\mathbf{v}}_1$	0011
0110	$\mathbf{v}_3 + \mathbf{v}_2$	4	4	$\hat{\mathbf{v}}_3 + \hat{\mathbf{v}}_2$	0110
0111	$\mathbf{v}_3 + \mathbf{v}_2 + \mathbf{v}_1$	5	5	$\hat{\mathbf{v}}_3 + \hat{\mathbf{v}}_2 + \hat{\mathbf{v}}_1$	0111
1000	$\mathbf{v}_4$	1	1	$\hat{\mathbf{v}}_4$	1000
1001	$\mathbf{v}_4 + \mathbf{v}_1$	4	4	$\hat{\mathbf{v}}_4 + \hat{\mathbf{v}}_1$	1001
1010	$\mathbf{v}_4 + \mathbf{v}_2$	2	2	$\hat{\mathbf{v}}_4 + \hat{\mathbf{v}}_2$	1100
1011	$\mathbf{v}_4 + \mathbf{v}_2 + \mathbf{v}_1$	5	5	$\hat{\mathbf{v}}_4 + \hat{\mathbf{v}}_2 + \hat{\mathbf{v}}_1$	1101
1100	$\mathbf{v}_4 + \mathbf{v}_3$	2	2	$\hat{\mathbf{v}}_4 + \hat{\mathbf{v}}_3$	1010
1011	$\mathbf{v}_4 + \mathbf{v}_2 + \mathbf{v}_1$	5	5	$\hat{\mathbf{v}}_4 + \hat{\mathbf{v}}_2 + \hat{\mathbf{v}}_1$	1101
1101	$\mathbf{v}_4 + \mathbf{v}_3 + \mathbf{v}_1$	5	5	$\hat{\mathbf{v}}_4 + \hat{\mathbf{v}}_3 + \hat{\mathbf{v}}_1$	1011
1110	$\mathbf{v}_4 + \mathbf{v}_3 + \mathbf{v}_2$	5	5	$\hat{\mathbf{v}}_4 + \hat{\mathbf{v}}_3 + \hat{\mathbf{v}}_2$	1110
1111	$\mathbf{v}_4 + \mathbf{v}_3 + \mathbf{v}_2 + \mathbf{v}_1$	8	8	$\hat{\mathbf{v}}_4 + \hat{\mathbf{v}}_3 + \hat{\mathbf{v}}_2 + \hat{\mathbf{v}}_1$	1111

Table 3.2: The values of d^2_{MIN} for two edge labelings, one using the original basis of (3.52) and the other using the new basis of (3.53).

043	176	171	045
045	176	171	043
043	171	176	045
045	171	176	043
176	043	045	171
171	043	045	176
176	045	043	171
171	045	043	176

Table 3.3: The eight permutations of the convolutional code (3.8) that result in distance equivalent trellis codes when used with the constellation in Figure 3.3.

The distance equivalent permutations of code polynomials $g_1(D)$, $\ldots$, $g_2(D)$ associated with the Gray coded constellations allow a search program to loop through distinct groups of polynomials and then check the interesting permutations. The distance equivalent codes associated with Ungerboeck labeled constellations do not have such an obvious code search implementation.

3.7 Summary

This chapter introduced trellis codes, beginning with a discussion of convolutional codes. Feedback-free convolutional encoders were represented in terms of the vector space spanned by the coefficients of the encoder polynomials.

With maximum likelihood sequence detection, the possible error sequences are the possible convolutional encoder output sequences. When the convolutional encoder is used in conjunction with a labeled constellation to form a trellis code, each error sequence $\mathbf{E}(D)$ can produce multiple sequences of symbol error distances depending on the input bit sequence that was transmitted. A worst case distance sequence $\{d_i^2(\mathbf{E}(D))\}$ was defined for each $\mathbf{E}(D)$.

There are three ways that two trellis codes can be considered equivalent. The weakest form of equivalence is distance equivalence, but even distance equivalent codes have the same values of the metrics presented in Chapter 2 for the rates and constellations examined in Chapter 6. Certain changes of basis result in distance equivalent codes, a fact that allows a significant decrease in search complexity.

Chapter 4

Constellation Labeling

The design of a trellis code involves the selection of a labeled constellation and the selection of a convolutional code to maximize some subset of the metrics presented in Chapter 2. This chapter investigates how a constellation of a given size and shape can be labeled so that the best possible metrics can be obtained. A single labeling is shown to work well for all the metrics of Chapter 2.

Section 4.1 defines three ways that labeled constellations can be considered equivalent corresponding to the three ways that trellis codes can be equivalent presented in Section 3.5 of the previous chapter.

Section 4.2 defines ways that one labeled constellation can be considered superior to another: *distance superiority* and *superiority in profile*. Section 4.3 identifies 4-PSK, 8-PSK, and 16-QAM labeled constellations that are superior in profile to all labeled constellations of the same size and shape, except those that are distance equivalent. These labeled constellations are used in the trellis code searches of Chapter 6.

Two common constellation labeling strategies, Gray coding and Ungerboeck labeling, are considered in Section 4.4. These strategies are compared with each other in the context of the ideas presented in the previous sections of this chapter.

The section begins with a demonstration of how Ungerboeck labeling and Gray coding are often equivalent. For all 2^n-PSK and square 4^n-QAM constellations, a pair of strictly equivalent constellations (i.e., constellations that produce strictly equivalent trellis codes) can be found such that one is Gray coded but the other is Ungerboeck labeled.

Not all constellations have equivalent labelings with both strategies. As examples, the 8-QAM and 32-QAM constellations are considered. This section demonstrates that although Ungerboeck labeling is possible for these two constellations, Gray coding is not.

Furthermore, constellations of the same size and shape that are Ungerboeck labeled are not necessarily distance equivalent. Likewise, two Gray labeled constellations of the same size and shape need not be distance equivalent. Examples of this phenomenon are given for Ungerboeck labeled 16-QAM and Gray labeled 16-PSK constellations. The distance-superior 16-QAM labeling is also an isometric labeling of the half integer grid.

The examples mentioned in the previous paragraph demonstrate that not all Gray coded or Ungerboeck labeled constellations can be superior in profile to any other labeling. With either strategy, some labelings are distance superior to others. However, the section concludes by proving that the "right" Gray coding will always produce a labeled constellation that is superior in profile to all others whenever such a dominant labeling exists.

4.1 Equivalent Labeled Constellations

The three ways that constellations can be considered equivalent correspond to the three ways that trellis codes can be equivalent defined in Section 3.5.

Definition 20 *Two labeled constellations $\mathcal{C}_1$ and $\mathcal{C}_2$ are strictly, range, or distance equivalent if for any trellis code that uses labeled constellation $\mathcal{C}_1$ there is respectively a strictly, range, or distance equivalent trellis code that uses labeled constellation $\mathcal{C}_2$.*

As will be discussed in Chapter 5, the primary consideration in designing a trellis code is minimization of the maximum pairwise error probability. For AWGN channels, even those with variations in SNR due to fading or frequency selectivity, this is accomplished with metrics that depend only on the distance sequences of the trellis code.

Distance equivalent trellis codes have the same distance sequences and thus the same values for these metrics. If $\mathcal{C}_1$ and $\mathcal{C}_2$ are distance equivalent, then for every trellis code that uses $\mathcal{C}_1$ there is a distance equivalent trellis code that uses $\mathcal{C}_2$. Thus a code search need consider only one member of a set of distance equivalent constellations.

The previous chapter showed that certain changes of basis applied to the convolutional code polynomials produce distance equivalent trellis codes. In this section, changes of basis applied to the constellation edge labels are shown to produce distance equivalent constellations.

Theorem 6 *Any two labeled constellations $\mathcal{C}_1$ and $\mathcal{C}_2$ whose edge labels are related by a change of basis are distance equivalent.*

Proof: Let F be the mapping that implements the change of basis relating the edge labels of $\mathcal{C}_1$ to those of $\mathcal{C}_2$. Select any trellis code T_1 having constellation $\mathcal{C}_1$ and a feedback-free rate k/n convolutional encoder with polynomials $\mathbf{G}_1, \ldots, \mathbf{G}_k$. Construct a trellis code T_2 by using constellation $\mathcal{C}_2$ with the convolutional encoder that results from applying F to the coefficients of $\mathbf{G}_1, \ldots, \mathbf{G}_k$.

Consider any set of input bit error sequences $e_1(D), \ldots, e_k(D)$. The output error sequence produced by T_1 is

$$\mathbf{E}^{(1)}(D) = \sum_{j=1}^{k} e_j(D)\mathbf{G}_j(D) \tag{4.1}$$

$$= \sum_{i=0}^{\infty} \mathbf{E}_i^{(1)} D^i. \tag{4.2}$$

The corresponding output error sequence produced by T_2 is

$$\mathbf{E}^{(2)}(D) = \sum_{j=1}^{k} e_j(D)F(\mathbf{G}_j(D)) \tag{4.3}$$

$$= \sum_{j=1}^{k} F(e_j(D)\mathbf{G}_j(D)) \tag{4.4}$$

$$= \sum_{i=0}^{\infty} F(\mathbf{E}_i^{(1)})D^i. \tag{4.5}$$

Equality in (4.4) follows from the additive property of F (see Lemma 3.28).

Every edge labeled $\mathbf{E}_i$ in $\mathcal{C}_1$ is labeled $F(\mathbf{E}_i)$ in $\mathcal{C}_2$, and so $d_{\text{MIN}}(\mathbf{E}_i)$ in $\mathcal{C}_1$ is equal to $d_{\text{MIN}}(F(\mathbf{E}_i))$ in $\mathcal{C}_2$ for every i. Thus T_1 and T_2 are distance equivalent and the theorem is proved. $\square$

The construction of T_2 above can also be used to demonstrate how constellations related by a change of basis that share the same zero labeled point are strictly equivalent.

Theorem 7 *Two labeled constellations $\mathcal{C}_1$ and $\mathcal{C}_2$ whose edge labels are related by a change of basis are strictly equivalent $\Longleftrightarrow$ they label the same point with the zero label.*

Proof: Consider the trellis code T_2 constructed in the previous proof. Now replace input error sequences $e_j(D)$ with input bit sequences $b_j(D)$ and output error sequences $E(D)$ with encoder output sequences $\mathbf{C}(D)$ in the previous proof.

If two constellations with edge labels related by a change of basis label the same point with zero, then the point labels as well as the edge labels are related by that change of basis. Thus C_i labels the same point in $\mathcal{C}_1$ that output $F(C_i)$ labels in $\mathcal{C}_2$. Thus T_1 and T_2 are strictly equivalent, and the $\Longrightarrow$ direction is proved.

The all zeros input sequence maps to the output sequence that consists entirely of symbols equal to the constellation point having the zero label. Two constellations with different points having the zero label cannot be strictly equivalent since the all zeros input sequence does not map to the same output sequence for both constellations. This proves the $\Longleftarrow$ direction. $\square$

The design of a trellis code includes the selection of a labeled constellation. The choice of labeling for a given constellation is simplified by the fact that many different labelings produce strictly equivalent constellations. Consider a labeled 16-QAM constellation such as the one in Figure 3.3 of the Chapter 3. Maintaining the same zero labeled point, any change of basis produces a strictly equivalent constellation. The number of these mappings is the same as the number of bases for the space of four bit vectors, which is $15 \times 14 \times 12 \times 8 = 20\,160$.

The number of labelings that produce distance equivalent constellations is, of course, even larger. For distance equivalence there is freedom to place the zero label on any of the points. Thus the number of distance equivalent labelings is at least the number of strictly equivalent labelings above multiplied by the size of the constellation. Thus for the 16-QAM constellation, there are at least $16 \times 20\,160 = 322\,560$ distance equivalent constellations.

That still leaves $16!/322\,560 = 64\,864\,800$ distinct labelings that do not necessarily produce distance equivalent codes. These remaining distinct labelings are the subject of the next section.

4.2 The Edge Length Profile

In the previous section many different labeled constellations were shown to produce distance equivalent and even strictly equivalent trellis codes. In this section, labeled constellations that do not produce distance equivalent codes are compared. The goal is to remove inferior labelings from consideration and thus reduce search complexity.

Two ways in which one constellation can be superior to another are discussed below. This discussion refers to three constellations $\mathcal{C}_1$, $\hat{\mathcal{C}}_1$, and $\mathcal{C}_2$ of the same size and the same shape. The size of a constellation $\mathcal{C}$ is the number of points in $\mathcal{C}$ and is denoted by $|\mathcal{C}|$. $\mathcal{C}_1$, $\hat{\mathcal{C}}_1$, and $\mathcal{C}_2$ have edge labels $\mathbf{E}_i^{(1)}$, $\hat{\mathbf{E}}_i^{(1)}$, and $\mathbf{E}_i^{(2)}$, respectively.

Definition 21 *$\mathcal{C}_1$ is distance superior to $\mathcal{C}_2$ (or $\mathcal{C}_2$ is distance inferior to $\mathcal{C}_1$) if for every trellis code T_2 designed using $\mathcal{C}_2$ there is a distance superior (see Definition 18) trellis code T_1 defined using $\mathcal{C}_1$.*

Lemma 2 *$\mathcal{C}_1$ is distance superior to $\mathcal{C}_2$ if there exists a constellation $\hat{\mathcal{C}}_1$ distance equivalent to $\mathcal{C}_1$ such that for $i = 1, \ldots, |\mathcal{C}_1|$:*

$$d_{\text{MIN}}(\hat{\mathbf{E}}_i^{(1)}) \geq d_{\text{MIN}}(\mathbf{E}_i^{(2)}) \,, \tag{4.6}$$

with at least one strict inequality.

Proof: By (4.6), any trellis code T_2 designed with $\mathcal{C}_2$ is distance inferior to the trellis code $\hat{T}_1$ that uses the convolutional code of T_2 with the constellation $\hat{\mathcal{C}}_1$. T_2 must also be distance inferior to some trellis code T_1 designed using $\mathcal{C}_1$ since $\mathcal{C}_1$ is distance equivalent to $\hat{\mathcal{C}}_1$. Thus for any trellis code T_2 using $\mathcal{C}_2$ there is a distance superior trellis code T_1 using $\mathcal{C}_1$. □

The metrics (to be introduced in Chapter 5) that will used for designing trellis codes are larger (better) for distance superior trellis codes. Thus any trellis code designed using $\mathcal{C}_2$ is inferior in these metrics to some code designed using $\mathcal{C}_1$. The inferior constellation $\mathcal{C}_2$ may be removed from consideration in a trellis code search based on these metrics.

Any set of possible constellation labelings is partially ordered under Definition 2 since distance superiority obeys the transitive property. However, there may be two constellation labelings that are not distance equivalent with neither being distance superior. Thus Definition 2 does not necessarily provide a complete ordering for the set of possible labelings. Even when one labeling is distance superior to another, demonstrating this using Lemma 2 requires effort.

A weaker form of superiority that is easily applied to any pair of labelings uses a labeled constellation's edge length profile.

Definition 22 *The edge length profile $\{p_i\}$ of a labeled constellation is defined to be the list of $d^2_{\text{MIN}}(\mathbf{E})$ values for all nonzero edge labels E listed in increasing order.*

To demonstrate, an example edge length profile is constructed for the labeled 16-QAM constellation shown in Figure 3.3 of Chapter 2. Table 3.2 shows the $d^2_{\text{MIN}}(\mathbf{E})$ values listed lexicographically according to E for that constellation. The corresponding edge length profile is given in Table 4.1. Superiority based on the edge length profile is defined in terms of $\{p_i^{(1)}\}$ and $\{p_i^{(2)}\}$, the edge length profiles for $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively.

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
p_i	1	1	1	1	2	2	2	2	4	4	5	5	5	5	8

Table 4.1: Edge length profile for 16-QAM constellation in Figure 3.3.

Definition 23 *$\mathcal{C}$ is superior in profile to $\hat{\mathcal{C}}$ if $p_i^{(1)} \geq p_i^{(2)}$ for $i = 1, \ldots, |\mathcal{C}_1|$ with at least one strict inequality.*

Often, if $\mathcal{C}_1$ is superior in profile to $\mathcal{C}_2$, then $\mathcal{C}_1$ is also distance superior to $\mathcal{C}_2$. In fact, no example has been found where $\mathcal{C}_1$ is superior in profile to $\mathcal{C}_2$ and not distance superior to $\mathcal{C}_2$. However, it remains to be shown whether superiority in profile always implies distance superiority. Furthermore, superiority in profile still may not completely order the set of possible labelings.

The code searches in this dissertation restrict attention to labelings that are superior in profile to all others. The next section identifies these constellations for 4-PSK, 8-PSK, and 16-QAM.

4.3 Superior in profile constellations

This section derives labeled constellations that are superior in profile to all other labeled constellations of the same size and shape (except those that are distance equivalent). This section considers only 4-PSK, 8-PSK, and 16-QAM constellations. However, the strategy of optimizing the edge length profile can be applied to larger constellations as well.

The derivations are given in order of difficulty rather than constellation size. 4-PSK has the simplest derivation while 16-QAM requires more discussion, and 8-PSK is the most involved of the three derivations. Three lemmas about labeled constellations are used extensively in the following discussion. These lemmas are direct results of the fact that no two constellation points have the same label.

Lemma 3 *All the edges emanating from a given point must have different labels.*

Proof: Edge labels are computed as the difference of the labels of the two connected constellation points. Since all the constellation points have different labels, all the edges emanating from a given point must have different labels. □

A path consists of one or more sequentially connected edges. The next lemma concerns the sums of edge labels along different paths emanating from the same constellation point.

Definition 24 *A path's edge label sum is the sum of the edge labels in that path.*

Lemma 4 *Two different paths emanating from the same initial constellation point have the same edge label sum if they end at the same final constellation point.*

Proof: Each edge label is the difference of the associated point labels. All the intermediate point labels along a path cancel in the edge label sum leaving only the difference of the initial point and the final point regardless of the particular path taken.

The two paths share the same initial point, and every constellation point has a different label. Thus the edge label sums are the same if and only if the final points are the same. □

Lemma 5 *If all points in a constellation are connected by paths consisting only of edges with the minimum edge length, then the set of edge labels of the minimum length edges must contain a basis for the set of all edge labels.*

Proof: Fix an initial point. Every other point can be reached by a path consisting only of minimum length edges. Thus every edge label can be expressed as the edge label sum of minimum length edges. Hence a subset of the minimum length edge labels is a basis. □

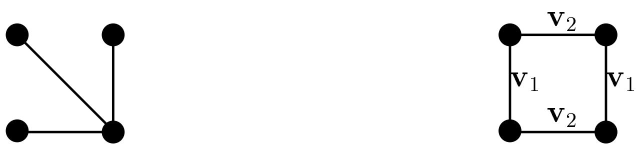
Lemma 5 applies to all PSK and QAM constellations considered in this dissertation.

4.3.1 4-PSK

This section upper bounds the edge length profile for 4-PSK and identifies the labelings that achieve this bound.

To construct the edge length profile bound, the set of edges emanating from any single point are considered. Regardless of how the constellation is labeled, each edge in this set has a distinct label by Lemma 3, and every edge label appears exactly once among this set of edges. As a result, the d^2_{MIN} of each edge label is upper bounded by the squared length of one of the edges in the set. Thus these edges produce an upper bound on the edge length profile.

For any labeled constellation of the same size and shape, no element in its edge length profile can be larger than the corresponding element in the bounding edge length profile described above. Figure 4.1(a) shows the edges used to construct this edge length profile bound for the 4-PSK constellation.



(a) Edges for length profile bound.

(b) Labeling that achieves the bound.

Figure 4.1: 4-PSK constellation shown with edges for edge length profile bound (left) and edge labeling that achieves the bound.

The bounding edge length profile resulting from the edges in Figure 4.1(a) is shown below in Table 4.2, assuming nearest neighbors are separated by 1.

i	1	2	3
p_i	1	1	2

Table 4.2: A bounding 4-PSK edge length profile.

To achieve this bound on the edge length profile, there can only be two edge labels with $d^2_{\text{MIN}} = 1$. The only way to have only two such edge labels and satisfy Lemma 3 is for the edge labels to alternate around the ring as shown in Figure 4.1(b).

The remaining choices are selecting the basis vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ and selecting the point to be labeled 00. All possibilities for these two choices produce distance equivalent constellations. Thus any 4-PSK constellation that has edge labels as in Figure 4.1(b) is superior in profile to all but distance equivalent 4-PSK constellations.

4.3.2 16-QAM

The 16-QAM constellation is considered next since a simple edge length profile bound is easily achieved as with 4-PSK. Figure 4.2 shows the edges used to construct an edge length profile bound for the 16-QAM constellation using the same technique as for 4-PSK.

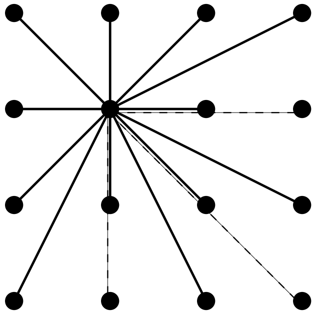


Figure 4.2: Edges emanating from one point are used to construct a 16-QAM edge length profile that cannot be surpassed.

The bounding edge length profile resulting from the edges in Figure 4.2 is shown below in Table 4.3. Recall that the nearest neighbor distance is assumed to be 1. Table 4.3 is exactly the edge length profile of Table 4.1. Thus this bounding profile is achieved by the Gray coded constellation shown in Figure 3.3. All of the 16-QAM constellations that achieve this bound are distance equivalent, as shown by the following argument, which is illustrated in Figure 4.3.

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
p_i	1	1	1	1	2	2	2	2	4	4	5	5	5	5	8

Table 4.3: Bounding edge length profile for edges in Figure 4.2.

There are exactly four $d^2_{\text{MIN}} = 1$ edge labels in a labeled constellation that achieves the edge length profile of Table 4.3. The following argument shows that all square 16-QAM constellations that have exactly four $d^2_{\text{MIN}} = 1$ edge labels are distance equivalent.

Lemma 5 implies that the $d^2_{\text{MIN}} = 1$ edge labels must form a basis. Choose any basis $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4\}$ to be the edge labels for the four minimum d^2_{MIN} edges. The particular choice of basis is unimportant since all choices lead to distance equivalent constellations.

Choose one of the four central points and label the four minimum d^2_{MIN} edges connected to this point. Step 1 (Figure 4.3(a)) shows a labeling for these four edges.

Each of the four edge labels must be different by Lemma 3. Only one of the 4! ways of matching the four basis vectors to the four edges is considered since the other choices can be attained by the appropriate permutation of the basis.

Now consider the edges labeled **a** and **b** in Step 2 (Figure 4.3(b)). Lemma 4 gives

$$\mathbf{v}_2 + \mathbf{v}_3 = \mathbf{a} + \mathbf{b}. \tag{4.7}$$

Recall that each vector $\mathbf{v}_i$ is its own additive inverse. If $\mathbf{a} = \mathbf{v}_1$ then $\mathbf{b}$ must equal $\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3$ to satisfy (4.7). However, $\mathbf{b}$ would then be a fifth edge label with $d^2_{\text{MIN}} = 1$, and the edge length profile would not be achieved. Similarly, $\mathbf{a} = \mathbf{v}_4$ would force a fifth edge with $d^2_{\text{MIN}} = 1$. Lemma 3 prevents $\mathbf{a} = \mathbf{v}_2$.

This leaves $\mathbf{a} = \mathbf{v}_3$, and (4.7) forces $\mathbf{b} = \mathbf{v}_2$. The same reasoning provides labels for the six remaining unlabeled edges shown in Step 2. Step 3 (Figure 4.3(c)) shows the resulting labeling for these edges.

Step 3 introduces two new edges labeled **c** and **d**. Lemma 3 forces $\mathbf{c} = \mathbf{v}_4$ and $\mathbf{d} = \mathbf{v}_3$ or the introduction of additional edges with $d^2_{\text{MIN}} = 1$. The remaining unlabeled edges in Step 3 have their labels forced exactly as in Step 2. The two unlabeled edges in Step 4 (Figure 4.3(d)) also have their labeling forced as in Step 2. Step 5 (Figure 4.3(e)) shows the constellation with all the minimum d^2_{MIN} edges labeled.

Note that the edge labeling of Figure 4.3(e) was entirely forced after the initial labeling of Step 1. Thus all 16-QAM constellations that achieve the edge length profile bound of Table 4.3 are distance equivalent, being related by a change of basis.

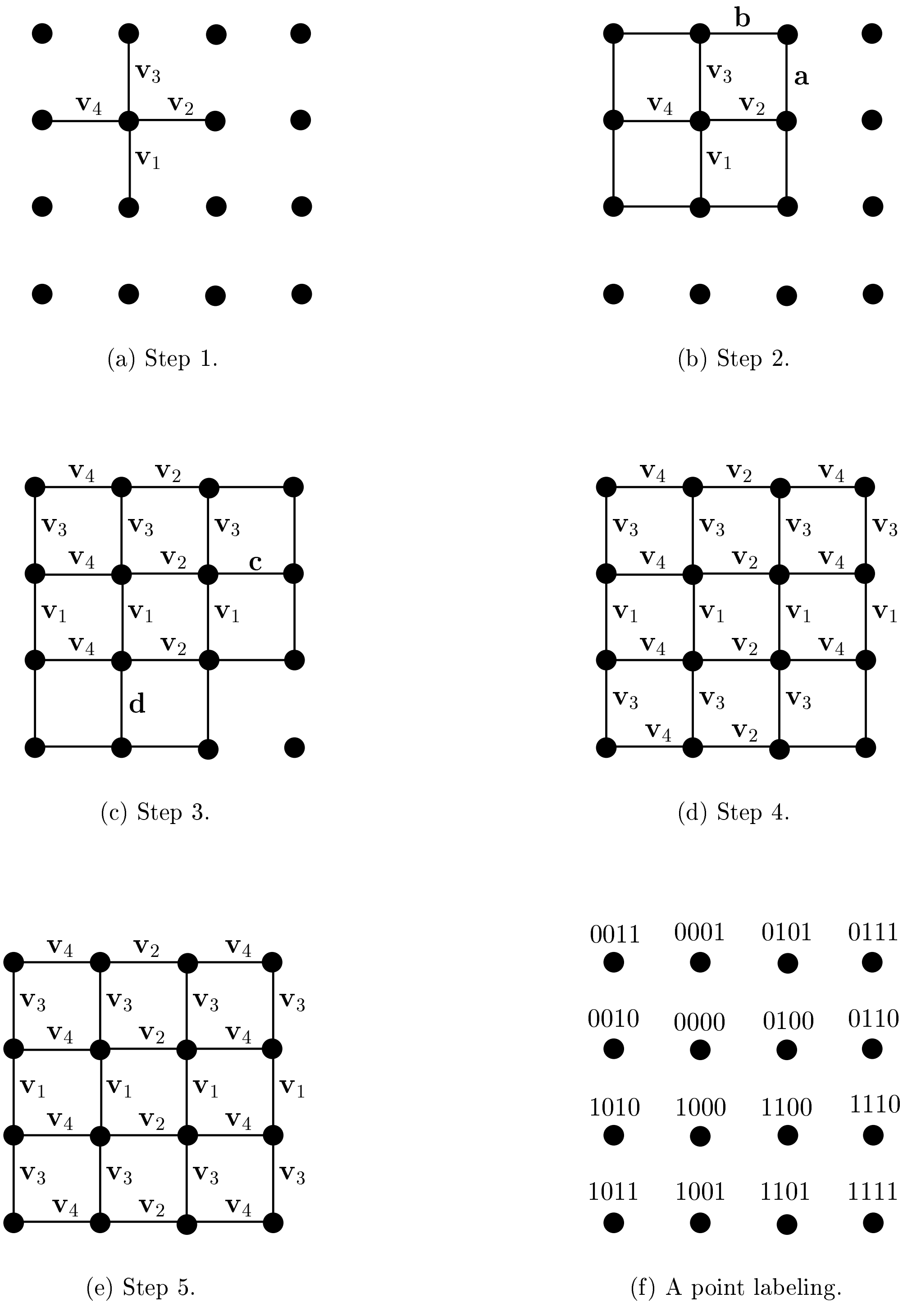
Any 16-QAM constellation that achieves the length profile bound can be constructed by choosing a point to have the zero label and selecting a basis $\mathbf{v}_1, \ldots, \mathbf{v}_4$. Figure 4.3(f) shows one example. This labeling is exactly the same as the Gray coded 16-QAM constellation shown in Figure 3.3 of the previous chapter. It was achieved by choosing

$$\mathbf{v}_1 = 1000 \tag{4.8}$$

$$\mathbf{v}_2 = 0100 \tag{4.9}$$

$$\mathbf{v}_3 = 0001 \tag{4.10}$$

$$\mathbf{v}_4 = 0010. \tag{4.11}$$



(a) Step 1.

(b) Step 2.

(c) Step 3.

(d) Step 4.

(e) Step 5.

(f) A point labeling.

Figure 4.3: Steps for achieving the best possible edge length profile for a 16-QAM constellation.

4.3.3 8-PSK

In this subsection, the set of labeled 8-PSK constellations that are superior in profile to all others is constructed. As with 4-PSK and 16-QAM, all these superior in profile constellations are shown to be distance equivalent. The line of argument in this subsection is slightly different from that of the previous two. First, a set of distance equivalent constellations is constructed. Then the constellations in this set are shown to superior in profile to all other 8-PSK constellations.

The distance between nearest neighbors in an 8-PSK constellation is referred to as Δ_1. Lemma 5 implies that there must be a basis containing only edge labels with $d^2_{\text{MIN}} = \Delta_1$. The vector space of 3-bit vectors that label the edges of 8-PSK constellations require three vectors to form a basis. Thus there must be three distinct labels that have $d^2_{\text{MIN}} = \Delta_1$.

All labelings that have the smallest possible number (three) of labels with $d^2_{\text{MIN}} = \Delta_1$ are now shown to be distance equivalent. The steps in this argument are shown in Figure 4.4.

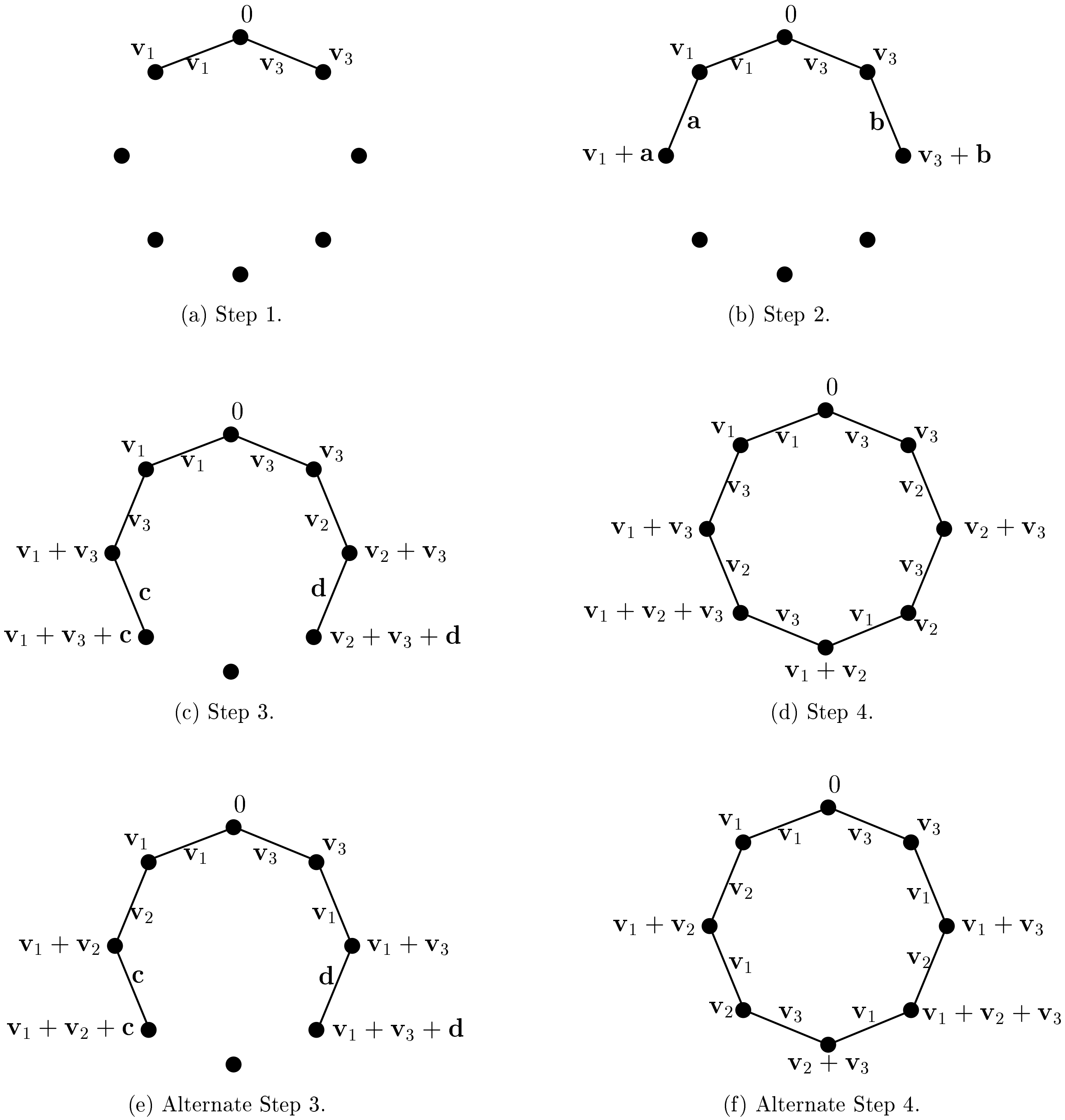
Choose any basis $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$ to be the three edge labels with $d^2_{\text{MIN}} = \Delta_1$. All the $d^2_{\text{MIN}} = \Delta_1$ edges will be labeled with one of the three basis vectors. In Step 1 (Figure 4.4(a)) an arbitrary point is labeled 0 and the two minimum d^2_{MIN} edges emanating from this point are labeled $\mathbf{v}_1$ and $\mathbf{v}_3$. (The other possible labels for these edges from $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$ could be obtained by a permutation of the choice of initial basis and thus lead to distance equivalent constellations.)

Consider the two edges labeled $\mathbf{a}$ and $\mathbf{b}$ in Step 2 (Figure 4.4(b)). By Lemma 3, $\mathbf{a} \neq \mathbf{v}_1$ and $\mathbf{b} \neq \mathbf{v}_3$. Thus $\mathbf{a} \in \{\mathbf{v}_2, \mathbf{v}_3\}$ and $\mathbf{b} \in \{\mathbf{v}_1, \mathbf{v}_2\}$. If both $\mathbf{a}$ and $\mathbf{b}$ are chosen to be $\mathbf{v}_2$, it is impossible to label the rest of the $d^2_{\text{MIN}} = \Delta_1$ edges with labels in $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$ and have a different label for each point. It is also impossible to have both $\mathbf{a} = \mathbf{v}_3$ and $\mathbf{b} = \mathbf{v}_1$ since this would lead to the points labeled $\mathbf{v}_1 + \mathbf{a}$ and $\mathbf{v}_3 + \mathbf{b}$ having the same label. There remain two choices:

1. $\mathbf{a} = \mathbf{v}_3$ and $\mathbf{b} = \mathbf{v}_2$.

2. $\mathbf{a} = \mathbf{v}_2$ and $\mathbf{b} = \mathbf{v}_1$.

These two choices produce distance equivalent constellations. For the moment, select the first choice. Step 3 (Figure 4.4(c)) shows the resulting labels and identifies



(a) Step 1.

(b) Step 2.

(c) Step 3.

(d) Step 4.

(e) Alternate Step 3.

(f) Alternate Step 4.

Figure 4.4: Steps for achieving the best possible edge length profile for an 8-PSK constellation.

the edges labeled $\mathbf{c}$ and $\mathbf{d}$ as the next edges to be considered. The point labeled $\mathbf{v}_1 + \mathbf{v}_3 + \mathbf{c}$ cannot be labeled $\mathbf{v}_1$ or $\mathbf{v}_3$. There are already points with those labels. Thus $\mathbf{c}$ cannot be $\mathbf{v}_3$ or $\mathbf{v}_1$, leaving only $\mathbf{c} = \mathbf{v}_2$. The point in question is labeled $\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3$.

Now consider the point labeled $\mathbf{v}_3 + \mathbf{v}_2 + \mathbf{d}$ in Figure 4.4(c). The choice of $d = \mathbf{v}_1$ is not possible since the point label $\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3$ is no longer available. Similarly, $\mathbf{d} = \mathbf{v}_2$ is not possible. The only choice is $d = \mathbf{v}_3$ producing the point label $\mathbf{v}_2$.

In Step 4 (Figure 4.4(d)), there is one remaining point and one remaining point label. The remaining point label $\mathbf{v}_1 + \mathbf{v}_2$ forces the two remaining edge labels to be $\mathbf{v}_3$ and $\mathbf{v}_1$.

Now suppose that in Step 2 the second choice is made so that $a = \mathbf{v}_2$ and $b = \mathbf{v}_1$. This choice leads to the alternate Steps 3 and 4 shown in Figures 4.4(e) and 4.4(f). These alternate steps follows by symmetry from the original Steps 3 and 4.

The resulting constellations shown in Step 4 and Alternate Step 4 (Figures 4.4(d) and 4.4(f)) are distance equivalent. To see this, consider changing the basis of the constellation in Step 4 using the following mapping:

$$\mathbf{v}_1 \rightarrow \mathbf{v}_3 \qquad (4.12)$$

$$\mathbf{v}_2 \rightarrow \mathbf{v}_2 \qquad (4.13)$$

$$\mathbf{v}_3 \rightarrow \mathbf{v}_1 \qquad (4.14)$$

This produces exactly the edge labeling of Alternate Step 4 rotated by 45 degrees.

Every 8-PSK constellation must have at least three edge labels with $d^2_{\text{MIN}} = \Delta_1$. The above construction demonstrates that all the constellations with exactly three such edge labels are distance equivalent. The last step is to show that this set of labeled constellations is superior in profile to all other labeled 8-PSK constellations.

Figure 4.5 shows the edges emanating from a single point in an 8-PSK constellation. The four possible distances between two points in an 8-PSK constellation are identified in Figure 4.5 as $\Delta_1, \Delta_2, \Delta_3$, and Δ_4. As with the 16-QAM constellation, these edges produce an edge length profile bound. Table 4.4 shows the edge length profile bound corresponding to Figure 4.5. For comparison, Table 4.5 shows the edge length profile of the $d^2_{\text{MIN}}(\mathbf{E})$ values for the labeled constellation shown in Figure 4.4(d).

