Part I

Modulation and Canonical Reception

Discrete Data Transmission Basics

I Modulation and Canonical Reception 1

1 Discrete Data-Transmission Basics 4

1.1 Discrete Data-Message Encoding and Decoding . . . 7
1.1.1 The Vector-Symbol Channel Model . . . 7
1.1.1.1 Vector-Symbol Communication Examples . . . 8
1.1.1.2 Energy . . . 13
1.1.2 Optimum Data Detection . . . 14
1.1.2.1 The MAP Detector . . . 14
1.1.2.2 The Maximum Likelihood (ML) Detector . . . 15
1.1.3 Decision Regions . . . 16
1.1.4 Optimum Average Error Probability Calculation . . . 17
1.1.5 Irrelevant Components of the Channel Output . . . 19
1.1.5.1 Reversibility . . . 21
1.1.6 Optimum Bit-Error Probability and Log Likelihood Decoding . . . 22
1.1.7 P_e Calcuation and The Bhattacharrya Bound . . . 23
1.2 Data Modulation and Demodulation for Continuous-Time Channels . . . 26
1.2.1 Signal Waveform Representation by Vectors . . . 27
1.2.2 Modulator Examples . . . 30
1.2.3 Vector-Space Interpretation of the Modulated Waveforms . . . 34
1.2.4 Demodulation . . . 36
1.2.5 MIMO Channel Basics . . . 38
1.3 The Additive White Gaussian Noise (AWGN) Channel . . . 41
1.3.1 Continuous-Time AWGN Conversion to a Vector AWGN Channel . . . 42
1.3.1.1 Optimum Detection with the AWGN Channel . . . 44
1.3.1.2 General Receiver Implementation . . . 45
1.3.1.3 Signal-to-Noise Ratio (SNR) Maximization with a Matched Filter . . . 47
1.3.2 Error Probability for the AWGN Channel . . . 48
1.3.2.1 AWGN Invariance to Rotation and Translation . . . 49
1.3.2.2 Union Bounding . . . 52
1.3.2.3 The Nearest Neighbor Union Bound . . . 57
1.3.2.4 Alternative Performance Measures . . . 58
1.3.2.5 Block Error-Rate Measures . . . 61
1.3.3 General Classes of Constellations and Modulation . . . 62
1.3.3.1 Fair Comparisons . . . 65
1.3.3.2 Cubic Constellations . . . 65
1.3.3.3 Binary Antipodal Constellations . . . 66
1.3.3.4 On-Off Keying (OOK) . . . 67
1.3.3.5 Vertices of a Hypercube (Block Binary) . . . 68
1.3.3.6 Orthogonal Constellations . . . 69
1.3.3.7 Circular Constellations - M-ary Phase Shift Keying . . . 75
1.3.4 Rectangular (and Hexagonal) Signal Constellations . . . 75
1.3.4.1 Pulse Amplitude Modulation (PAM) . . . 76
1.3.4.2 Quadrature Amplitude Modulation (QAM) . . . 78

1.3.4.3 Constellations based on lattices . . . 88
1.3.4.4 Hexagonal Signal Constellations . . . 92
1.3.5 Baseband Modulation . . . 92
1.3.5.1 Passband Representations and Terminology . . . 93
1.3.5.2 Baseband-Equivalent AWGN Channel . . . 101
1.3.5.3 Conversion to a baseband equivalent channel . . . 104
1.3.6 Passband Analysis for QAM alternatives . . . 111
1.3.6.1 Passband VSB Analysis . . . 111
1.3.6.2 Passband CAP Analysis . . . 112
1.3.6.3 OQAM or "Staggered QAM" . . . 113
1.3.6.4 The difference between complex and baseband equivalent channels . . . 113
1.3.7 Additive Self-Correlated Noise . . . 114
1.3.7.1 The Filtered (One-Shot) AWGN Channel . . . 115
1.3.7.2 Optimum Detection with Self-Correlated Noise . . . 117
1.3.7.3 Performance of Suboptimal Detection with Self-Correlated Noise . . . 121
1.3.8 The Matrix AWGN . . . 122
1.4 Discrete Channels . . . 124
1.4.1 Discrete Memoryless Channels . . . 124
1.4.2 Symmetric DMCs . . . 126
1.4.2.1 The Binary Symmetric Channel (BSC) . . . 127
1.4.2.2 The Binary Erasure Channel (BEC) . . . 128
1.4.2.3 The symmetric q'ary-DMC . . . 128
1.5 Linear and Non-Linear Single-Message Channel Models . . . 130
1.5.1 The Nonlinear Channel State and Consequent Decoders . . . 131
1.5.1.1 Multivariate Taylor Series and the Volterra Series . . . 134
1.5.1.2 Continuous Models . . . 135
1.5.2 The relationship of the nonlinear model to neural-network modeling . . . 135
1.6 Statistically parameterized Variable Channels . . . 139
1.6.1 Memoryless Channels and Dimensionality . . . 142
1.6.2 Statistically Variable AWGNs . . . 143
1.6.2.1 Some channel-gain distributions/densities . . . 143
1.6.2.2 Outage Probability . . . 147
1.6.2.3 The Wireless Designer's Paradox . . . 147
1.6.3 Channels with Parameter Memory . . . 148
1.6.3.1 Coherence Time . . . 148
1.6.3.2 Coherence Bandwidth . . . 150
1.6.3.3 Spatial Coherence . . . 151
1.6.3.4 Discretization of probability densities . . . 152
1.6.3.5 The Ergodic Markov model . . . 153
1.7 Disguised Channels . . . 155
1.7.1 Facial Recognition . . . 155
1.7.2 Distance Measurement . . . 155
1.7.3 Speaker Recognition . . . 156
1.7.4 Deep-Packet Inspection / Eavesdropping . . . 156
1.7.5 Search and the PageRank Algorithm as Ergodic Markov Channels . . . 156
Chapter 1 Exercises . . . 158

Bibliography **193**

Index **194**

Chapter 1

Discrete Data-Transmission Basics

This chapter introduces this textbook's foundation for digital data transmission that links basic mathematical concepts in probability theory, discrete and continuous fields, and basic optimum-decision concepts. The consequent foundational understanding readies the designer to comprehend many simple wireline and wireless transmission systems, while also enabling the designer to progress to increasingly sophisticated methods in later chapters. Simple vector-symbol representation of transmitters and receivers with definitions of optima and corresponding performance analysis basics appear here. These basics allow calculation of data rates achieved, corresponding to defined quality measures like error probability, and to a common framework for system comparisons using nearest-neighbor and minimum-distance concepts. This chapter concludes with more than 50 exercises that help the reader examine these foundational basics and illustrate the topics' utility in a number of both practical designs and/or curious challenges.

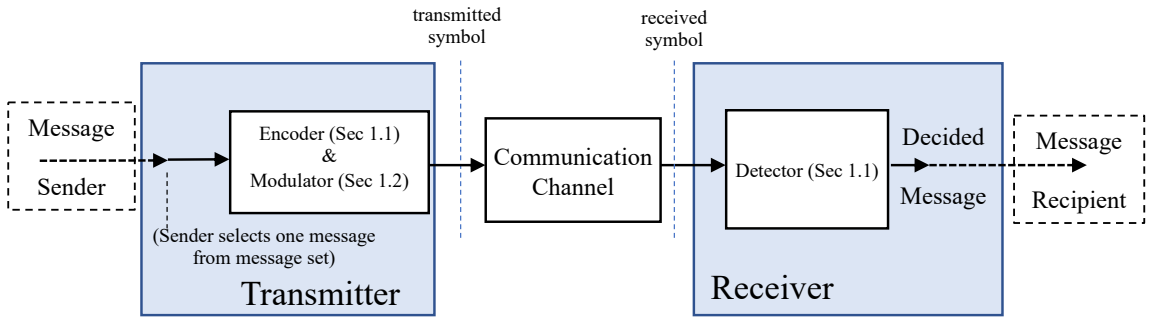


Figure 1.1: Discrete data transmission.

Figure 1.1 illustrates **discrete data transmission**, which is the transmission of one message from a finite set of messages through a communication channel. A **message sender** at the **transmitter** communicates with a **message recipient** at the **receiver**. The sender selects one message from the finite set, and the transmitter sends a corresponding **transmit symbol** that uniquely represents this message through the communication channel. The receiver decides the message sent by observing the **received symbol**, which may not be the same as the transmitted symbol, and passes that decided message to the recipient

This chapter concentrates on optimal detection for a single message transmission through the chan-

nel[1]. Such single-message analysis is often called **one-shot** analysis. When a single message is sent, Section 1.1's **encoder** maps that message to the transmit symbol.

Similarly with a single message, Section 1.1's **detector** is the receiver device that makes the receiver's decision. Section 1.1 also develops optimum detection that minimizes the probability of an erroneous receiver decision on which message was transmitted. A decoder maps the decision into the corresponding message. Chapter 2 addresses encoders that expand multiple successive coordinated symbol transmissions, while Chapter 3 addresses channel-induced inter-symbol interference.

A single bit, a digital sequence of bits, bytes (8-bit groups), or other groupings of a finite number of bits represent the message to be sent. More bits means a larger finite set of possible choices for the single message sent. The different bit combinations each uniquely represent individually the distinct discrete messages that pass to the channel. The bits themselves are usually not compatible with direct message transmission through most communication channels. Thus the encoder converts the messages' bits into appropriate **symbols** that the transmitter can send through the channel. The symbols depend on the permissible types of channel inputs, which are typically modeled as within a field or vector space. Section 1.1 models the channel with a conditional probability distribution on received symbols for each given transmitted symbol value.

The channel distorts the transmitted symbols both deterministically and randomly to produce the received symbols. Because the received symbol will usually not exactly equal the transmitted symbol, Section 1.1 develops the criteria for the detector that makes a optimum decision based on the observed received symbol. Optimal decisions minimize the probability of message/symbol decision error. The transmitted symbols have probabilities equal to the probabilities of the messages that they represent. The optimal decision will depend only on the probabilistic model for the channel and the channel-input-message's probability distribution. The general optimal decision specializes in many later sections' important practical cases of interest. This probabilistic approach allows conceptual extension beyond data "transmission" to all types of recognition, detection, and matching problems that often go under more exotic modern names like "machine learning," "search engine," and/or "facial recognition," as in Section 1.7 on disguised channels. A good part of life and education tries to learn, infer, or understand/receive some communication or information (as well as to transmit or store it so it is more easily understood by another), and this chapter provides basics that apply to all these basic communication areas.

Section 1.2 then expands the transmitter model to include **modulation**. A modulator converts the encoder-output symbols into continuous-time signals for transmission through a continuous-time channel. This chapter develops a theory of modulation and corresponding demodulation that links to Section reffund2.1's discrete vector representation for any set of continuous-time signals. This "vector-channel" approach was pioneered for educational purposes by Jack Wozencraft and Irwin Jacobs in Chapter 4 of their classic text [1] (Chapter 4). In fact, the first two sections of this chapter closely parallel their development (with some updating and rearrangement), before diverging in Sections 1.3 – 1.7 and in the remainder of this text. Section 1.2's last subsection introduces the **multiple-input multiple-output (MIMO)** modulation that may generate multiple continuous-time signals for coordinated transmission through separate antennas' or wires' channels.

Section 1.3 investigates continuous-time channels, particularly the most common case of the additive Gaussian-noise channel, which maps easily into Section 1.1's discrete-time vector model without loss of generality. Section 1.3 also develops simpler widely applicable methods to calculate and estimate average error probability, P_e, for a vector channel with **Additive White Gaussian Noise (AWGN)**, particularly introducing and using nearest-neighbor and minimum-distance concepts. Section 1.3 also discusses several popular modulation formats and determine bounds for their error probability with the AWGN, including signals derived from rectangular lattices, a popular and practical signal-transmission method. Section 1.3 also addresses the extension to carrier-modulated signals. Section 1.4 progresses to discrete channels where the inputs and outputs belong to discrete finite sets and the concept of noise necessarily becomes discrete and part of the channel's general conditional-probability model.

Section 1.5 introduces linear and nonlinear one-shot /matrix channels. Later chapters revisit these models for increasingly sophisticated transmission. Section 1.6 then models the additive-noise channel's

[1]Dependencies between successive message transmissions can be important also, but the study of such inter-message dependency is deferred to later chapters.

gain as random, which allows an average analysis of time-varying channels that is useful for wireless data transmission.

1.1 Discrete Data-Message Encoding and Decoding

This section, particularly Subsection 1.1.1, mathematically and statistically models the basic transmitter, channel, and receiver through symbol vectors. Most results correspond to the transmitted symbol vector $\boldsymbol{x}$ and corresponding received symbol vector $\boldsymbol{y}$ being N-dimensional real-valued vectors, $\boldsymbol{x} \in \mathbb{R}^N$ and $\boldsymbol{y} \in \mathbb{R}^N$ with $N \in \mathbb{Z}$, while Sections 1.3 and 1.4 will expand to complex vectors and other possible finite fields for the symbol values. These N vector dimensions all are temporal, which means they occupy time (and/or frequency within a given time interval). Some channels produce temporal output vectors $\boldsymbol{y}$ that have a different dimensionality than the input $N + \nu$ where $\nu \in \mathbb{Z}$.

Section 1.2 defines further **MIMO (multiple-input-multiple-output)** vector channels that fit precisely also into this section's framework, but extend the vector dimensionality index to correspond to simultaneous transmission of symbols over L_x multiple-input, parallel, N-dimensional channels to L_y multiple received symbols that may each also be N-dimensional. The transmitted symbols may be viewed in aggregate then as a single $L_x \cdot N$-dimensional transmitted symbol (chosen from a larger set of possible symbols over a larger dimensionality), and similarly the received symbols can also be viewed as single received symbol, also of larger dimensionality, $L_y \cdot N$. With MIMO, $\boldsymbol{x}$'s N individual elements each themselves become an $L_x \times 1$ vector (and similarly each element of $\boldsymbol{y}$ is $L_y \times 1$). In this text, $L_x = L_y = 1$ is the initial default, and MIMO expansions to larger L_x or L_y appear explicitly throughout.

Of importance also is Subsection 1.1.2's study of the optimal **detector**. The optimal detector decides which of Figure 1.2's discrete symbol N-dimensional vectors $\boldsymbol{x}_i$ $i = 0, ..., M-1$ was most likely transmitted based on the single observation of Figure 1.2's received symbol vector $\boldsymbol{y}$. Subsection 1.1.3 introduces a well known pictorial description, a decision region, for the optimum detector. These concepts then allow Subsection 1.1.4's symbol-error-probability definition and calculation and Subsection **??**'s important, useful, and well-known , but amusingly named irrelevancy concepts. The remaining subsections (1.1.6 and 1.1.7) further enhance and describe data-transmission performance measures.

1.1.1 The Vector-Symbol Channel Model

The **vector-symbol channel model** appears in Figure 1.2. A message from the set of M possible messages m_i $i = 0, ..., M-1$ is sent every T seconds, where T is the **symbol period** for the discrete data transmission system. Thus, messages are sent at the **symbol rate** of $1/T$ messages per second. The number of messages that can be sent is often measured in bits so that $b \triangleq \log_2(M)$ bits are sent every symbol period. Thus, the **data rate** is $R = b/T$ bits per second. The message is often considered to be a real integer equal to the index i, in which case the message is abbreviated as m with possible values $m \in \{0, ...M-1\}$. This chapter's one-shot analysis will focus attention on a single symbol period over time $t \in [0, T]$.

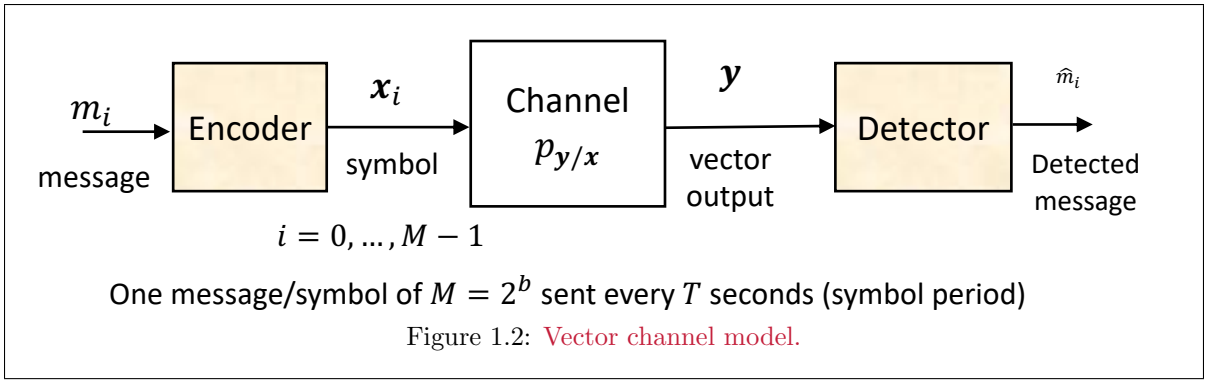


Figure 1.2: Vector channel model.