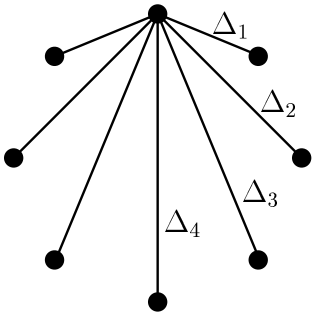


Figure 4.5: Edges for an 8-PSK edge length profile bound.

i	1	2	3	4	5	6	7
p_i	Δ_1	Δ_1	Δ_2	Δ_2	Δ_3	Δ_3	Δ_4

Table 4.4: Edge length profile bound for edges in Figure 4.5.

The edge length profile bound of Table 4.4 bounds each element p_i independently, and the edge length profile of Table 4.5 achieves the bounding values of Table 4.4 for p_1, p_2, p_4, p_6, and p_7. Thus if p_3 and p_5 in Table 4.5 can be shown to be as large as possible for any 8-PSK labeled constellation, then the set of distance equivalent constellations constructed above is superior in profile to all other 8-PSK constellations.

Every 8-PSK constellation must have three edges with $d^2_{\text{MIN}} = \Delta_1$, hence $p_3 = \Delta_1$ for every 8-PSK constellation. Thus the value of p_3 in Table 4.5 is as large as possible, and the bound on p_3 in Table 4.4 is not achievable by any 8-PSK constellation.

The last step is to show that p_5 in Table 4.5 is also as large as possible for any 8-PSK constellation. It was shown above that 8-PSK constellations must have at least three edges with $d^2_{\text{MIN}} = \Delta_1$. All 8-PSK constellations with exactly three such edges are distance equivalent and have the profile of Table 4.5. The only possibility for a p_5 that is larger than the Δ_2 in Table 4.5 is to have more than three edges with $d^2_{\text{MIN}} = \Delta_1$.

If there are five or more edges with $d^2_{\text{MIN}} = \Delta_1$, then $p_5 = \Delta_1 < \Delta_2$. Suppose that there are exactly four edge labels with $d^2_{\text{MIN}} = \Delta_1$. Since three of these edge labels form a basis, there is at most one subset of three of these edge labels that sum to zero. Equivalently, at most one pair of these edge labels will sum to produce another one of these four edge labels.

i	1	2	3	4	5	6	7
E	$\mathbf{v}_1$	$\mathbf{v}_2$	$\mathbf{v}_3$	$\mathbf{v}_1 + \mathbf{v}_3$	$\mathbf{v}_2 + \mathbf{v}_3$	$\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3$	$\mathbf{v}_1 + \mathbf{v}_2$
$p_i = d^2_{\text{MIN}}(\mathbf{E})$	Δ_1	Δ_1	Δ_1	Δ_2	Δ_2	Δ_3	Δ_4

Table 4.5: Edge length profile for constellation in Figure 4.4(d).

There are at least two edge labels for edges with squared distance Δ_2 by Lemma 4, but these edge labels could actually have $d^2_{\text{MIN}} = \Delta_1$. The Δ_2 edges always have labels that are the sum of a pair of edge labels with $d^2_{\text{MIN}} = \Delta_1$. From the previous paragraph, only one of these edge labels can have $d^2_{\text{MIN}} = \Delta_1$. Thus there is at least one edge label with $d^2_{\text{MIN}} = \Delta_2$ if exactly four edge labels have $d^2_{\text{MIN}} = \Delta_1$. In this case, $p_5 = \Delta_2$.

Thus no 8-PSK labeling can achieve $p_5 > \Delta_2$, and the edge length profile of Table 4.5 cannot be surpassed. All the 8-PSK constellations that achieve the length profile of Table 4.5 must have exactly three edge labels with $d^2_{\text{MIN}} = \Delta_1$. As shown above, all these constellations are distance equivalent.

4.4 Gray Coding and Ungerboeck Labeling

Noticeably missing from the first three sections of this chapter was any mention of the two most popular strategies for labeling constellations, Ungerboeck labeling and Gray coding. Using the concepts of distance equivalence and distance superiority, some new insights can be gained about these standard labeling techniques. In Section 4.4.1 Gray coding and Ungerboeck labeling are introduced and formally defined.

Section 4.4.2 shows how Ungerboeck labeling and Gray coding can often be strictly equivalent. For all 2^n-PSK and square 4^n-QAM constellations, pairs of strictly equivalent labeled constellations are constructed such that one constellation meets the definition of Gray coding and the other meets the definition of Ungerboeck labeling. The superior-in-profile edge labelings identified in Section 4.3 can be achieved by such pairs of strictly equivalent constellations, one Gray coded and the other Ungerboeck labeled.

Not all constellations have equivalent labelings with both strategies. As examples, the 8-QAM and 32-QAM constellations are considered. Section 4.4.3 demonstrates that although Ungerboeck labeling is possible for these two constellations, Gray coding is not.

Surprisingly, constellations of the same size and shape that are Ungerboeck labeled are not necessarily distance equivalent. Likewise, two Gray labeled constellations of the same size and shape need not be distance equivalent. Examples are given in Section 4.4.4 for Ungerboeck labeled 16-QAM and Gray labeled 16-PSK constellations.

The examples mentioned in the previous paragraph demonstrate that not all Gray coded or Ungerboeck labeled constellations can be superior in profile to every other labeling. With either strategy, some labelings are distance superior to others. However, Section 4.4.7 concludes that the "right" Gray coding will always produce a labeled constellation that is superior in profile to all others whenever such a dominant labeling exists.

4.4.1 Definitions

Gray coding was first introduced in the context of uncoded transmission. To minimize the effect of noise on the BER, Gray coding enforces the rule that all nearest neighbors differ by exactly one bit. In this way, the most common symbol errors in uncoded transmission (those between nearest neighbors) lead to one bit error.

Definition 25 *A labeled constellation is Gray coded if and only if all nearest neighbors differ by exactly one bit.*

This motivation for using Gray coding in the context of uncoded transmission does not apply to the context of coded transmission. As pointed out by Ungerboeck in [5], the strategy of combining Gray coded constellations with maximum Hamming distance convolutional codes does not always provide the maximum Euclidean distance. Ungerboeck labeling was introduced by Ungerboeck in [5] as part of a technique for designing maximum Euclidean distance trellis codes.

Ungerboeck labeling (also called set partitioning) partitions the constellation into a hierarchy of collections of mutually exclusive, collectively exhaustive subsets with the largest possible separation between nearest neighbors in the same subset. Following the approach set forth in [5], these subsets (often called cosets because of their algebraic structure) will be identified by the appropriate number of least significant bits.

Definition 26 *A 2^n-point constellation is called Ungerboeck labeled if for every $m \in \{1, \ldots, n\}$ the minimum distance of two points sharing the same m least significant bits is as large as possible.*

Ungerboeck labeling was invented for a code design technique that features the use of uncoded bits. Since uncoded bits result in poor performance on the fading channels of interest in this dissertation, Ungerboeck labeling is not the obvious choice for code designs for these channels.

4.4.2 Strictly equivalent pairs

In this subsection, pairs of strictly equivalent labeled constellations such that one meets the definition of Ungerboeck labeling and the other meets the definition of Gray coding are constructed for all 2^n-PSK and square QAM constellations. The PSK and square QAM cases are proved separately, but both proofs use similar inductive arguments.

Theorem 8 *For any positive integer n, there is a pair of distance equivalent 2^n-PSK constellations such that one meets the definition of Gray coding and the other meets the definition of Ungerboeck labeling.*

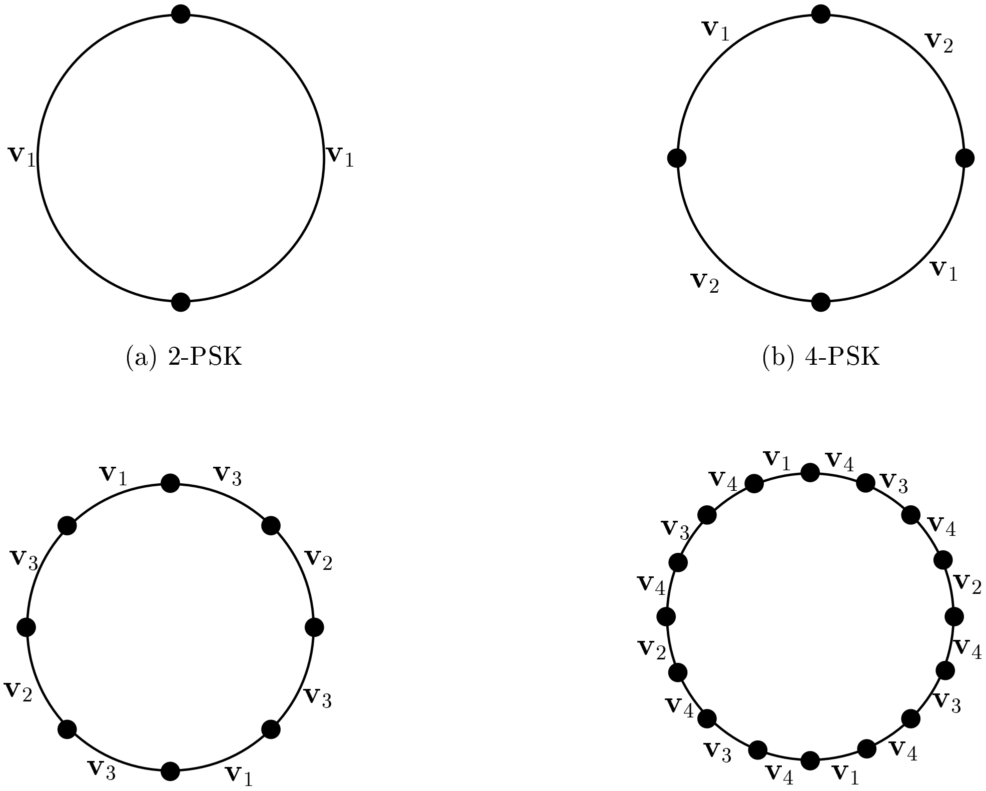
Proof: The base case of 2-PSK obviously meets either definition. Now consider the following inductive edge labeling strategy, illustrated in Figure 4.6. When going from the $n-1$ case to the n case, label the 2^n minimum distance edges by alternating around the ring between using a new basis vector $\mathbf{v}_n$ and using the $n-1$ case basis vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_{n-1}\}$ in their original pattern around the ring.

Choosing the n case basis $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ to be the vectors with exactly one bit equal to 1 (and all the rest 0s) labels the constellation to meet the Gray coding definition.

Suppose that $n-1$ case basis $\{\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_{n-1}\}$ provide an edge labeling for the 2^{n-1}-PSK constellation that meets the definition for Ungerboeck labeling. Choose the n case basis in the following way: For $i \in \{1, \ldots, n-1\}$ set

$$\mathbf{v}_i = (\hat{\mathbf{v}}_i << 1) + 1. \tag{4.15}$$

That is, left shift $\hat{\mathbf{v}}_i$ by one bit and make the least significant bit a 1. Set $\mathbf{v}_n = 0 \ldots 01$.



(a) 2-PSK

(b) 4-PSK

(c) 8-PSK

(d) 16-PSK

Figure 4.6: Inductive edge labeling strategy for 2^n-PSK.

All the minimum distance edge labels have a 1 in the least significant bit. Thus all points sharing the same least significant bit are separated by paths consisting of an even number (at least two) minimum distance edge labels. The resulting separation of at least $2\pi/2^{n-1}$ radians is the largest separation possible for 2^{n-1} points on a ring.

The points that share $m \geq 2$ least significant bits are either all in the $n-1$ case Ungerboeck labeled constellation or a $\pi/2^{n-1}$ rotation of it. These points are separated by the largest distance possible ($2\pi/2^{n-m}$ radians) because of the labeling structure inherited from the $n-1$ case.

The Gray labeled and Ungerboeck labeled PSK constellations described above differ only in the choice of basis. In either case, any point can be chosen to have the zero label. When the same point is chosen to have the zero label, the Ungerboeck labeled and Gray coded constellations are strictly equivalent by Theorem 7. □

Theorem 9 *For any positive integer n, there is a pair of distance equivalent square 4^n-QAM constellations such that one meets the definition of Gray coding and the other meets the definition of Ungerboeck labeling.*

Proof: The square QAM constellations under consideration all have the structure of the Z^2 lattice — the 2-dimensional integer grid. The base case of 4-QAM is shown in Figure 4.7(a). Since 4-QAM is identical to 4-PSK, Theorem 8 applied to this constellation guarantees choices of $\mathbf{v}_1$ and $\mathbf{v}_2$ that satisfy each of the two definitions.

Consider the following inductive edge labeling strategy illustrated in Figure 4.7 for 4-, 16-, and 64-QAM. For all these square QAM constellations, the vertical edges in any one row all have the same edge label, and the horizontal edges in any one column all have the same edge label.

The induction determines the labeling for these rows and columns. The top row of minimum distance vertical edges are labeled with the new basis vector $\mathbf{v}_{2n-1}$. Thereafter, rows of vertical edges are alternately labeled with the vectors used on the rows of vertical edges in the $n-1$ case (in the same pattern) and $\mathbf{v}_{2n-1}$. The columns of horizontal edges are labeled in the same way, using the new basis vector $\mathbf{v}_{2n}$.

Selecting the basis vectors $\mathbf{v}_1, \ldots, \mathbf{v}_{2n}$ to be the vectors having exactly one bit equal to 1 produces a labeled constellation that meets the definition of Gray labeling.

Suppose that $\{\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_{2n-2}\}$ are the edge labels of a square 4^{n-1}-QAM constellation that meets the definition for Ungerboeck labeling. For $i \in \{1, \ldots, 2(n-1)\}$ set

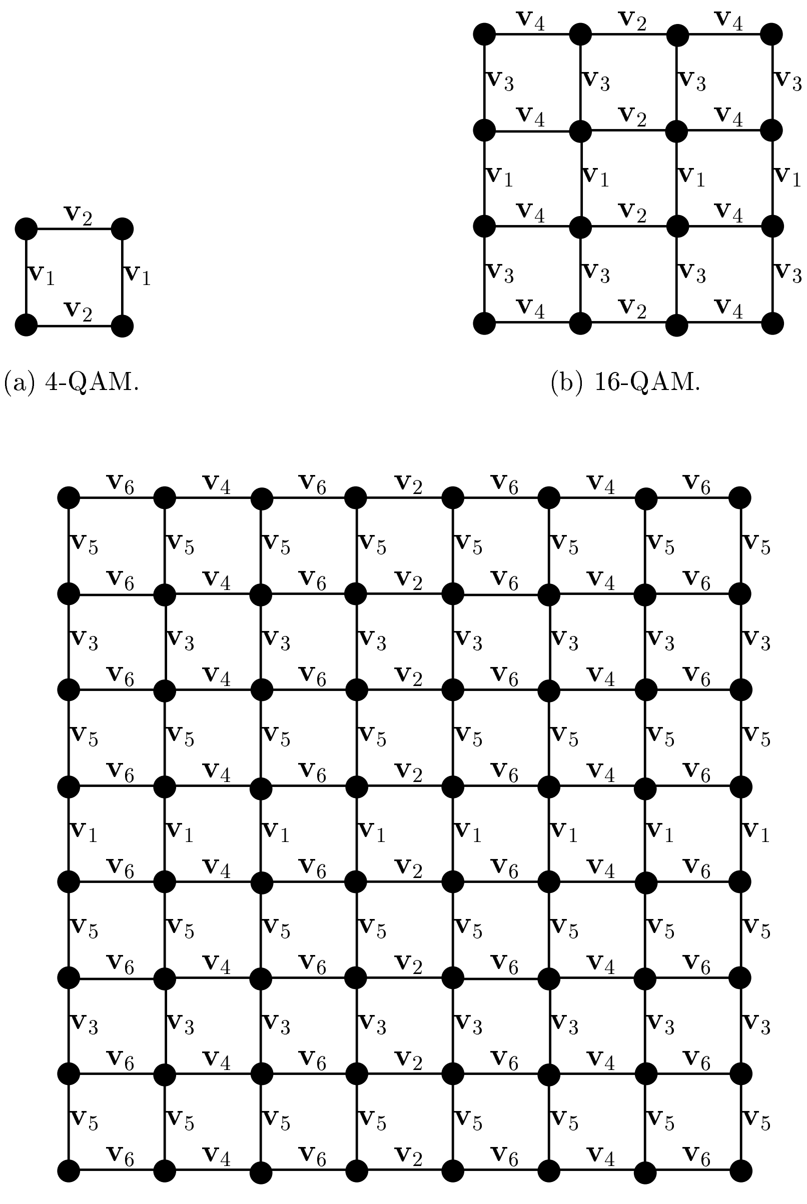
$$\mathbf{v}_i = (\hat{\mathbf{v}}_i << 2) + 1 \qquad \text{for } i \text{ odd} \tag{4.16}$$

$$\mathbf{v}_i = (\hat{\mathbf{v}}_i << 2) + 3 \qquad \text{for } i \text{ even.} \tag{4.17}$$

That is, left shift $\hat{\mathbf{v}}_i$ by two bits and make the two least significant bits 01 for i odd and 11 for i even. Also set

$$\mathbf{v}_{2n-1} = 0\ldots01 \tag{4.18}$$

$$\mathbf{v}_{2n} = 0\ldots11. \tag{4.19}$$


$\mathbf{v}_2$
$\mathbf{v}_1$ $\mathbf{v}_1$
$\mathbf{v}_2$
$\mathbf{v}_4$ $\mathbf{v}_2$ $\mathbf{v}_4$
$\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$
$\mathbf{v}_4$ $\mathbf{v}_2$ $\mathbf{v}_4$
$\mathbf{v}_1$ $\mathbf{v}_1$ $\mathbf{v}_1$ $\mathbf{v}_1$
$\mathbf{v}_4$ $\mathbf{v}_2$ $\mathbf{v}_4$
$\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$
$\mathbf{v}_4$ $\mathbf{v}_2$ $\mathbf{v}_4$
$\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$ $\mathbf{v}_2$ $\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$
$\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$
$\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$ $\mathbf{v}_2$ $\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$
$\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$
$\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$ $\mathbf{v}_2$ $\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$
$\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$
$\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$ $\mathbf{v}_2$ $\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$
$\mathbf{v}_1$ $\mathbf{v}_1$ $\mathbf{v}_1$ $\mathbf{v}_1$ $\mathbf{v}_1$ $\mathbf{v}_1$ $\mathbf{v}_1$ $\mathbf{v}_1$
$\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$ $\mathbf{v}_2$ $\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$
$\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$
$\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$ $\mathbf{v}_2$ $\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$
$\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$ $\mathbf{v}_3$
$\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$ $\mathbf{v}_2$ $\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$
$\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$ $\mathbf{v}_5$
$\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$ $\mathbf{v}_2$ $\mathbf{v}_6$ $\mathbf{v}_4$ $\mathbf{v}_6$

(a) 4-QAM.

(b) 16-QAM.

(c) 64-QAM.

Figure 4.7: Inductive edge labeling strategy for 4^n-QAM.

All minimum distance edge labels have a 1 in the least significant bit. Thus the shortest path between two points sharing the same significant bit must have at least two minimum distance edges. Let the distance between nearest neighbors be 1. All the points sharing the same least significant bit are separated by a distance of at least $\sqrt{2}$, which is the largest separation possible for half the points in a Z^2 lattice.

The horizontal minimum distance edges all have a 1 in the next to least significant bit, and the vertical minimum distance edges all have a 0 in the next to least significant bit. Thus points that share the two least significant bits must either be in the same row or in a row that is not adjacent. In either case, the two points are separated by a distance of at least 2, which is the largest separation possible for one fourth of the points in a Z^2 lattice.

The points that share $m \geq 2$ least significant bits all belong to a scaled, shifted $n-1$ case Ungerboeck labeled constellation. These points are separated by the largest distance possible because of the labeling structure inherited from the $n-1$ case.

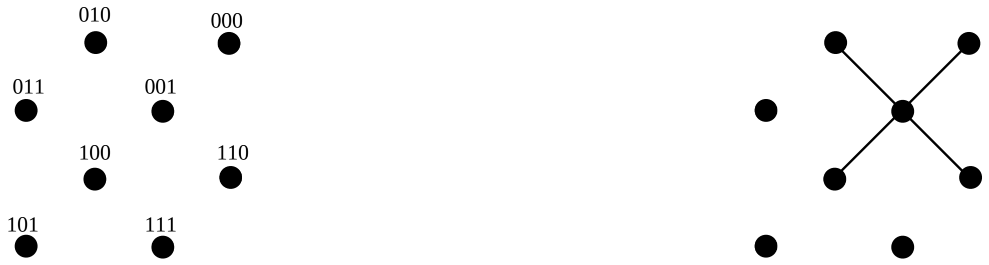
The Gray labeled and Ungerboeck labeled square QAM constellations described above differed only in the choice of basis. In either case, any point can be chosen to have the zero label. When the same point is chosen in both cases to have the zero label, the Ungerboeck labeled and Gray coded constellations are strictly equivalent by Theorem 7. □

The 4-QAM, 8-PSK, and 16-QAM labelings found in Section 4.3 are the same as those produced by the labeling strategies presented in the two proofs above. Thus both Gray coded and Ungerboeck labeled constellations exist that are superior in profile to all but distance equivalent labeled constellations for these cases.

4.4.3 Constellations that cannot be Gray coded

The previous subsection demonstrated that strictly equivalent Ungerboeck labeled and Gray coded constellations can be found for all 2^n-PSK and square 4^n-QAM constellations. However, the two labeling strategies are not entirely equivalent. Ungerboeck labeling can be applied to constellations where Gray coding is not possible. Two examples are given below.

An 8-QAM constellation (also called 8-AMPM or 8-Cross) is shown in Figure 4.8(a) with Ungerboeck labeling. This constellation cannot be labeled with Gray coding. As shown in Figure 4.8(b), some points have four nearest neighbors, but there are only three 3-bit edge labels that have Hamming weight one. By Lemma 3 it is impossible to label every minimum distance edge with a Hamming weight one label.



(a) Ungerboeck labeling.

(b) Four nearest neighbors.

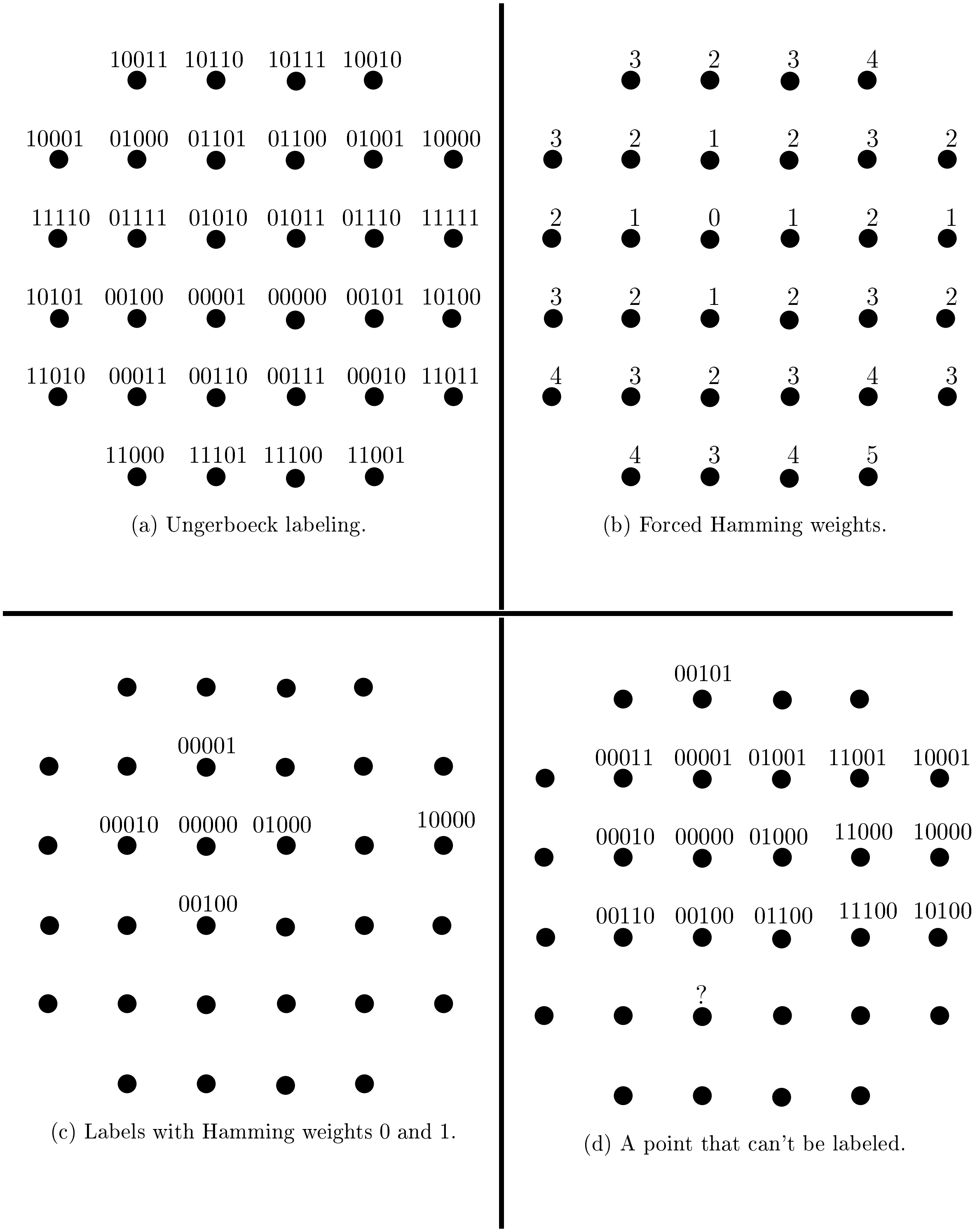
Figure 4.8: An 8-QAM constellation that can be Ungerboeck labeled but not Gray coded.

A common 32-QAM constellation is shown in Figure 4.9(a) with Ungerboeck labeling. As with the 8-QAM constellation above, this constellation cannot be labeled with Gray coding. However, the 32-QAM demonstration is more involved.

The Gray coding definition is based on the edge labels, and is entirely independent of which point is labeled zero for a given edge labeling. There are four symmetric points at the center of the 32-QAM constellation. Without loss of generality, the top left center point is chosen to have the zero label (00000). Having selected this zero label, the Hamming weights of all the labels are forced to be either those shown in Figure 4.9(b) or a reflection of those weights about the diagonal from top left to bottom right.

Without loss of generality, assign the five Hamming weight 1 points in Figure 4.9(b) with labels as shown in Figure 4.9(c). All the labels shown in Figure 4.9(d) are forced by the choice of the Hamming weight 1 labels and the following two requirements:

1. Every minimum distance edge label must have Hamming weight one. (Otherwise the constellation is not Gray coded.)



(a) Ungerboeck labeling.

(b) Forced Hamming weights.

(c) Labels with Hamming weights 0 and 1.

(d) A point that can't be labeled.

Figure 4.9: A 32-QAM constellation that can be Ungerboeck labeled but not Gray coded.

2. Every point must have a different label.
   (Otherwise the constellation is not useful for a trellis code.)

The point labeled “?” must have one of the five labels Hamming distance 1 away from 00100, the point immediately above it, to satisfy requirement 1 above. However, all five of these labels have already been used. Thus the two requirements listed above cannot be simultaneously satisfied, and the 32-QAM constellation cannot be Gray coded.

4.4.4 A Distance Inferior Ungerboeck Labeling

As demonstrated in the previous subsection, there are not always two strictly equivalent constellations, one Gray coded and the other Ungerboeck labeled, for every constellation size and shape. This section demonstrates a more surprising fact about these labeling strategies. An Ungerboeck labeled constellation of a given size and shape need not be distance equivalent to a another Ungerboeck labeled constellation of the same size and shape. Similarly, a Gray coded constellation need not be distance equivalent to another Gray coded constellation of the same size and shape.

Figure 4.10 shows two Ungerboeck labeled 16-QAM constellations that are not distance equivalent; the labeling in Figure 4.10(a) is distance superior to that of Figure 4.10(b). To show this, a labeling is found that is distance equivalent to Figure 4.10(a) but has values of $d^2_{\text{MIN}}(\mathbf{E})$ that are greater than or equal to those of Figure 4.10(b) for every edge label E (with two strict inequalities).

Table 4.6 shows the edge length profile of the constellation in Figure 4.10(a). It is the best possible 16-QAM edge length profile identified in Table 4.3, achieved by applying the edge labeling of Figure 4.3(e) with

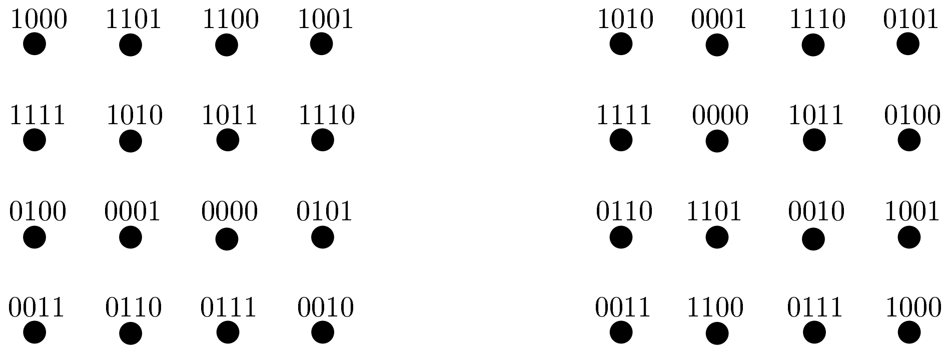
$$\mathbf{v}_1 = 1011 \tag{4.20}$$

$$\mathbf{v}_2 = 0001 \tag{4.21}$$

$$\mathbf{v}_3 = 0111 \tag{4.22}$$

$$\mathbf{v}_4 = 0101. \tag{4.23}$$

Table 4.6 shows the edge labels E for each $d^2_{\text{MIN}}(\mathbf{E})$ for the labeling shown in Figure 4.10(a). Also shown are the edge labels $\hat{\mathbf{E}}$ for each $d^2_{\text{MIN}}(\hat{\mathbf{E}})$ for the distance



(a) Distance Superior.

(b) Distance Inferior.

Figure 4.10: Two Ungerboeck labeled 16-QAM constellations, one distance superior to the other.

$d^2_{\mathrm{MIN}}(\mathbf{E})$	1	1	1	1	2	2	2	2	4	4	5	5	5	5	8
E	1	5	7	b	2	6	a	e	4	c	3	9	d	f	8
$\hat{\mathbf{E}}$	b	f	1	d	e	a	6	2	4	c	5	3	7	9	8

Table 4.6: Edge length profile for Figure 4.10(a). The original edge labels E and those resulting from a change of basis $\hat{\mathbf{E}}$ are given in hexadecimal notation.

$d^2_{\mathrm{MIN}}(\mathbf{E}) = d^2_{\mathrm{MIN}}(\hat{\mathbf{E}})$	1	1	1	1	2	2	2	2	4	4	1	5	5	1	8
E	b	f	1	d	e	a	6	2	4	c	5	3	7	9	8

Table 4.7: Edge lengths for Figure 4.10(b) ordered so that edges E correspond to edges $\hat{\mathbf{E}}$ in Table 4.6.

equivalent labeling that results from the change of basis $\mathbf{v}_i \to \hat{\mathbf{v}}_i$ with

$$\hat{\mathbf{v}}_1 = 1101 \tag{4.24}$$

$$\hat{\mathbf{v}}_2 = 1011 \tag{4.25}$$

$$\hat{\mathbf{v}}_3 = 0001 \tag{4.26}$$

$$\hat{\mathbf{v}}_4 = 1111. \tag{4.27}$$

Table 4.7 shows the values of d^2_{MIN} for the Ungerboeck labeled constellation of Figure 4.10(b). The edge labels E in Table 4.7 have the same order as the edge labels $\hat{\mathbf{E}}$ in Table 4.6. Comparing these two tables confirms that the labeled constellation in Figure 4.10(a) is distance superior to the one shown in Figure 4.10(b) even though both are Ungerboeck labeled.

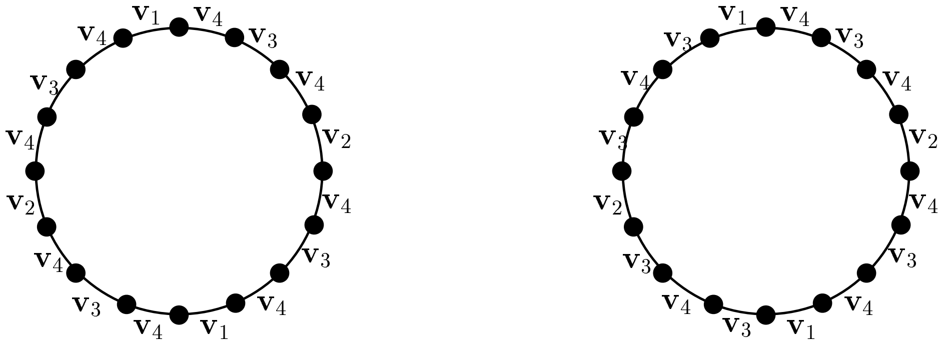
4.4.5 Isometric Labelings

In [22] (see also [23]), Forney introduced the concept of geometrically uniform codes. Many good codes are geometrically uniform. A full treatment of geometric uniformity is beyond the scope of this dissertation. However, the constellation labeling required for a trellis code to be geometrically uniform is of interest.

For a trellis code to be geometrically uniform, the constellation must have an *isometric labeling.* In [22] and [23] the definition of isometric labeling applies only to the coset labels for a constellation that has been partitioned into cosets. Below is a paraphrasing of Proposition 2 in [22], which defines isometric labeling.

Definition 27 *The coset labels of a constellation are an isometric labeling if and only if for any selected coset label* $\mathbf{v}$*, exclusive-oring all the coset labels with* $\mathbf{v}$ *has the same effect as applying a geometric isometry (some combination of translations, rotations, and reflections) to the constellation.*

Isometric labelings make sense in the context of *geometrically uniform signal sets* in which all points are related by isometries that leave the constellation invariant. The 16-QAM constellation is not geometrically uniform since the corner points are not related to the central points by such an isometry. However, the entire half-integer grid (of which the 16-QAM constellation may be considered a subset) is a



(a) Distance Superior.

(b) Distance Inferior.

Figure 4.11: Two Gray coded 16-PSK constellations, one distance superior to the other.

geometrically uniform signal set.

Consider replicating the labelings of Figures 4.10(a) and 4.10(b) over the entire half integer grid. Replication of the distance superior labeling of Figure 4.10(a) would produce an isometric labeling of the half integer grid. However, replication of the distance inferior labeling of Figure 4.10(b) would not. This suggests that there may be a connection between labelings which maximize the edge length profile and isometric labelings. However, the concept of isometric labeling applies only to the coset labelings of certain constellations. The edge length profile is easily computed for the full labeling of any constellation.

4.4.6 A Distance Inferior Gray Coding

As demonstrated in Section 4.3, all Gray coded 16-QAM constellations are distance equivalent, but two Gray coded 16-PSK constellations are not necessarily distance equivalent. Figure 4.11 shows two edge labelings that produce Gray coded 16-PSK constellations whenever $\mathbf{v}_1, \ldots, \mathbf{v}_4$ are chosen to be the four Hamming weight one 4-bit vectors. The edge labeling in Figure 4.11(a) is distance superior to the labeling in Figure 4.11(b) as demonstrated in Table 4.8, for which the distances Δ_i are defined in Figure 4.12.

Edge Label	d^2_{MIN}	
	Figure 4.11(a)	Figure 4.11(b)
$\mathbf{v}_1$	Δ_1	Δ_1
$\mathbf{v}_2$	Δ_1	Δ_1
$\mathbf{v}_3$	Δ_1	Δ_1
$\mathbf{v}_4$	Δ_1	Δ_1
$\mathbf{v}_1 + \mathbf{v}_2$	Δ_8	Δ_6
$\mathbf{v}_1 + \mathbf{v}_3$	Δ_4	Δ_2
$\mathbf{v}_1 + \mathbf{v}_4$	Δ_2	Δ_2
$\mathbf{v}_2 + \mathbf{v}_3$	Δ_4	Δ_2
$\mathbf{v}_2 + \mathbf{v}_4$	Δ_2	Δ_2
$\mathbf{v}_3 + \mathbf{v}_4$	Δ_2	Δ_2
$\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3$	Δ_5	Δ_5
$\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_4$	Δ_7	Δ_5
$\mathbf{v}_1 + \mathbf{v}_3 + \mathbf{v}_4$	Δ_3	Δ_3
$\mathbf{v}_2 + \mathbf{v}_3 + \mathbf{v}_4$	Δ_3	Δ_3
$\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3 + \mathbf{v}_4$	Δ_6	Δ_6

Table 4.8: The edge lengths for the labelings of Figures 4.11(a) and 4.11(b) with the four strict inequalities underlined.

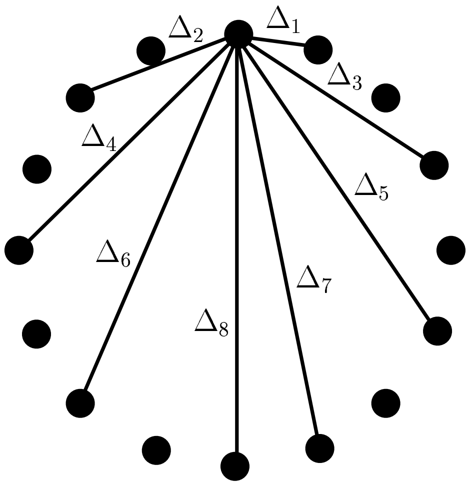


Figure 4.12: Distances between points in a 16-PSK constellation.

4.4.7 Superior in profile Gray coded constellations

The previous subsection identified distance inferior labelings that met the definitions of Gray coding and Ungerboeck labeling. In this subsection it is shown that if any labeling for a 2^n-point PSK or square QAM constellation is superior in profile to all other labelings (except distance equivalent ones), then a Gray coded constellation exists that has this optimality property.