The encoder formats the messages for transmission over the vector-symbol channel by uniquely mapping each message m_i into its specific corresponding symbol vector $\boldsymbol{x}_i$, typically an N-dimensional real data symbol chosen from a **signal constellation** C that is the set of $|C| \geq M$ distinct points

Name	Definition
$M = 2^b \in \mathbb{Z}^+$	**number of messages**, corresponding to $b \geq 0$ information bits
$T \in \mathbb{R}$	**symbol period**; $1/T$ is the **symbol rate**
$R = \frac{b}{T} \in \mathbb{R}$	**data rate**.
$\boldsymbol{x}$	transmitted **symbol** value (typically a real or complex vector)
$C \subset \mathbb{R}^N$	**constellation** that consists of all possible symbol values $\{\boldsymbol{x}_i \; i = 0, ...\|C\| - 1\}$ with $\|C\| \geq M$ and in this chapter, always equal.
$p_m(i) \in \mathbb{R}$	**message's probability distribution**, $i = 0, ..., M - 1$
$p_{\boldsymbol{x}}(i) \in \mathbb{R}$	**symbol value's probability distribution** $= p_m(i)$
$N \in Z^+$	**number** of real temporal **dimensions** per transmit symbol the receive symbol has $N + \nu$ dimensions with default $\nu = 0$ and $\nu \in \mathbb{Z}$
$L_x \in \mathbb{Z}^+$	number of **transmit spatial dimensions**
$L_y \in \mathbb{Z}^+$	number of **receive spatial dimensions**

Table 1.1: Table of transmitted-symbol quantities' definitions.

$C = \{\boldsymbol{x}_i \; i = 0, ..., |C| - 1\}$. In this chapter $|C| = M$, but it is possible in Chapter 2's coded systems for the signal constellation to have more possible points than there are messages. The detector decides which message $\hat{m}_i$ was sent from among the set of M possible messages $\{m_i \quad i = 0, \ldots, M - 1\}$ that could have been transmitted over the vector channel. In the vector channel, $\boldsymbol{x}$ is a random vector, with discrete probability distribution $p_{\boldsymbol{x}}(i), \; i = 0, ..., |C| - 1$.

Definition 1.1.1 (Symbol Transmission Definition Summary) *Table 1.1 summarizes the symbol-related definitions:*

1.1.1.1 Vector-Symbol Communication Examples

This subsection tries both to provide some examples and to illustrate how many problems today viewed in computer science and other fields are often fundamentally discrete data transmission examples. Such view then motivates discrete data transmission as a fundamental core discipline through many fields of signal processing, imaging, recognition, movement or object detection, and even search engines as special cases.

EXAMPLE 1.1.1 [Light-Pulse Bit Transmission]

Figure 1.3 illustrates basic fiber-optic transmission conceptually. Simple fiber-optic transmission systems transmit bits as light pulses through a fiber[2]. Thus, a light pulse's presence indicates message "1," and its absence indicates message "0." Figure 1.3's encoder "gate" passes light to the fiber waveguide when the message input is $m = 1$ and blocks light when $m = 0$. Figure 1.3's single light pulse corresponds to a single "1" between a previous and a succeeding "0." For the simple binary case, the index of i essentially becomes m to simplify notation, and the inputs will be equally likely with probability $p_m = 1/2$. Figure 1.3 receiver sensor accepts the received light pulse.

[2]Typical laser fiber-transmission wavelengths are 1.490 μm (201 THz) and 1.310 μm (229 THz), perhaps one for each transmission direction. Such a pulse-energy/light-detection system has the name **non-coherent**. **Coherent** optical receivers phase-lock (see Chapter 6) to the laser frequency while non-coherent essentially ignore the carrier frequency and sense only energy without regard to phase. This example describes only the simpler non-coherent.

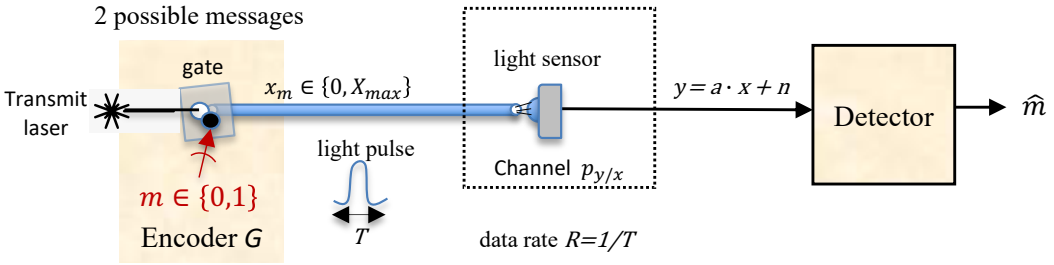


Figure 1.3: [Light-pulse bit-transmission system.

The laser physics and semiconductor technologies supporting such transmission evolve with time so that very narrow pulses are possible, allowing narrow values for the symbol period T. The x_1 transmit-symbol value will be some optical intensity level $X_{max} > 0$ launched into the fiber, while x_0 will be a lower (hopefully zero, so $x_0 = 0$) intensity level. A corresponding optical device senses the received-symbol value y. This y value may not fully represent the optical energy because that energy attenuates in traversing the fiber. The receive sensor also has imperfections that introduce additive noise n. These imperfections' energy levels increase relative to the optical energy as T narrows. Thus

$$y = a \cdot x + n \;\; ; \;\; 0 < a < 1 \;\; .$$

While more narrow T means higher data rate $R = 1/T$, it also increases the risk that the relatively larger noise component will confuse the light sensor. A longer fiber channel will also increase this risk because that increases the light pulse's attenuation. The detector provides the output $\hat{m}$ with as much reliability as possible. That is specifically an output of 1 when optical energy is sensed and 0 otherwise in each symbol period.

Figure 1.3 also represents this entire system as an overlay to Figurer 1.2's generic functions. The designer must obtain the channel's probability distribution $p_{y/x}$ by measurement or analysis prior to analysis of the system's performance. In this additive noise case, $p_{y/x} = p_n$, so the noise's distribution determines the channel. Since the fiber has attenuation a, the channel is $p_{y/x} = p_n\left(\frac{y}{a} - x\right)$ with the value $x \in \{0, X_{max}\}$.

Another example is wireless transmission:

EXAMPLE 1.1.2 [Wireless Transmission] *Wireless transmission occupies some band of frequencies, often centered on a carrier frequency f_c. Figure 1.4 illustrates the basics for transmission of $b = 6$ bits/symbol with one of 64 possible discrete amplitude and phase settings a sinusoid with the carrier frequency $x_i(t) = cost(2\pi f_c \cdot t + \theta_i)$. The carrier frequency example is 5.8 GHz, so probably supporting Wi-Fi transmission. Section 1.3 later illustrates the conversion of such carrier-modulated waveforms to vectors of input $\boldsymbol{x}$ that are independent of the exact frequency. Symbols are sent every T seconds. The multipath channel will have a response $h(t)$ that includes any transmit and receive filtering. When T is short, successive symbols will overlap causing Chapter 3's intersymbol interference. Nonetheless, there exists a probabilistic channel model $p_{\boldsymbol{y}/\boldsymbol{x}}$ for all such wireless data-transmission systems, allowing common analysis methodology.*

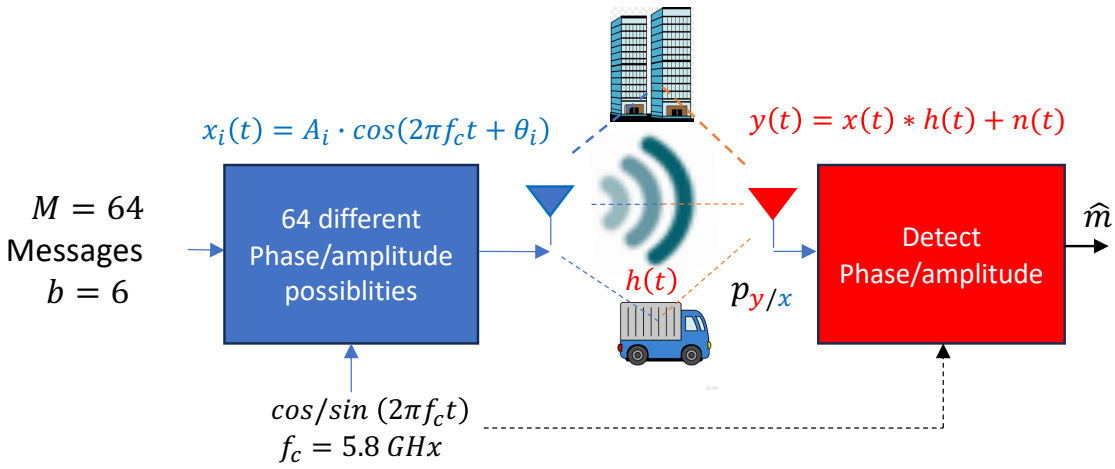


Figure 1.4: Multipath Wireless Channel Example.

While Example 1.1.1 almost directly aligns with intuition, many common computing and engineering problems are also basic communication problems as the next example illustrates.