Theorem 10 *For a given 2^n-point PSK or square QAM constellation, if a labeling exists that is superior in profile to all but distance equivalent labelings, then there is a Gray coded labeling that is strictly equivalent to that superior-in-profile labeling.*

Proof: A Gray coded constellation of the specified size and shape exists by the inductive constructions of Section 4.4.2. This constellation is an example of a labeling that has only n minimum d^2_{MIN} edge labels. Thus the superior-in-profile labeled constellation must have at most n minimum d^2_{MIN} edge labels as well. It cannot have fewer than n since the minimum d^2_{MIN} edge labels must contain a basis by Lemma 5. Thus the superior-in-profile constellation must have exactly n minimum d^2_{MIN} edge labels.

Changing the basis comprising the minimum d^2_{MIN} edge labels of the superior-in-profile labeled constellation to the n vectors having Hamming weight 1 produces a distance equivalent Gray coded constellation. This Gray coded constellation is strictly equivalent to the superior-in-profile labeling by Theorem 7. □

Note that not all Gray codings necessarily have this optimality property, since not all Gray codings are distance equivalent.

4.5 Summary

This chapter investigated how best to label a constellation of a given size and shape for use in a trellis code. Constellations related by a change of basis are distance equivalent and lead to distance equivalent trellis codes that have the same values of the design metrics discussed in Chapter 5.

The edge length profile was introduced, and 4-PSK, 8-PSK, and 16-QAM constellations superior in profile to all others were identified. These superior-in-profile constellations have Ungerboeck labeled and Gray coded representations that are distance equivalent. In fact, for all 2^n-PSK and 4^n-QAM constellations, a pair of distance equivalent constellations, one Ungerboeck labeled and the other Gray coded, can always be identified.

However, Gray coding and Ungerboeck labeling are not entirely equivalent. For some constellations such as 8-QAM and 32-QAM, Ungerboeck labeling is possible while Gray coding is not.

It was demonstrated that two constellations of the same size and shape need not be distance equivalent simply because they are both Ungerboeck labeled or because they are both Gray coded. Thus one Ungerboeck labeled constellation may be distance superior to another Ungerboeck labeled constellation of the same size but strictly equivalent to a Gray coded constellation. From this observation, it appears that the edge length profile is more informative about the potential trellis code performance than whether it is Ungerboeck labeled or Gray coded.

This chapter restricted attention to the three constellations used in the code searches of Chapter 6. However, the edge length profile based constellation labeling strategy outlined in this chapter can be extended to larger constellations and constellations in more than two dimensions.

Chapter 5

Capacity and Constellation Size

The metrics presented in Chapter 2 are used in Chapter 6 to search for good trellis codes with 4-PSK (or 4-QAM), 8-PSK, and 16-QAM constellations. Chapter 4 demonstrated how to label these constellations properly, but did not address the choice of constellation size and shape.

This chapter compares the information rates possible using square QAM constellations and PSK constellations of various sizes to the information rate achieved by the optimal complex Gaussian input distribution. The performance of these constellations is also compared with appropriate limiting cases. The limiting case of $M \times M$ QAM constellations is the uniform distribution on a square. The limiting case of M-PSK constellations is the uniform distribution on a ring.

The *capacity* C is the maximum information rate that can be reliably transmitted over a channel [10]. It is computed by maximizing the *mutual information* $I(x; y)$ over the set of possible input distributions P_x:

$$C = \max_{P_x} I(x; y) \tag{5.1}$$

The mutual information $I(x; y)$ for a given input distribution P_x is the maximum rate that can be transmitted reliably using a coding scheme for which the marginal distribution of transmitted values is P_x. For all the channels considered in this dissertation, the P_x that maximizes $I(x; y)$ (i.e. achieves capacity C) is Gaussian. Consider a suboptimal P_x that is a uniform distribution over some particular constellation. Computing $I(x; y)$ for this suboptimal P_x indicates how well codes designed with that constellation can perform relative to the optimal Gaussian codebooks.

Ungerboeck examined $I(x;y)$ in [5] for a variety of QAM and PSK constellations on the AWGN channel. This chapter begins with a review in Section 5.1 of Ungerboeck's analysis of constellation mutual informations on the AWGN channel. (Ungerboeck's analysis did not include the comparisons with the limiting case distributions found here.) Section 5.2 applies this method of analysis to AWGN channels with independent Rayleigh fading.

Four examples of AWGN channels with ISI employing multicarrier modulation are explored in Section 5.3. This section demonstrates that the *high* SNR subchannels determine the constellation size required for good performance on a given ISI channel.

In the limit of high SNR$= \bar{\mathcal{E}}_x/\sigma^2$, the mutual information achievable with a particular constellation $\mathcal{C}$ is simply $\log_2(|\mathcal{C}|)$ where $|\mathcal{C}|$ is the number of points in the constellation. As seen in the plots that follow, the mutual information achievable with a particular constellation is almost exactly that of its limiting case distribution (uniform on the square for QAM or on the ring for PSK) for very low SNR. The area of interest is the transition region between the low SNR and high SNR behavior.

Some care is required in the interpretation of the results presented in this chapter. The mutual information curves that follow indicate how well a constellation can perform in the context of a code with unconstrained complexity and decoding delay. In general, constraints on complexity or decoding delay might make a different constellation size than indicated by these mutual information curves desirable. However, the next chapter presents an example where these mutual information curves are a useful guide to choosing the correct constellation size.

5.1 AWGN Channel

In this section Ungerboeck's mutual information analysis of QAM and PSK is reviewed for AWGN channels. Additional insight is provided by computing the mutual information curves for the limiting case distributions of these constellations. For the AWGN channel, the mutual information associated with any input distribution P_x

can be written as

$$\begin{aligned} I(x;y) &= I(x;x+n) & (5.2)\\ &= h(x+n) - h(x+n|x) & (5.3)\\ &= h(x+n) - h(n) & (5.4)\\ &= h(x+n) - \log_2(2\pi e\sigma^2) & (5.5) \end{aligned}$$

where σ^2 is the variance of the complex Gaussian noise in any one dimension. The *differential entropy* $h(q)$ [10] of the random variable q is defined to be

$$h(q) = -\int f(q)\log_2\left(f(q)\right)dq \qquad (5.6)$$

where $f(q)$ is the probability density function for q.

$I(x;y)$ is found by computing $h(x+n)$ where x and n are both complex random variables. When P_x is a complex Gaussian with covariance matrix

$$\begin{bmatrix} E[x_{\mathrm{R}}x_{\mathrm{R}}] & E[x_{\mathrm{R}}x_{\mathrm{I}}] \\ E[x_{\mathrm{I}}x_{\mathrm{R}}] & E[x_{\mathrm{I}}x_{\mathrm{I}}] \end{bmatrix} = \begin{bmatrix} \bar{\mathcal{E}}_x & 0 \\ 0 & \bar{\mathcal{E}}_x \end{bmatrix}, \qquad (5.7)$$

the differential entropy has a well known expression that leads to the well known complex AWGN channel capacity.

$$\begin{aligned} h(x+n) &= \log_2(2\pi e(\sigma^2 + \bar{\mathcal{E}}_x)) & (5.8)\\ C &= \log_2\left(1 + \frac{\bar{\mathcal{E}}_x}{\sigma^2}\right) & (5.9) \end{aligned}$$

For the other input distributions studied here — QAM, PSK, uniform square, and uniform ring — numerical integration is required to compute $h(x+n)$. These numerical integrations were performed using Mathematica.

5.1.1 QAM Constellations on AWGN Channels

In this subsection the performance on AWGN channels of $M \times M$ square constellations ($M = 2$, 3, 4, 6, and 8) is compared with the performance of the uniform square input distribution and the optimal Gaussian input distribution. Figure 5.1 shows the five

QAM constellations that are considered along with the limiting case of the uniform distribution on a square.

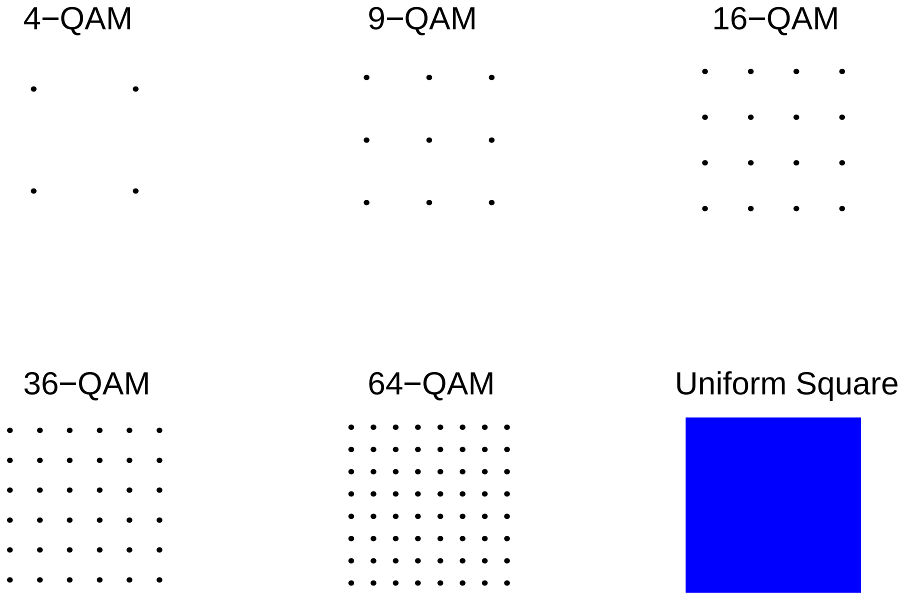


Figure 5.1: Five QAM constellations and the limiting case of the uniform distribution on a square.

For the constellations shown in Figure 5.1, the real and imaginary parts of x are independent. The real and imaginary parts of n are independent regardless of the constellation choice. For both x and n, the real and imaginary parts have identical distributions. Thus the two dimensional integral computing $h(x+n)$ equals twice the one dimensional integral that computes $h(x_{\mathrm{R}} + n_{\mathrm{R}})$.

These one dimensional integrals were computed numerically to produce the curves shown in Figure 5.2. Figure 5.2 shows curves of mutual information vs. SNR for the five constellations and the uniform distribution on a square shown in Figure 5.1 for a range of SNRs. For comparison, the AWGN capacity (5.9) is shown as well.

For a given constellation $\mathcal{C}$, compare its mutual information $I_{\mathcal{C}}$ with channel capacity in Figure 5.2 at the information rate of $\log_2(|\mathcal{C}|) - 1$. As Ungerboeck observed in [5], a given square QAM constellation $\mathcal{C}$ at information rate of $\log_2(|\mathcal{C}|) - 1$ has a mutual information within 1–2 dB of the capacity curve. Thus a QAM constellation of size 2^{R+1} has the potential to yield a code with performance close to optimal at

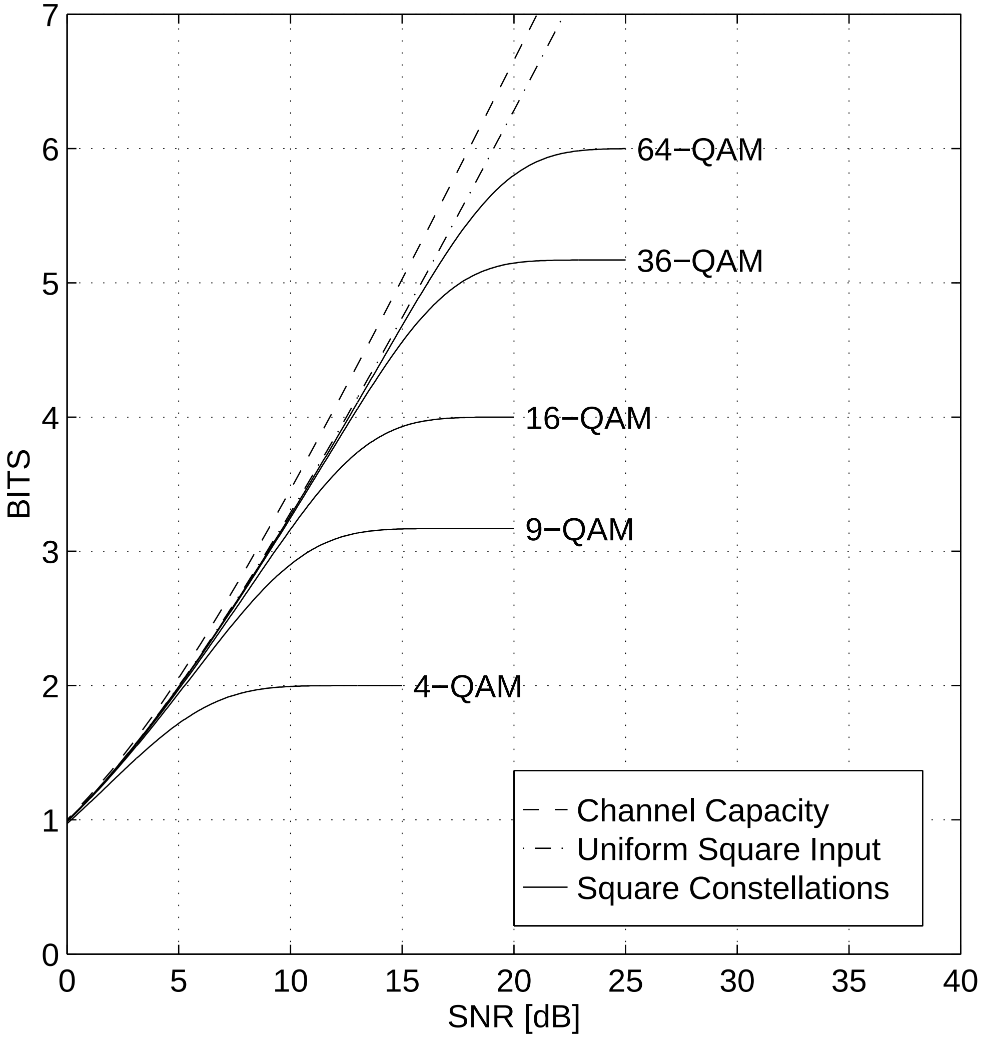


Figure 5.2: Mutual information as a function of SNR for five QAM constellations, uniform distribution on a square, and Gaussian distribution for the AWGN channel.

rate R.

The curves of Figure 5.2 suggest that the trend will continue for larger constellation sizes. That is, for any rate R transmission, a constellation of size 2^{R+1} will have mutual information performance within 1–2 dB of capacity. Ozarow and Wyner confirmed that intuition in [24] by deriving mutual information bounds on the rates achievable by PAM constellations on the real AWGN channel. These bounds immediately apply to the $M \times M$ QAM constellations studied here for the complex AWGN channel by considering the real and imaginary components as independent channels.

No uniform QAM constellation will ever outperform the limiting case of the uniform square input distribution. However, comparing I_{C} with I_{square} at the information rate of $\log_2(|\mathcal{C}|) - 1$, the performance difference is only 0.1 dB. Thus a QAM constellation with 2^{R+1} points will give performance *very* close to that of the uniform square input at rate R.

Compare I_{square} to the channel capacity curve at high SNR. These two lines become asymptotically parallel with a well known separation along the y-axis of $\log_2(\pi e/6) = 0.50923$ bits. This corresponds to a separation along the x-axis of 1.53 dB. This distance has come to be known as the "shaping gain" [25] since it is the amount of gain that can be obtained over square QAM at high SNR by improving the constellation shape.

In the last few years, impressive progress has been made in the quest for the 1.53 dB of shaping gain. Some of the highlights can be found in [25, 26, 27]. To obtain a large part of the 1.53 dB of possible shaping gain, the marginal input distribution must be made less like the uniform and more like a Gaussian. As in [25], this requirement can be stated geometrically as follows.

Consider each valid sequence of constellation points as a single point in a higher dimensional space. If the constellation points are chosen uniformly from a square, then the sequences are uniformly distributed in the volume of an N-cube in the higher dimensional space. However, if the constellation points are chosen from a Gaussian distribution, then the sequences are approximately uniformly distributed in the volume of an N-sphere.

The codes designed in the next chapter transmit one information bit per symbol. At this rate, the possible shaping gain is only about 0.2 dB and need not distract attention from the other issues that truly dominate performance in this region. Furthermore, recent work indicates that shaping gain and coding gain are separable. Thus existing good shaping schemes likely could be combined with the periodic metrics for codes designed to operate at higher bit rates.

5.1.2 PSK Constellations on AWGN Channels

In this subsection, the performance on AWGN channels of M-PSK constellations (M = 4, 8, 16, and 32) is compared with the performance of the uniform ring input distribution and the optimal Gaussian input. Figure 5.3 shows the four PSK constellations that are considered along with the limiting case of the uniform distribution on a ring.

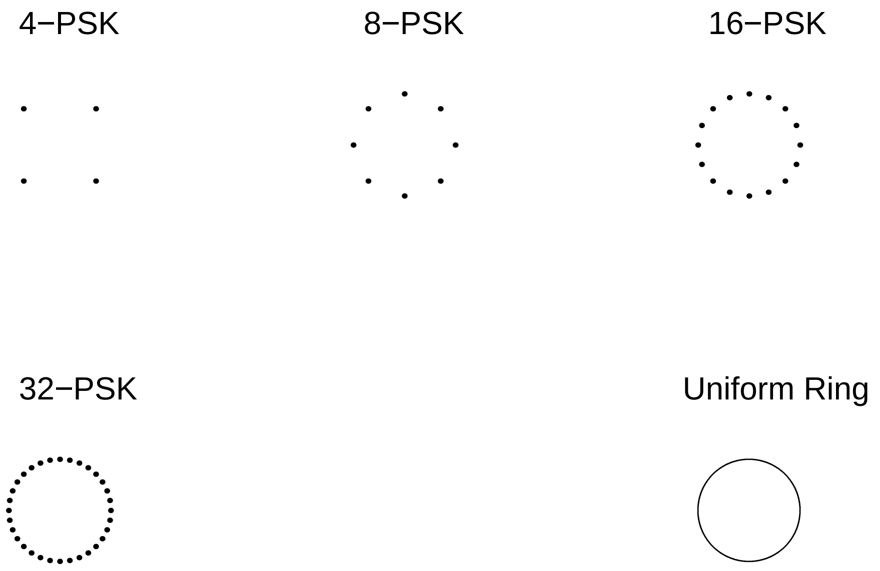


Figure 5.3: Four PSK constellations and the limiting case of the uniform distribution on a ring.

For PSK constellations with more than four points, the computation of $h(x+n)$ cannot be accomplished with a one dimensional numerical integration. By converting to polar coordinates, symmetry allows integration over only π/M radians for an M-PSK constellation. This is an easily computed numerical integration over a smooth, non-oscillatory surface. The value of $h(x+n)$ is then obtained by multiplying the result of this integration by $2M$.

To compute the mutual information for the uniform ring input, the formula derived by Wyner in [28] is applied. This formula is given below (in nats).

$$I_{\text{ring}} = -\int_0^\infty f(x) \ln\left(\frac{f(x)}{x}\right) dx + \ln\left(\frac{2\text{SNR}}{e}\right) \tag{5.10}$$

where

$$f(x) = 2x\text{SNR}e^{-\text{SNR}(1+x^2)} I_0(2x\text{SNR}) \tag{5.11}$$

and I_0 is the modified Bessel function of order zero. This integral has no closed form solution, and again numerical integration is required.

Figure 5.4 shows the mutual information curves for the four PSK constellations shown in Figure 5.3. For comparison, the AWGN channel capacity and the mutual information for the uniform distribution on a ring are also shown.

Examine the performance of the 8-PSK constellation at $\log_2 8 - 1 = 2$ bits. As Ungerboeck observed in [5], the 8-PSK mutual information at 2 bits is only 1.2 dB away from the capacity curve. However, unlike the situation with QAM constellations, Figure 5.4 shows the loss from capacity to be increasing steadily with constellation size.

Similar to the QAM scenario, the PSK constellations never outperform the limiting case (here the uniform ring input distribution). The performance difference between I_{C} for any of the PSK constellations and I_{ring} is negligible at the information rate of $\log_2(|\mathcal{C}|) - 1$. In fact, the difference is negligible for rates less than half a bit below $\log_2(|\mathcal{C}|)$. Thus for a fixed constellation size, PSK constellation performance remains close to I_{ring} at higher rates than those for which QAM constellation performance remains close to I_{square}.

Unfortunately, the limiting case curve I_{ring} for the PSK constellations is much farther from capacity than I_{square} for the QAM constellations. Compare the channel capacity curve to I_{ring} at high SNR. These two lines do not become asymptotically parallel. Instead, the slope of I_{ring} becomes half that of the channel capacity curve.

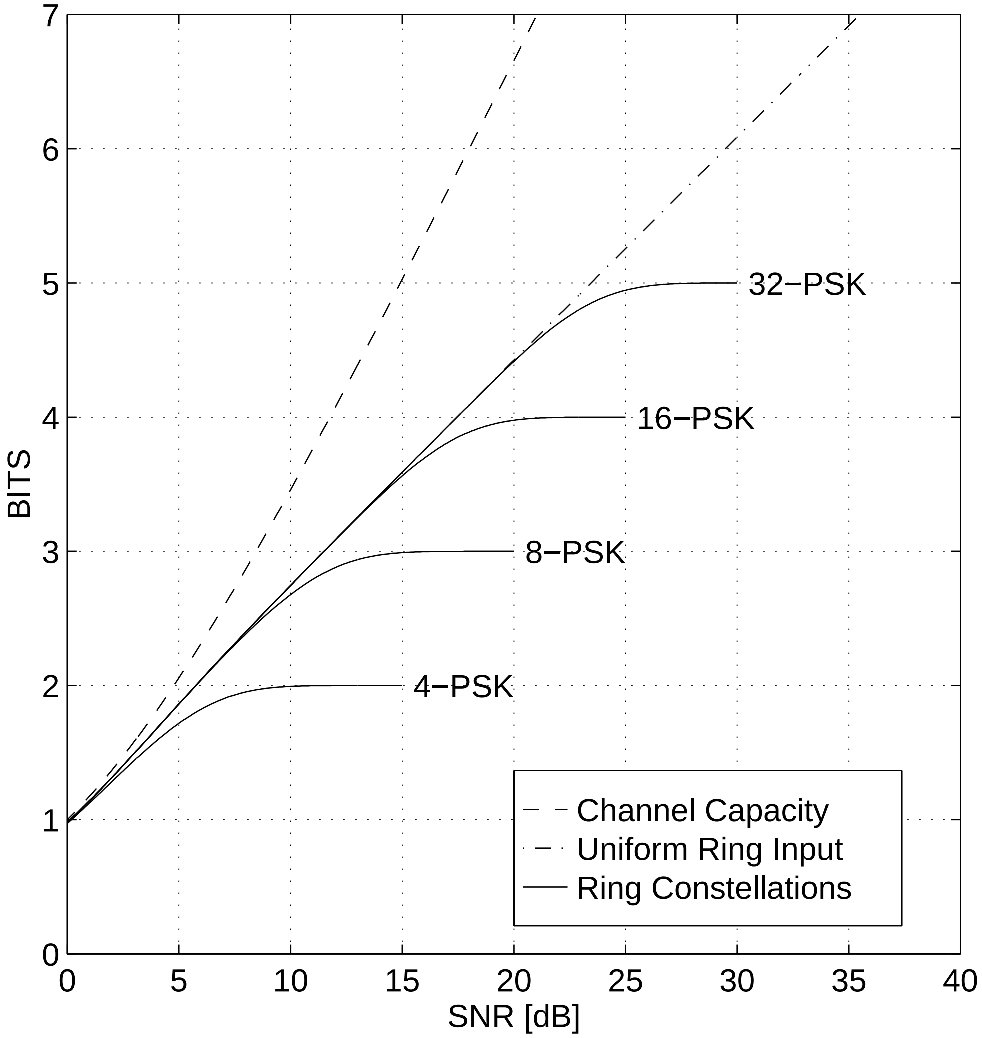


Figure 5.4: Channel capacity and mutual information for four PSK constellations and the uniform distribution on a ring as a function of SNR for the AWGN channel.

In [28], Wyner derived a high SNR approximation of I_{ring} that is useful here:

$$\begin{aligned} I_{\text{ring}} &\approx \frac{1}{2}\log_2\left(\frac{4\pi\text{SNR}}{e}\right) \qquad \text{from [28]} && (5.12) \\ &= \frac{1}{2}\log_2(\text{SNR}) + \frac{1}{2}\log_2\left(\frac{4\pi}{e}\right) && (5.13) \\ &\approx \frac{1}{2}C + 1.1044 \text{ bits} && (5.14) \end{aligned}$$

where C is the AWGN channel capacity of (5.9).

Thus when restricted to use a PSK constellation, a constellation of size 2^{R+1} is certainly sufficient to provide performance close to that of the uniform ring input. However, this input allows transmission only at rates about half those possible for QAM constellations of the same size at given large SNR.

Given this analysis, it is necessary to comment on situations where it might be desirable to use a PSK constellation despite its inferior mutual information performance in Figure 5.4. A PSK constellation can be received without regard for amplitude ambiguity at the receiver. Furthermore, with differential encoding, the transmission can also be made insensitive to slowly varying phase ambiguity. Finally, a constant amplitude transmission such as PSK allows the use of economical, power efficient amplifiers.

Often, the use of nonlinear amplifiers places dual constraints of both average power and peak power on the transmitted signal. Shamai has derived the optimal input distributions for transmission under concurrent average power and peak power constraints in [29]. The optimal distributions under these dual constraints are always concentric circles. Under certain conditions when peak power is the dominant constraint, the optimal distribution is the uniform distribution on the peak power ring.

5.2 Independent Rayleigh Fading

Consider again the independent Rayleigh fading channel discussed in Section 2.3. The apparent independence is the result of interleaving. Thus, the fading is sometimes slow enough that the fading scale factors a_i can be estimated. As has been the case

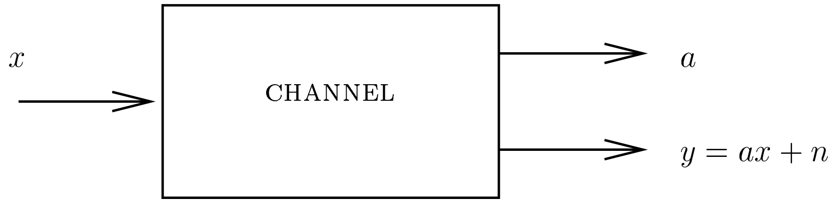


Figure 5.5: Model of a fading channel with perfect channel state information at the receiver.

throughout this dissertation, estimation error is neglected; it is assumed that the scale factors a_i are known exactly at the receiver, as illustrated in the fading channel model shown in Figure 5.5.

Applying the chain rule for mutual information (Theorem 2.5.2 in [10]):

$$\begin{aligned} I(x; y, a) &= I(x; a) + I(x; y|a) && (5.15) \\ &= I(x; y|a) \qquad \text{since } x - a && (5.16) \end{aligned}$$

Using the definition of conditional mutual information in [10] and the fact that scaling preserves information,

$$\begin{aligned} I(x; y|a) &= \int_a I(x; ax + n)p(a)da \qquad \text{by definition} && (5.17) \\ &= \int_a I(x; x + \frac{n}{a})p(a)da. && (5.18) \end{aligned}$$

For any fixed scalar a, the mutual information $I(x; x + n/a)$ is given by (5.5)

$$I\left(x; x + \frac{n}{a}\right) = h\left(x + \frac{n}{a}\right) - \log_2\left(\frac{2\pi e\sigma^2}{|a|^2}\right) \tag{5.19}$$

Note that $I(x; x + n/a) = I(x; x + n/|a|)$. The conditional mutual information in (5.18) depends on the magnitude $r = |a|$, which has the Rayleigh PDF given in (2.16), but not on the phase of a. Thus,

$$I(x; y, a) = \int_r^\infty I(x; x + n/r)\frac{r\ e^{r^2/2\gamma^2}}{\gamma^2}dr\,. \tag{5.20}$$

The integral in (5.20) is approximated for complex Gaussian inputs x in [30] (but see also [31] for an important correction) where it is considered an "average capacity." However, for the channel described in Figure 5.5 it is in fact the usual Shannon capacity since complex Gaussian inputs maximize the mutual information in (5.20).

Reliable transmission at this capacity does *not* require that the transmitter know the fading sequence $\mathbf{a}$. If the transmitter does know the fading sequence $\mathbf{a}$, an even higher capacity is achievable as shown in [32]. However, this dissertation restricts attention to the case where the transmitter does not have this information.

See [33] for a discussion of how decoding delay constraints affect the achievability of the capacity obtained by maximizing (5.20). If the decoding delay is small enough relative to the time constant of the fading, then no capacity is achievable. The possibility of a severe fade lasting longer than the acceptable coding delay precludes completely reliable transmission at any positive rate. In these cases, the probability of outage as a function of data rate becomes the performance measure of choice.

The approximations in [30, 31] are useful for analysis. However, for the visual comparisons of this chapter, computing (5.20) numerically provides better accuracy. For the PSK and QAM constellations, such analytic approximations are unavailable. With the numerical integration of (5.20) in mind, the $I(x; x+n)$ curves shown in the previous section were computed at closely spaced SNR values so that interpolated curves could be used as kernels for numerical integrations computing the mutual informations for those constellations in the presence of Rayleigh fading.

Figures 5.6 and 5.7 show the mutual information curves for the QAM and PSK constellations used on an AWGN channel with Rayleigh fading. For both QAM and PSK constellations, the mutual information curves deviate from the limiting case mutual information curves at lower rates than on the AWGN channel.

Examining Figure 5.6, the performance loss from I_{square} at $\log_2(|\mathcal{C}|) - 1$ is around 0.5 dB for the 4-QAM constellation and grows as the constellation size grows. For the 64-QAM constellation, the loss is around 1.5 dB at $I_{\text{64-QAM}} = 5$ bits. This performance loss is relatively small, but does indicate that larger constellations may be useful in the Rayleigh fading environment. For the QAM constellations, a size 2^{R+2} constellation is required for performance to be within 0.1 dB of I_{square} at rate R for an AWGN channel with independent Rayleigh fading. This is twice the size

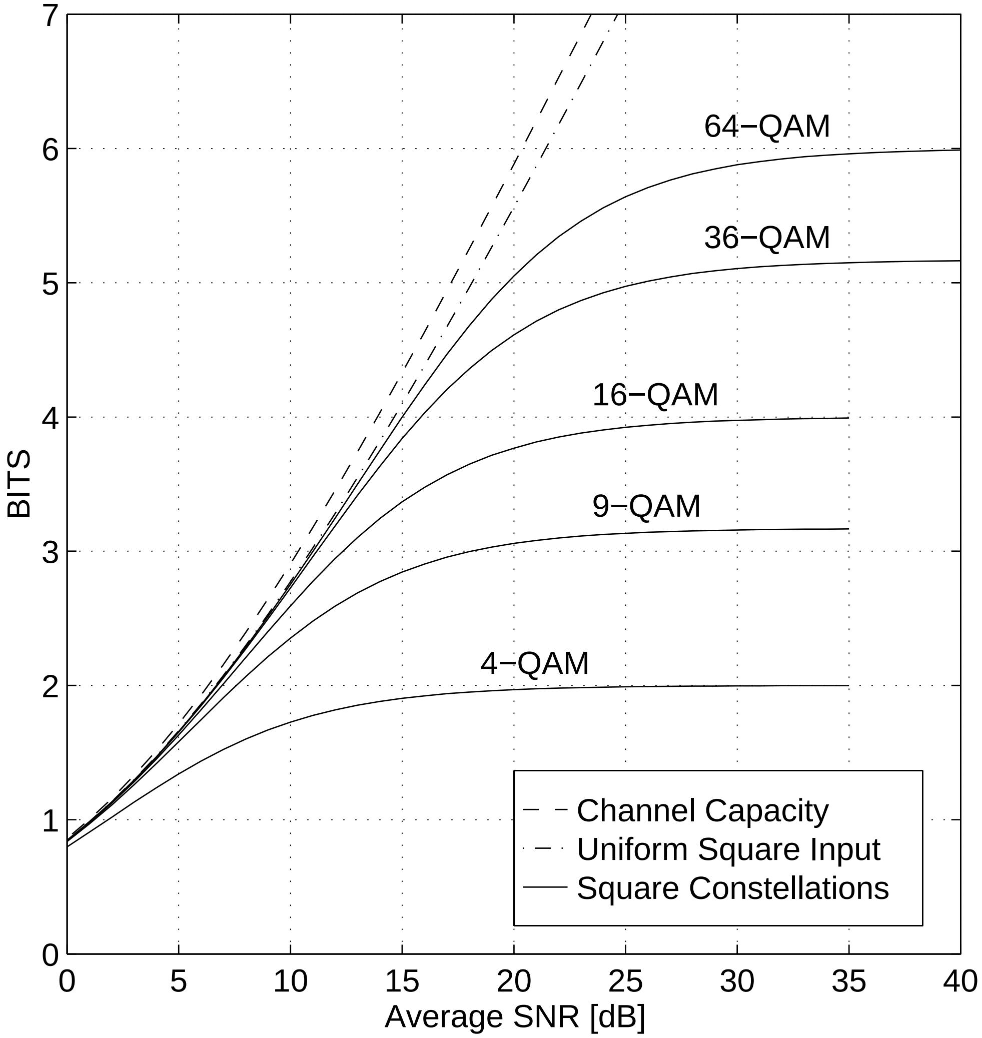


Figure 5.6: Mutual information on an AWGN channel with independent Rayleigh fading as a function of average SNR for five QAM constellations, uniform distribution on a square, and Gaussian distribution.

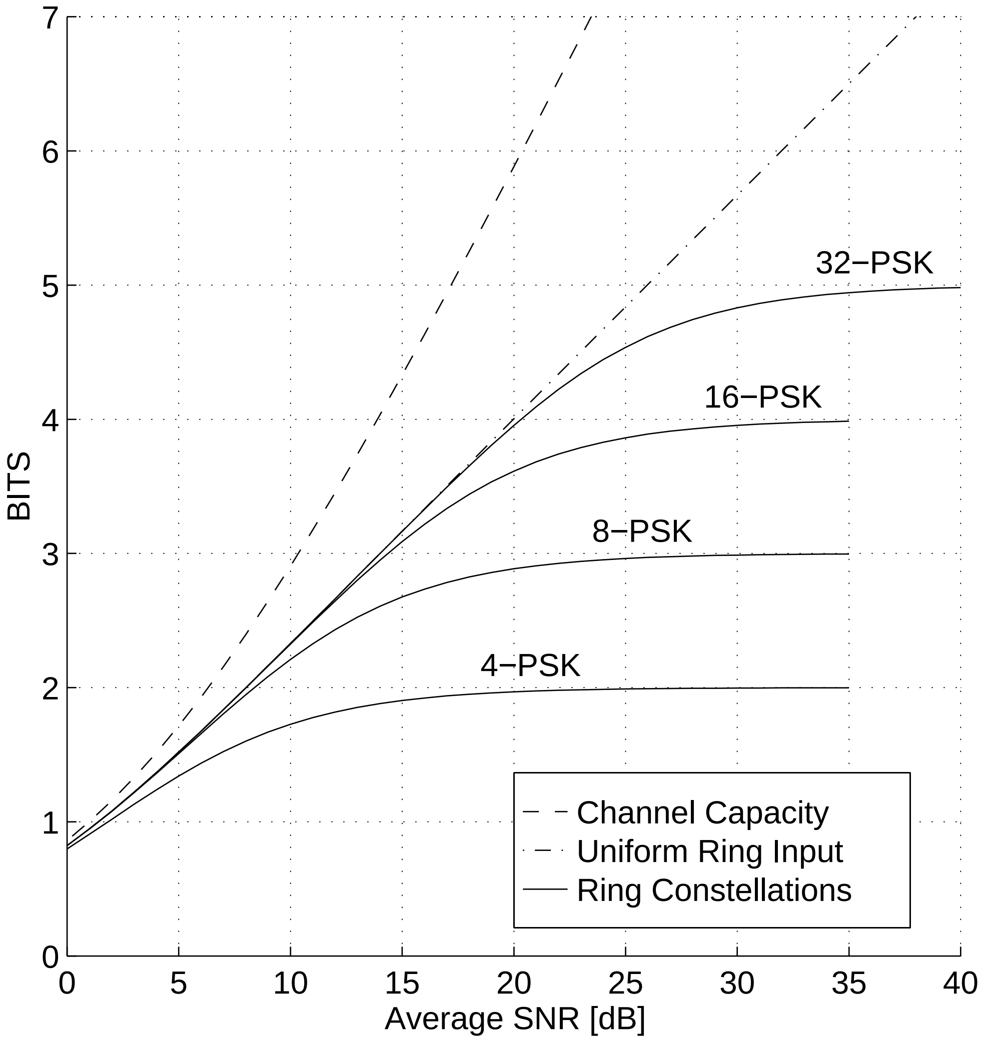


Figure 5.7: Mutual information on an AWGN channel with independent Rayleigh fading as a function of average SNR for four PSK constellations, uniform distribution on a ring, and Gaussian distribution.

required for an AWGN channel without Rayleigh fading.

Ho, Cavers, and Varaldi suggested in [34] that larger constellation sizes would be useful for Rayleigh fading channels. Specifically, they claimed that going from an 8-PSK to a 16-QAM constellation for transmission of 2 bits/symbol (i.e., going from 2^{R+1} to 2^{R+2}) provided a 5 dB performance improvement on a Rayleigh fading channel. Figures 5.6 and 5.7 support the increase from 2^{R+1} to 2^{R+2}, but show only about 1 dB of improvement at $R = 2$ bits going from 8-PSK to 16-QAM. Perhaps the mutual information analysis underestimates the improvement possible going from 8-PSK to 16-QAM since limits on complexity and decoding delay are neglected.

Still, the 5 dB claim by Ho *et al.* requires careful interpretation. This claim was based on a comparison of a standard 8-PSK trellis code with 8 states (3 memory elements in the encoder) to a specially designed 16-QAM code with 16 states (4 memory elements). The 16-QAM code was designed so that its decoding complexity was comparable to that of the 8-PSK code despite having an extra memory element. Thus, in a sense, this was a fair comparison.

There is no question that the 16-QAM code has good performance. However, since different code design strategies were applied to the two constellations, the comparison is not entirely conclusive about whether or not 8-PSK codes with similar performance exist. On this point, the different number of memory elements used by the two encoders is an important consideration.

As discussed in Chapter 2, effective code length ECL has a large impact on performance in independent Rayleigh fading scenarios. Theorem 1 in Chapter 3 showed that

$$\text{ECL} \leq \left\lfloor \frac{\nu}{k} \right\rfloor + 1, \tag{5.21}$$

where k is the number of information bits transmitted per symbol and ν is the number of memory elements in the encoder.

The codes compared by Ho *et al.* both achieved their respective ECL bounds; the 8-PSK code had ECL$= 2$ while the 16-QAM code had ECL$= 3$. The different ECL values imply a large performance gap between the two codes. The remaining question is whether a code with 4 memory elements could be designed for 8-PSK to have ECL$= 3$ and to be similar in decoding complexity to the 16-QAM code. Such a code might have a performance much better than that of the 8-PSK code with 3 memory elements considered in [34]

In the next chapter Rayleigh fading channel codes are designed with a fixed number of memory elements for transmission of 1 bit per symbol using three different constellation sizes. The same code design strategy is used for all three constellations. In this search, all three constellations had the same ECL, each achieving the bound of (5.21). Consistent with Figures 5.6 and 5.7 and with the claims of [34], there was improvement (in terms of code product distance CPD) when the constellation was increased in size from 2^{R+1} to 2^{R+2}. However, no further improvement in the Rayleigh fading metrics was seen when the constellation was further increased to 2^{R+3}.

5.3 Multicarrier Modulation for Intersymbol Interference

In this section the broadcast scenario where one information sequence is transmitted to several receivers is considered. Figure 5.8 shows an example of such a scenario where digital video signals are broadcast to many television sets. The physical path between the transmitter and each receiver is different, hence a different impulse response is associated with each receiver. In practical broadcast scenarios, the transmitter usually does not know the impulse responses of the various receivers.

A common way to deal with the multiplicity of impulse responses is for each receiver to use equalization to convert the intersymbol interference (ISI) channel approximately to an AWGN channel. However, these equalization schemes work best when some type of channel specific precoding is performed by the transmitter, as explored in Chapter 7. This type of transmitter specialization is difficult when the transmitter is broadcasting to several receivers.

This section explores the effect of constellation size on mutual information when multicarrier modulation (MCM) is used for broadcast transmission over multiple impulse responses. Tutorials on the details of MCM can be found in [35] and [36]. For this dissertation, the trellis code design problem presented by MCM is of primary interest. This problem can be understood without exploring all the details of MCM.

MCM operates on blocks of N encoder outputs $x_1, \ldots, x_N$ as shown in Figure 5.9.

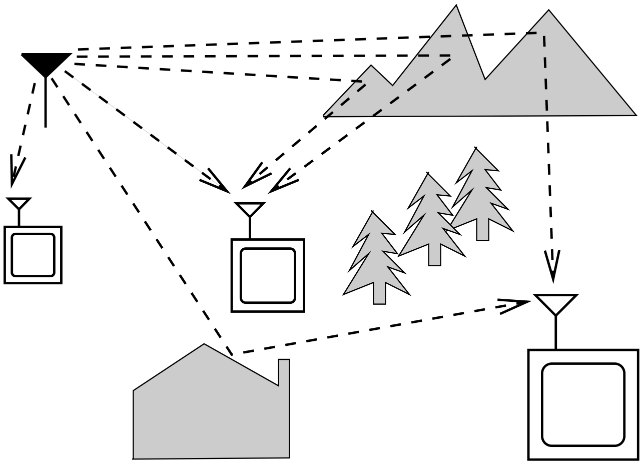


Figure 5.8: A digital video signal is broadcast to several television sets.

The block $x_1, \dots, x_N$ is a block of frequency domain subsymbols, which are constellation points. This block is converted to a block of time domain samples by an inverse Fourier transform (IDFT). The block of time domain samples (along with a cyclic prefix not shown in Figure 5.9) is transmitted over an ISI channel with AWGN. The receiver performs a DFT on the received block (after removing the cyclic prefix) to produce the received block of frequency domain symbols $y_1, \dots, y_N$, which are scaled constellation points distorted by AWGN.

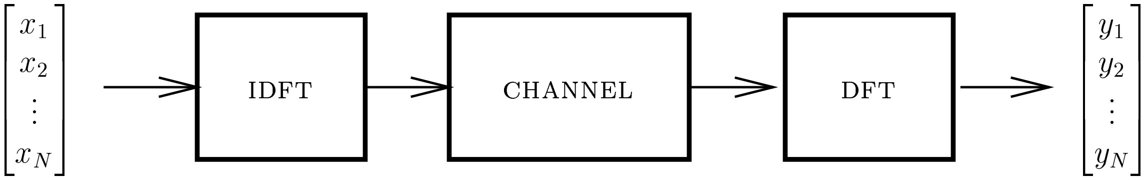

$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$
IDFT
CHANNEL
DFT
$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$

Figure 5.9: Multicarrier modulation system.

The combined effect of IDFT, ISI channel, and DFT produces the set of N parallel subchannels shown in Figure 5.10. The complex scale factors are the values of the DFT of the channel impulse response. The complex noise terms n_i are complex

AWGN with independent real and imaginary parts each having variance σ^2. Figure 5.10 is equivalent to Figure 5.5 with the values of a_i repeating every N symbols.

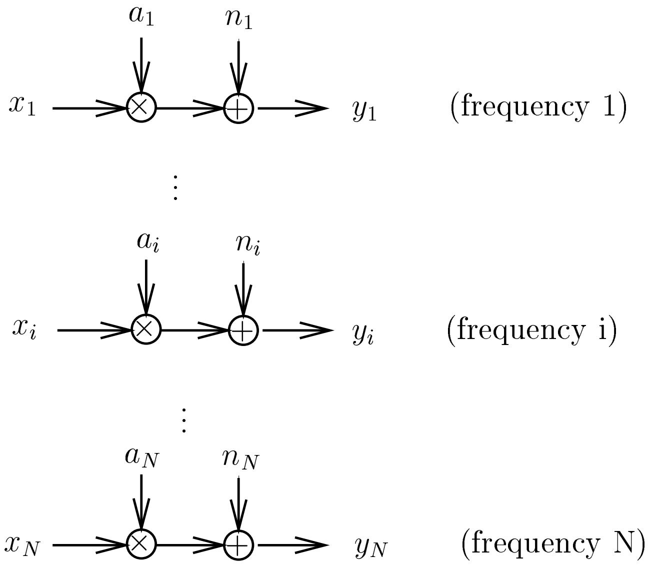


Figure 5.10: Parallel subchannels in frequency.

The mutual information per block for the channel shown in Figure 5.10 is

$$\begin{aligned} I &= \sum_{i=1}^{N} I\left(x_i; x_i + \frac{n_i}{a_i}\right) && (5.22)\\ &= \sum_{i=1}^{N} \log_2\left(1 + \frac{|a_i|^2 E|x_i|^2}{\sigma^2}\right) \qquad \text{for Gaussian } x_i. && (5.23)\end{aligned}$$

In a point to point communication scenario where the transmitter knows the values $a_1, \ldots, a_N$, the mutual information in (5.23) can be maximized by using the water-filling distribution of $E|x_i|^2$ [37]. In practice, a constant value of $E|x_i|^2$ is usually maintained except for sufficiently poor subchannels where $E|x_i|^2$ is set to zero. When the transmitter knows $a_1, \ldots, a_N$, often denser constellation are used on channels with larger values of $|a_i|^2/\sigma^2$ as in [38].

In the point-to-multipoint scenario illustrated in Figure 5.8 (or a point-to-point scenario where the transmitter does not know the values of $a_1, \ldots, a_N$), specialization of $E|x_i|^2$ or the constellation size is not possible. Usually, one symbol power $E|x_i|^2 =$

$\mathcal{E}_x$ and one constellation is used for all the symbols [39, 40]. In this case, the mutual information per block is

$$I = \sum_{i=1}^{N} \log_2 \left(1 + \frac{|a_i|^2 \mathcal{E}_x}{2\sigma^2}\right) \tag{5.24}$$

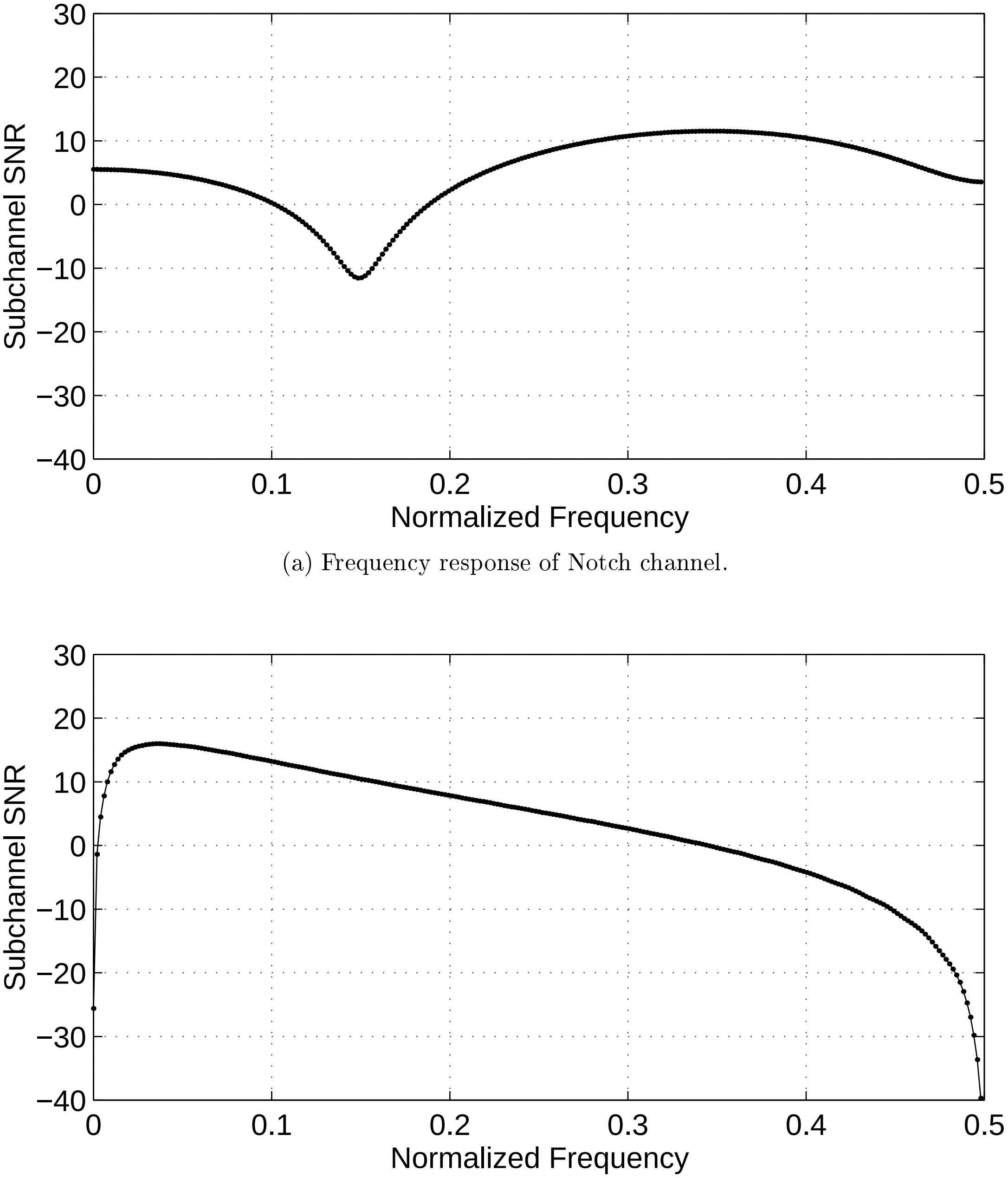
for the optimal case of Gaussian inputs x_i. This is less than or equal to the capacity achieved by waterfilling, but waterfilling is not possible when the values $a_1, \ldots, a_N$ are not known.

Shannon's fundamental coding theorems [41, 37] ensure that for each set of values $a_1, \ldots, a_N$ there is a code with $E|x_i|^2 = \mathcal{E}_x$ that achieves reliable transmission at the per block rate given in (5.24). However, this fact alone is not enough since the transmitter does not know $a_1, \ldots, a_N$ and thus cannot choose between different codes meant for different cases of $a_1, \ldots, a_N$. Fortunately, Root and Variaya's compound Gaussian channel results [42] show that there must exist a "universal" code that reliably transmits at rate NR per block over all sets of values $a_1, \ldots, a_N$ for which

$$R \leq \frac{1}{N} \sum_{i=1}^{N} \log_2 \left(1 + \frac{|a_i|^2 \mathcal{E}_x}{2\sigma^2}\right). \tag{5.25}$$

The "universal" code would require at least 2^{NR} points in its constellation to transmit reliably in the case where all the values of a_i are zero except for one extremely large value. This case is not of practical concern, and the plots that follow do show that reasonable constellation sizes can support transmission over a wide range of frequency selective channels. Mutual information curves for the previously considered QAM and PSK constellations are now shown for four examples of ISI channels.

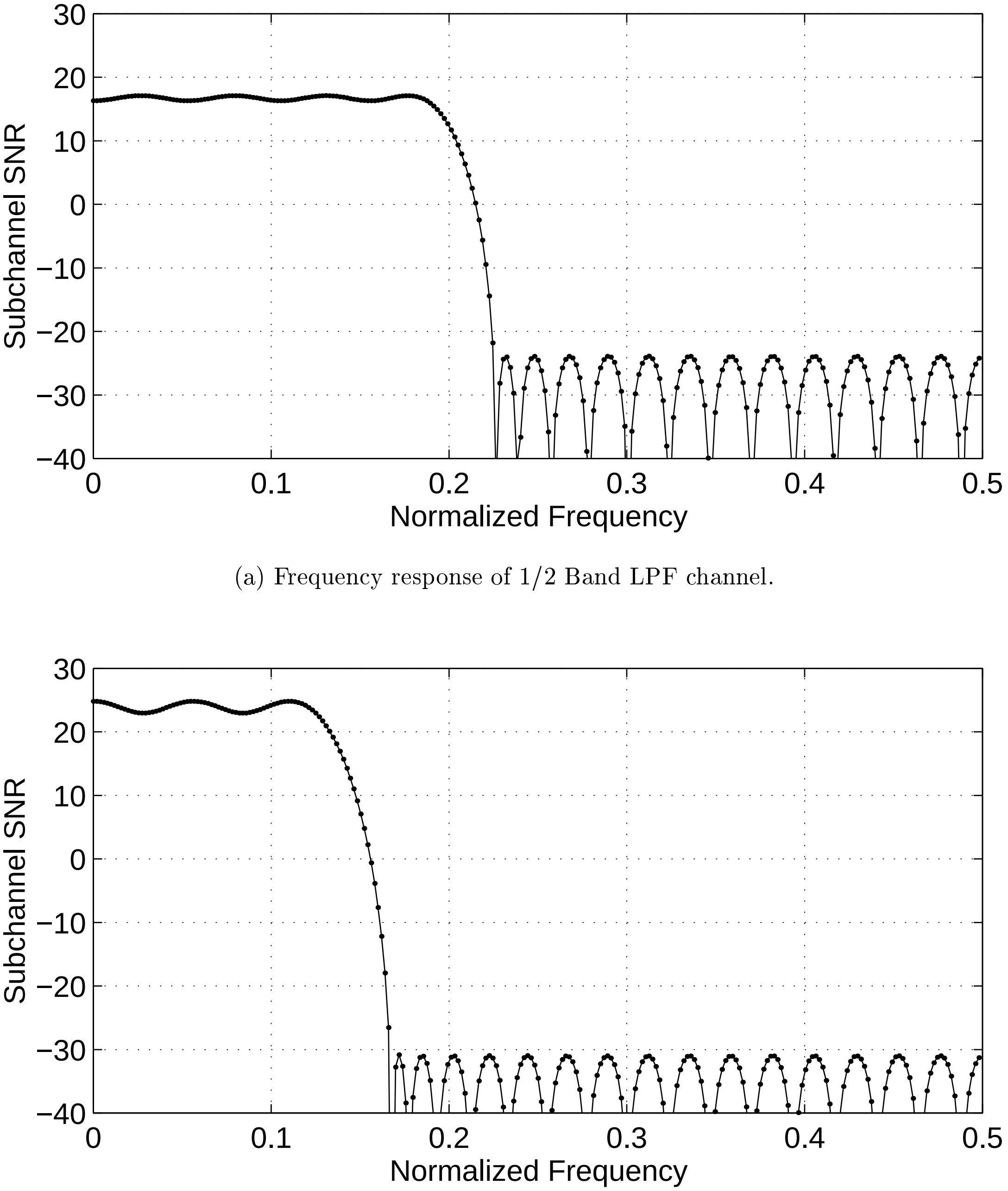
For these four ISI channels, Figures 5.11 and 5.12 show the subchannel SNRs $\mathcal{E}_x|a_i|^2/2\sigma^2$ for MCM with $N = 512$. Figure 5.13 shows the corresponding four impulse responses. The Notch channel (studied by Sari in [39]) has a mild notch at a normalized frequency of 0.15. The Ramp channel (was used in by Cioffi in [36]) shows a monotonic decrease in SNR except for a notch at DC. The 1/2 and 1/3 Band low pass filters (LPFs) were designed using the Remez exchange algorithm of Parks and McClellen. They are included here as examples of extremely severe frequency selectivity.



(a) Frequency response of Notch channel.

(b) Frequency response of Ramp channel.

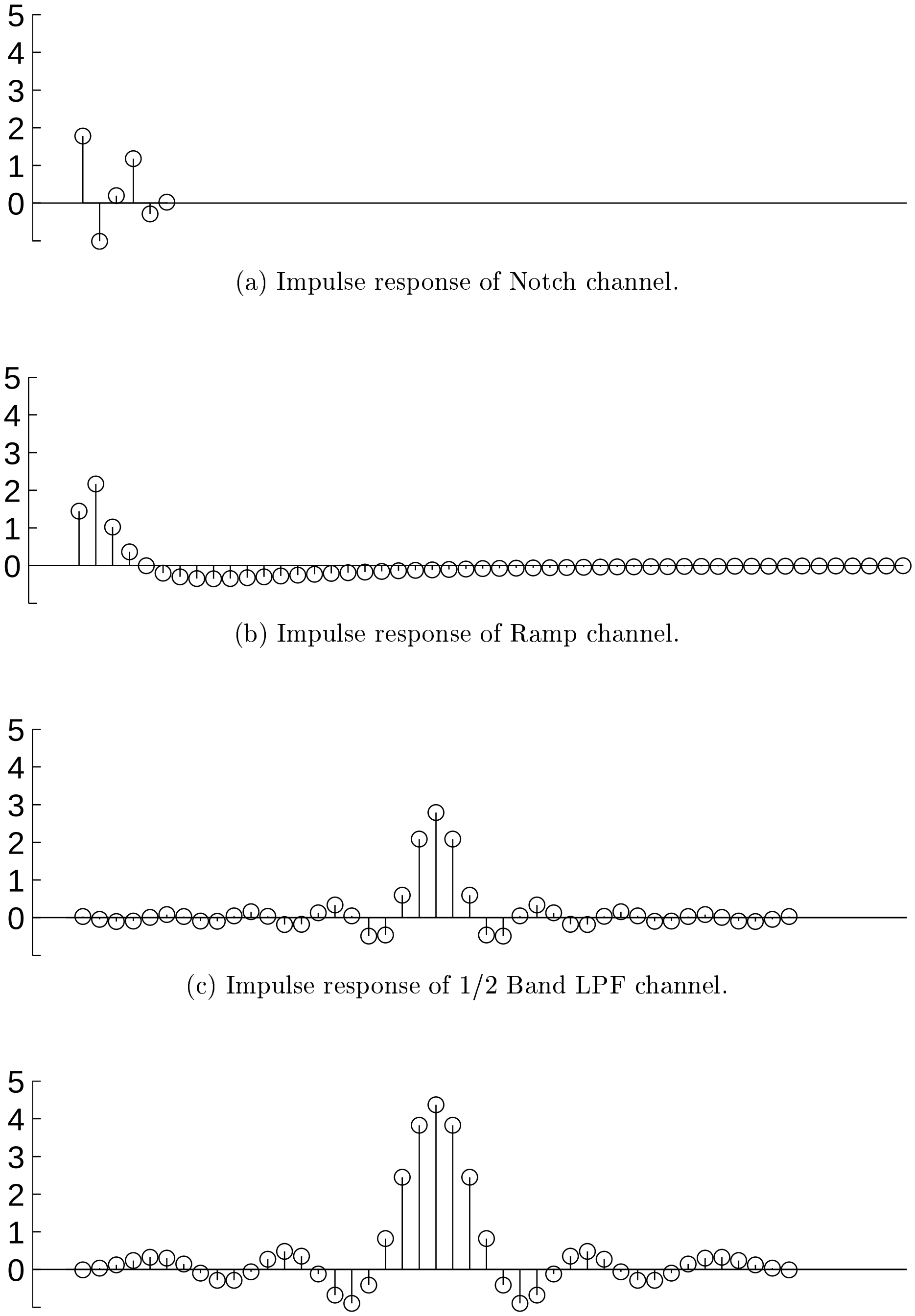
Figure 5.11: Frequency response (subchannel SNRs) for Notch and Ramp channels with an overall equivalent AWGN SNR of 6 dB.



(a) Frequency response of 1/2 Band LPF channel.

(b) Frequency response of 1/3 Band LPF channel.

Figure 5.12: Frequency response (subchannel SNRs) for 1/2 Band and 1/3 Band LPF channels with an overall equivalent AWGN SNR of 6 dB.



(a) Impulse response of Notch channel.

(b) Impulse response of Ramp channel.

(c) Impulse response of 1/2 Band LPF channel.

(d) Impulse response of 1/3 Band LPF channel.

Figure 5.13: Impulse responses for the Notch, 1/2 Band LPF, and 1/3 Band LPF channels.

Figures 5.14–5.17 show the QAM mutual information curves for multicarrier modulation with 512 subchannels on the four channels studied. Figures 5.18–5.21 show the PSK mutual information curves for these channels. As in the Rayleigh fading mutual information integrals, the AWGN mutual information curves of Figures 5.2 and 5.4 are used as kernels in summation of (5.22) to compute the mutual informations of the various constellations on these channels.

To plot the mutual information curves in a way that allows comparison between channels, those channels having the same maximum achievable rate for a given transmitter subchannel power level are considered equivalent. These channels will in general have different input SNRs and different output SNRs. The SNR assigned to these equivalent channels for plotting purposes is the SNR of an AWGN channel with the same capacity. This SNR is called the *equivalent AWGN SNR*.

If the values $a_1, \ldots, a_N$ are the same, then the mutual information curves are those for the AWGN channel shown in Section 5.1. When there is variation in the values of $a_1, \ldots, a_N$, some of the subchannel SNRs will be above the equivalent AWGN SNR and some will be below. The overall loss from the mutual information of the limiting case (uniform square or ring) will be an average of the losses on each subchannel. The loss from the limiting case increases with SNR so that the increased loss on the high SNR (higher than the equivalent AWGN SNR) subchannels dominates the decreased loss on the lower SNR subchannels. Thus the mutual information curves will be below those for the AWGN channel.

The amount by which the mutual information curves are below those for the AWGN channels depends a great deal on the specifics of the values $a_1, \ldots, a_N$. In general, more variation causes more degradation in the mutual information curves.

The channels are ordered so that the SNR variation is increasing. For the Notch channel, the mutual information curves are similar to those for Rayleigh fading seen in the previous section. For the Ramp channel, further degradation is apparent, and for the low pass filter channels the degradation is dramatic.

To further explore the reasons for this degradation, consider the 4-QAM (equivalently 4-PSK) constellation used on the subchannels of the 1/2 Band LPF channel at an equivalent AWGN SNR of 6 dB. These subchannel SNRs are shown in Figure 5.12(a). Since the channel is essentially a step in the frequency domain, the subchannels fall into two categories. There is a set of low SNR subchannels (less than -20 dB) for which the 4-QAM constellation performs close to the limiting case. Actually, these subchannels are so poor that they support practically no information transmission.

However, there is also a set of high SNR subchannels (above 15 dB) for which the 4-QAM constellation's limitation to 2 bits is a severe handicap. (Compare the 4-QAM mutual information to capacity at 15 dB in Figure 5.2.) This is generally the case. A constellation's mutual information performance under SNR variation is limited by the high SNR subchannels rather than the low SNR subchannels. In general, the constellation size required to handle large SNR variations is determined by how high the SNR can be rather than by how low it can be.

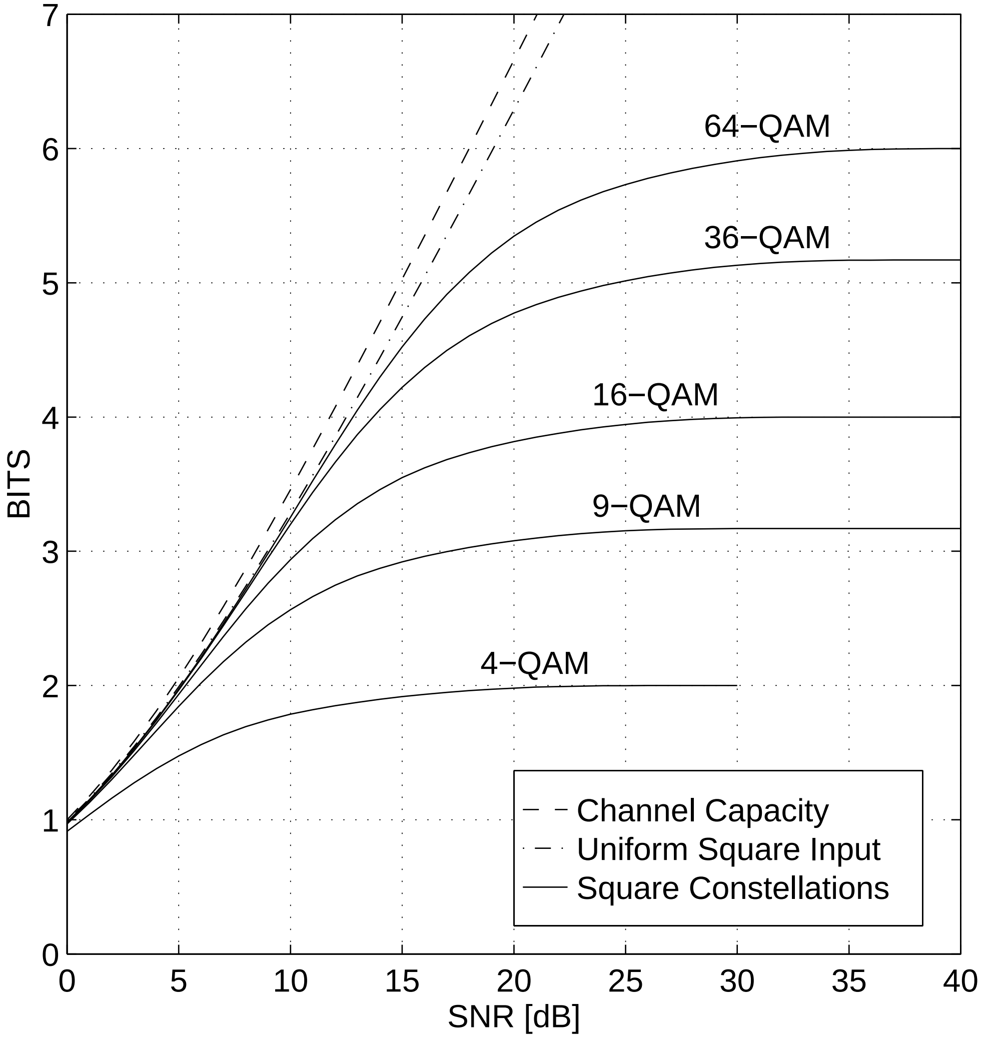


Figure 5.14: Mutual information (using MCM) on the Notch channel as a function of equivalent AWGN SNR for five QAM constellations, uniform distribution on a square, and Gaussian distribution.

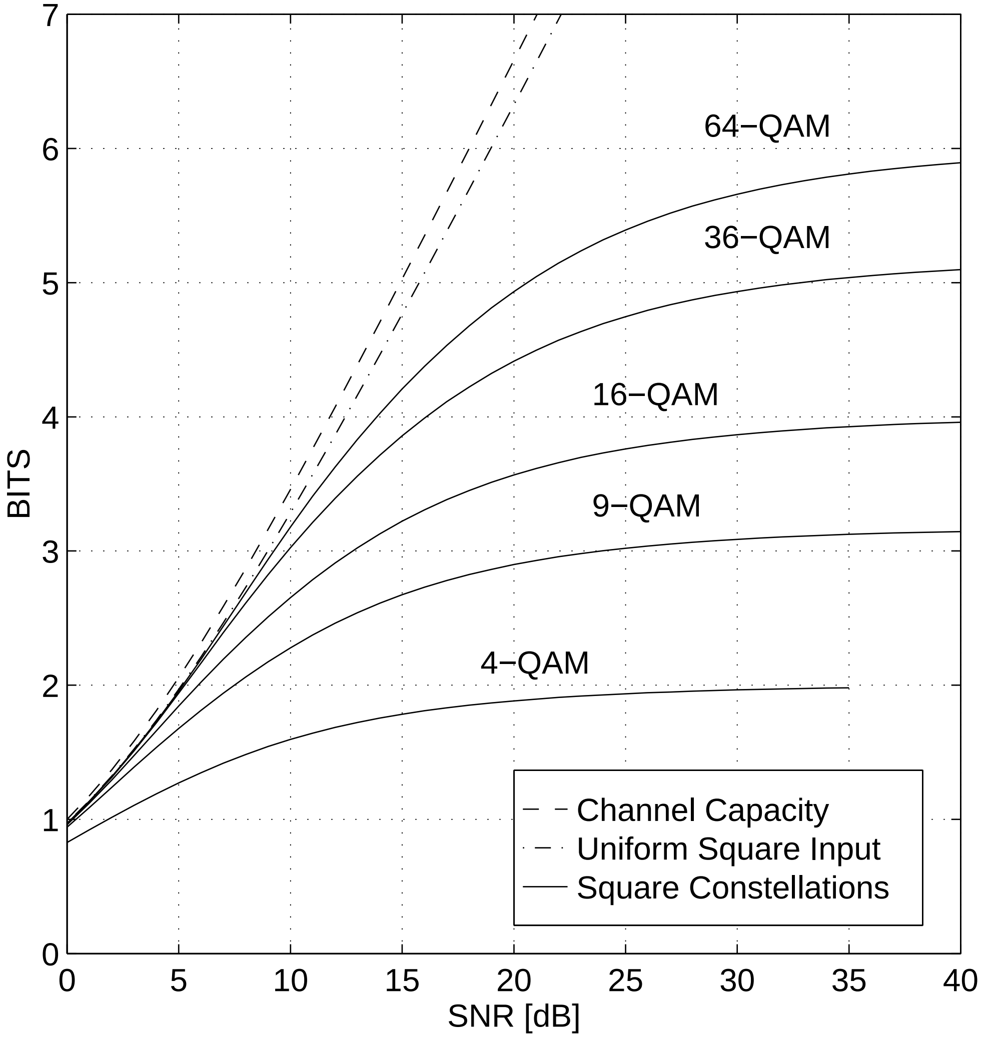


Figure 5.15: Mutual information (using MCM) on the Ramp channel as a function of equivalent AWGN SNR for five QAM constellations, uniform distribution on a square, and Gaussian distribution.

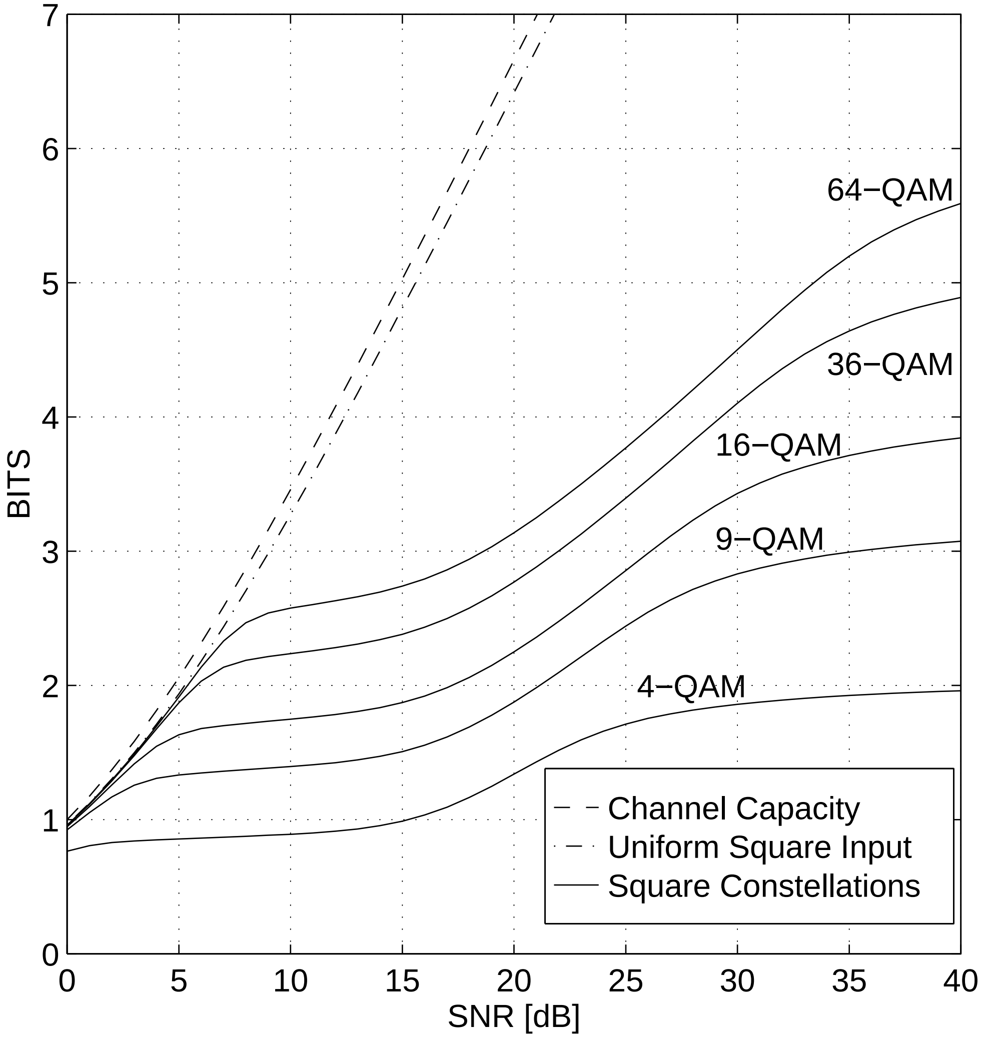


Figure 5.16: Mutual information (using MCM) on the 1/2 Band LPF channel as a function of equivalent AWGN SNR for five QAM constellations, uniform distribution on a square, and Gaussian distribution.

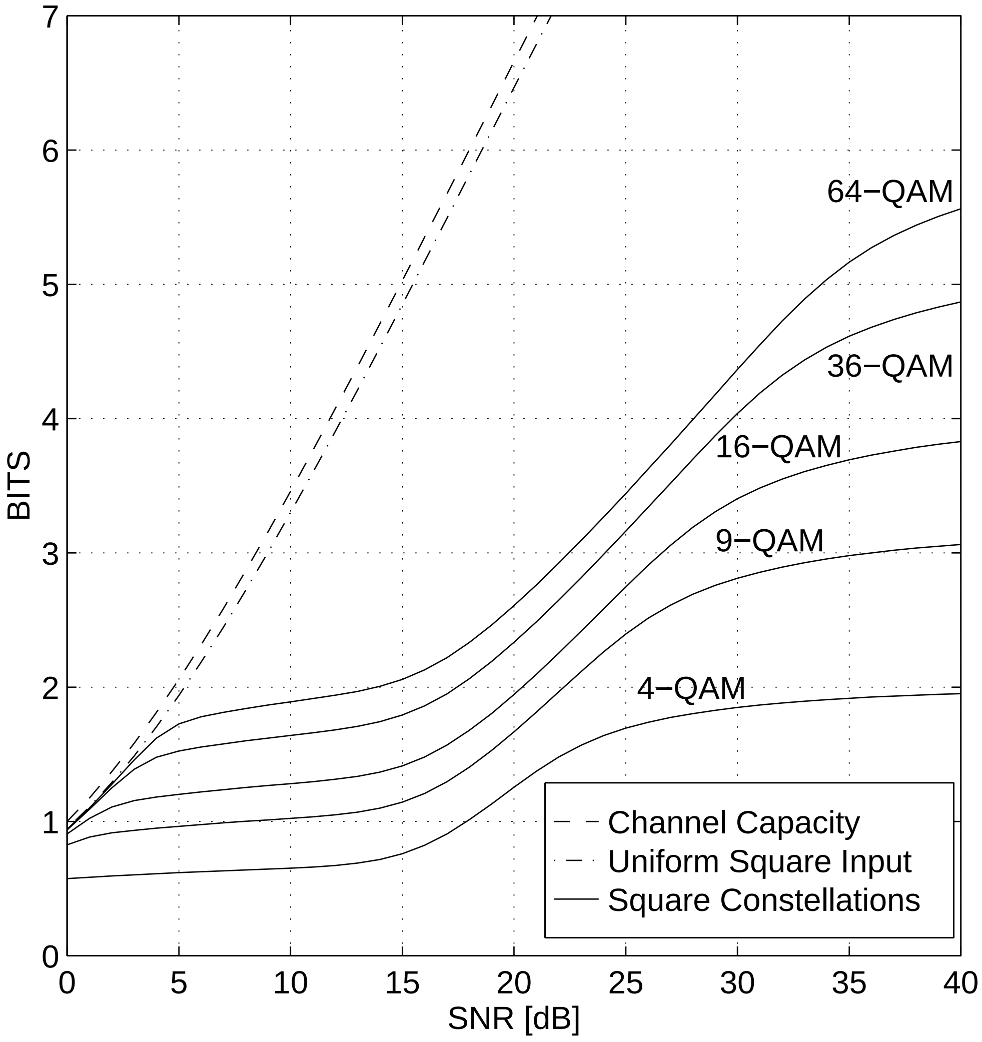


Figure 5.17: Mutual information (using MCM) on the 1/3 Band LPF channel as a function of equivalent AWGN SNR for five QAM constellations, uniform distribution on a square, and Gaussian distribution.