EXAMPLE 1.1.3 [Facial recognition is data transmission]

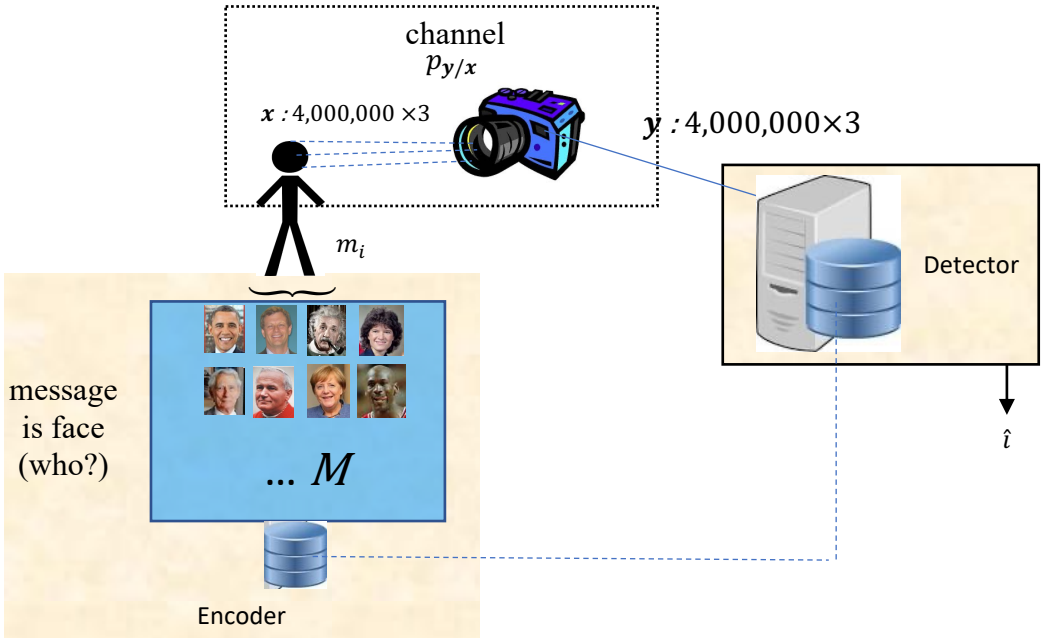


Figure 1.5: Facial recognition is data transmission.

Figure 1.5 illustrates facial recognition as discrete data transmission. The message set contains all the possible faces that a digital camera's photograph might capture. This set may

be quite large. Eight faces appear, but the set could be larger or smaller; perhaps as large as M being the number of earthly humans or as small as simply $M = 2$ images that might correspond only to if an object is present or absent from a constant background. Photograph dimensionality is usually quite large. $N \approx 4 \times 10^6 \times 3$ is shown, but this number could vary with the camera's resolution. Typically the dimensions correspond each to "pixels" that a digital camera captures in a two-dimensional grid, for instance $N = 2280 \times 1640$. Each pixel itself may have a few dimensional components (like red, green, and blue intensities) that are real numbers (Figure 1.5 shows then 3). Together these pixels form a large symbol vector for each possible face.

The encoder is tacit, effectively implied by light falling on one of the possible faces causing it to be encoded into the reflected color intensities. The digital camera is part of the channel that coverts that light into the pixel's intensity numbers. Depending on the aperture and distance, the channel output photographed face may be turned or scaled relative to the stored facial image. Thus, the possible message set may need to expand by various scalings and turnings so that there are sufficiently many message-set images to cover the population of recognizable people. The facial-recognition server attempts to use knowledge of the channel $p_{\boldsymbol{y}/\boldsymbol{x}}$ that provides for every possible camera-photo-output received-symbol vector $\boldsymbol{y}$ its probability given each and every possible transmitted-symbol vector $\boldsymbol{x}_i$ (the stored photos). Of course, as in all data-transmission systems, the input message set is known to the detector/receiver as well as the transmitter. Association of a set of a sufficiently large photo message-set with each and every individual is actually a dual of data transmission, often called data compression. Design of reasonable message sets in general, and would certainly be needed for practical facial recognition beyond this example, is more generally known as 'coding." Choice of good codes can simplify the detection problem enormously while allowing reliably message transmission.

A third example translates position to message.

EXAMPLE 1.1.4 [Distance and movement estimation as data transmission]

Simplest radar[3] and lidar[4] systems measure the delay time for a pulse or "ping" reflection to return, then divide the delay time by twice the speed of light to estimate the reflecting object's distance, as in Figure 1.6. Successive time measurements may be used to estimate speed of movement also as $\frac{d_2-d_1}{t_2-t_2}$ where t_1 and t_2 are the successive measurement times for d_1 and d_2, respectively. Figure 1.6 shows only one distance measurement. Radar systems' lower-frequencies can traverse longer distances with less attenuation than can lidar systems[5]. However, lidar's smaller wavelengths allow smaller objects to be identified and can[6] provide a more accurate "3D" mapping of the object.

Figure 1.6 simplifies both these systems for distance estimation of a driver-less car approaching a stop sign. The messages corresponding to a quantization of distance between the car and the stop sign, with each message index i corresponding to a distance. Figure 1.6 shows $M = 10$ distances from 10 meters to 100 meters The receiver, which is in the same location (on the car) as the transmitter, detects the reflected pulse's delay τ and converts that into a distance $\hat{m}_i$, or effectively deciding one of the distances/messages.

Figure 1.6 shows only one object, but of course there could be multiple reflectors, in which case the previous example's image recognition could be a first problem solved by a camera system collaborating with the distance estimation. Both together, or either by itself, are basic data transmission problems.

[3]RAdio Detection And Ranging (RADAR) uses sinusoidal energy bursts at "radio" frequencies between 3 MHz (100m wavelength) to 300 GHz (1mm wavelength).

[4]LIght Detection And Ranging (LIDAR), which is similar to radar but at higher frequencies "light" frequencies from 30 THz (10μm wavelenght) to 1200 THz (250nm wavelength)

[5]Generally speaking, the lower the frequency, the lower the attenuation so 3 MHz travels much further than 30 THz.

[6]When the lidar source moves for successive measurements or has multiple antennas participating, the known transmit positions and corresponding reflections can map a small stationary object's shape or equivalent be used to create a multi-dimensional image reflection.

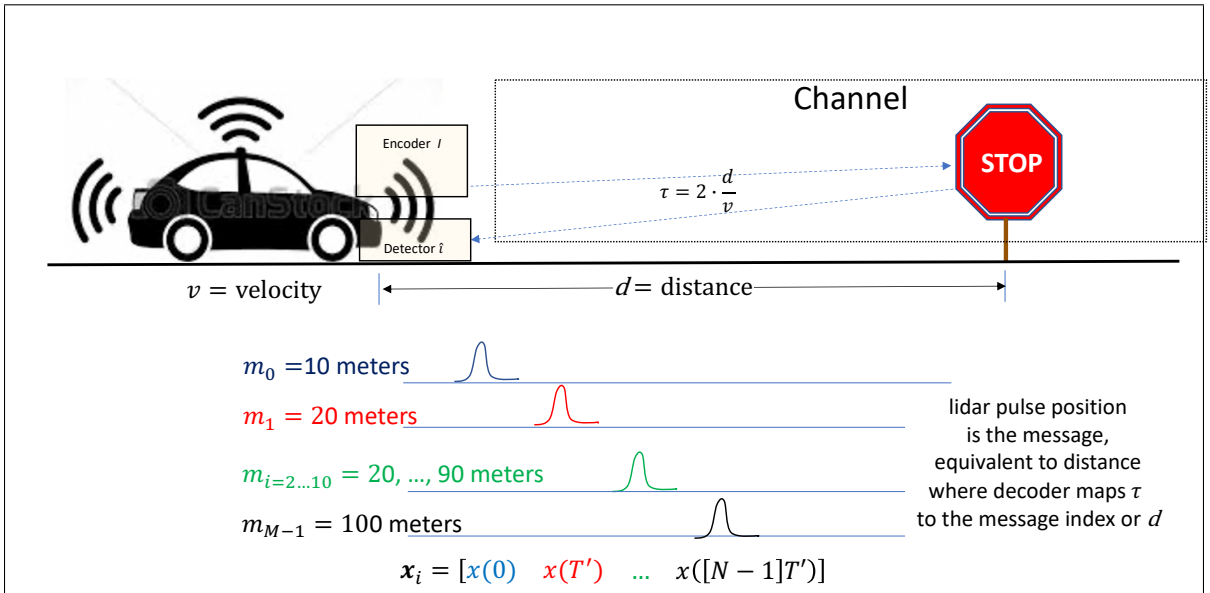


Figure 1.6: Self-driving lidar uses data transmission.

The next example further expands the perspective on data-transmission applicability:

EXAMPLE 1.1.5 [El Goog search engine is data transmission/detection]

Yes, a search engine is a receiver/detector. The message set $\{m_0, ..., m_M$ may have M as large as all the world's URL's[7]. The searcher types the channel output "Stanford" that corresponds to their own brain and the keyboard attempting to approximate what they seek. What they seek is the web-page or information m_i that is URL www.stanford.edu . The search engine's detector models the channel-describing conditional probabilities for possible typed queries (the channel outputs, a vector of values corresponding to perhaps ASCII characters), $\boldsymbol{y}$, given what the searcher really might have wanted, $\boldsymbol{x}$. Search engines estimate this set of conditional probabilities by observing users' page-hit frequencies in going from one web page to another. The matrix formed is a probability matrix, for whose largest eigenvalue's corresponding eigenvector is the channel conditional-probability distribution that the detector subsequently uses. The calculation of the eigenvector is updated often, typically with each query and results. This distribution's highest probability for the specific $\boldsymbol{y}$ value is the displayed top-line result (ignoring paid advertisements). This search is the "PageRank" algorithm (originally conceived and patented by Stanford graduate-student communications engineers L. Page and S. Brin). Other next-best, and next-to-next best, possible decodings are also often displayed in order. This multi-decision display is a form of "soft decoding or information" that appears in later chapters and in Subsection 1.1.6's discussion of log-likelihood ratios.

[7]This may not be a "large" number for transmission engineers. Typical wireless codes' decoders actually search more possibilities, but the code/symbols use structure, see Chapter 2, that allows search-complexity reduction.

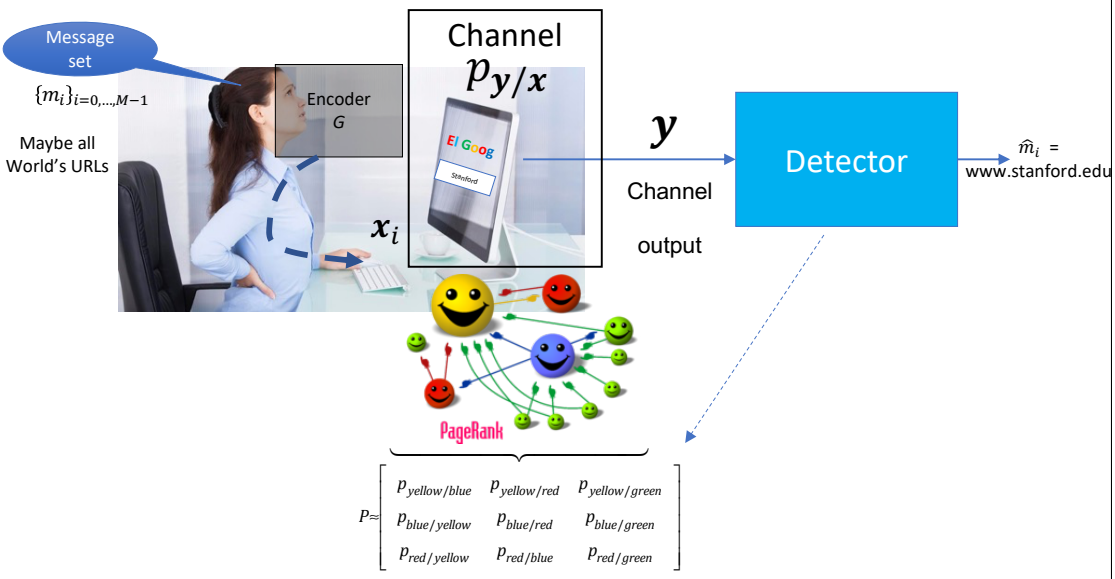


Figure 1.7: Search engine is data transmission.

1.1.1.2 Energy

An important concept for a real-valued signal constellation is its average energy[8]:

Definition 1.1.2 [Average Energy] *A signal constellation's* **average energy** *is*

$$\mathcal{E}_{\boldsymbol{x}} \triangleq \mathbb{E}\left[||\boldsymbol{x}||^2\right] = \sum_{i=0}^{|C|-1} ||\boldsymbol{x_i}||^2 \cdot p_{\boldsymbol{x}}(i) \quad , \tag{1.1}$$

where $||\boldsymbol{x_i}||^2$ *is the squared-length of the vector* $\boldsymbol{x_i}$, $||\boldsymbol{x_i}||^2 \triangleq \sum_{n=1}^{N} x_{in}^2$. *"E" denotes expected or mean value. The average energy is also closely related to the concept of* **average power**, *which is*

$$P_{\boldsymbol{x}} \triangleq \frac{\mathcal{E}_{\boldsymbol{x}}}{T} \quad , \tag{1.2}$$

corresponding to the amount of energy per symbol period.

In the same symbol period, the transmitted symbol vector $\boldsymbol{x}$ corresponds to a received symbol vector $\boldsymbol{y}$, which is also an N-dimensional real vector.[9] The received symbol's conditional probability (given the input symbol), $p_{\boldsymbol{y}/\boldsymbol{x}}$, completely models the discrete data channel. The detector then translates the received symbol vector $\boldsymbol{y}$ into a decision on which symbol $\hat{\boldsymbol{x}} \in \{\boldsymbol{x}_0, ..., \boldsymbol{x}_{M-1}\}$ was transmitted. A **decoder** (which is part of the decision device) reverses the encoder process and converts the detector output $\hat{\boldsymbol{x}}$ into the message corresponding to the decision $\hat{m}$.

[8] Electrical engineers may note power (and therefore energy) necessarily are also a function of line/antenna impedance. That impedance's square root is presumed absorbed into the symbol's value in cases where the symbol is viewed as a voltage level. This scaling would also be implied for received symbols.

[9] Section 1.2 will address the transformation of $y(t) \rightarrow \boldsymbol{y}$ for continuous-time channels.

The particular message symbol vector corresponding to m_i is $\boldsymbol{x_i}$ and has n^{th} component x_{in}. The n^{th} component of $\boldsymbol{y}$ is denoted y_n , $n = 1, ..., N$. The random received symbol vector $\boldsymbol{y}$ may have a continuous probability density or a discrete probability distribution $p_{\boldsymbol{y}}(\boldsymbol{v})$, where $\boldsymbol{v}$ is a dummy variable spanning all the possible N-dimensional vector values for $\boldsymbol{y}$. The received symbol's distribution is a function of the transmit-symbol and channel-transition-probability distributions:

$$p_{\boldsymbol{y}}(\boldsymbol{v}) = \sum_{i=0}^{|C|-1} p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, i) \cdot p_{\boldsymbol{x}}(i) \quad . \tag{1.3}$$

The received symbol's average energy is

$$\mathcal{E}_{\boldsymbol{y}} = \sum_{\boldsymbol{v}} \|\boldsymbol{v}\|^2 \cdot p_{\boldsymbol{y}}(\boldsymbol{v}) \quad . \tag{1.4}$$

An integral replaces[10] the sum in (1.3) and (1.4) for the case of a continuous density function $p_{\boldsymbol{y}}(\boldsymbol{v})$. As an example, consider the simple additive noise channel $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n}$. In this case $p_{\boldsymbol{y}/\boldsymbol{x}} = p_{\boldsymbol{n}}(\boldsymbol{y} - \boldsymbol{x})$, where $p_{\boldsymbol{n}}(\bullet)$ is the noise probability distribution, when $\boldsymbol{n}$ is independent of the input $\boldsymbol{x}$.

1.1.2 Optimum Data Detection

For Figure 1.2's vector channel, the error probability is defined as the probability that the decoded message $\hat{m}$ is not equal to the transmitted message:

Definition 1.1.3 [Error Probability] *The* **Error Probability** *is*

$$P_e \triangleq P\{\hat{m} \neq m\} \quad . \tag{1.5}$$

The corresponding **probability of being correct** *is therefore*

$$P_c = 1 - P_e = 1 - P\{\hat{m} \neq m\} = P\{\hat{m} = m\} \quad . \tag{1.6}$$

The optimum data detector chooses $\hat{m}$ to minimize P_e, or equivalently, to maximize P_c. The probability of being correct is a function of the particular transmitted message, m_i.

1.1.2.1 The MAP Detector

The probability of a correct decision $\hat{m} = m_i$, given the specific channel output vector $\boldsymbol{y} = \boldsymbol{v}$, is

$$P_c(\hat{m} = m_i,\ \boldsymbol{y} = \boldsymbol{v}) = p_{m/\boldsymbol{y}}(m_i, \boldsymbol{v}) \cdot p_{\boldsymbol{y}}(\boldsymbol{v}) = p_{\boldsymbol{x}/\boldsymbol{y}}(\boldsymbol{x}_i, \boldsymbol{v}) \cdot p_{\boldsymbol{y}}(\boldsymbol{v}) = {}^{11}\ p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{x}_i, \boldsymbol{v}) \cdot p_{\boldsymbol{x}_i} \quad . \tag{1.7}$$

Thus the optimum decision device observes the particular received symbol $\boldsymbol{y} = \boldsymbol{v}$ and, as a function of that symbol, chooses an $\hat{m} = m_i$, $i = 0, ..., M-1$ that maximizes the probability of a correct decision in (1.7). This quantity $p_{m/\boldsymbol{y}}$ is referred to as the *à posteriori* probability for the vector channel. Summing (discrete $\boldsymbol{v}$ components, or equivalently integrating when continuous $\boldsymbol{v}$) over all $\boldsymbol{v}$ values, $P_{m/\boldsymbol{y}}(i, \boldsymbol{v}) \cdot p_{\boldsymbol{y}}(\boldsymbol{v})$ yields P_c, which is maximized overall then too (since $p_{\boldsymbol{y}}(\boldsymbol{v}) \geq 0$). P_e is then minimized. Thus, the optimum detector for Figure 1.2's vector channel is the *Maximum à Posteriori* detector:

> **Theorem 1.1.1** [MAP Detector] *The* **Maximum à Posteriori (MAP) Detector** *that chooses the message index i to maximize the à posteriori probability $p_{m/\boldsymbol{y}}(i, |\boldsymbol{v})$ given a received symbol $\boldsymbol{y} = \boldsymbol{v}$ minimizes the error probability P_e.*

[10] The replacement of a continuous probability distribution function by a discrete probability distribution function (sometimes called a density mass function) is, in strictest mathematical terms, not advisable; however, we do so here, as this particular substitution prevents a preponderance of additional notation, and it has long been conventional in the data transmission literature. The reader is thus forewarned to keep the continuous or discrete nature of the probability density in mind in the analysis of any particular vector channel.