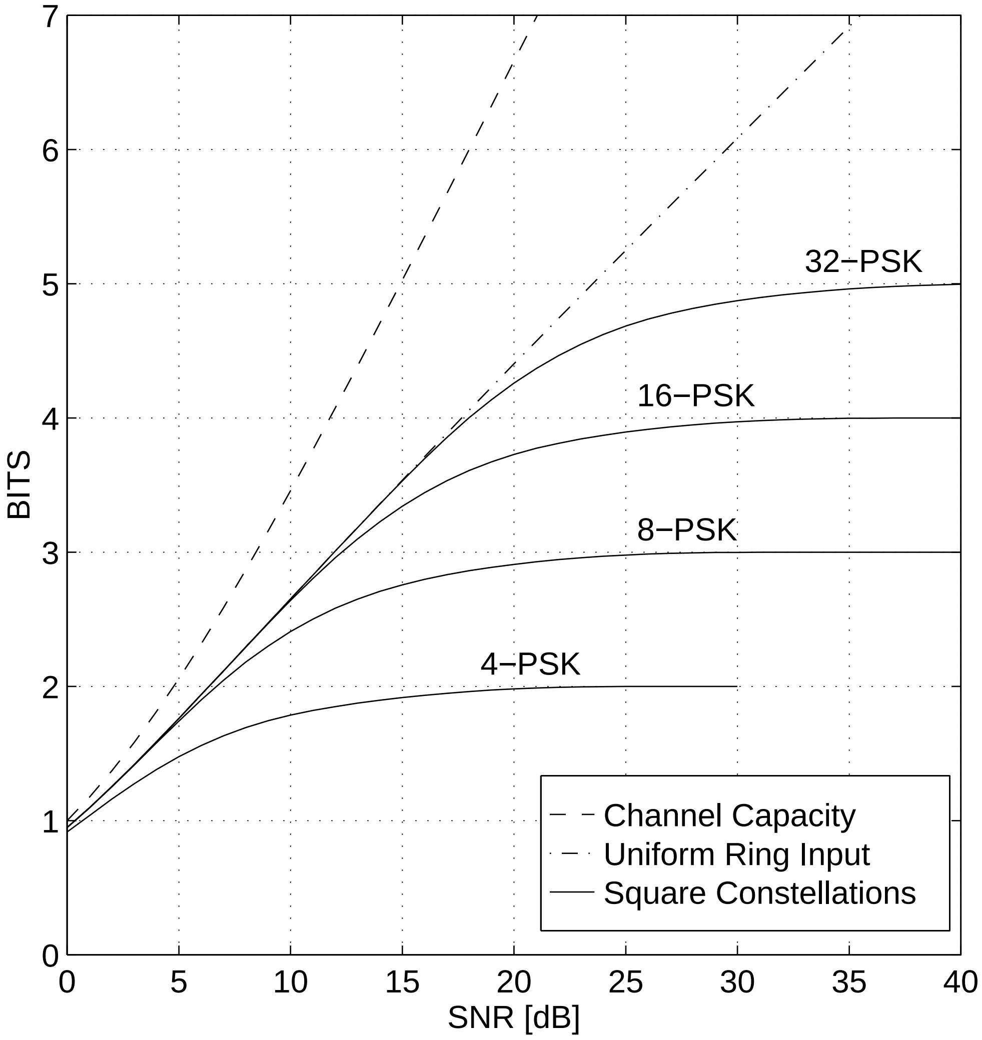


Figure 5.18: Mutual information (using MCM) on the Notch channel as a function of equivalent AWGN SNR for four PSK constellations, uniform distribution on a ring, and Gaussian distribution.

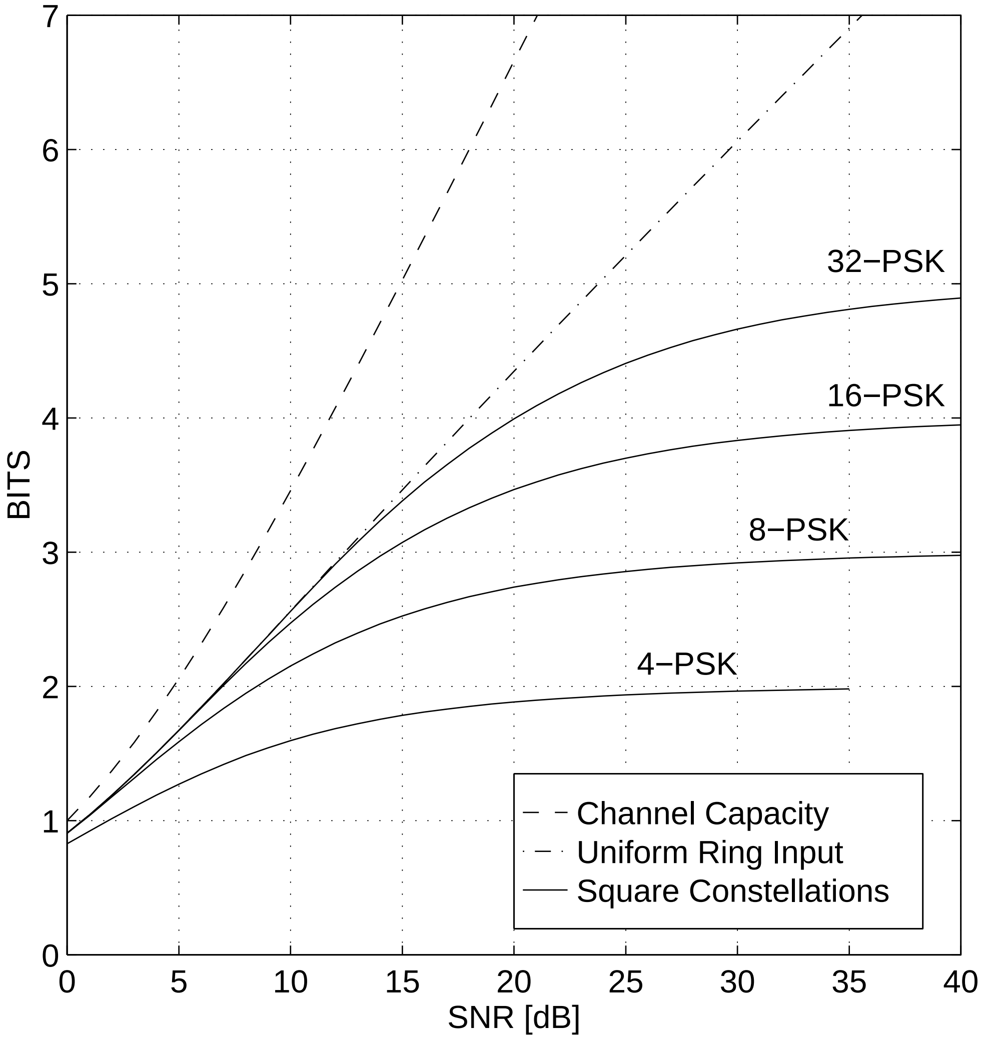


Figure 5.19: Mutual information (using MCM) on the Ramp channel as a function of equivalent AWGN SNR for four PSK constellations, uniform distribution on a ring, and Gaussian distribution.

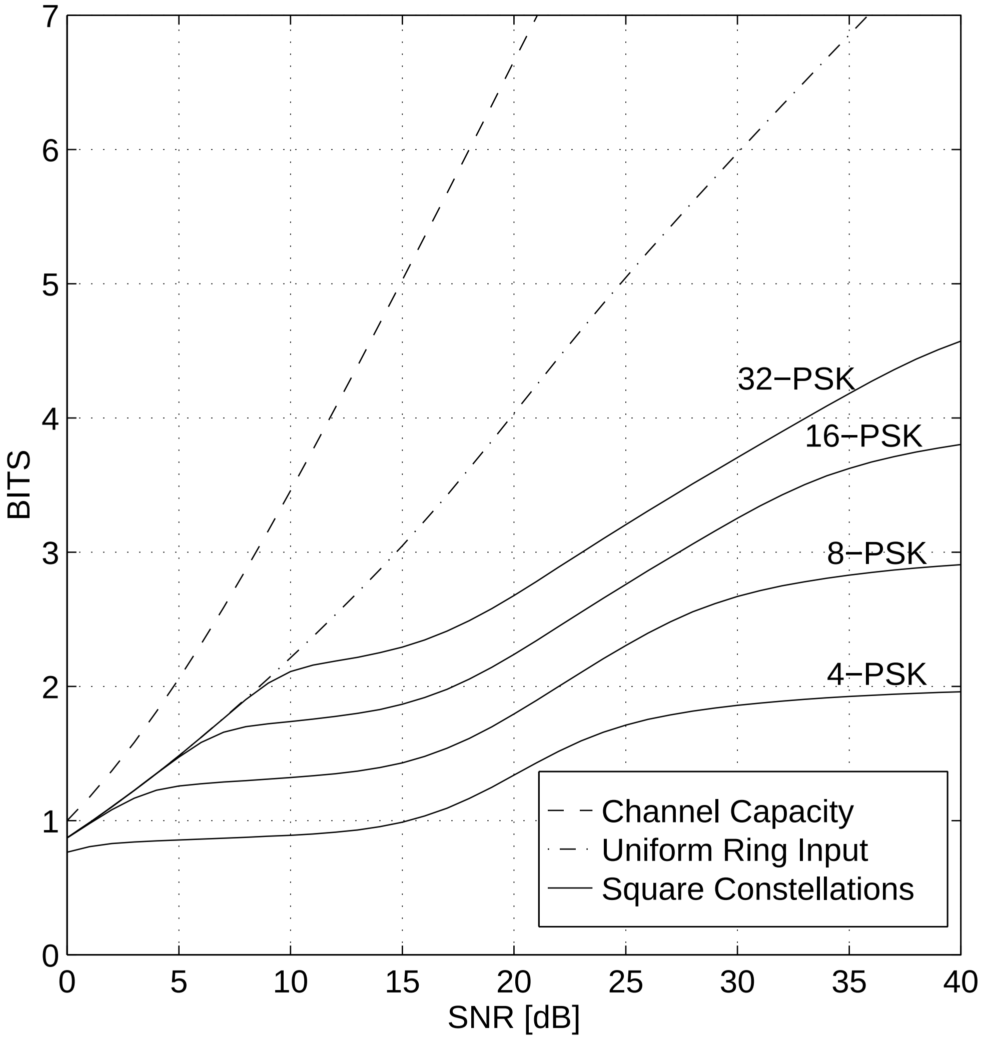


Figure 5.20: Mutual information (using MCM) on the 1/2 Band LPF channel as a function of equivalent AWGN SNR for four PSK constellations, uniform distribution on a ring, and Gaussian distribution.

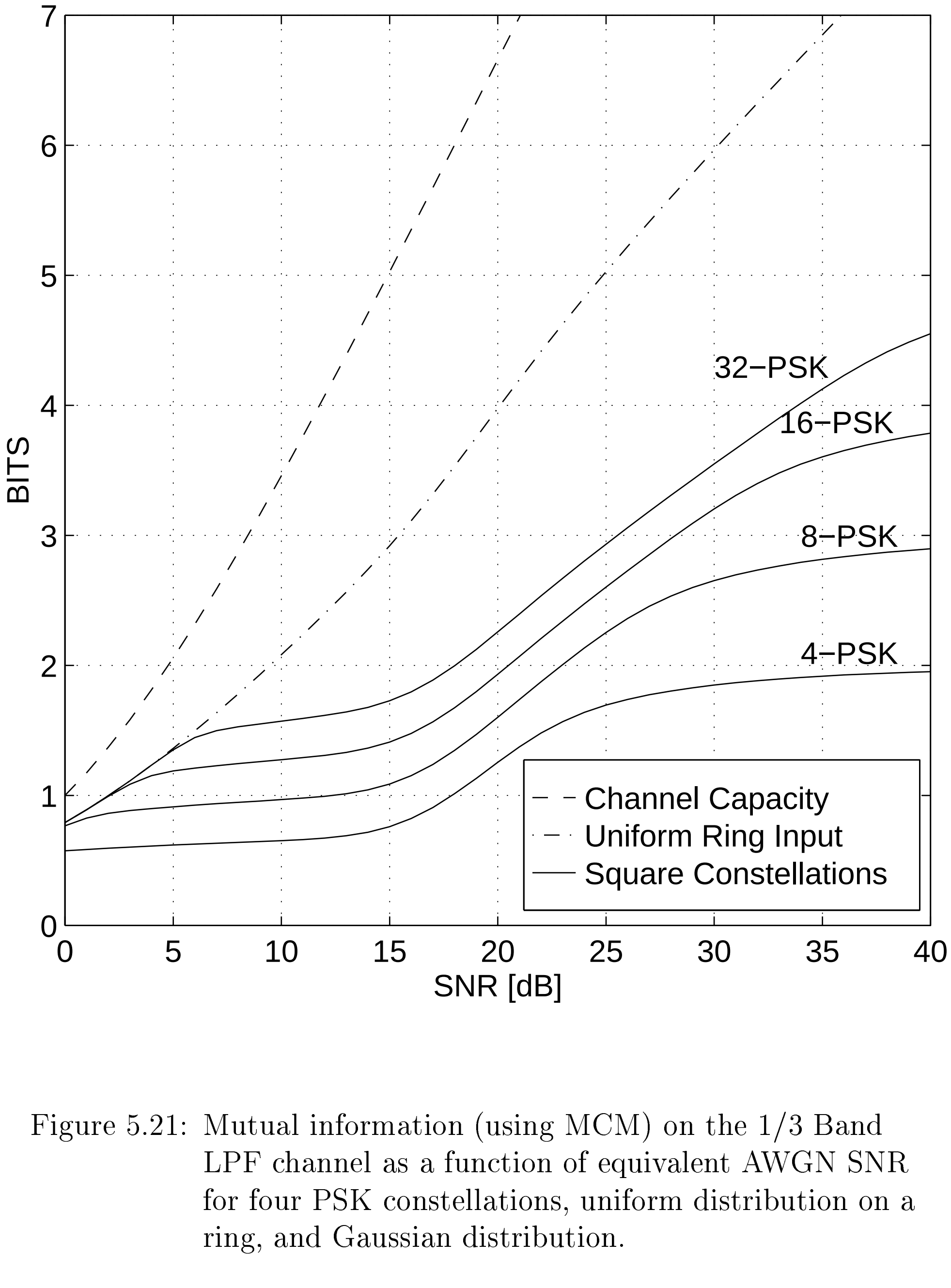


Figure 5.21: Mutual information (using MCM) on the 1/3 Band LPF channel as a function of equivalent AWGN SNR for four PSK constellations, uniform distribution on a ring, and Gaussian distribution.

5.4 Summary

In this chapter the mutual information performance of $M \times M$ QAM and M-PSK constellations was explored for AWGN channels, Rayleigh fading channels, and multicarrier modulation in ISI channels. A QAM constellation of size 2^{R+1} is within about 1.5 dB of the optimal Gaussian input sequence for the AWGN channel. A PSK constellation of size 2^R supports about half the rate of a QAM constellation of the same size for large R.

An $M \times M$ QAM constellation will never outperform the limiting case of the uniform input on a square. The PSK constellations will never outperform the limiting case of a uniform distribution on a ring. For the AWGN channel, the QAM constellations are very close to their limiting case curve (I_{square}) for rates more than one bit below $\log_2$ of the constellation size. The PSK constellations are very close to their limiting case curve (I_{ring}) for rates more than half a bit below $\log_2$ of the constellation size.

For independent Rayleigh fading, lower rates are required for performance close to the limiting case curves. The QAM constellations are very close to I_{square} for rates more than 2 bits below $\log_2$ of the constellation size. The PSK constellations are very close to I_{ring} for rates more than 1 bit below $\log_2$ of the constellation size.

The third type of communication channel explored was the ISI channel decomposed into parallel subchannels (without ISI) by MCM. For these channels, the constellation size required for performance close to I_{square} or I_{ring} depends largely on the amount of SNR variation. For a particular channel, the constellation size must be chosen large enough that the subchannels with the highest SNRs can be used efficiently. Thus it is the *high* SNR subchannels that determine what constellation size is required.

Chapter 6

A Code Search Example

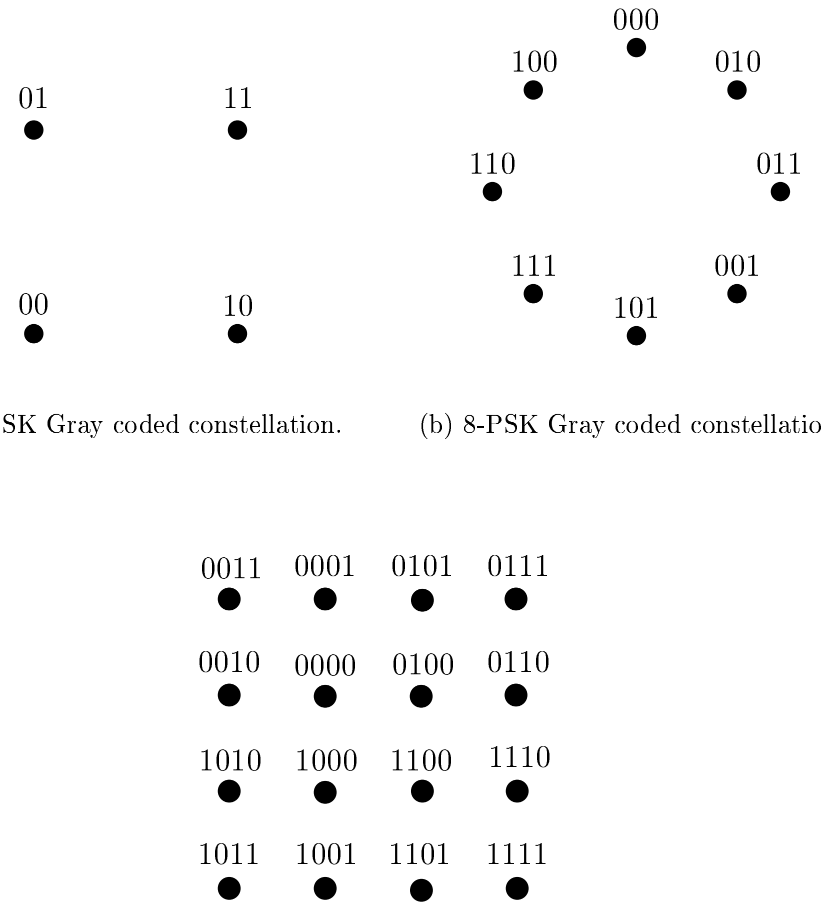
This chapter brings together the ideas of the previous four chapters to design a trellis code for periodically interleaved broadcast multicarrier modulation (MCM). The periodic metrics — periodic effective code length and code periodic product distance — derived at the end of Chapter 2 are well suited for this scenario, since periodic interleaving is used to disperse the strongly correlated adjacent subchannel gains. For comparison, codes are also designed using the other metrics presented in Chapter 2 (code Euclidean distance, effective code length, and code product distance).

The codes designed in this chapter use 6 memory elements ($\nu = 6$) and transmit 1 information bit per symbol ($k = 1$). These parameters allow comparison with the standard rate-1/2, $\nu = 6$ code [43]. This standard code was used with a 4-PSK constellation for broadcast MCM with periodic interleaving by Sari [39] (see also [44]) and others.

Section 6.1 identifies the standard code mentioned above as the $k = 1$, $\nu = 6$ code that maximizes CED. Section 6.2 finds codes that maximize ECL and CPD, and Section 6.3 identifies codes that maximize the periodic metrics. The codes found in each of these three sections are simulated using MCM with periodic interleaving on five different frequency selective channels in Section 6.4.

Three different constellations are considered in the trellis code searches that follow: 4-PSK, 8-PSK, and 16-QAM. For each constellation, a Gray labeling that achieves the maximum possible edge length profile (as demonstrated in Section 3 of Chapter 4) is employed. Figure 6.1 shows these three labeled constellations.

In each case a strictly equivalent Ungerboeck labeling exists by Theorems 8 and 9



(a) 4-PSK Gray coded constellation.

(b) 8-PSK Gray coded constellation.

(c) 16-QAM Gray coded constellation.

Figure 6.1: The three superior-in-profile labeled constellations used for the code searches and simulations of this chapter.

of Chapter 4. A strictly equivalent code results regardless of the choice between the Gray coding shown in Figure 6.1 and the associated Ungerboeck labeling.

Gray coding allows the code searches to avoid distance equivalent codes by avoiding certain permutations of encoder polynomials according to the procedure outlined on page 45 of Chapter 3. In fact, the example application on pages 45–45 of that procedure is for the Gray coded 16-QAM constellation in Figure 6.1(c).

The metrics derived in Chapter 2 are based on distance sequences $\{d_i^2(b \rightarrow \hat{b})\}$. The searches in this chapter reduce complexity by considering only a worst case distance sequence $\{d_i^2(\mathbf{E}(D))\}$ for each $E(D)$ (see Definition 14 in Chapter 3). For the constellations considered in this chapter, the resulting metrics are the same whether computed using $\{d_i^2(b \rightarrow \hat{b})\}$ or $\{d_i^2(\mathbf{E}(D))\}$ because for every $\mathbf{E}(D)$ there is a pair $(b(d), \hat{b}(D))$ that satisfies the conditions (3.25) and (3.26) on page 37.

For the 4-PSK and 16-QAM labeled constellations in Figure 6.1, one such set of pairs is obtained by fixing $b(D) = 0$ and considering all sequences $\hat{b}(D)$. For these constellations every edge emanating from the zero labeled constellation point has the minimum distance associated with its edge label. (See the achievable edge length profile bounds shown in Figures 4.1(a) and 4.5.)

For the 8-PSK constellation in Figure 6.1, only two edge labels, 001 and 100, have more than one possible distance associated with them. Let X represent an unspecified bit. Constellation points labeled X1X have the minimum distance for edge label 001 but not 100. Similarly, points labeled X0X have the minimum distance for edge label 100 but not 001. For any $\mathbf{E}(D)$, an input sequence $b(D)$ can be found that sends $\mathbf{C}_i$=X1X whenever $\mathbf{E}_i$=001 and $\mathbf{C}_i$=X0X whenever $\mathbf{E}_i$=100. Thus for every $\mathbf{E}(D)$ there is a pair $(b(d), \hat{b}(D))$ that satisfies (3.25) and (3.26) on page 37.

A *catastrophic* convolutional code has an error sequence with a finite EL that produces an infinite number of information bit errors. Catastrophic codes give poor performance even when they have large values for the metrics presented in Chapter 2. The code searches in this chapter remove catastrophic codes from consideration using the invariant factors test resulting from Theorem 1 of [21] (see also [45]).

Reversing the time order of the generator polynomials reverses the time order of the distance sequences, but does not affect the metrics presented in Chapter 2. An encoder is removed from consideration if it is the time reversal of an encoder already examined in the code search.

6.1 Euclidean Distance

In this section, a code is sought that maximizes code Euclidean distance (CED) with secondary consideration given to the number of Euclidean nearest neighbors (N_{CED}). These metrics were defined on page 1. For each 2^n point constellation ($n = 2, 3, 4$), an exhaustive search of feedback-free rate $1/n$ convolutional codes was performed (excluding catastrophic codes, distance equivalent permutations, and time reversals).

For each code considered by the search, CED and N_{CED} were computed with an adaptation of the bidirectional search algorithm proposed initially by Bahl et al. [46] for computing the free Hamming distance of a convolutional code. In [47], Larsen corrected some oversights in [46], and the corrected algorithm has come to be known as the "Larsen algorithm." The basic approach of this algorithm applies whenever the metric $J(\mathbf{E}(D))$ can be expressed as the sum of positive values that are functions of the error sequence elements $\mathbf{E}_i$:

$$J = \sum_{i=1}^{\infty} f(\mathbf{E}_i). \tag{6.1}$$

As originally described in [46, 47], $f(\mathbf{E}_i)$ was the Hamming weight of $\mathbf{E}_i$, but the algorithm applies equally well (computing CED) when $f(\mathbf{E}_i) = d^2_{\text{MIN}}(\mathbf{E}_i)$. In [5], Ungerboeck follows this approach except that $f(\mathbf{E}_i)$ was defined so that

$$f(\mathbf{E}_i) < d^2_{\text{MIN}}(\mathbf{E}_i) \tag{6.2}$$

for certain values of $\mathbf{E}_i$, yielding pessimistic metrics for some codes.

Tables 6.1–6.3 present the codes with largest CED for 4-PSK (rate 1/2), 8-PSK (rate 1/3), and 16-QAM (rate 1/4), respectively. All three tables actually show the *same* two trellis codes as having the maximum CED. It turns out that for $k = 1$ and $\nu = 6$, forcing the constellation to be 8-PSK or 16-QAM reduces CED. In these two cases, the maximum CED trellis codes use 4-PSK subsets of the larger constellations.

Recall that the previous chapter suggested using a constellation having *at least* 2^{R+1} points for the AWGN channel, but showed that the larger constellations should perform at least as well. This observation holds for large complexity and large delay.

However, for fixed ν and k, the code searches of this section demonstrate that

#	polynomial		CED^2	N_{CED}
1	133	171	20.0	11
2	135	163	20.0	11

Table 6.1: Rate-1/2 64-state 4-PSK codes with maximum Euclidean distance. All codes shown use the full 4-PSK constellation.

#	polynomial			Actual Constellation	CED^2	N_{CED}
1	133	42	171	4-PSK	20.0	11
2	135	56	163	4-PSK	20.0	11
3	107	32	127	8-PSK	18.0	1

Table 6.2: Rate-1/3 64-state 8-PSK codes with maximum Euclidean distance.

#	polynomial				Actual Constellation	CED^2	N_{CED}
1	171	133	133	171	4-PSK	20.0	11
2	135	163	163	135	4-PSK	20.0	11
					⋮		
3	117	135	135	157	2×4 QAM	16.44	1
					⋮		
4	117	155	145	137	16-QAM	14.4	2

Table 6.3: Rate-1/4 64-state 16-QAM codes with maximum Euclidean distance.

#	ECL	PECL	$\log_2(\cdot)$					
			CPD	CPPD_4	CPPD_5	CPPD_6	CPPD_7	CPPD_8
1	6	5	16.0	∞	9.0	9.17	9.0	11.17
2	6	4	16.0	9.17	10.59	9.0	10.0	11.17

Table 6.4: Other properties of the codes in Table 6.1.

trellis codes using constellations with 2^{R+2} and 2^{R+3} points are sometimes unable to match the CED achieved by trellis codes using a 2^{R+1} point constellation. Thus for practical complexity trellis codes, a constellation can be "too big" for the AWGN channel. Note, however, that this search was restricted to trellis codes; a different coding structure might use the constellation more effectively.

As discussed above, the 4-PSK codes listed in Table 6.1 are the best found in terms of CED. The only choice is between codes #1 and #2 in that table; no other codes with CED $= 20$ were found. Since both codes have the same N_{CED}, the other metrics shown in Table 6.4 are used to differentiate between the two codes. Code #1 is selected for simulation because of its larger value of PECL (and because of its status as an often simulated code).

This code (Code #1 in Table 6.1) is commonly used for 4-PSK constellations [43]. In particular, it has been studied for transmission of one bit per symbol in the context of broadcast MCM [39, 44], which is the environment simulated in this chapter.

6.2 Effective Length and Product Distance

In this section, a code is sought that maximizes effective code length (ECL) with secondary consideration given to the code product distance (CPD). These metrics were defined on page 13. As in the previous section, exhaustive searches of feedback-free convolutional codes were performed for the rate associated with each constellation. The searches excluded catastrophic codes, distance equivalent permutations, and time reversals.

Once again, the bidirectional search algorithm can be applied to compute the metrics of interest. The computation of ECL is accomplished by the standard algorithm with $f(\mathbf{E}_i)$ defined as the indicator function of $\mathbf{E}_i \neq 0$. The CPD is computed by continuing to explore error sequences with EL $=$ ECL until the CPD is identified. The computation of ECL and CPD in this way is described fully in [48].

From Theorem 1 on page 35, the maximum possible ECL is

$$\text{ECL} \leq \left\lfloor \frac{\nu}{k} \right\rfloor + 1 \tag{6.3}$$

$$= \frac{6}{1} + 1 \tag{6.4}$$

$$= 7. \tag{6.5}$$

The search identified trellis codes for all three constellations that achieved ECL = 7. Tables 6.5, 6.6, and 6.7 present the codes with ECL = 7 having the largest CPD for 4-PSK (rate 1/2), 8-PSK (rate 1/3), and 16-QAM (rate 1/4) respectively. The largest value for the CPD was produced by a trellis code using the 8-PSK constellation. The tables list $\log_2$(CPD) rather than CPD because the pairwise error expression in (2.27) on page 2.27 is exponential in EL and $\log_2$(PD) (rather than PD).

This situation is another example where a metric (CPD) became worse for a larger constellation, contrary to the intuition of the previous chapter. For the largest constellation (16-QAM), the best CPD for an ECL = 7 code was produced by a convolutional encoder that used only 8 of the 16 available points. This CPD was still lower than that found with 8-PSK.

Note that the largest CPD was found with the 2^{R+2} point constellation. All of the code searches by Du and Vucetic [11, 12, 13, 14] seeking to maximize CPD considered only 2^{R+1} point constellations. The code searches in this section suggest that larger values of CPD might be obtained with larger (2^{R+2} point) constellations.

Five rate-1/3 8-PSK codes were found that had ECL = 7 and had the maximum CPD of $2^{11.54}$. From these five codes, code #1 in Table 6.6 was chosen because it had the the largest CED. Table 6.8 shows the periodic metrics for the five codes in Table 6.6

6.3 Periodic Metrics

In this section, a code is sought that maximizes the periodic effective code length (PECL) with secondary consideration given to the code periodic product distances CPPD$_i$. The periodic metrics assume a period P for the periodic interleaver. The period $P = 8$ used to compute the metrics in this section is consistent with the $P = 8$

#	polynomial	ECL	N_{ECL}	$\log_2(\text{CPD})$	CED^2	N_{CED}
1	45 173	7	5	9.0	18.0	2
2	123 155	7	5	9.0	18.0	3
3	133 175	7	6	9.0	18.0	1
4	135 157	7	6	9.0	18.0	2
5	55 163	7	6	9.0	18.0	3
6	75 123	7	7	9.0	18.0	5
7	75 113	7	8	9.0	18.0	3

Table 6.5: Rate-1/2 64-state 4-PSK codes with largest CED among those with maximum ECL and CPD. All codes shown use the full 4-PSK constellation.

#	polynomial	ECL	N_{ECL}	$\log_2(\text{CPD})$	CED^2	N_{CED}
1	135 66 177	7	1	11.54	14.34	1
2	133 65 177	7	1	11.54	13.51	1
3	153 66 177	7	1	11.54	12.69	1
4	166 55 177	7	2	11.54	12.34	1
5	156 65 177	7	1	11.54	12.34	1

Table 6.6: Rate-1/3 64-state 8-PSK codes with largest CED among those with maximum ECL and CPD. All codes shown use the full 8-PSK constellation.

#	polynomial	Actual Constellation	ECL	N_{ECL}	$\log_2(\text{CPD})$	CED^2	N_{CED}
1	135 177 177 153	$2 \times 4QAM$	7	1	10.09	12.44	1
2	135 177 173 157	16-QAM	7	1	9.03	10.4	1
3	135 177 167 157	16-QAM	7	1	9.03	10.4	1
4	135 167 173 177	16-QAM	7	1	9.03	10.4	1
5	135 157 167 177	16-QAM	7	1	9.03	10.4	1
6	133 177 175 167	16-QAM	7	1	9.03	10.4	1
7	133 167 177 175	16-QAM	7	1	9.03	10.4	1

Table 6.7: Rate-1/4 64-state 16-QAM codes with largest CED among those with maximum ECL and CPD.

#	PECL	$\log_2(\cdot)$			
		CPPD$_5$	CPPD$_6$	CPPD$_7$	CPPD$_8$
1	6	∞	7.17	5.46	2.91
2	6	∞	6.18	4.58	0.00
3	6	∞	6.24	5.08	-0.23
4	5	10.21	8.41	2.55	1.50
5	6	∞	6.17	4.45	1.10

Table 6.8: Periodic metrics for Rate-1/3 64-state 8-PSK codes listed in Table 6.6

periodic interleaving used in the simulations in the next section.

The two bounds on PECL derived on page 25 are repeated here for convenience.

$$\text{PECL} \leq \left\lfloor P\left(1 - \frac{k}{n}\right)\right\rfloor + 1\,. \tag{6.6}$$

$$\text{PECL} \leq \left\lfloor \frac{\nu}{k} \right\rfloor + 1 \tag{6.7}$$

Recalling that our search is for $\nu = 6$ and $k = 1$, (6.7) implies

$$\text{PECL} \leq 7 \tag{6.8}$$

for all the codes considered. Since n varies with constellation size, the bound resulting from (6.6) varies with constellation as follows (assuming $k = 1$):

$$\text{PECL} \leq 5 \qquad \text{for 4-PSK} \tag{6.9}$$

$$\text{PECL} \leq 6 \qquad \text{for 8-PSK} \tag{6.10}$$

$$\text{PECL} \leq 7 \qquad \text{for 16-QAM.} \tag{6.11}$$

The primary goal of this section's search is to maximize PECL. Thus the rate-1/4 16-QAM trellis codes are considered first to see if the bound in (6.11) can be achieved. This search, as with the previous searches, excludes catastrophic codes, distance equivalent permutations, and time reversals. The algorithm proposed by Lapidoth in Theorem 1 of [18] was used to check whether each candidate code had PECL ≥ 7. This was done by checking that no erasure pattern U with $|U| < 7$ results in an error on the periodic block erasure channel with $M = P = 8$.

Many rate-1/4 16-QAM codes were found that had $\text{PECL} = 7$. The 4-PSK and 8-PSK constellations were never considered since they cannot produce a code with $\text{PECL} = 7$.

The code periodic product distances CPPD_7 and CPPD_8 are used to select a code from among those with $\text{PECL} = 7$. The bidirectional search algorithm used to compute CED, ECL, and CPD does not immediately apply to the computation of CPPD_i. Instead, a computationally intensive direct trellis search was used. Further investigation could lead to a more efficient method for computing CPPD. However, the values of CPPD_7 and CPPD_8 were required only for codes that had $\text{PECL} = 7$, and so a more efficient algorithm was not required for this search.

It is not obvious how to weigh the relative importance of CPPD_7 and CPPD_8. However, some codes can be immediately removed from consideration using the following definition, which is specific to the current situation.

Definition 28 *A rate-1/4, $\nu = 6$, 16-QAM trellis code with* $\text{PECL} = 7$ *is dominated if there exists another rate-1/4, $\nu = 6$, 16-QAM trellis code with* $\text{PECL} = 7$ *that has larger values of both* CPPD_7 *and* CPPD_8.

A code that is not dominated is called *Pareto optimal* [16]. When all of the dominated trellis codes are removed from consideration, only eleven Pareto optimal codes remain. These eleven undominated codes are listed in Table 6.9 along with their periodic metrics.

Normalizing CPPD_i by taking the ith root (or dividing $\log_2 \text{CPPD}_i$ by i) is suggested by (2.41) on page 17. Because of this, the last column in Table 6.9 lists the smaller value (choosing between $i = 7$ and $i = 8$) of $i^{-1} \log_2 \text{CPPD}_i$. Table 6.10 lists the other metrics for these eleven codes.

With only eleven Pareto optimal codes from which to choose, it is realistic at this point to simulate each code and choose the one with the best error rate of interest. However, only one code is selected for simulation in Section 6.4.

Figure 6.2 plots $\log_2 \text{CPPD}_8$ vs. $\log_2 \text{CPPD}_7$ for the eleven Pareto optimal codes. There are several strategies for selecting one of the eleven codes, but two of the codes stand out as desirable.

Code #9 has the largest minimum value of $\log_2 \text{CPPD}_i$ (minimizing over $i = 7, 8$).

#	polynomial				PECL	$\log_2(\cdot)$		Min $i^{-1}\log_2(\text{CPPD}_i)$ for $i \in \{7, 8\}$
						CPPD$_7$	CPPD$_8$	
1	33	177	127	35	7	5.20	−0.87	−.108
2	31	173	163	57	7	4.99	−0.58	−.073
3	105	177	157	131	7	4.84	0.75	.094
4	107	175	143	127	7	4.39	1.75	.219
5	43	176	171	45	7	4.07	2.46	.308
6	43	175	155	103	7	3.67	2.62	.328
7	63	175	135	103	7	3.39	2.82	.352
8	55	173	107	61	7	2.98	2.88	.360
9	71	103	137	145	7	2.94	3.50	.420
10	73	105	135	143	7	2.77	3.76	.396
11	43	121	135	143	7	2.07	3.77	.296

Table 6.9: Rate-1/4 64-state 16-QAM codes with undominated pairs (CPPD$_7$, CPPD$_8$) given the maximum PECL of 7.

#	CED^2	N_{CED}	ECL	N_{ECL}	$\log_2(\text{CPD})$
1	10.4	1	7	1	5.71
2	10.4	1	7	1	5.03
3	12.8	1	7	1	6.39
4	12.8	1	7	1	4.39
5	11.2	1	7	2	4.07
6	12.4	1	7	2	4.07
7	12.8	2	7	2	3.39
8	10.8	1	7	1	3.07
9	10.8	1	7	2	3.07
10	11.6	1	7	2	4.07
11	10.8	1	7	2	2.07

Table 6.10: Other properties of rate-1/4 64-state 16-QAM codes listed in Table 6.9.