[11] The more general form of this identity is called "Bayes Theorem", [2].

Proof: See the above paragraph.**QED.**

Subsection 1.1.4 describes the calculation of the corresponding optimum average error probability.

The MAP detector thus simply chooses the index i with the highest conditional probability $p_{m_i|\boldsymbol{y}}(i|\boldsymbol{v})$. When m and $\boldsymbol{x}$ are in 1-to-1 correspondence (as always in this chapter), then $|C| = M$ and then $p_{\boldsymbol{x}/\boldsymbol{y}}(i,\boldsymbol{v}) = p_{m/\boldsymbol{y}}(i,\boldsymbol{v})$ and $p_{\boldsymbol{x}}(i) = p_m(i)$. It is often convenient to represent the message by $\boldsymbol{x}$ when this is true. For every possible received vector $\boldsymbol{y}$, the designer of the detector can calculate the corresponding best index i, which depends on the input distribution $p_{\boldsymbol{x}}(i)$.

Thus, Rule 1.1.1 below summarizes the following **MAP** detector rule in terms of the known probability densities of the channel ($p_{\boldsymbol{y}/\boldsymbol{x}}$) and of the input vector ($p_{\boldsymbol{x}}$):

Rule 1.1.1 [MAP Detection Rule]

$$\hat{m} \Rightarrow m_i \quad if \quad p_{\boldsymbol{y}/m}(\boldsymbol{v},i) \cdot p_m(i) \geq p_{\boldsymbol{y}/m}(\boldsymbol{v},j) \cdot p_m(j) \quad \forall \quad j \neq i \tag{1.8}$$

If equality holds in (1.8), then the decision can be assigned to either message m_i or m_j without changing the minimized error probability.

1.1.2.2 The Maximum Likelihood (ML) Detector

If all transmitted messages are equally probable, that is if

$$p_m(i) = \frac{1}{M} \quad \forall\, i = 0, ..., M-1 \quad , \tag{1.9}$$

then the MAP Detection Rule becomes the Maximum Likelihood Detection Rule:

Rule 1.1.2 [ML Detection Rule]

$$\hat{m} \Rightarrow m_i \quad if \quad p_{\boldsymbol{y}/m}(\boldsymbol{v},i) \geq p_{\boldsymbol{y}/m}(\boldsymbol{v},j) \quad \forall \quad j \neq i \quad . \tag{1.10}$$

If equality holds in (1.10), then the decision can be assigned to either message m_i or m_j without changing the error probability.

As with the MAP detector, the ML detector also chooses an index i for each possible received vector $\boldsymbol{y} = \boldsymbol{v}$, but this index now only depends on the channel transition probabilities and is independent of the input distribution (by assumption). The ML detector essentially cancels the $1/M$ factor on both sides of (1.8) to get (1.10). This type of detector only minimizes P_e when the input data messages have equal probability of occurrence. As this requirement is often met in practice, ML detection is often used. Even when the input distribution is not uniform, ML detection is still often employed as a detection rule, because the input distribution may be unknown and thus assumed to be uniform. The **Minimax Theorem** sometimes justifies this uniform assumption:

Theorem 1.1.2 [Minimax Theorem for Symmetric Channels] *The ML detector minimizes the maximum possible average error probability when the input distribution is unknown if the conditional ML error probability $P_{e,ML/m=m_i}$ is independent of i (which also has the name* **symmetric channel**, *see Subsection 1.4.2).*

Proof: First, if $P_{e,ML/i}$ is independent of i, then

$$\begin{aligned} P_{e,ML} &= \sum_{i=0}^{M-1} p_{\boldsymbol{x}}(i) \cdot P_{e,ML/i} \\ &= P_{e,ML/i} \end{aligned}$$

And so,

$$\begin{aligned}\max_{\{p_{\boldsymbol{x}}\}} P_{e,ML} &= \max_{\{p_{\boldsymbol{x}}\}} \sum_{i=0}^{M-1} p_{\boldsymbol{x}}(i) \cdot P_{e,ML/i} \\ &= P_{e,ML} \sum_{i=0}^{M-1} p_{\boldsymbol{x}}(i) \\ &= P_{e,ML} \qquad (1.11)\end{aligned}$$

If R is any receiver other than the ML receiver, then

$$\begin{aligned}\max_{\{p_{\boldsymbol{x}}\}} P_{e,R} &= \max_{\{p_{\boldsymbol{x}}\}} \sum_{i=0}^{M-1} p_{\boldsymbol{x}}(i) \cdot P_{e,R/i} \\ &\geq \sum_{i=0}^{M-1} \frac{1}{M} P_{e,R/i} \text{ (because } \max_{\{p_{\boldsymbol{x}}\}} \{P_{e,R} \geq P_{e,R}\} \text{ for given } \{p_{\boldsymbol{x}}\}\text{, specifically uniform)} \\ &\geq \sum_{i=0}^{M-1} \frac{1}{M} P_{e,ML/i} \text{ (because the ML minimizes } P_e \text{ when } p_{\boldsymbol{x}}(i) = \tfrac{1}{M} \text{ for } i = 0, \ldots, M-1) \\ &= P_{e,ML}\end{aligned}$$

So,

$$\begin{aligned}\max_{\{p_{\boldsymbol{x}}\}} P_{e,R} &\geq P_{e,ML} \\ &= \max_{\{p_{\boldsymbol{x}}\}} P_{e,ML}. \text{ from (1.11)}\end{aligned}$$

Thus, the ML receiver minimizes the maximum P_e over all possible receivers. **QED.**

The symmetry condition imposed by the Minimax Theorem is not always satisfied in practical situations; but the likelihood of an application where both the inputs are nonuniform in distribution and the ML conditional error probabilities are not symmetric is rare. Thus, ML receivers have come to be of nearly ubiquitous use in place of MAP receivers when detecting symbols. If the input probability distribution is not uniform, compression methods can be used to reduce the sender's bit rate so that $p_{\boldsymbol{x}}$ appears uniform at the new lower data rate; however such compression is beyond this text's scope.

1.1.3 Decision Regions

In the case of either the MAP Rule in (1.8) or the ML Rule in (1.10), each and every possible value for the channel output $\boldsymbol{y}$ maps into one of the M possible transmitted messages. Thus, the vector space (or more generally the field of values for the transmitted symbol values) for $\boldsymbol{y}$ is partitioned into M regions corresponding to the M possible decisions. Simple communication systems have well-defined boundaries (to be shown later), so the decision regions often coincide with intuition. Nevertheless, in some well-designed communications systems, the decoding function and the regions can be more difficult to visualize.

Definition 1.1.4 [Decision Region] *The* **decision region** *using a MAP detector for each message m_i, $i = 0, ..., M-1$ is defined as*

$$\mathcal{D}_i \triangleq \{\boldsymbol{v} \mid p_{\boldsymbol{y}/m}(\boldsymbol{v}, i) \cdot p_m(i) \geq p_{\boldsymbol{y}/m}(\boldsymbol{v}, j) \cdot p_m(j) \ \forall \ j \neq i\} \quad . \qquad (1.12)$$

With uniformly distributed input messages, the decision regions reduce to

$$\mathcal{D}_i \triangleq \{\boldsymbol{v} \mid p_{\boldsymbol{y}/m}(\boldsymbol{v}, i) \geq p_{\boldsymbol{y}/m}(\boldsymbol{v}, j) \ \ \forall \ \ j \neq i\} \quad . \tag{1.13}$$

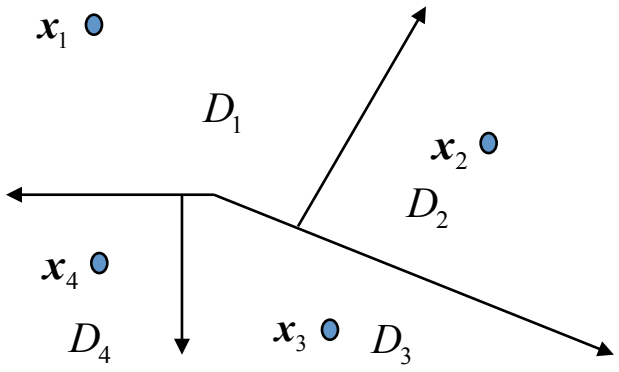


Figure 1.8: **Decision regions.**

In Figure 1.8, each of the four different two-dimensional transmitted vectors $\boldsymbol{x}_i$ (corresponding to the messages m_i) has a surrounding decision region in which any received value for $\boldsymbol{y} = \boldsymbol{v}$ is mapped to the message m_i. In general, the decision regions need not be connected, and although such situations are rare in practice, they can occur (see Problem 1.12). Section 1.3 illustrates several example AWGN decision regions.

1.1.4 Optimum Average Error Probability Calculation

Foillowing [1], the probability of a correct decision, P_c, in Equation (1.7) is for a specific value of m_i. MAP-detector use corresponds to a specific (optimum, maximum) probability of correct decision, and corresponding minimum P_e for those values of $\boldsymbol{v} \in \mathcal{D}_i$, and so can also be rewritten

$$P_c(\hat{m} = m_i, \ \boldsymbol{y} = \boldsymbol{v} \in \mathcal{D}_i) = P_{\boldsymbol{y}=\boldsymbol{v}/m=m_i}(\boldsymbol{v} \in \mathcal{D}_i, m_i) \cdot p_{m_i} \quad . \tag{1.14}$$

The average error probability for a detector $\hat{m} = m_i$ with a optimum-decision-region (or really any decision region corresponding to the a specific) rule $\mathcal{D}_i$ and corresponding $P_{\boldsymbol{y}/m=m_i}(\boldsymbol{v}, m_i)$ would then be

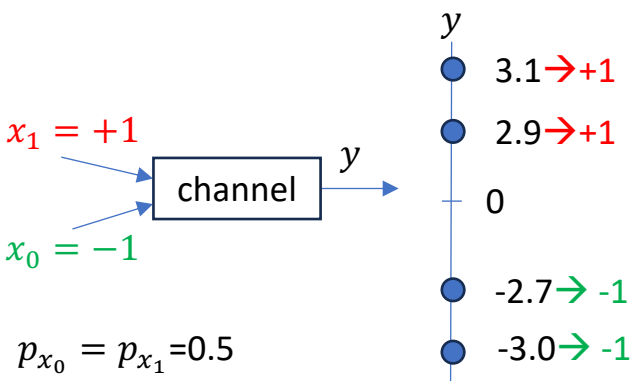
$$P_{c,max} \triangleq \mathbb{E}\left[P_c\right] = \sum_{i=0}^{M-1} \left\{ \sum_{\boldsymbol{v} \in \mathcal{D}_i} P_{\boldsymbol{y}|m=m_i}(\boldsymbol{v}|m_i) \right\} \cdot p_{m_i} \quad . \tag{1.15}$$

Thus the minimum average P_e for the MAP detector can be computed as

$$P_{e,min} \triangleq 1 - P_{c,max} = 1 - \sum_{i=0}^{M-1} \left\{ \sum_{\boldsymbol{v} \in \mathcal{D}_i} p_{\boldsymbol{y}|m_i}(\boldsymbol{v}|i) \right\} \cdot p_{m_i} \quad . \tag{1.16}$$

Several examples of (1.16)'s computation will occur for specific channels later in this chapter. Often in specific cases, the double sum/integration can be tightly bounded and simplified to a simple expressing involving the minimum separation between transmitted symbols and the average number of nearest neighboring transmitted symbols.

EXAMPLE 1.1.6 *[Pe calculation for simple channel]* Figure 1.9 illustrates a channel with binary equiprobable inputs ± 1 and 4 possible outputs with $p_{x/y}$ provided in a table for each input/output combination. The MAP detector, or equivalently in this case the ML detector, selects the largest value of $p_{x/y}(i)$ to i from the corresponding position in each column of the table.



$p_{x/y}$	3.1	2.9	-2.7	-3.0
+1	.9	.85	.1	.2
-1	.1	.15	.9	.8

Figure 1.9: **Simple Channel P_e Example**

Equation (1.16)'s P_e calculation uses $p_{y/x}$. The following equations (after scaling each by 2) implement $p_x = \sum_y p_{x/y} \cdot p_y$:

$$1 = 1.8 \cdot p_{3.1} + 1.7 \cdot p_{2.7} + .2 \cdot p_{-2.7} + .4 \cdot p_{-3} \quad (1.17)$$

$$1 = .2 \cdot p_{3.1} + .3 \cdot p_{2.7} + 1.8 \cdot p_{-2.7} + 1.6 \cdot p_{-3} \quad (1.18)$$

The sum of these equations restates that p_y is a probability distribution and must sum to 1. Using this fact and that $0 \le p_y(i) \le 1$ for all outputs, and observing that for fractions less than or equal to one the sum of the absolute values (equal to probabilities since all nonnegative) must exceed the sum of the squared values. A pseudoinverse solution minimizes the sum of squared values to the undetermined equations (1.17) and (1.18): The following matlab commands find p_y.

```
>> right=[1.8 1.7 .2 .4 ; .2 .3 1.8 1.6]
    1.8000    1.7000    0.2000    0.4000
    0.2000    0.3000    1.8000    1.6000
>> left=[1 ; 1]
     1
     1
py=pinv(right)*left
    0.2409
    0.2421
    0.2597
    0.2573
    >> sum(py)  =    1.0000
```

The probabilities are all positive and sum to unity. They are a unique solution for these non-negative and unity sum constraints because any other solution would sum to greater than 1 (because the pseudoinverse has smallest sum-squared solution and this solution attains that upper bound)[12]. Continuing, $p_{y/x=+1}$ and $p_{y/x=-1}$ follow from $p_{x/y} \cdot p_y / p_x$:

```
>> pygx1=py.*[1.8 ; 1.7 ; .2 ; .4] = % quantity on the right is (px/y / px) for x=+1
```

[12]Generally any all positive-coefficient set of equations for probabilities that sum to 1 would indeed also have pseudoinverse solution. See Appendix C for more on matrix calculus and linear algebra, including the pseudoinverse, which has the property of producing a solution to underdetermined linear equations (more unknowns than linearly independent equations) that has minimum norm.

```
    0.4336
    0.4115
    0.0519
    0.1029
>> pygxm1=py.*[.2 ; .3 ; 1.8 ; 1.6] % quantity on the right is (px/y / px) for x=-1
pygxm1 =
    0.0482
    0.0726
    0.4675
    0.4118
>> pygx1*.5 +pygxm1*.5
    0.2409
    0.2421
    0.2597
    0.2573       checks!
```

With these values for $p_{y/x}$, then Equation (1.15)'s $P_c = \sum_x \sum_{y \in \mathcal{D}_x} p_{y/x} \cdot p_x$ produces:

```
>> pygx=[pygx1' ; pygxm1'] =
    0.4336    0.4115    0.0519    0.1029
    0.0482    0.0726    0.4675    0.4118
>> pxgy=[.9 .85 .1 .2 ; .1 .15 .9 .8] =
    0.9000    0.8500    0.1000    0.2000
    0.1000    0.1500    0.9000    0.8000
>> pxgy*py =
    0.5000
    0.5000  (checks)
>> pxy = pxgy.*[py' ; py'] =
    0.2168    0.2058    0.0260    0.0515
    0.0241    0.0363    0.2337    0.2059
>> sum(pxy) =
    0.2409    0.2421    0.2597    0.2573
>> sum(sum(pxy)) =     1 % checks
>> px=[0.5 ; 0.5];
>> pygx.*[px px px px] =
    0.2168    0.2058    0.0260    0.0515
    0.0241    0.0363    0.2337    0.2059  %checks again
>> Pc= (pygx(1,1)+pygx(1,2)+pgyx(2,3)+pygx(2,4))*.5 =    0.8622 % sum over only y in decision region (each x)
>> Pe=1-ans =    0.1378
```

So, there is roughly 14% chance of bit error with an optimum detector for this simple channel. This is usually unacceptably high error probability for most systems. With Chapter 2's codes, this channel can have very low error probability ($P_e \to 0$) if strings of large-N input $\pm$ values, or "codewords" in have $+1$ slightly more probable ($p_{+1} = .524$) than -1, leading to reliably decodable average bit rates as high as roughly $\bar{b} = b/N = .998$ for this channel. Usually, more channel output possibilities than channel input possibilities enhances the ability to exploit coding, although increasing constellation size $|C| > 2^b$ also can assist coding.