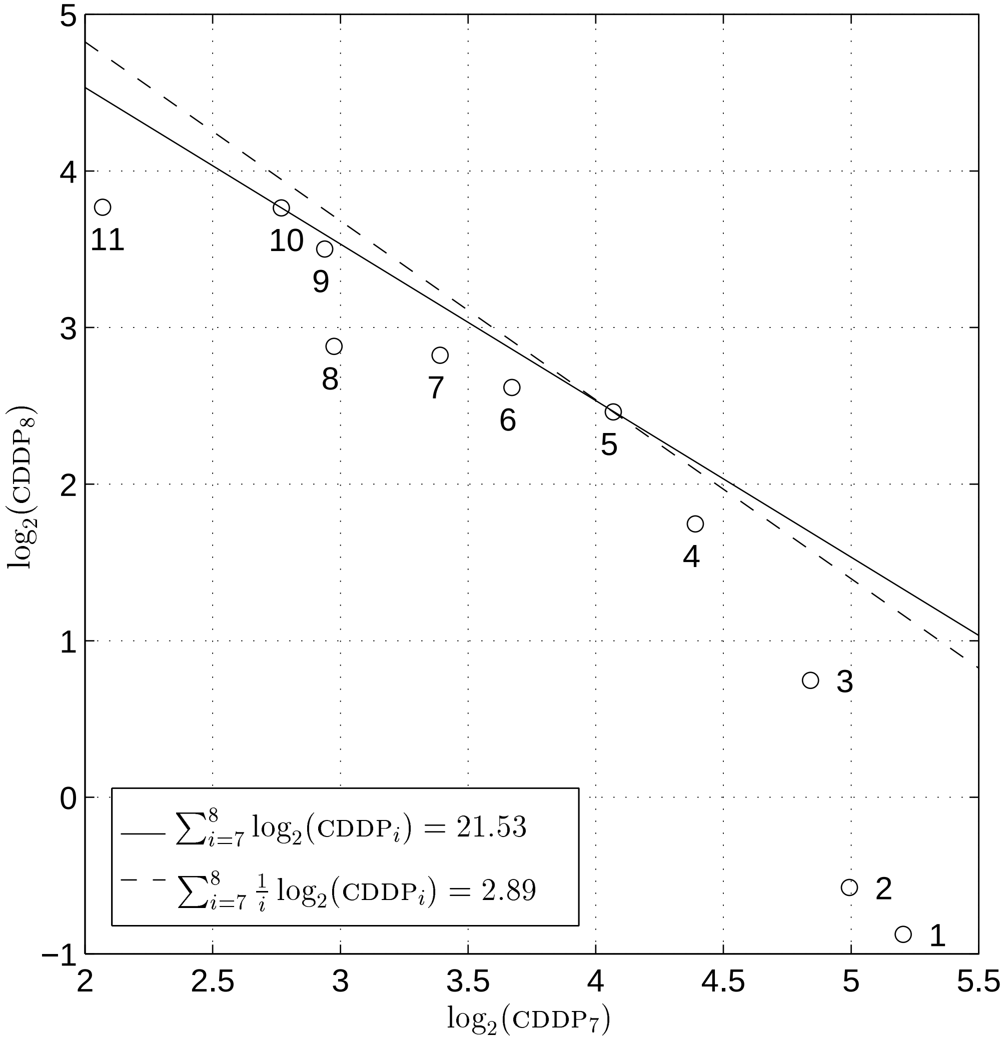


Figure 6.2: $\log_2(\text{CDDP}_8)$ vs. $\log_2(\text{CDDP}_7)$ for codes in Table 6.9.

This code also has the largest minimum value of $i^{-1}\log_2 \textsc{cppd}_i$.

However, Code #5 maximizes both $\sum \log_2 \textsc{cppd}_i$ and $\sum i^{-1}\log_2 \textsc{cppd}_i$. Code #5 was chosen for simulation, partly because it has a larger value of $\log_2 \textsc{cppd}_7$ than Code #9. Because events with periodic event lengths of 7 have inherently less diversity, it was decided to place slightly more emphasis on $\textsc{cppd}_7$.

Code #1 maximizes $\log_2 \textsc{cppd}_7$ outright, but its low value of $\log_2 \textsc{cppd}_8$ makes it clearly undesirable.

As clear from this example, maximizing the periodic metrics is complicated by the tradeoff between values of $\textsc{cppd}_i$ for various indexes. However, the same tradeoff exists even for the regular product distance. In that case, the issue is completely ignored and only the minimum effective length products are considered. In fact, there is an infinite number of possible effective lengths. Thus the tradeoff problem for regular product distance is actually more complicated than in the periodic case where only a few periodic effective lengths (two for the example above) are possible.

6.4 Simulations

The major elements of the simulation setup have already been described in the previous chapters. Multicarrier modulation was used to decompose intersymbol interference channels into 512 subchannels as illustrated in Figures 5.9 and 5.10 on pages 97–98. In these simulations, the overhead introduced by the cyclic prefix was neglected.

Periodic interleaving and deinterleaving was performed with $P = 8$ and $B = 512$, as illustrated on page 20. Maximum likelihood sequence detection was done using the Viterbi algorithm, while assuming that the receiver had perfectly estimated the channel frequency response.

Five channels were simulated: the AWGN channel and the four channels whose impulse and frequency responses were plotted on page 102 and pages 100–101 respectively.

Each of Figures 6.3, 6.4, and 6.5 plots the BER performance of one code on all five channels. All the BER curves in this section are plotted vs. the equivalent AWGN SNR discussed on page 103. Each point plotted represents a simulation that ran until 50 trellis error events occurred. Each trellis event involved at least 2 bit errors, so each plotted point represents at least 100 bit errors.

Figures 6.3–6.5 show simulation results for the codes identified in Sections 6.1–6.3 respectively. Figure 6.5 shows that the code designed in Section 6.3 to maximize the periodic metrics performs consistently well on all five channels. Figures 6.3–6.4 show that the other two codes have good performance on some channels but not others.

Note in Figure 6.5 that the best BER curve is the curve for the 1/3 Band LPF channel (on which the other two codes failed completely). This BER curve is similar to that of the maximum CED code on the AWGN channel (see Figure 6.3). This emphasizes the fact that, from a mutual information point of view, the 1/3 Band LPF channel is no worse than the other channels studied.

Figures 6.6–6.10 each plot the BER performance of all three codes on one of the five channels. The channels are examined in the following order in these figures: AWGN, Notch, Ramp, 1/2 Band LPF, and 1/3 Band LPF.

Figure 6.6 shows that for the AWGN channel the maximum CED and ECL codes perform almost identically, while the periodic metric code is 1 dB worse in performance than the other two at BER= 10^{-6}.

Figure 6.7 shows all three codes having similar performance on the Notch channel. For the Ramp channel in Figure 6.8, the maximum ECL and PECL codes perform almost identically, but the maximum ED code is more than 2 dB worse in performance than the other two at BER= 10^{-6}.

With the 1/2 Band LPF channel, the periodic metric code has the best BER performance. The maximum ECL code is 1 dB worse at BER= 10^{-6}. The maximum CED code doesn't achieve a BER below 0.1 for the entire SNR range simulated. This shows the limitations of CED as a metric for severely frequency selective channels. It is also consistent with Figure 5.20 on page 111. Figure 5.20 predicts a loss of at least 15 dB when the 4-PSK constellation is used to send one bit per symbol on this channel.

Finally, the 1/3 Band LPF channel demonstrates the very robust performance of the code designed to maximize the periodic metrics. The periodic metric code has a BER of 10^{-6} with an equivalent AWGN SNR of less than 5 dB, while the other two codes have a BER above 0.2 even at 9 dB of AWGN SNR.

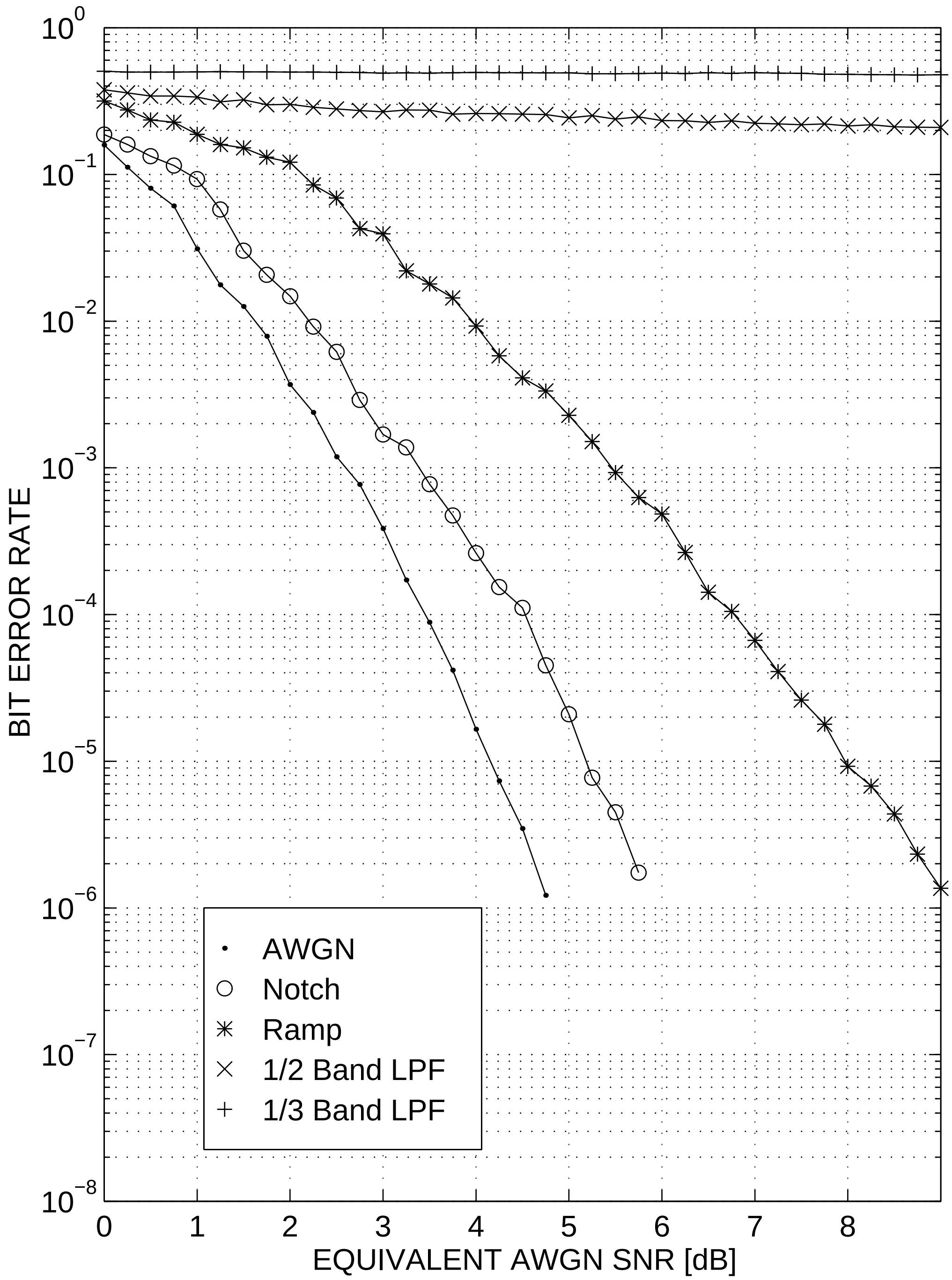


Figure 6.3: BER vs. equivalent AWGN SNR on five MCM channels for maximum CED code (Table 6.1 #1).

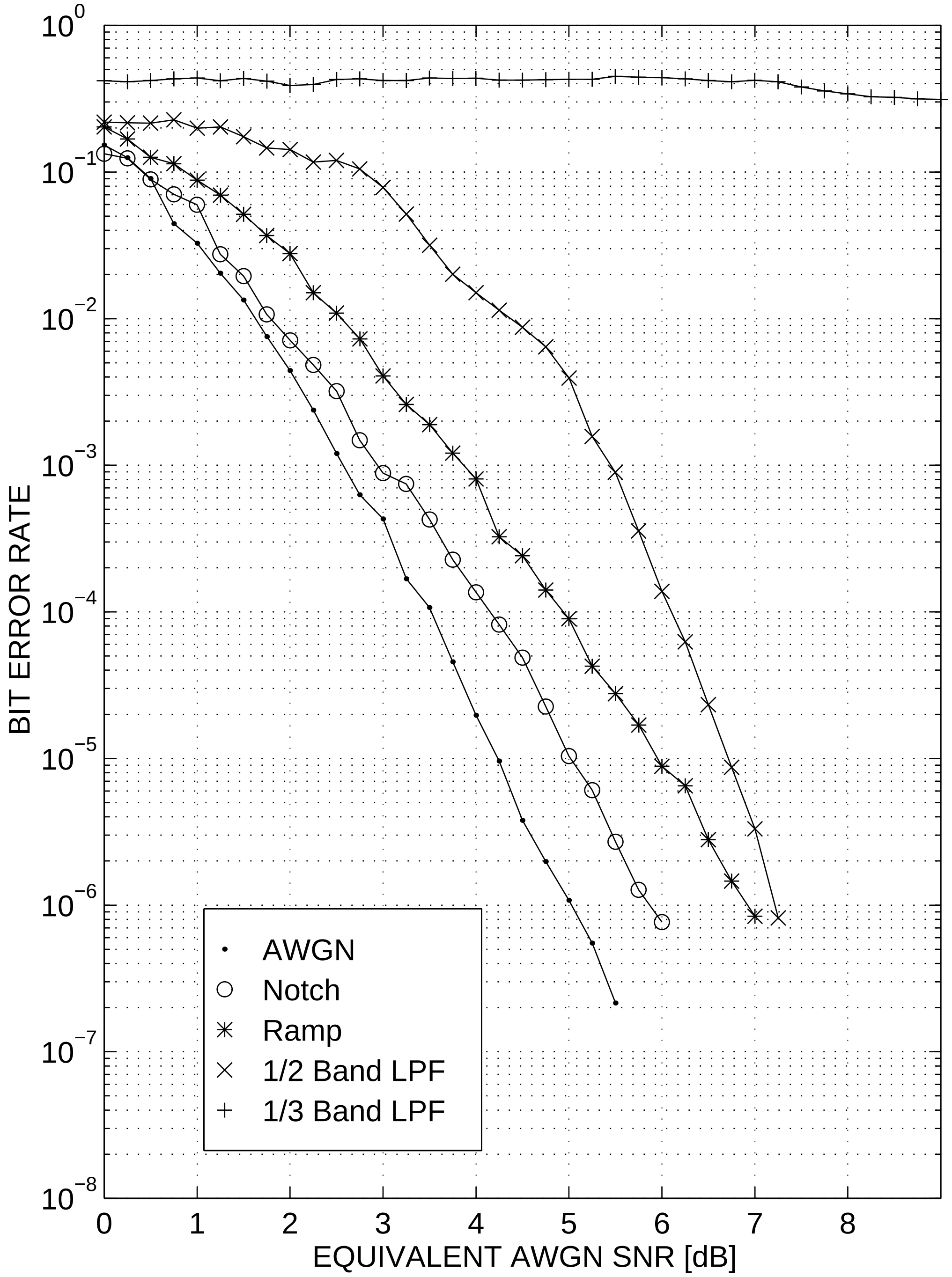


Figure 6.4: BER vs. equivalent AWGN SNR on five MCM channels for code optimizing ECL, CPD, and CED in that order (Table 6.6 #1).

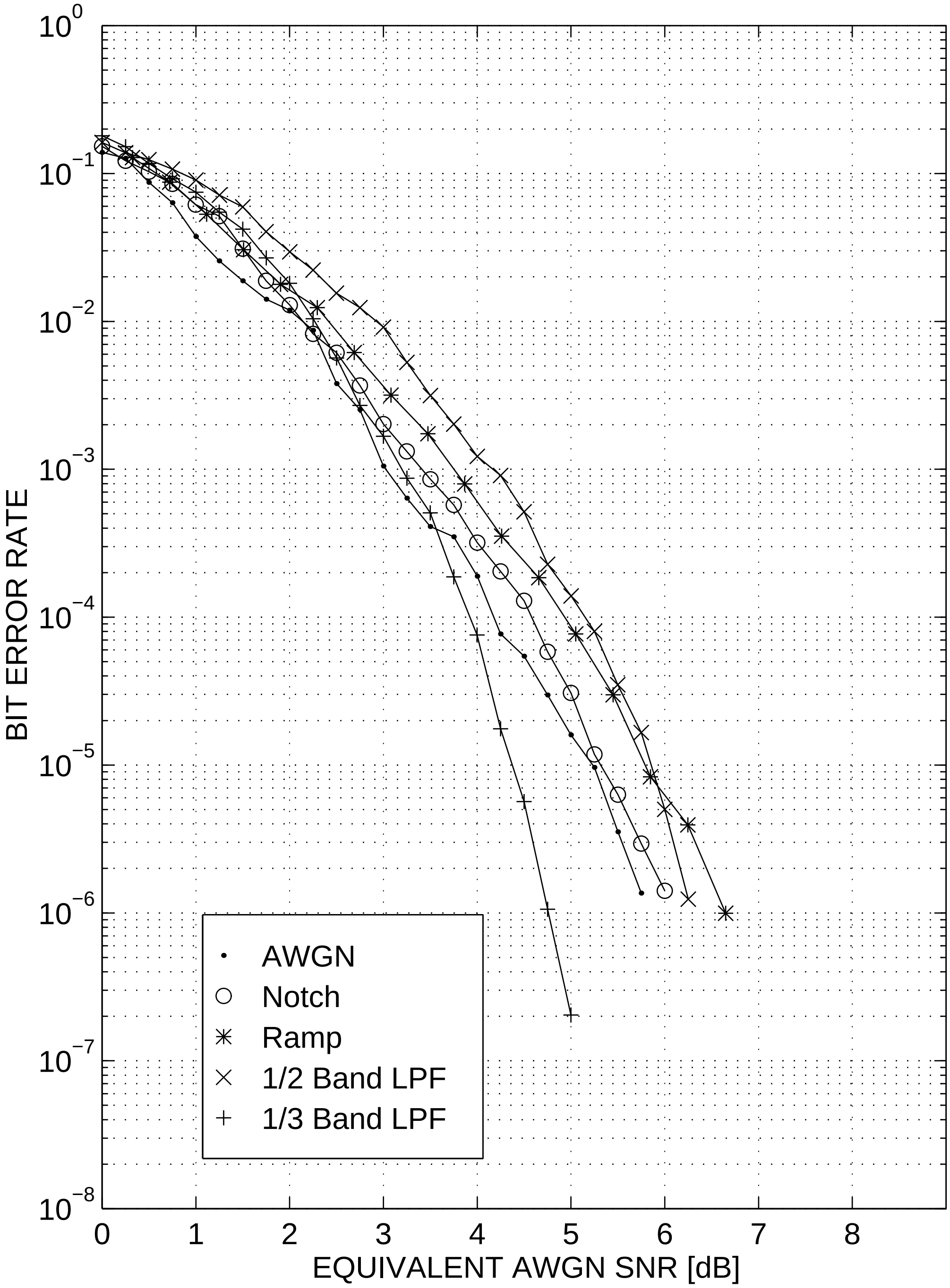


Figure 6.5: BER vs. equivalent AWGN SNR on five MCM channels for code optimizing PECL and CPPD (Table 6.9 #5).

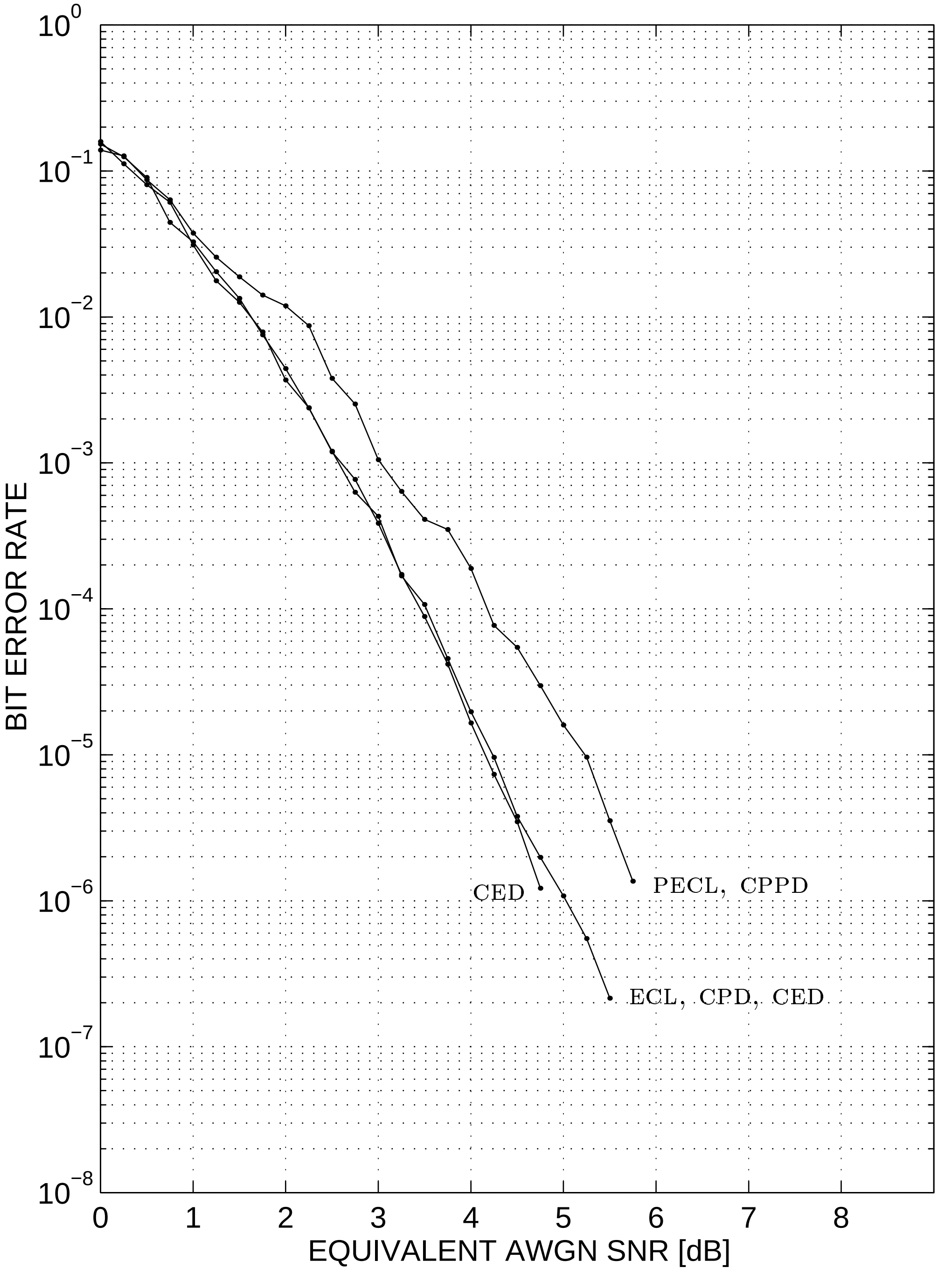


Figure 6.6: BER vs. equivalent AWGN SNR on the AWGN channel for all three codes.

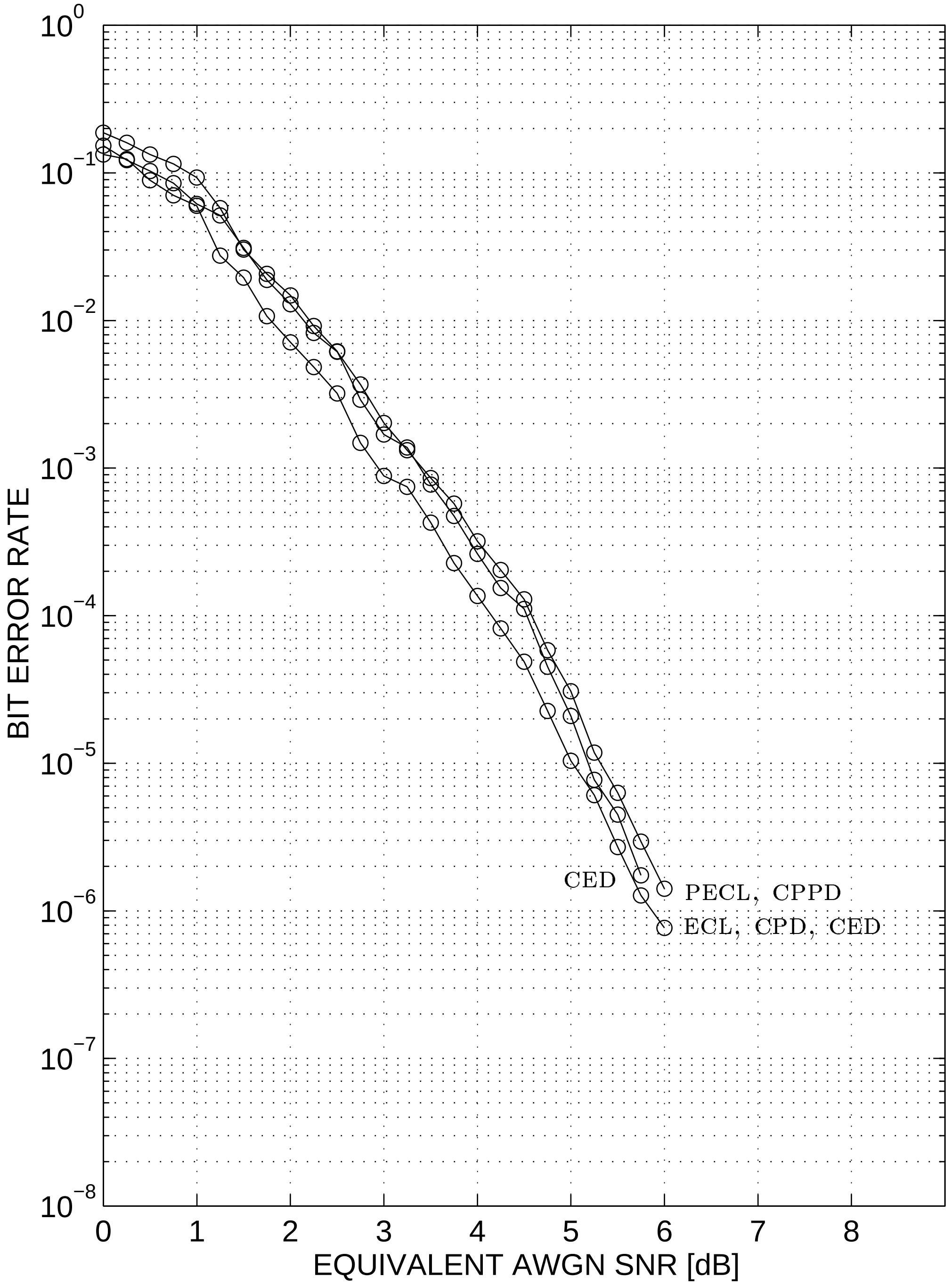


Figure 6.7: BER vs. equivalent AWGN SNR on the Notch channel for all three codes.

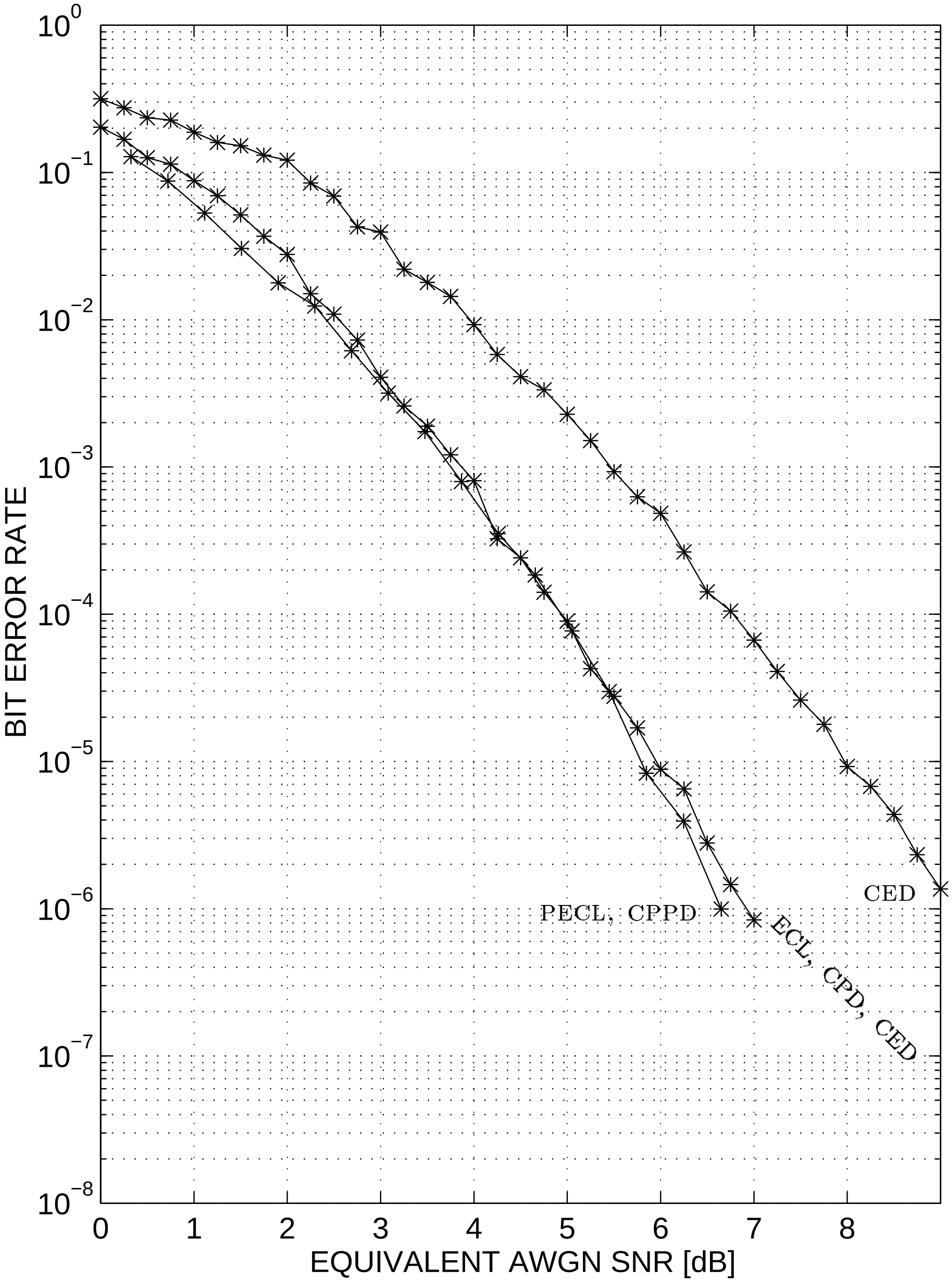


Figure 6.8: BER vs. equivalent AWGN SNR on the Ramp channel for all three codes.

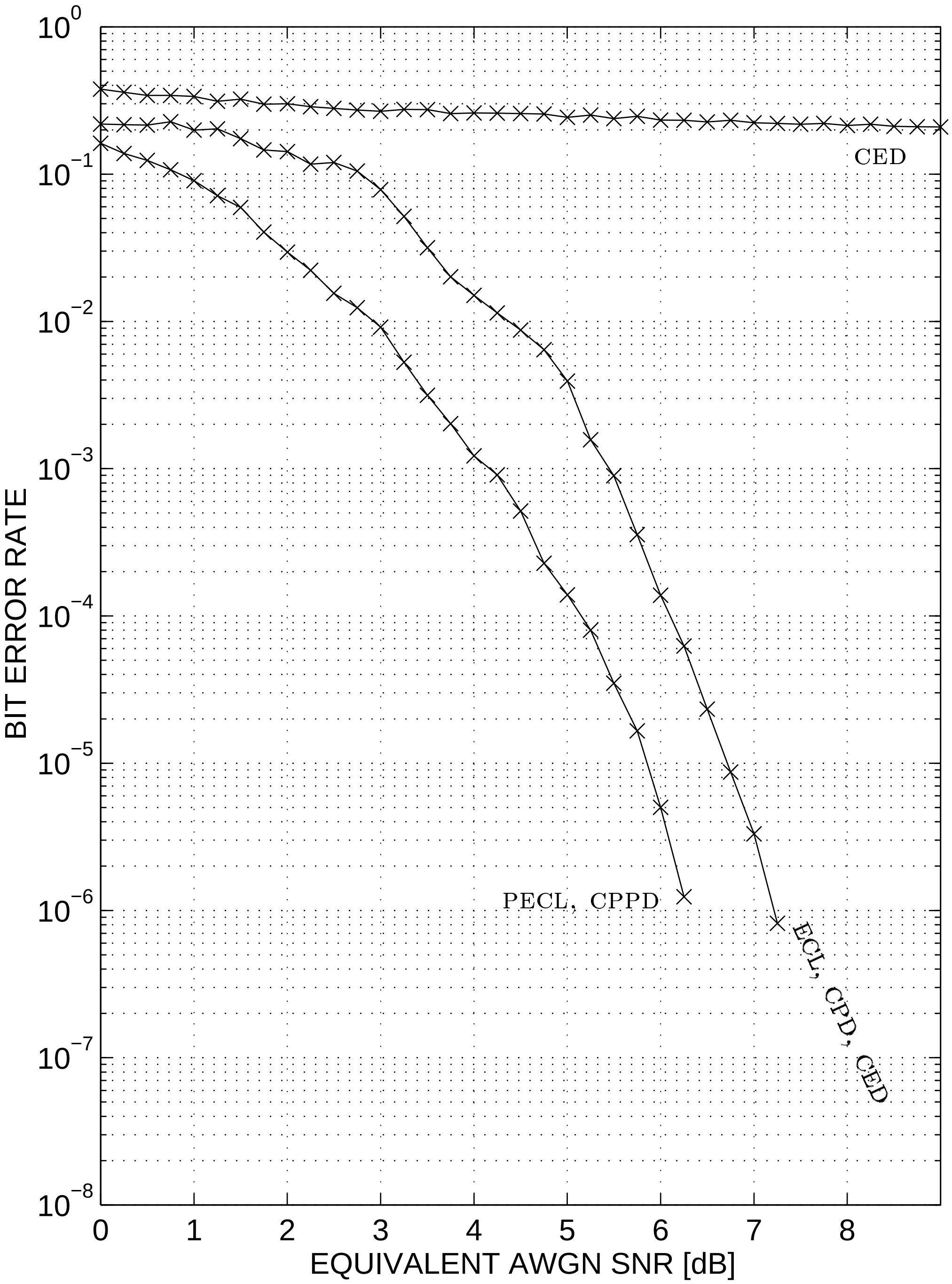


Figure 6.9: BER vs. equivalent AWGN SNR on the 1/2 Band LPF channel for all three codes.

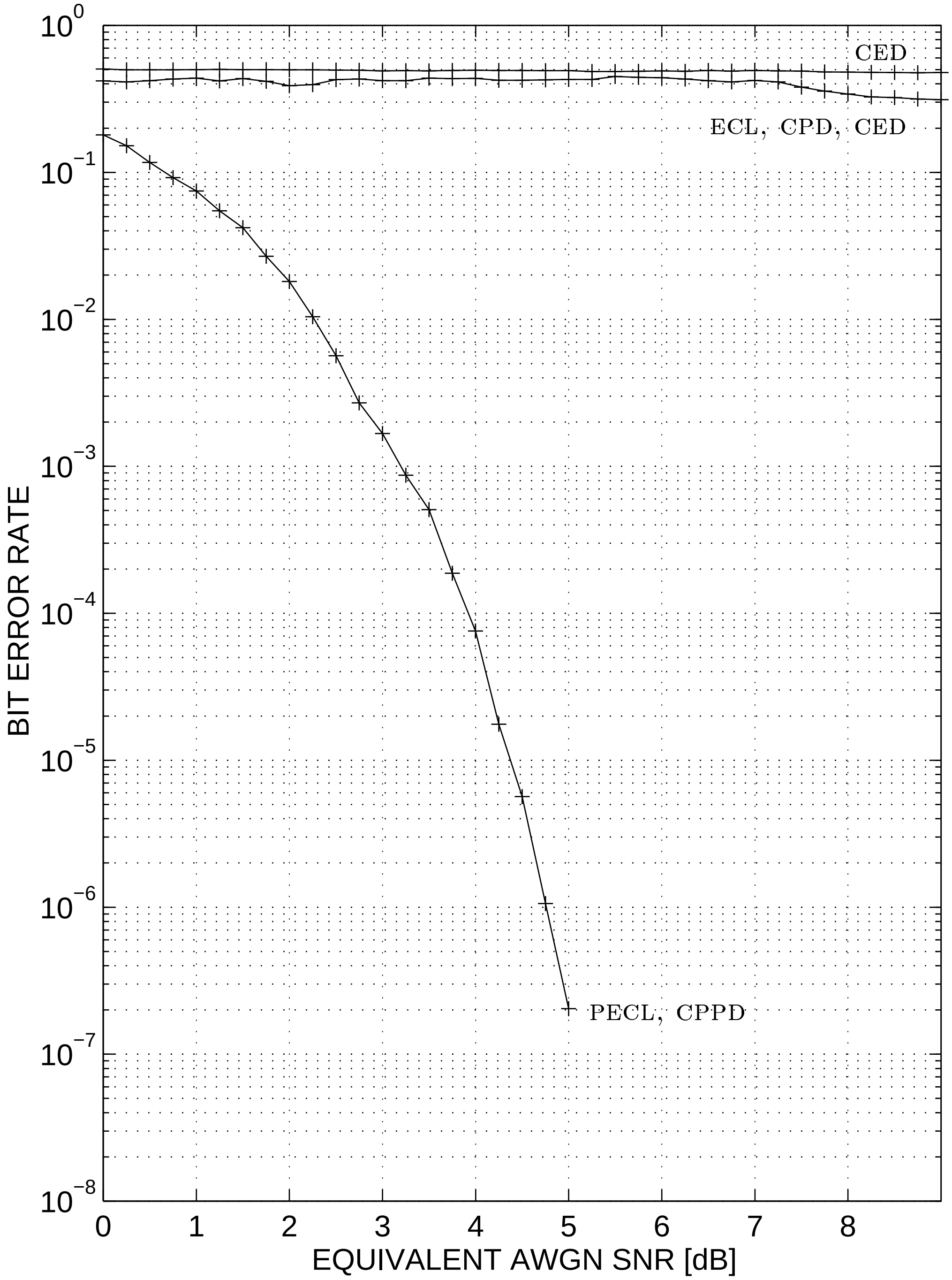


Figure 6.10: BER vs. equivalent AWGN SNR on the 1/3 Band LPF channel for all three codes.

6.5 Summary

This chapter concludes five chapters of investigation into how a trellis code should be designed for robust broadcast multicarrier modulation. The metrics presented in Chapter 2 were applied to a specific transmission scenario where $k = 1$ bits are transmitted per symbol and a complexity of $\nu = 6$ memory elements is tolerable. The searches used the superior-in-profile constellations identified in Chapter 4 and were completed efficiently by avoiding the distance equivalent codes identified in Chapter 3.

The prediction in Chapter 6 of poor performance by 4-PSK and 8-PSK on the lowpass filter channels was confirmed by the simulations in Section 6.4. However, the information-theoretic analysis of Chapter 6 could not have predicted that the metrics of CED and CPD actually decreased when the constellation size was increased beyond an optimal size.

Section 6.4 clearly demonstrated the superiority of the periodic metrics in designing codes for severely frequency selective channels.

Chapter 7

Achievable Rates for Tomlinson-Harashima Precoding

This chapter examines Tomlinson-Harashima precoding (THP) on discrete time channels having intersymbol interference and additive white Gaussian noise. An exact expression for the maximum achievable information rate of the zero-forcing (ZF) Tomlinson-Harashima precoder (THP) is derived as a function of the channel impulse response, the input power constraint, and the additive white Gaussian noise variance. Information rate bounds are provided for the minimum mean square error (MMSE) THP. The performance of ZF-THP and MMSE-THP relative to each other and to channel capacity is explored in general and for some example channels.

Consider power-constrained additive white Gaussian noise (AWGN) communication channels with intersymbol interference (ISI). One approach for transmission over such channels is to combine a coding scheme designed for ISI-free AWGN channels with signal processing that makes the channel appear ISI-free. This processing might be a linear equalizer, a decision feedback equalizer (DFE), or some form of precoding. In each case, the processing can be optimized to satisfy a zero-forcing (ZF) or a minimum mean square error (MMSE) criterion.

Any real application of the signal processing techniques discussed above either preserves the channel capacity or introduces some loss in capacity. Invertible signal processing, such as a linear equalizer using an invertible filter, will preserve the channel capacity if used with appropriate channel coding and decoding techniques.

Decision feedback equalization also preserves capacity since the input to the decision device can be processed to undo the decision feedback. The resulting signal can then be filtered by the inverse of the feedforward filter, assuming that the feedforward filter is invertible. In practice, the symbol-by-symbol decisions made at the output of a linear equalizer or a decision feedback equalizer may preclude reliable transmission at rates approaching capacity.

For analysis purposes, the decisions in the feedback section of a DFE are assumed to be always correct. The structure resulting from this assumption, the *ideal* DFE, is not invertible since the correct decisions are not available to undo the ideal decision feedback. In fact, the ideal DFE is a different ISI channel with its own capacity that can be below [49], equal [50], or above [51] the original channel capacity, depending on the choice of filters and whether interleaving is employed. Note that the data processing inequality does not apply to the ideal DFE because of the extra information in the correct decisions.

In practice, the decisions used in the feedback section of an actual DFE are not always correct. Once an incorrect decision is made, its presence in the feedback section causes additional errors. This effect, known as error propagation [52], is ignored by the ideal analysis discussed above. Error propagation is exacerbated when DFE is combined with coded modulation techniques using dense constellations. As the constellation distances become smaller, symbol-by-symbol decisions become less accurate. Decisions are required by the DFE long before the redundancy in the coded modulation can improve their accuracy.

Precoding suffers no effect analogous to error propagation, but provides error variances similar to those of the ideal DFE. It can be combined easily with coded modulation schemes as in [53, 54, 55, 56, 57] and other papers. Tomlinson [58] and Harashima [59, 60] independently introduced the concept of precoding. The structure that they presented will be referred to as the Tomlinson-Harashima Precoder (THP). There are other precoding structures [55, 56, 57], but this dissertation is concerned only with quantifying the capacity loss introduced by THP.

Price [61] and Harashima et al. [60] showed that ZF-THP achieves the maximum possible mutual information at high SNR for pulse amplitude modulation (PAM) inputs. Miyakawa et al. [59] computed the rates achievable with ZF-THP for a specific coaxial cable channel using high SNR approximations. Mazo and Salz [62] characterized the power difference between the inputs and outputs of a THP transmitter and extended the ZF-THP from real-valued signals and filters to complex-valued signals and filters. Cioffi and Dudevoir [63] introduced the MMSE-THP and compared its output SNR with that of the ZF-THP.

Unlike linear equalization and DFE, THP employs nonlinear processing that is not invertible. Thus some loss from channel capacity is expected in a communication system employing THP. This dissertation quantifies that loss from capacity for any given ISI channel and AWGN variance. An exact formula is derived for the ZF-THP information rate. Upper and lower bounds are provided for the MMSE-THP information rate. These information rate characterizations do not rely on high SNR approximations; they are valid for any SNR.

The loss from capacity at a particular SNR depends on the specific channel impulse response. Several impulse responses will be studied as examples. For channels with severe ISI, the MMSE-THP can provide a significant performance improvement over the ZF-THP for low to mid-range SNR. When the SNR becomes sufficiently large, these two techniques become identical. At high SNR the only loss from capacity incurred by THP is the shaping loss described by Forney [64].

In Section 7.1, Tomlinson-Harashima precoding will be reviewed. Section 7.2 presents the information rate characterizations for ZF-THP and MMSE-THP. Initially the analysis is confined to real-valued inputs and impulse responses. However, the section concludes by extending the results to the complex case. Section 7.3 applies the rate characterizations to five example impulse responses.

7.1 Tomlinson-Harashima Precoding

7.1.1 The Channel Model

In this dissertation, impulse responses will be denoted by their formal D-transforms. For example, the channel impulse response $\{h_k\}$ will be referred to as $H(D)$, where $H(D) = \sum_k h_k D^k$. For an AWGN channel with ISI, the input sequence $\{x_k\}$ is filtered by the channel $H(D)$. White Gaussian noise n_k is added to produce the channel output. The input sequence must obey an average power constraint $\frac{1}{N}\sum_{k=1}^{N} E[x_k^2] \leq$

P, where N is the number of symbols in a codeword. In this dissertation, the input sequence $\{x_k\}$ will usually be independent and identically distributed. In this case, the power constraint is simply $E[x_k^2] \leq P$.

7.1.2 The Modulo Operator Γ_t

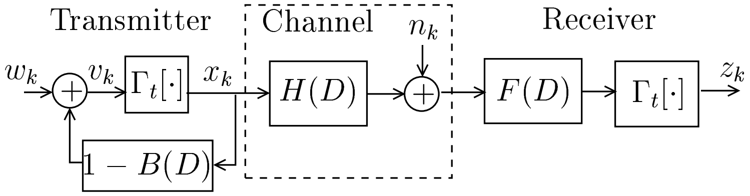


Figure 7.1: Communication system using Tomlinson-Harashima Precoding

Figure 7.1 shows the general THP system. Here Γ_t is a mapping from the real numbers $\mathcal{R}$ to $(-t/2, t/2]$ where t is a positive real number. Specifically,

$$\Gamma_t[v_k] = x_k = v_k + a_k \tag{7.1}$$

where a_k is the unique integer multiple of t for which $x_k \in (-t/2, t/2]$. If w_k is an independent, identically distributed (i.i.d.) sequence with a uniform distribution on $(-t/2, t/2]$, then x_k will also be i.i.d. with a uniform distribution on $(-t/2, t/2]$. This can be seen by considering the conditional distribution of x_k in Figure 7.1 given any particular value at the output of the filter $1 - B(D)$. Since this conditional distribution is always uniform over $(-t/2, t/2]$, the marginal distribution of x_k will also be uniform over $(-t/2, t/2]$.

For i.i.d. PAM w_k, as the alphabet size increases $E[x^2]$ converges to $t^2/12$, the second moment of a uniform distribution on $(-t/2, t/2]$ [62]. In this dissertation t is chosen based on this large alphabet PAM approximation of $E[x^2]$. Thus, to satisfy the power constraint, $E[x_k^2] \leq P$, with equality,

$$t = \sqrt{12P}. \tag{7.2}$$

Smaller values of t than this choice give a lower achievable information rate. Larger values of t cause the power constraint to be violated by an i.i.d. PAM w_k sequence. Thus, larger values of t would require w_k to be carefully constructed to satisfy $E[x^2] \leq P$.

7.1.3 Selection of $F(D)$ and $B(D)$

The system in Figure 7.2 is equivalent to that shown in Figure 7.1. The sequence a_k is a sequence of integer multiples of t chosen so that $x_k \in (-t/2, t/2]$ for all k. The noise $\hat{n}_k$ in Figure 2 is n_k filtered by $F(D)$. $B(D)$ and $F(D)$ are linear time invariant filters chosen to minimize either zero-forcing or minimum mean square error optimality criteria. Figure 7.2 will be useful in the information rate derivations that follow.

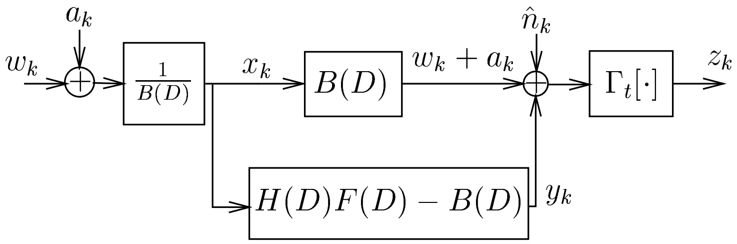


Figure 7.2: Communication system equivalent to Figure 7.1.

$B(D)$ must be causal and monic ($b_0 = 1$) so that the feedback filter $1 - B(D)$ requires only previous values of x_k. B(D) is closely related to the feedback filter of a DFE. The filter $F(D)$ corresponds to the combination of the sampled matched filter and feedforward filter in a DFE.

Choosing $B(D)$ and $F(D)$ to satisfy a zero-forcing criterion produces the scheme originally proposed by Tomlinson [58] and Harashima [60]. In the context of THP, zero-forcing implies forcing $y_k = 0$ (see Figure 7.2) for all k. Spectral factorization techniques [65] satisfy this criterion by producing an allpass $F(D)$ such that $H(D)F(D)$ is causal, monic, and minimum phase. $B(D)$ is chosen to equal $H(D)F(D)$. These are exactly the same values of $B(D)$ and $F(D)$ used in a zero-forcing DFE.

The (unbiased) MMSE-THP [63] is obtained by choosing $F(D)$ and $B(D)$ to minimize $\text{VAR}(\hat{n}_k + y_k)$ under the constraints that $B(D)$ and $H(D)F(D)$ are monic.

As with the ZF-THP, spectral factorization techniques provide the desired filters, which are exactly the values of $B(D)$ and $F(D)$ used in an unbiased MMSE-DFE. Unlike the ZF-THP, $H(D)F(D)$ need not be causal or minimum phase, and $F(D)$ need not be allpass.

7.1.4 THP and DFE Transfer Functions

Following [55], it is interesting to compare the THP transfer function with $a_k = 0$ in Figure 7.2 with the transfer function of the DFE shown in Figure 7.3.

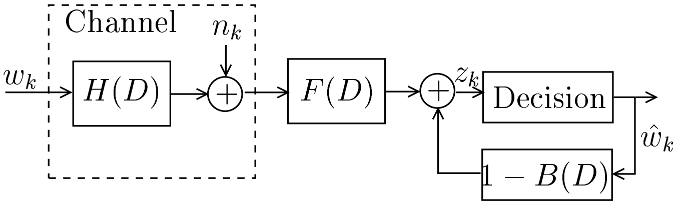


Figure 7.3: Communication System Using a Decision Feedback Equalizer

With the ideal assumption that $\hat{w}_k = w_k$, the DFE transfer function is

$$\frac{Z(D)}{W(D)} = H(D)F(D) + 1 - B(D). \tag{7.3}$$

With $a_k = 0$ in Figure 7.2 (and neglecting the receiver $\Gamma_t[\cdot]$ operator) the transfer function of the THP is

$$\frac{Z(D)}{W(D)} = \frac{H(D)F(D)}{B(D)}. \tag{7.4}$$

As noted in [55], the zero-forcing choices of $B(D)$ and $F(D)$ make the two transfer functions (7.3) and (7.4) identical; they both equal 1. However, this equivalence holds only for the zero-forcing case; the MMSE choices of $B(D)$ and $F(D)$ produce transfer functions that are not identical.

7.2 Information Rates for THP

This section characterizes the maximum information rates achievable by both ZF-THP and MMSE-THP. The notation used in this section is that shown in Figures 7.1 and 7.2. For simplicity, initial analyses are confined to real x_k and real coefficients of $H(D)$. However, the section concludes with an extension to the complex case.

7.2.1 Zero-Forcing

As discussed on page 141, the ZF-THP has $F(D)$ and $B(D)$ chosen so that $y_k = 0$. Recall that $\hat{n}_k$ is n_k filtered by $F(D)$. Since $F(D)$ is an allpass filter, $\hat{n}_k$ is white. While the original AWGN channel with ISI has memory, the overall ZF-THP system is a memoryless channel with input w and output z given by

$$z = \Gamma_t[w + a + \hat{n}_{\mathrm{Z}}]. \tag{7.5}$$

The index k will be suppressed as above when it is not needed for clarity. Furthermore, $\hat{n}_{\mathrm{Z}}$ and $\hat{n}_{\mathrm{M}}$ will be used to denote explicitly the filtered Gaussian noise produced by the ZF and MMSE choices of $F(D)$ respectively. The a term in (7.5) is an integer multiple of t and does not affect the output of $\Gamma_t[\cdot]$. Thus

$$z = \Gamma_t[w + \hat{n}_{\mathrm{Z}}]. \tag{7.6}$$

The ZF-THP memoryless channel has mutual information

$$\begin{aligned} I(w; z) &= h(z) - h(z|w) && (7.7)\\ &= h(\Gamma_t[w + \hat{n}_{\mathrm{Z}}]) - h(\Gamma_t[\hat{n}_{\mathrm{Z}}]) && (7.8)\\ &\leq \log_2(t) - h(\Gamma_t[\hat{n}_{\mathrm{Z}}]) && (7.9) \end{aligned}$$

where $h(\cdot)$ denotes differential entropy. The upper bound of (7.9) follows from the maximum differential entropy of a random variable with support constrained to an interval. This bound can be achieved by choosing w to be i.i.d. uniform over the interval $(-t/2, t/2]$ giving

$$C_{\text{ZF-THP}} = \log_2(t) - h(\Gamma_t[\hat{n}_{\mathrm{Z}}]). \tag{7.10}$$

7.2.2 Minimum Mean Square Error

As discussed in the introduction, precoding and equalization in general seek to make the channel ISI-free. However, the channel produced by the MMSE-THP still has nonzero ISI since $y_k \neq 0$. The MMSE choice of $F(D)$ also produces a non-white $\hat{n}_k$.

Consistent with the goal of producing an ISI-free channel, ideal interleaving will be assumed in the following analysis to produce a memoryless channel. Ideal interleaving indicates interleaving deep enough that

$$\hat{n}_i - \hat{n}_j, \quad y_i - y_j, \quad w_i - y_j \tag{7.11}$$

when indices i and j are associated with the same codeword. The symbol $-$ is used to indicate statistical independence. Ideal interleaving in this context is information-lossy since information contained by future and past values of z about the current values of w and $\hat{n}$ is neglected. Thus higher information rates would be possible without interleaving.

With interleaving, the inputs and outputs associated with any particular codeword behave as those of a memoryless channel with peak limited input w and output

$$\begin{aligned} z &= \Gamma_t[w + a + y + \hat{n}_{\mathrm{M}}] && (7.12)\\ &= \Gamma_t[w + y + \hat{n}_{\mathrm{M}}]. && (7.13) \end{aligned}$$

The mutual information of this channel is given below.

$$\begin{aligned} I(w; z) &= h(z) - h(z \mid w) && (7.14)\\ &= h(\Gamma_t[w + y + \hat{n}_{\mathrm{M}}]) - h(\Gamma_t[y + \hat{n}_{\mathrm{M}}] \mid w). && (7.15) \end{aligned}$$

An upper bound on the capacity of the channel produced by MMSE-THP with ideal interleaving can be found from the following two inequalities:

$$\begin{aligned} h(\Gamma_t[w + y + \hat{n}_{\mathrm{M}}]) &\leq \log_2(t) && (7.16)\\ h(\Gamma_t[y + \hat{n}_{\mathrm{M}}] \mid w) &\geq h(\Gamma_t[y + \hat{n}_{\mathrm{M}}] \mid w, y) && (7.17)\\ &= h(\Gamma_t[\hat{n}_{\mathrm{M}} \mid w, y] && (7.18)\\ &= h(\Gamma_t[\hat{n}_{\mathrm{M}}]). && (7.19) \end{aligned}$$

For (7.19), note that $\hat{n}_{\mathrm{M}}$ is independent of (w, y). The above inequalities give

$$C_{\text{MMSE-THP}} \leq \log_2(t) - h(\Gamma_t[\hat{n}_{\mathrm{M}}]). \tag{7.20}$$

Note that the upper bound was obtained in part by neglecting the ISI term y. The tightness of this bound depends largely on the variance of y.

A lower bound on $C_{\text{MMSE-THP}}$ can be found by assigning w an i.i.d. uniform distribution on $(-t/2, t/2]$. For such a w, the output z is i.i.d. uniform on $(-t/2, t/2]$. As with x, this is seen by noting that all the relevent conditionals are uniforms. Thus

$$h(z) = \log_2(t). \tag{7.21}$$

Removing conditioning always increases entropy, so

$$h(\Gamma_t[y + \hat{n}_{\mathrm{M}}] \mid w) \leq h(\Gamma_t[y + \hat{n}_{\mathrm{M}}]). \tag{7.22}$$

The region of support of $\Gamma_t[y+\hat{n}_{\mathrm{M}}]$ is $(-t/2, t/2]$. Thus an upper bound on $h(\Gamma_t[y+\hat{n}_{\mathrm{M}}])$ is the maximum differential entropy for a distribution with region of support $(-t/2, t/2]$ and variance equal to $\mathrm{VAR}(y + \hat{n}_{\mathrm{M}})$. A truncated Gaussian distribution [66] achieves this maximum differential entropy. Let $\mathcal{T}(\sigma^2, t)$ be a zero mean Gaussian truncated to $(-t/2, t/2]$ with variance (after truncation) of σ^2. Then,

$$h(\Gamma_t[y + \hat{n}_{\mathrm{M}}]) \leq h(\mathcal{T}(\sigma^2, t)). \tag{7.23}$$

Combining (7.21) and (7.23) produces the desired lower bound on $C_{\text{MMSE-THP}}$,

$$C_{\text{MMSE-THP}} \geq \log_2(t) - h(\mathcal{T}(\sigma^2, t)). \tag{7.24}$$

Following [66], $h(\mathcal{T}(\sigma^2, t))$ (in bits) is computed as follows:

$$h(\mathcal{T}(\sigma^2, t)) = \frac{1}{2}\log_2(2\pi\gamma^2 e) + \log_2(2\,\mathrm{erf}(t/2\gamma)) - \frac{t\log_2(e)\exp(-t^2/8\gamma^2)}{4\gamma\sqrt{2\pi}\ \mathrm{erf}(t/2\gamma)}.$$

The variance γ^2 of the Gaussian before truncation can be found from σ^2 by solving the equation

$$\sigma^2 = \gamma^2 - \frac{\gamma t \exp(-t^2/8\gamma^2)}{2\sqrt{2\pi}\ \mathrm{erf}(t/2\gamma)}, \tag{7.25}$$

where

$$\mathrm{erf}(t) = \frac{1}{\sqrt{2\pi}} \int_0^t \exp(-u^2/2)du.$$

Note that $\Gamma_t[y + \hat{n}_{\mathrm{M}}] = \Gamma_t[y + \Gamma_t[\hat{n}_{\mathrm{M}}]]$. A truncated Gaussian with $\sigma^2 = \mathrm{VAR}(y + \Gamma_t[\hat{n}_{\mathrm{M}})])$ produces a slightly tighter bound, but requires the computation of $\mathrm{VAR}(\Gamma_t[\hat{n}_{\mathrm{M}}])$. This tighter bound was used in the plots shown in Section 5.

In situations where the receiver complexity is to be minimized, the feedforward filter $F(D)$ can be moved to the transmitter as discussed in [67]. In this case, $\hat{n}_k$ is simply n_k/s, where s is the real scalar for which $sF(D)$ does not change the transmitter power. The bounds derived above apply to this case directly, since the variance of $\hat{n}_k$ has not changed.

7.2.3 Extension to Complex Values

In [62] the real-valued ZF-THP of Tomlinson and Harashima was extended to a complex-valued ZF-THP suitable for QAM transmission. An analogous extension follows for the MMSE-THP. Figures 7.1 and 7.2 are valid block diagrams for the general complex THP provided that Γ_t is redefined to be the appropriate many to one mapping in the space of complex numbers. Let v be the complex number with real part v_{R} and imaginary part v_{I}. Then

$$\Gamma_t[v] = v + a = x\,, \tag{7.26}$$

where a_{R} and a_{I} are chosen so that x_{R} and x_{I} are in the interval $(-t/2, t/2]$. To avoid confusion, a superscript will be used henceforth to denote the dimensionality of Γ_t. For a complex v,

$$\Gamma_t^2[v] = \Gamma_t^1[v_{\mathrm{R}}] + i\Gamma_t^1[v_{\mathrm{I}}]. \tag{7.27}$$

This mapping is the natural extension of the real-valued THP. Other mappings are possible and may even be desirable [56, 57, 64], but they will not be discussed here.

Let the input power constraint be $E[|x|^2] \leq 2P$. The following bounds for the complex case result from arguments that parallel the arguments presented above for the real case.

$$
\begin{aligned}
C_{\text{ZF-THP}} &= 2\log_2(t) - h(\Gamma_t^2[\hat{n}_{\text{Z}}]) \qquad (7.28)\\
C_{\text{MMSE-THP}} &\leq 2\log_2(t) - h(\Gamma_t^2[\hat{n}_{\text{M}}]) \qquad (7.29)\\
C_{\text{MMSE-THP}} &\geq 2\log_2(t) - h(\mathcal{T}(\sigma_{\text{R}}^2, t)) - h(\mathcal{T}(\sigma_{\text{I}}^2, t)) \qquad (7.30)\\
&= 2\log_2(t) - 2h(\mathcal{T}(\sigma_{\text{R}}^2, t)) \qquad (7.31)
\end{aligned}
$$

In (7.30) and (7.31), $\sigma_{\text{R}}^2 = \textsc{var}(y_{\text{R}}+\Gamma_t^1[\hat{n}_{\text{M,R}}])$ and $\sigma_{\text{I}}^2 = \textsc{var}(y_{\text{I}}+\Gamma_t^1[\hat{n}_{\text{M,I}}])$. Similar to the real case, the lower bound on $C_{\text{MMSE-THP}}$ assumes that w_{R} and w_{I} are are independent of each other and that both are uniform on $(-t/2, t/2]$. As a result, $\sigma_{\text{R}}^2 = \sigma_{\text{I}}^2$.

7.3 Performance Comparisons

In this section, the rate characterizations derived in the previous section are used to examine how much loss from channel capacity is experienced by THP. The well known "shaping loss" is demonstrated to be the only loss experienced by THP at high SNR. Information rates achieved by MMSE-THP and ZF-THP are compared with each other and to capacity for AWGN with no ISI. THP information rates are explored for three different ISI channel examples. Finally, a downsampled version of the third ISI channel is examined to show that lower symbol rates can improve THP performance at low SNR.

7.3.1 Shaping Loss

The relationships between the original channel capacity, the capacity using MMSE-THP, and the capacity using ZF-THP generally depend on the ISI. However, regardless of the particular ISI, the achievable information rates of both the MMSE-THP and ZF-THP structures converge in the limit of high SNR to 0.255 bits (the shaping loss) less than the maximum achievable information rate for an i.i.d. input sequence.

In the limit of high SNR, it is well known that the filters $B(D)$ and $F(D)$ used in the ZF-DFE are exactly those used in the MMSE-DFE. Since the filters used in a DFE are the same as those used in the corresponding THP, the ZF-THP and MMSE-THP structures become identical in the limit of high SNR. This behavior is evident in the bounds presented previously since the upper and lower bounds on $C_{\text{MMSE-THP}}$ converge to $C_{\text{ZF-THP}}$ in the limit of high SNR. In the limit of high SNR, all three expressions, (7.10), (7.20), and (7.24) converge to

$$\log_2(t) - \frac{1}{2}\log_2(2\pi e E[\hat{n}^2]). \tag{7.32}$$

From [50] the maximum information rate achievable on a real AWGN channel with ISI using an i.i.d. input sequence can be expressed as

$$\frac{1}{2}\log_2(1 + \text{SNR}_{\text{MMSE-U}}), \tag{7.33}$$

with

$$\text{SNR}_{\text{MMSE-U}} = \frac{t^2/12}{E[\hat{n}^2] + E[y^2]}, \tag{7.34}$$

where $\hat{n}$ and y are those occurring in the MMSE-THP. Subtracting (7.32) from (7.33) yields

$$\frac{1}{2}\log_2\frac{\pi e}{6} = 0.255 \text{ bits},$$

plus terms that go to zero for high SNR.

Thus at high SNR the difference between the THP information rate and the maximum achievable rate using an i.i.d. input sequence is exactly the well known shaping loss of 0.255 bits (1.53 dB of SNR loss on an AWGN channel). Forney [64] identified this loss as being a result of the peak limitation introduced by precoding. The above derivation demonstrates that this is the *only* loss introduced by Tomlinson-Harashima precoding at high SNR.

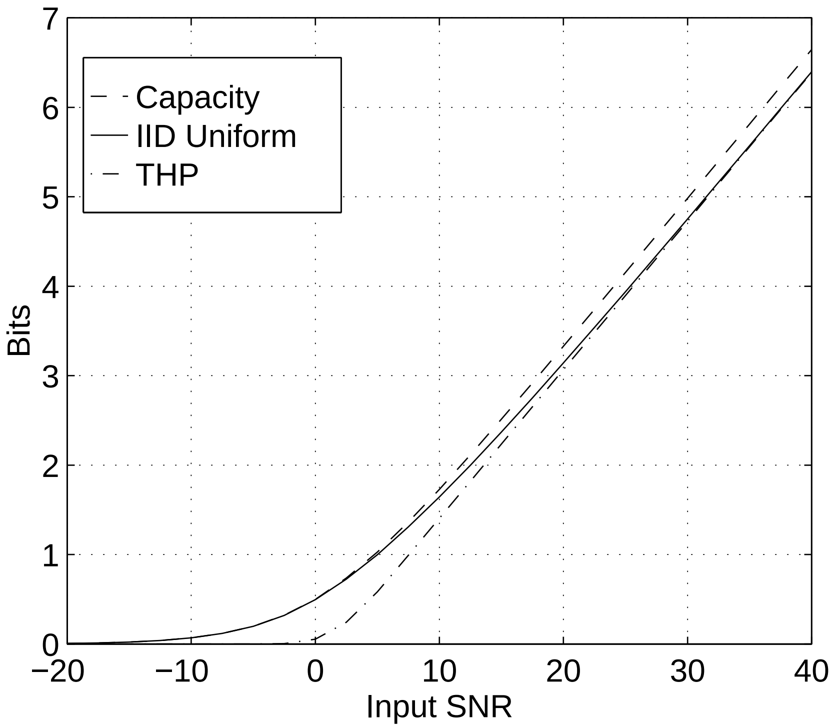


Figure 7.4: Information Rates on the AWGN Channel

7.3.2 THP on the AWGN Channel

Figure 7.4 shows information rate curves of interest for the AWGN channel with no ISI. For this channel, the ZF-THP and MMSE-THP are exactly the same with $B(D) = F(D) = 1$. The capacity for the channel produced by this structure is

$$C_{\text{THP}} = \log_2(t) - h(\Gamma_t[n]). \tag{7.35}$$

The well known AWGN channel capacity is the same as the mutual information for an i.i.d. Gaussian x_k. It is

$$C = \frac{1}{2}\log_2\left(1 + \frac{t^2}{12E[n^2]}\right), \tag{7.36}$$

where $t^2/12$ is the average transmitter energy. The solid line plotted in Figure 7.4 is the mutual information for x_k that is i.i.d. with a uniform distribution on the modulo interval $(-t/2, t/2]$, which is

$$I_{\text{IID-UNIFORM}} = h(x+n) - \frac{1}{2}\log_2(2\pi e E[\hat{n}^2]). \tag{7.37}$$

Using THP with w i.i.d. uniform over modulo interval produces x with a uniform distribution over the modulo interval. Thus (7.37) is a lower bound on the capacity of the THP structure without the receiver modulo operation for the AWGN channel.

Note that the IID-UNIFORM mutual information converges to the original channel capacity at low SNR. This indicates that the receiver modulo operation is responsible for all of the loss from capacity at low SNR for the AWGN channel. At high SNR this behavior is reversed. As shown in the previous subsection, all of the loss from channel capacity at high SNR can be explained by the peak constraint imposed by the transmitter modulo operation.

7.3.3 Example Channels with ISI

Three ISI channels are used in the following study of THP performance: the Two-Tap channel, the Ramp channel, and the Step channel. The impulse and frequency responses of these channels are shown in Figures 7.5 and 7.6 respectively. All three impulse responses shown are minimum phase. Thus the ZF-THP will cancel all the taps except the first. These cancelled taps can be viewed as an opportunity for the MMSE-THP to improve upon the ZF-THP performance.

Figures 7.7–7.9 show THP performance as a function of input SNR for the three impulse responses. For each of these channels the ZF-THP capacity is plotted as well as the upper and lower bounds on the MMSE-THP capacity. These information rates are compared with curves showing the original channel capacity and the mutual information for an i.i.d. Gaussian x_k with $E[x^2] = t^2/12$. The IID-GAUSSIAN curve is displayed to indicate the loss due to the fact that the THP transmitter produces approximately i.i.d. inputs.

Capacity was approximated by discrete waterfilling using an 8192-point FFT. The THP information rates required the differential entropy and the variance of the modulo of a Gaussian random variable. These values were computed by numerical integration using Mathematica. The variances of $\hat{n} + y$ and $\hat{n}$ were computed using

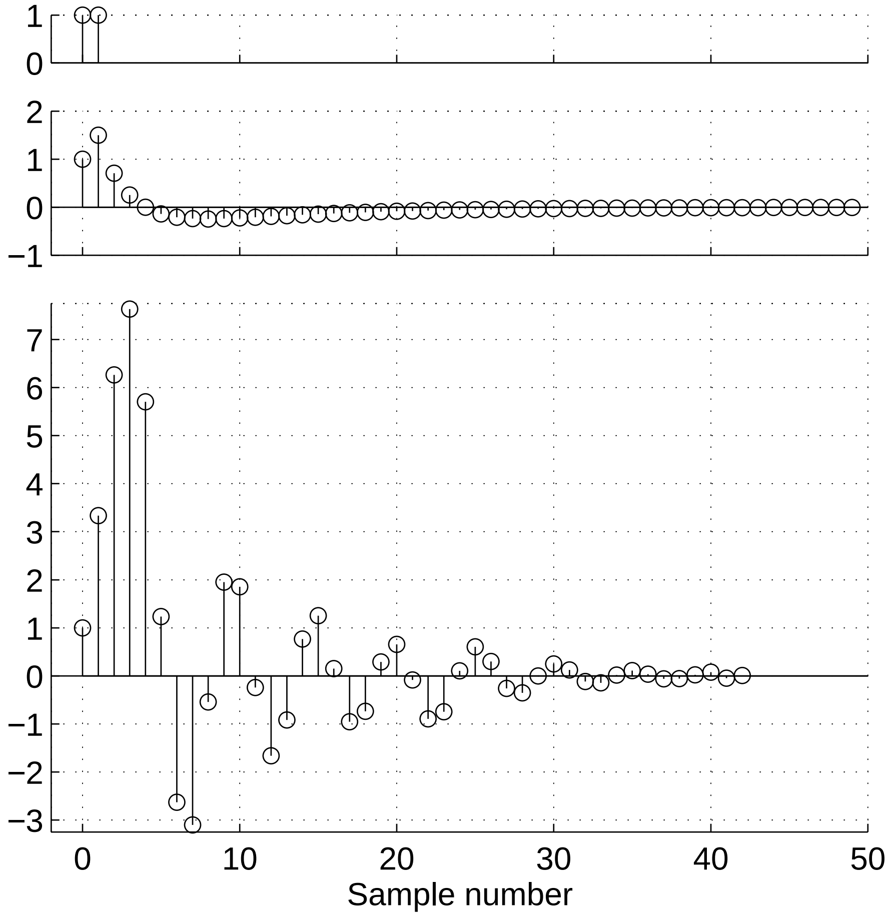


Figure 7.5: Impulse responses for the Two-Tap (top), Ramp (middle), and Step (bottom) channels.

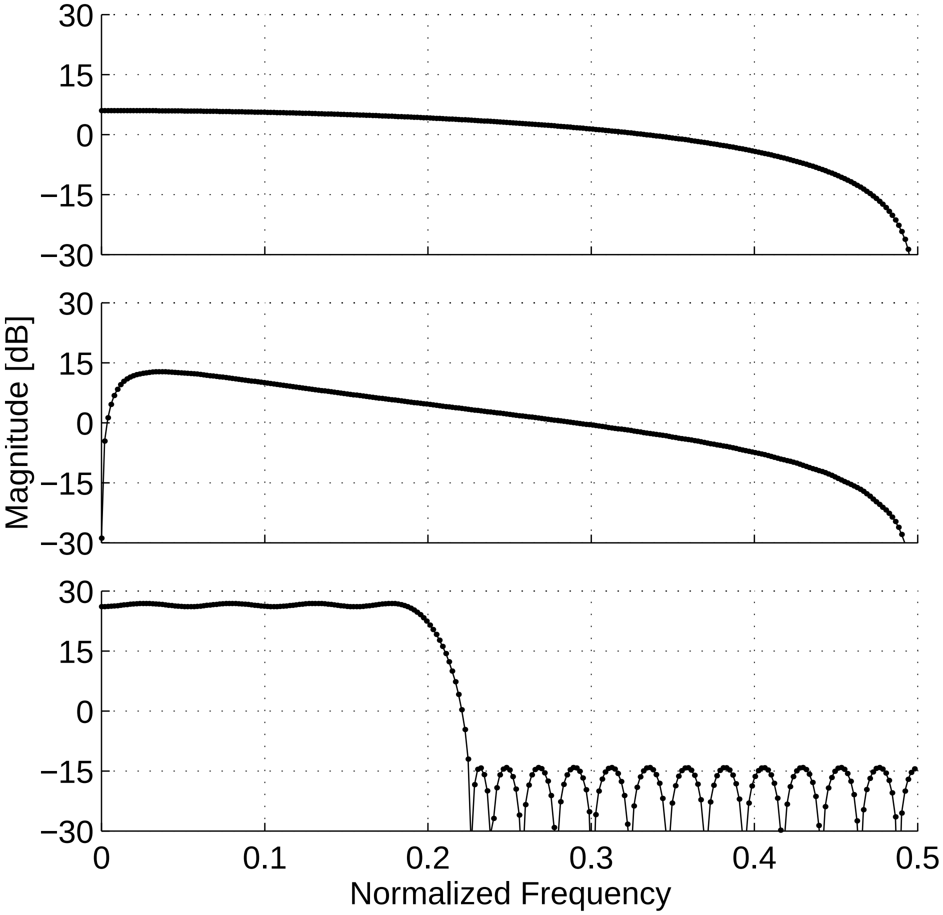


Figure 7.6: Frequency responses for the Two-Tap (top), Ramp (middle), and Step (bottom) channels.

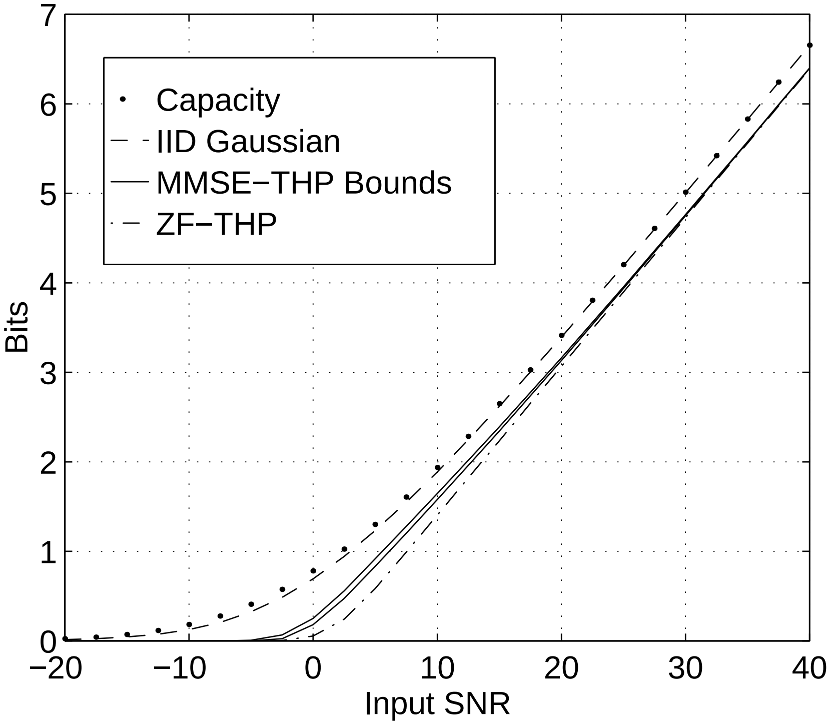


Figure 7.7: Information Rates on the Two-Tap Channel.

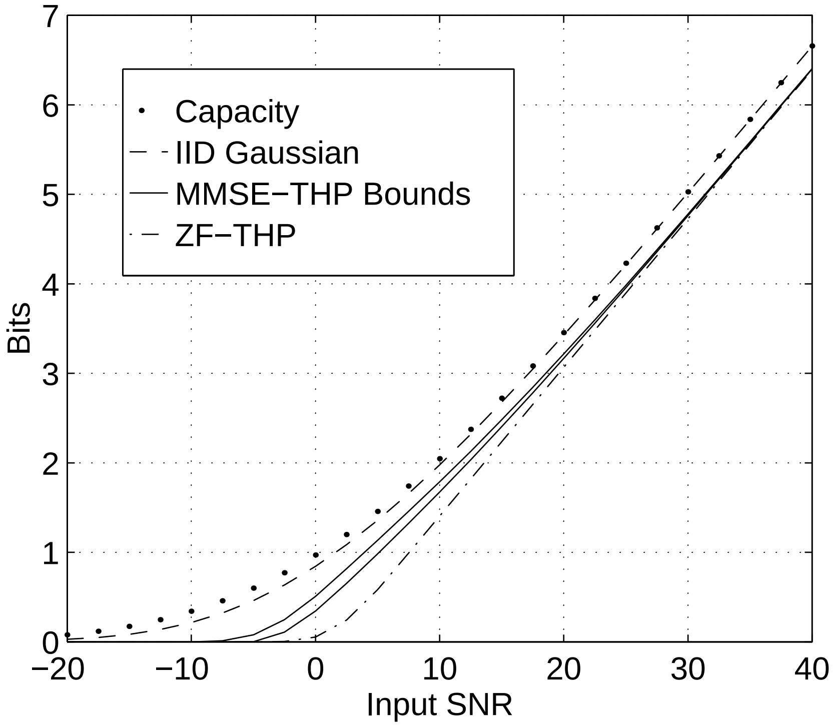


Figure 7.8: Information Rates on the Ramp Channel.

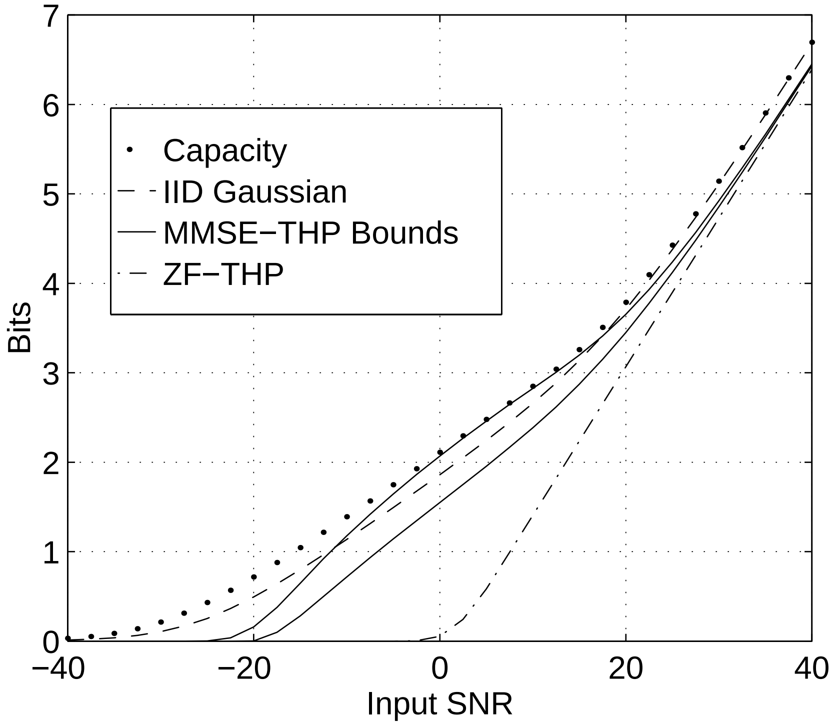


Figure 7.9: Information Rates on the Step Channel.

standard spectral factorization and partial fractions techniques [65, 63, 50, 45]. The variance of y was then computed as the difference of these two variances.

At low SNR, the MMSE-THP outperforms the ZF-THP for all three ISI channels studied here. The MMSE-THP may not strictly dominate the ZF-THP. This is unlike the DFE, where ideal MMSE-DFE performance strictly dominates ideal ZF-DFE performance. However, it does follow from a comparison of $C_{\text{ZF-THP}}$ to the lower bound on $C_{\text{MMSE-THP}}$ that the ZF-THP capacity can be only negligibly higher (0.08 bits) than the MMSE-THP capacity. By virtue of the MMSE cost function,

$$\operatorname{var}(\hat{n}_{\text{M}} + y_{\text{M}}) \leq \operatorname{var}(\hat{n}_{\text{Z}}). \tag{7.38}$$

Thus σ^2 in (7.24) is always less than or equal to the variance of $\hat{n}_{\text{Z}}$ in (7.10). Maximizing the difference between (7.24) and (7.10) with the error variances set equal provides the $H(D)$ independent bound of 0.08 bits. For $t/\sigma > 2$ the difference between (7.24) and (7.10) is negligible.

As the ISI becomes more severe, the performance gap between MMSE-THP and ZF-THP at low SNR becomes more pronounced. Also, the MMSE upper bound becomes less tight because the neglected y term becomes more significant. This upper bound behavior is quite evident in Figure 7.9.

7.3.4 Symbol Rate Optimization

For any fixed symbol rate, performance can be improved by adding a transmitter filter that maintains the power constraint but shapes the transmitter power spectrum to be optimal [50, 63]. The information rates derived in Section 3 apply to the resulting system simply by replacing $H(D)$ with the cascade of the transmit optimization filter and $H(D)$.

The symbol rate should also be optimized. Certainly this requires that the symbol rate be high enough to use the available bandwidth. However, in the context of THP, the symbol rate can be *too* high. As an example, consider the Step channel with a symbol rate half as high as considered previously. The effective impulse response was obtained by lowpass filtering and downsampling the original Step channel impulse response. Figure 7.10 shows this impulse response (converted to minimum phase) and the corresponding frequency response.

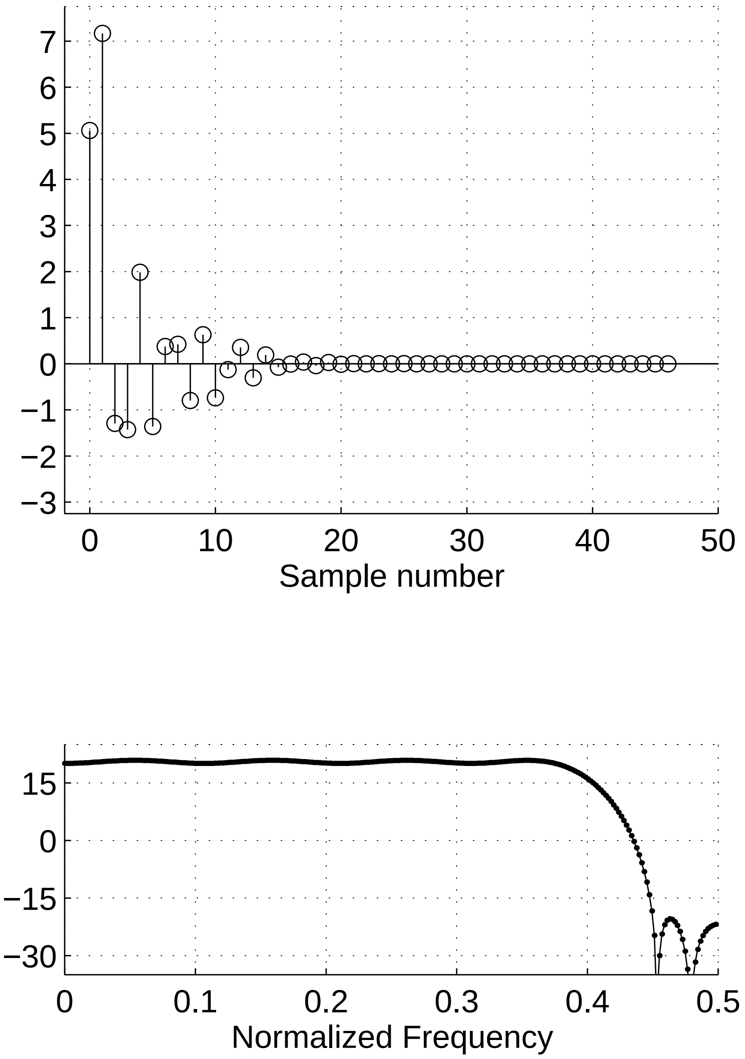


Figure 7.10: Minimum Phase Impulse Response (top) and Frequency Response (Bottom) for the Step channel with a Halved Symbol Rate.

Figures 7.11, 7.12, 7.13, 7.14 compare the original Step channel with the channel produced by halving the symbol rate. To allow a fair comparison of information rate per unit time, the downsampled Step channel rate values were divided by 2. For a fair comparison of energy use per unit time, the SNR values for the downsampled Step channel were decreased by 3 dB. Thus information rate and transmitter power are normalized to be per symbol at the original symbol rate.

Figure 7.11 shows that the discrete time channel produced by the lower symbol rate has a higher i.i.d. Gaussian input information rate than the original Step channel at low SNR. For low SNR, the i.i.d. Gaussian input is closer to the optimal waterfilling spectrum with the lower symbol rate than with the original symbol rate. For large SNR values, the original symbol rate provides much better performance than the lower symbol rate.

Figure 7.12 demonstrates that for a large region of low SNR, the lower symbol rate provides a dramatic improvement in the ZF-THP information rate. Most of this performance improvement is due to the milder ISI produced by the lower symbol rate rather than a more optimal transmit spectrum. When the SNR becomes large enough, the original symbol rate again provides higher information rates.

Figure 7.13 compares the MMSE-THP bounds for the two channels. At low SNR the lower symbol rate provides some improvement, but this improvement is not as substantial as in the ZF-THP case. Note that the MMSE-THP upper bound is tighter for the channel produced by the lower symbol rate due to a smaller residual ISI term.

Figure 7.14 shows all of the curves in the previous three figures. This figure summarizes the low SNR advantages of the lower symbol rate. Note that the gap between MMSE-THP and ZF-THP performance is much smaller for the lower symbol rate.

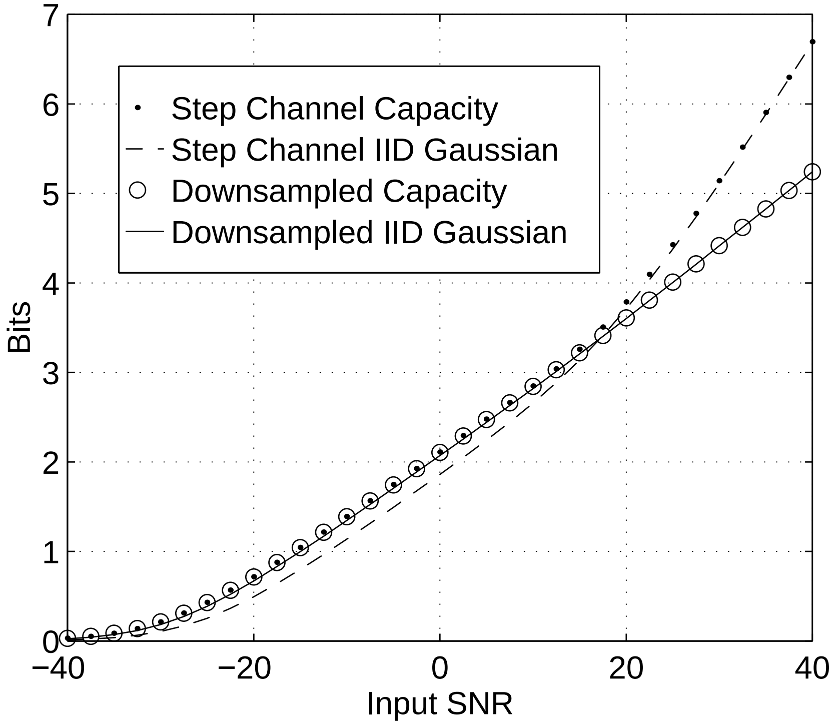


Figure 7.11: Capacities and Gaussian i.i.d. input information rates for Step channel with original and halved symbol rates. Information rate and transmitter power are normalized to be per symbol at the original symbol rate.

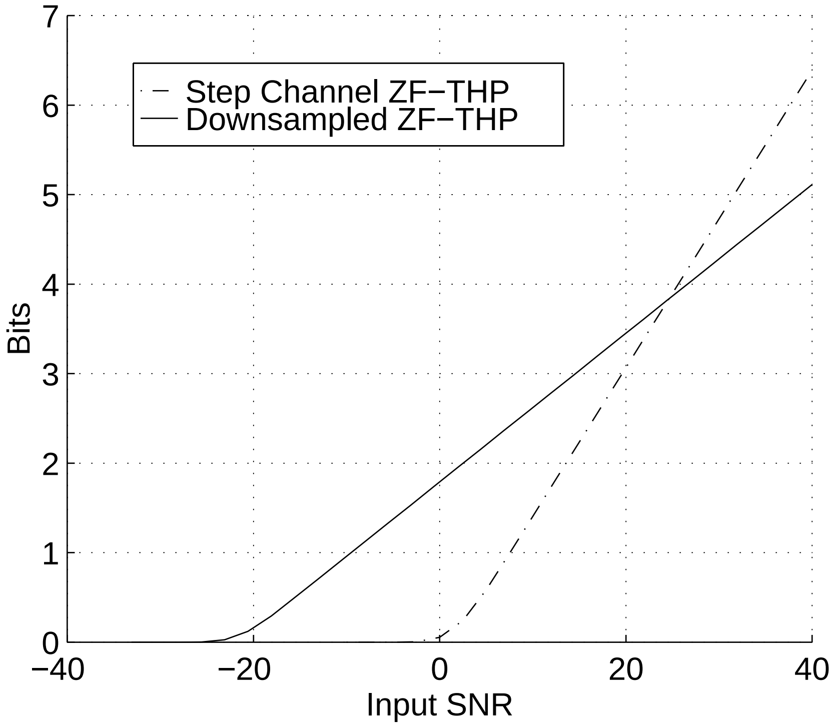


Figure 7.12: ZF-THP information rates for Step channel with original and halved symbol rates. Information rate and transmitter power are normalized to be per symbol at the original symbol rate.

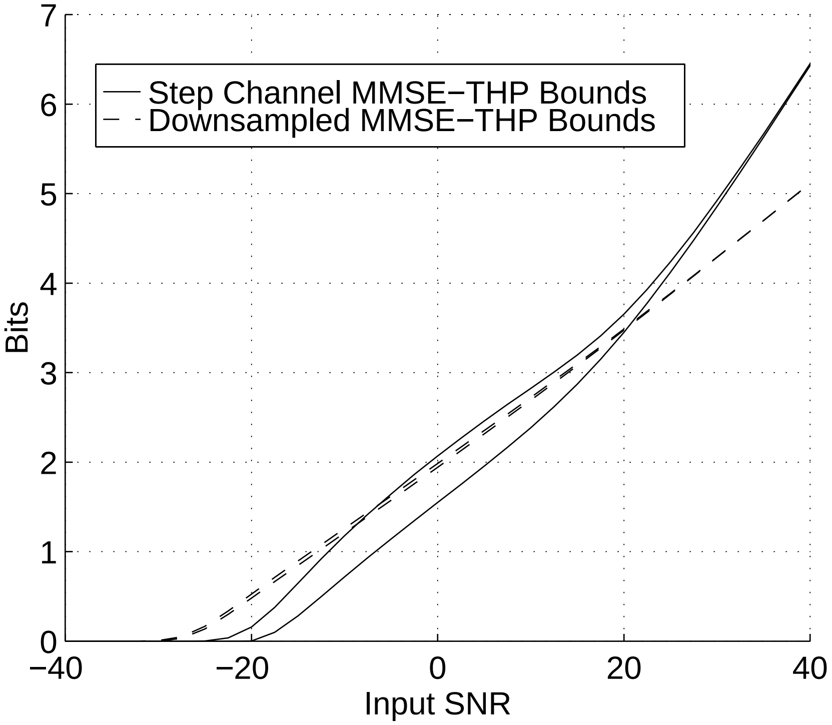


Figure 7.13: MMSE-THP information rate bounds for Step channel with original and halved symbol rates. Information rate and transmitter power are normalized to be per symbol at the original symbol rate.

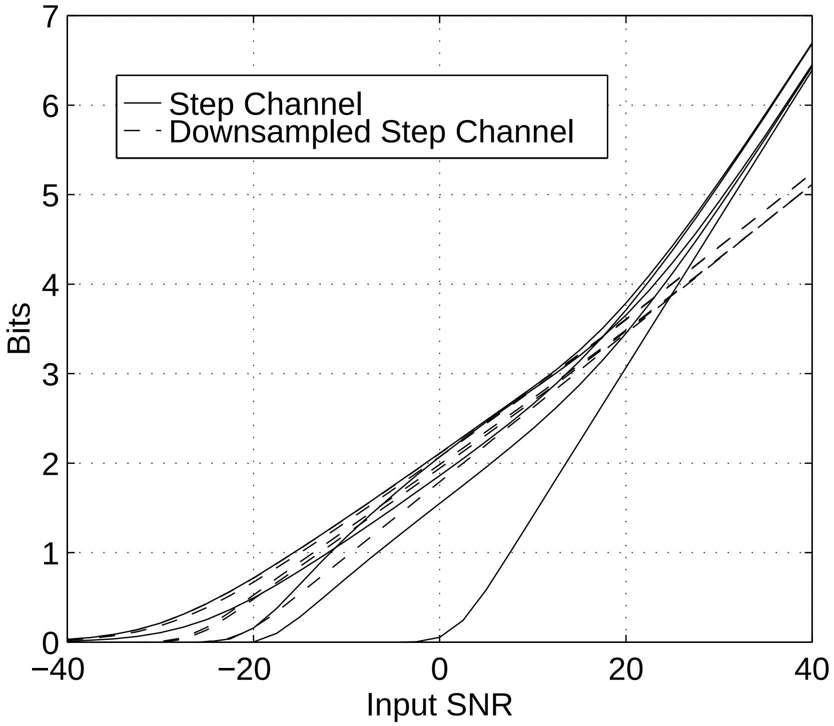


Figure 7.14: Information rates for Step channel with original and halved symbol rates. Information rate and transmitter power are normalized to be per symbol at the original symbol rate.

7.4 Summary

The maximum achievable information rate of the zero-forcing THP was derived as a function of the channel impulse response, the input power constraint, and the AWGN variance. Bounds were provided for the minimum mean square error (MMSE) THP. The tightness of the upper bound depends on the variance of the residual ISI term.

Regardless of the ISI for a particular channel, the performance of both the MMSE-THP and the ZF-THP become identical at high SNR. Both structures suffer exactly the shaping loss of 0.255 bits or 1.53 dB due to the peak constraint at the transmitter. At high SNR there is no additional loss due to the many-to-one mapping by the receiver modulo operation. For the AWGN channel at low SNR, the situation is reversed; the loss from capacity is due entirely to the receiver modulo with no loss due to the peak constraint.

At low SNR, there is a performance difference between the MMSE-THP and the ZF-THP for channels with ISI. In general the MMSE-THP performs better than the ZF-THP, and the performance gap becomes more pronounced as ISI becomes more severe. In some cases of severe ISI a lower symbol rate (perhaps used with a different carrier frequency) can improve the achievable rate of THP and decrease the gap between MMSE-THP and ZF-THP performance.

Chapter 8

Conclusion

8.1 Thesis Summary

Chapters 2–6 present a trellis code design strategy for channels with correlated fading. Chapter 7 characterize achievable rates for Tomlinson-Harashima precoding.

Chapter 2 began the investigation of trellis code design with a discussion of the design metrics used to select codes for various channel environments. This chapter introduced *periodic effective code length* and *code periodic product distance* as metrics for selecting a trellis code for channels with correlated fading.

Chapter 3 formally introduced trellis codes and their principal building block, convolutional codes. A method was presented for identifying trellis codes that are equivalent in the metrics of Chapter 2. This method reduces search complexity significantly since metrics are then computed for only one in a set of equivalent codes.

Chapter 4 investigated how a constellation should be labeled to obtain the largest values of the metrics of Chapter 2. A labeling strategy was introduced: that of maximizing the *edge length profile* of the labeled constellation. The edge length profile was demonstrated to be a better indicator of performance than whether the constellation is Ungerboeck labeled or Gray coded or neither. Some Ungerboeck and Gray labelings maximize the edge length profile, but others do not.

Chapter 5 applied Ungerboeck's mutual information comparison technique to single carrier transmissions with Rayleigh fading and broadcast multicarrier transmissions with frequency selective fading. Two bits of redundancy are sufficient for uncorrelated Rayleigh fading. The number of bits of redundancy useful for broadcast multicarrier transmissions depends on how high the subchannel SNRs can be.

Chapter 6 brought together the ideas of Chapters 2–5 to design three trellis codes based on the three sets of metrics introduced in Chapter 2. These three trellis codes were simulated in a multicarrier modulation system over five channels. The code designed with the new metrics of periodic effective code length and code periodic product distance performed well even on a severely frequency selective channel on which the codes designed according to the other metrics failed completely.

Chapter 7 presented an achievable rate characterization for Tomlinson-Harashima precoding. An exact expression was derived for the zero-forcing THP. Upper and lower bounds were provided for the minimum mean square error THP. The achievable rates for the zero-forcing and minimum mean square error formulations of THP were compared to each other and to channel capacity for some example channels. The importance of correctly choosing the symbol rate was demonstrated. Correct symbol rate selection was shown to be especially important for the zero-forcing THP.

8.2 Future Work

The new code design strategy outlined in Chapters 2–5 was used in Chapter 6 to design one code, a rate-1/4, 16-QAM trellis code with six memory elements. One direction for future work is the design of codes at other rates and complexity levels. Of particular interest are codes that transmit two or three bits per symbol rather than one. Such codes requires larger constellations such as 32-QAM and 64-QAM.

Chapter 6 simulated the newly designed code for multicarrier modulation over channels that were not changing over time. It would also be interesting to simulate performance for correlated Rayleigh fading channels and for frequency hopped transmissions with strong interference on certain hops.

The achievable rate analysis for THP raises the question of whether a similar analysis might be possible for the combined coding and precoding method of Laroia [57]. However, because the Laroia approach combines coding and precoding, simply computing a mutual information is not appropriate for this case.

Appendix A

Formal Definitions

Definition 1 *The (normalized) squared Euclidean distance* $\textsc{ed}^2$ *of the error sequence* $\mathbf{b} \to \hat{\mathbf{b}}$ *is*

$$\textsc{ed}^2 = \frac{\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2}{\mathcal{E}_x} \tag{A.1}$$

Definition 2 *The code Euclidean distance* $\textsc{ced}$ *of a code is the smallest* $\textsc{ed}$ *of an error sequence* $\mathbf{b} \to \hat{\mathbf{b}}$ *associated with that code.*

Definition 3 *The number of Euclidean nearest neighbors* $N_{\textsc{ced}}$ *is the number of sequences* $\hat{\mathbf{x}}$ *that are* $\textsc{ced}$ *away from a transmitted sequence* $\mathbf{x}$.

Definition 4 *The effective length* $\textsc{el}$ *of the error sequence* $\mathbf{b} \to \hat{\mathbf{b}}$ *is the cardinality of the set* $\{i | x_i - \hat{x}_i \neq 0\}$.

Definition 5 *The effective code length* $\textsc{ecl}$ *of a code is the smallest* $\textsc{el}$ *associated with that code.*

Definition 6 *The product distance* $\textsc{pd}$ *of the error sequence* $\mathbf{b} \to \hat{\mathbf{b}}$ *is defined to be*

$$\textsc{pd} = \prod_{i \in A} \frac{\|x_i - \hat{x}_i\|_2^2}{\mathcal{E}_x} \qquad \textit{where } A = \{i \mid x_i \neq \hat{x}_i\}. \tag{A.2}$$

Definition 7 *The code product distance of order* i *of a code,* $\textsc{cpd}_i$*, is the smallest product distance of an error sequence having* $\textsc{el}= i$ *associated with that code.*

Definition 8 *The periodic effective length* PEL *of an error sequence is the number of nonzero elements of the periodic distance vector.*

Definition 9 *The periodic effective code length* PECL *of a code is the smallest* PEL *for an error sequence associated with that code.*

Definition 10 *The periodic product distance* PPD *of an error sequence is the product of the nonzero elements of the periodic distance vector.*

Definition 11 *The code periodic product distance of order i of a code,* CPPD_i*, is the smallest* PPD *of an error sequence with* PEL$= i$.

Definition 12 *A rate k/n convolutional encoder is a time invariant finite state machine with k-bit inputs and n-bit outputs.*

Definition 13 *For a specified labeled constellation, $d^2_{\text{MIN}}(E)$ is the smallest squared Euclidean distance of an edge with label E.*

Definition 14 *For a specified labeled constellation and error sequence $\mathbf{E}(D)$,*

$$d_i^2(\mathbf{E}(D)) \triangleq d^2_{\text{MIN}}(\mathbf{E}_i) \tag{A.3}$$

Definition 15 *Two codes are strictly equivalent if they have the same mapping of input sequences to output sequences.*

Definition 16 *Two codes are range equivalent if they have the same set of possible output sequences.*

Definition 17 *Two trellis codes are distance equivalent if they have the same set of distance sequences $\{d_i^2(\mathbf{E}(D))\}$.*

Definition 18 *Trellis code T_1 is distance superior to trellis code T_2 if the distances sequences of T_1 can be paired (one to one) with those of T_2 such that each term in every distance sequence of T_1 is greater than or equal to the corresponding term in the paired sequence of T_2. There must be at least one strict inequality, otherwise the trellis codes are distance equivalent.*

Definition 19 *The mapping F implements a change of basis from $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ to $\{\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_n\}$ if for any $\mathbf{s} \in \mathcal{S}$ and any subset A of $\{1, 2, \ldots, n\}$:*

$$F(\mathbf{s}) = \sum_{i \in A} \hat{\mathbf{v}}_i \qquad \textit{when} \qquad \mathbf{s} = \sum_{i \in A} \mathbf{v}_i. \tag{A.4}$$

Definition 20 *Two labeled constellations $\mathcal{C}_1$ and $\mathcal{C}_2$ are strictly, range, or distance equivalent if for any trellis code that uses labeled constellation $\mathcal{C}_1$ there is respectively a strictly, range, or distance equivalent trellis code that uses labeled constellation $\mathcal{C}_2$.*

Definition 21 *$\mathcal{C}_1$ is distance superior to $\mathcal{C}_2$ (or $\mathcal{C}_2$ is distance inferior to $\mathcal{C}_1$) if for every trellis code T_2 designed using $\mathcal{C}_2$ there is a distance superior (see Definition 18) trellis code T_1 defined using $\mathcal{C}_1$.*

Definition 22 *The edge length profile $\{p_i\}$ of a labeled constellation is defined to be the list of $d^2_{\text{MIN}}(\mathbf{E})$ values for all nonzero edge labels E listed in increasing order.*

Definition 23 *$\mathcal{C}$ is superior in profile to $\hat{\mathcal{C}}$ if $p_i^{(1)} \geq p_i^{(2)}$ for $i = 1, \ldots, |\mathcal{C}_1|$ with at least one strict inequality.*

Definition 24 *A path's edge label sum is the sum of the edge labels in that path.*

Definition 25 *A labeled constellation is Gray coded if and only if all nearest neighbors differ by exactly one bit.*

Definition 26 *A 2^n-point constellation is called Ungerboeck labeled if for every $m \in \{1, \ldots, n\}$ the minimum distance of two points sharing the same m least significant bits is as large as possible.*

Definition 27 *The coset labels of a constellation are an isometric labeling if and only if for any selected coset label $\mathbf{v}$, exclusive-oring all the coset labels with $\mathbf{v}$ has the same effect as applying a geometric isometry (some combination of translations, rotations, and reflections) to the constellation.*

Definition 28 *A rate-1/4, $\nu = 6$, 16-QAM trellis code with $\text{PECL} = 7$ is dominated if there exists another rate-1/4, $\nu = 6$, 16-QAM trellis code with $\text{PECL} = 7$ that has larger values of both CPPD_7 and CPPD_8. (An undominated code is Pareto optimal.)*

Bibliography

[1] S. Haykin. *Adaptive Filter Theory, Second Edition.* Prentice Hall, 1991.

[2] S. K. Wilson, R. E. Khayata, and J. M. Cioffi. 16-QAM Modulation with Orthogonal Frequency Division Multiplexing in a Rayleigh Fading Environment. In *IEEE 44th Vehicular Technology Conference VTC–94*, pages 1665–1669, June 1994.

[3] A. J. Viterbi. Error Bounds for Convolutional Codes and an Asymptotically Optimal Decoding Algorithm. *IEEE Trans. on Inform. Theory*, 13:260–269, Apr. 1967.

[4] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv. Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate. *IEEE Trans. on Inform. Theory*, 20(2):284–287, March 1974.

[5] G. Ungerboeck. Channel Coding with Multilevel/Phase Signals. *IEEE Trans. on Inform. Theory*, 28(1):55–67, Jan. 1982.

[6] S. G. Wilson and Y. S. Leung. Trellis Coded Phase Modulation on Rayleigh Channels. In *Proceedings of ICC–87*, pages 739–743, June 1987.

[7] D. Divsalar and M. K. Simon. The Design of Trellis Coded MPSK for Fading Channels: Performance Criteria. *IEEE Trans. on Comm.*, 36(9):1004–1012, September 1988.

[8] E. Biglieri, D. Divsalar, P. J. McLane, and M. K. Simon. *Introduction to Trellis-Coded Modulation with Applications.* Macmillan Publishing Company, 1991.

[9] C.-E. W. Sundberg and N. Seshadri. Coded Modulation for Fading Channels: An Overview. *Eur. Trans. on Telecom.*, 4(3):309–323, May–June 1993.

[10] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* John Wiley & Sons, 1991.

[11] J. Du and B. Vucetic. New M-PSK Trellis Codes for Fading Channels. *Electronics Letters*, 26(16):1267–1269, Aug. 1990.

[12] J. Du and B. Vucetic. New 16-QAM Trellis Codes for Fading Channels. *Electronics Letters*, 27(12):1009–1010, June 1991.

[13] J. Du, Y. Kamio, H. Sasaoka, and B. Vucetic. New 32–QAM Trellis Codes for Fading Channels. *Electronics Letters*, 29(20):1745–1746, Sept. 1993.

[14] J. Du and B. Vucetic. Trellis Coded 16–QAM for Fading Channels. *Eur. Trans. on Telecom.*, 4(3):335–341, May–June 1993.

[15] S. K. Wilson and J. M. Cioffi. A Comparison of a Single-Carrier System Using a DFE and a Coded OFDM System in a Broadcast Rayleigh-Fading Channel. In *Proceedings of ISIT–95*, page 335, Sept. 1995.

[16] S. Boyd and L. Vandenberghe. Introduction to Convex Optimization with Engineering Applications. Lecture Notes for EE392x, Electrical Engineering Department, Stanford University, 1995.

[17] June 1995. Private Communications with Erik Ordentlich.

[18] A. Lapidoth. The Performance of Convolutional Codes on the Block Erasure Channel Using Various Finite Interleaving Techniques. *IEEE Trans. on Inform. Theory*, 40(5):1459–1473, Sept. 1994.

[19] R. J. McEleice and W. E. Stark. Channels with Block Interference. *IEEE Trans. on Inform. Theory*, 30:44–53, Jan. 1984.

[20] J. G. Proakis. *Digital Communications, Second Edition.* MacGraw Hill, 1989.

[21] G. D. Forney, Jr. Convolutional Codes I: Algebraic Structure. *IEEE Trans. on Inform. Theory*, 16(6):720–738, Nov. 1970.

[22] G. D. Forney, Jr. Geometrically Uniform Codes. *IEEE Trans. on Inform. Theory*, 37(5):1241–1260, Sept. 1991.

[23] Mitchell D. Trott. *The Algebraic Structure of Trellis Codes.* PhD thesis, Stanford University, Aug. 1992.

[24] L. H. Ozarow and A. D. Wyner. On the Capacity of the Gaussian Channel with a Finite Number of Input Levels. *IEEE Trans. on Inform. Theory*, 36(6):1426–1428, Nov. 1990.

[25] G. D. Forney, Jr. and L. F. Wei. Multidimensional Constellations – Part I: Introduction, Figures of Merit, and Generalized Cross Constellations. *IEEE J. Sel. Areas in Comm.*, 7:877–892, Aug. 1989.

[26] G. D. Forney. Trellis Shaping. *IEEE Trans. on Inform. Theory*, 38(2):281–300, March 1992.

[27] R. Laroia, N. Farvardin, and S. A. Tretter. On Optimal Shaping of Multidimensional Constellations. *IEEE Trans. on Inform. Theory*, 40(4):1044–1056, July 1994.

[28] A. D. Wyner. Bounds on Communication with Polyphase Coding. *Bell Syst. Tech. J.*, pages 523–559, Apr. 1966.

[29] S. Shamai (Shitz) and I. Bar-David. The Capacity of Average and Peak-Power-Limited Quadrature Gaussian Channels. *IEEE Trans. on Inform. Theory*, 41(4):1060–1071, July 1995.

[30] W. C. Lee. Estimate of Channel Capacity in a Rayleigh fading Environment. *IEEE Trans. on Veh. Tech.*, 39(3):187–189, Aug. 1990.

[31] C. G. Gunther. Comments on 'Estimate of Channel Capacity in a Rayleigh fading Environment'. *IEEE Trans. on Veh. Tech.*, 45(2):401–403, May 1996.

[32] A. Goldsmith. *Design and Performance of High-Speed Communication Systems in Time-Varying Radio Channels.* PhD thesis, University of California at Berkeley, 1995.

[33] L. H. Ozarow, Shlomo Shamai, and A. D. Wyner. Information Theoretic Considerations for Cellular Mobile Radio. *IEEE Trans. on Veh. Tech.*, 43(2):369–378, May 1994.

[34] P. K. Ho, J. K. Cavers, and J. L. Varaldi. The effect of Constellation Density on Trellis–Coded Modulation in Fading Channels. *IEEE Trans. on Veh. Tech.*, 42(3):318–325, Aug. 1993.

[35] J. A. C. Bingham. Multicarrier Modulation for Data Tranmsmission: An Idea Whose Time Has Come. *IEEE Comm. Mag.*, pages 5–14, May 1990.

[36] J.M. Cioffi. "A Multicarrier Primer". In *ANSI T1E1.4 Committee Contribution 91-157*, Boca Raton, FL, November 1991.

[37] C. E. Shannon. Communication in the Presence of Noise. *Proc. IRE*, 37:10–21, Jan. 1949.

[38] P. S. Chow, J. M. Cioffi, and J. A. C. Bingham. A practical discrete multitone tranceiver loading algorithm for data transmission over spectrally shaped channels. *IEEE Trans. on Comm.*, 43(3):773–775, March 1995.

[39] Hikmet Sari, Georges Karam, and Isabelle Jeanclaude. Transmission Techniques for Digital Terrestrial TV Broadcasting. *IEEE Comm. Mag.*, 33(2):100–109, Feb. 1995.

[40] B. LeFloch. Channel coding and modulation for DAB. In *First International Symposium on Digital Audio Broadcasting*, pages 99–110, June 1992.

[41] C. E. Shannon. A Mathematical Theory of Communication. *Bell Syst. Tech. J.*, 27:379–423, 623–656, July and Oct. 1948.

[42] W. L. Root and P. P. Varaiya. Capacity of Classes of Gaussian Channels. *SIAM J. of Applied Math*, 16(6):1350–1393, November 1968.

[43] A. J. Viterbi, J. K. Wolf, E. Zehavi, and R. Padovani. A pragmatic Approach to Trellis-Coded Modulation. *IEEE Comm. Mag.*, pages 11–19, July 1989.

[44] M. O. Polley, W. F. Schreiber, and S. J. Wee. Comments on 'Transmission Techniques for Digital Terrestrial TV Broadcasting'. *IEEE Comm. Mag.*, 33(11):22–26, Nov. 1995.

[45] John M. Cioffi. Signals and Detection. Stanford University Lecture Notes, 1994.

[46] L. R. Bahl, C. D. Cullum, W. D. Frazer, and F. Jelinek. An Efficient Algorithm for Computing Free Distance. *IEEE Trans. on Inform. Theory*, 18(3):437–439, May 1972.

[47] K. J. Larsen. Comments on 'An Efficient Algorithm for Computing Free Distance'. *IEEE Trans. on Inform. Theory*, 19(4):577–579, July 1973.

[48] J. Du and B. Vucetic. Algorithm for Computing Minimum Product Distance of TCM codes. *Electronics Letters*, 28(1):2–4, Jan. 1992.

[49] John R. Barry, Edward A. Lee, and David G. Messerschmitt. Capacity Penalty due to Ideal Zero–Forcing Decision–Feedback Equalization. *IEEE Trans. on Info. Theory*, 42(4):1062–1071, July 1996.

[50] John M. Cioffi, Glenn P. Dudevoir, M. Vedat Eyuboğlu, and G. David Forney, Jr. MMSE Decision-Feedback Equalizers and Coding – Parts I and II. *IEEE Trans. on Comm.*, 43(10):2582–2604, October 1995.

[51] Shlomo Shamai (Shitz) and Rajiv Laroia. The Intersymbol Interference Channel: Lower Bounds on Capacity and Precoding Loss. To appear in *IEEE Trans. on Info. Theory.*

[52] D. L. Duttweiler, J. E. Mazo, and D. G. Messerschmitt. An Upper Bound on the Error Probability in Decision–Feedback Equalization. *IEEE Trans. on Info. Theory*, 20(4):490–497, July 1974.

[53] A. K. Aman, R. L. Cupo, and N. A. Zervos. Combined Trellis Coding and DFE through Tomlinson Precoding. *IEEE J. Select. Areas Commun.*, 9(6):876–884, August 1991.

[54] G. J. Pottie and M. V. Eyuboğlu. Combined Coding and Precoding for PAM and QAM HDSL Systems. *IEEE J. Select. Areas Commun.*, 9(6):861–870, August 1991.

[55] M. V. Eyuboğlu and G. D. Forney, Jr. Trellis Precoding: Combining Coding, Precoding and Shaping for Intersymbol Interference Channels. *IEEE Trans. on Info. Theory*, 38(2):301–314, March 1992.

[56] R. Laroia. A Simple and effective precoding scheme for noise whitening on intersymbol interference channels. *IEEE Trans. on Comm.*, 41(10):1460–1463, October 1993.

[57] R. Laroia. Coding for Intersymbol Interference Channels – Combined Coding and Precoding. *IEEE Trans. on Info. Theory*, 42(4):1053–1061, July 1996.

[58] M. Tomlinson. New Automatic Equalizer employing Modulo Arithmetic. *Electronic Letters*, 7:138–139, March 1971.

[59] H. Miyakawa and H. Harashima. Information Transmission Rate in Matched Transmission Systems with Peak Transmitting Power Limitation. In *National Conference Record, Institute of Electronics, Information, and Communication Engineers of Japan*, page 1268, August 1969. Number 1269.

[60] H. Harashima and H. Miyakawa. Matched-Transmission Technique for Channels with Intersymbol Interference. *IEEE Trans. on Comm.*, 20(4):774–780, August 1972.

[61] R. Price. Nonlinearly feedback equalized PAM versus capacity for noisy channel filters. In *Proceedings ICC '72*, pages 22–12 to 22–17, June 1972.

[62] J. E. Mazo and J. Salz. On the Transmitted Power in Generalized Partial Response. *IEEE Trans. on Comm.*, 24(3):348–352, March 1976.

[63] John M. Cioffi and Glenn P. Dudevoir. Data Transmission with Mean–Square Partial Response. In *Proceedings of GLOBECOM '89*, Dallas, TX, November 1989.

[64] G. D. Forney, Jr. Trellis Shaping. *IEEE Trans. on Info. Theory*, 38(2):281–300, March 1992.

[65] J. Salz. Optimum Mean Square Decision Feedback Equalization. *Bell Syst. Tech. J.*, 52:1341–1373, October 1973.

[66] S. Shamai and A. Dembo. Bounds on the Symmetric Binary Cutoff Rate for Dispersive Gaussian Channels. *IEEE Trans. on Comm.*, 42(1):39–53, January 1994.

[67] A. B. Sesay and M. R. Gibbard. Asymmetric Signal Processing for Indoor Wireless LANs. In *Sixth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 6–10, Toronto, Canada, September 1995.