1.1.5 Irrelevant Components of the Channel Output

The discrete channel-output vector $\boldsymbol{y}$ may contain information that does not help determine which of the M messages has been transmitted. These irrelevant components may be discarded without loss of performance, i.e. the input detected and the associated error probability remain unchanged. The received symbol $\boldsymbol{y}$ can be separated into two sets of dimensions, those that carry useful information $\boldsymbol{y}_1$ and those that do not carry useful information $\boldsymbol{y}_2$. That is,

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{bmatrix} \quad . \tag{1.19}$$

Theorem 1.1.3 summarizes the condition on $\boldsymbol{y}_2$ that guarantees irrelevance [1]:

Theorem 1.1.3 [Theorem on Irrelevance] *If*

$$p_{\boldsymbol{x}/(\boldsymbol{y}_1,\boldsymbol{y}_2)} = p_{\boldsymbol{x}/\boldsymbol{y}_1} \tag{1.20}$$

or equivalently if the channel-related probability distribution

$$p_{\boldsymbol{y}_2/(\boldsymbol{y}_1,\boldsymbol{x})} = p_{\boldsymbol{y}_2/\boldsymbol{y}_1} \tag{1.21}$$

then $\boldsymbol{y}_2$ *is not needed in the optimum receiver, that is,* $\boldsymbol{y}_2$ *is* **irrelevant**.

Proof: For a MAP receiver, then clearly the value of $\boldsymbol{y}_2$ does not affect the maximization of $p_{\boldsymbol{x}|(\boldsymbol{y}_1,\boldsymbol{y}_2)}$ if $p_{\boldsymbol{x}|(\boldsymbol{y}_1,\boldsymbol{y}_2)} = p_{\boldsymbol{x}|(\boldsymbol{y}_1)}$ and thus $\boldsymbol{y}_2$ is irrelevant to the optimum receiver's decision. Equation (1.20) can be written as

$$\frac{p_{(\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2)}}{p_{(\boldsymbol{y}_1,\boldsymbol{y}_2)}} = \frac{p_{(\boldsymbol{x},\boldsymbol{y}_1)}}{p_{\boldsymbol{y}_1}} \tag{1.22}$$

or equivalently via "cross multiplication"

$$\frac{p_{(\boldsymbol{x},\boldsymbol{y}_1,\boldsymbol{y}_2)}}{p_{(\boldsymbol{x},\boldsymbol{y}_1)}} = \frac{p_{(\boldsymbol{y}_1,\boldsymbol{y}_2)}}{p_{\boldsymbol{y}_1}} \quad , \tag{1.23}$$

which is the same as (1.21). **QED**.

The reverse of the theorem of irrelevance is not necessarily true, as can be shown by counterexamples. Two examples (due to Wozencraft and Jacobs, [1]) reinforce the concept of irrelevance. In these examples, the two noise signals $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ are independent and the input is uniformly distributed.

EXAMPLE 1.1.7 *[Extra Irrelevant Noise]* Suppose $\boldsymbol{y}_1$ is the noisy channel output shown in Figure 1.10.

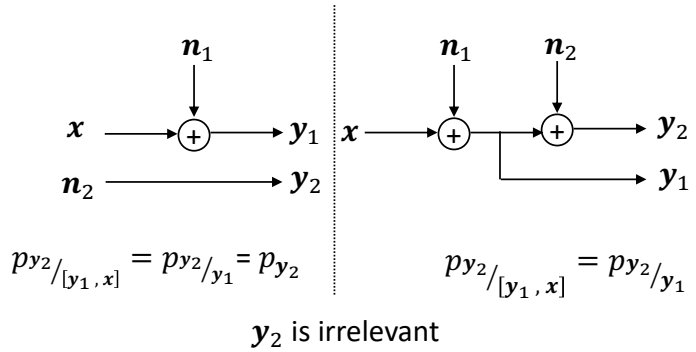


Figure 1.10: **Extra irrelevant noise.**

In the first example on the Figure 1.10's left, $\mathrm{p}_{\boldsymbol{y}_2|\boldsymbol{y}_1,\,\boldsymbol{x}} = \mathrm{p}_{\boldsymbol{n}_2} = \mathrm{p}_{\boldsymbol{y}_2|\boldsymbol{y}_1}$, thus satisfying the condition for $\boldsymbol{y}_2$ to be ignored, as might be obvious upon casual inspection. The extra independent noise signal $\boldsymbol{n}_2$ tells the receiver nothing given $\boldsymbol{y}_1$ about the transmitted message $\boldsymbol{x}$. In the second example on Figure 1.10's right, the irrelevance of $\boldsymbol{y}_2$ given $\boldsymbol{y}_1$ is not quite as obvious as the signal is present in both the received channel output components. Nevertheless, $\mathrm{p}_{\boldsymbol{y}_2|\boldsymbol{y}_1,\,\boldsymbol{x}} = \mathrm{p}_{\boldsymbol{n}_2}(v_2 - v_1) = \mathrm{p}_{\boldsymbol{y}_2|\boldsymbol{y}_1}$, so $\boldsymbol{y}_2$ can be ignored.

In some other cases the output component $\boldsymbol{y}_2$ should not be discarded. A classic example is the following case of "noise cancelation":

EXAMPLE 1.1.8 [Noise Cancelation] Suppose $\boldsymbol{y}_1$ is the noisy channel output shown in Figure 1.11

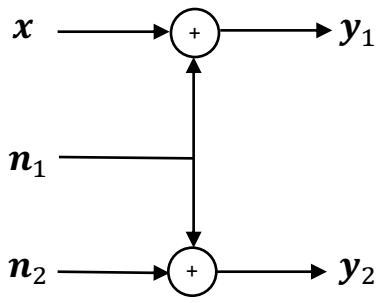


Figure 1.11: **Noise can be partially canceled.**

while $\boldsymbol{y}_2$ may appear to contain only useless noise, it is in fact possible to reduce the effect of $\boldsymbol{n}_1$ in $\boldsymbol{y}_1$ by constructing an estimate of $\boldsymbol{n}_1$ using $\boldsymbol{y}_2$. Correspondingly, $\mathrm{p}_{\boldsymbol{y}_2|\boldsymbol{y}_1,\,\boldsymbol{x}} = \mathrm{p}_{\boldsymbol{n}_2}(\boldsymbol{v}_2 - (\boldsymbol{v}_1 - \boldsymbol{x}_i)) \neq \mathrm{p}_{\boldsymbol{y}_2|\boldsymbol{y}_1}$.

1.1.5.1 Reversibility

An important result in digital communication is the **Reversibility Theorem**, which will be used several times in this text. This theorem is, in effect, a special case of the Theorem on Irrelevance:

Theorem 1.1.4 [Reversibility Theorem] *The application of an invertible transformation to the channel output vector $\boldsymbol{y}$ does not affect the performance of the MAP detector.*

Proof: Using the Theorem on Irrelevance, if the channel output is $\boldsymbol{y}_2$ and the result of the invertible transformation is $\boldsymbol{y}_1 = G(\boldsymbol{y}_2)$, with inverse $\boldsymbol{y}_2 = G^{-1}(\boldsymbol{y}_1)$, then $[\boldsymbol{y}_1\ \boldsymbol{y}_2] = \left[\boldsymbol{y}_1\ \ G^{-1}(\boldsymbol{y}_1)\right]$. Then, $\mathrm{p}_{\boldsymbol{x}/(\boldsymbol{y}_1,\boldsymbol{y}_2)} = \mathrm{p}_{\boldsymbol{x}/\boldsymbol{y}_1}$, which is the definition of irrelevance. Thus, either of $\boldsymbol{y}_1$ or $\boldsymbol{y}_2$ is sufficient to detect $\boldsymbol{x}$ optimally and attain the same minimum error probability or equivalently the same optimum performance.**QED**.

Equivalently, Figure 1.12 illustrates the reversibility theorem by constructing a MAP receiver for the output of the invertible transformation $\boldsymbol{y}_1$ as the cascade of the inverse filter G^{-1} and the MAP receiver for the input of the invertible transformation $\boldsymbol{y}_2$. The receiver for $\boldsymbol{y}_2$ can sometimes be simpler to design than one for $\boldsymbol{y}_1$. Later chapters will use this concept to produce equivalent optimum receivers that might not otherwise appear equivalent.

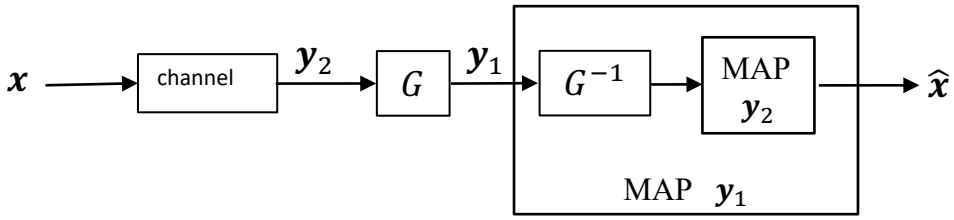


Figure 1.12: **Reversibility theorem illustration.**

1.1.6 Optimum Bit-Error Probability and Log Likelihood Decoding

Designers may be interested in minimizing the **bit-error probability** within a message instead of minimizing the symbol/message error. The message may then be viewed as having b bits specifically denoted by u_j $j = 1, ..., b$ with $u_j = 0$ or 1, and abbreviated by the vector $\boldsymbol{u}$. p_{u_j} is the probability distribution for u_j. A set of MAP (or ML) detectors, one for each bit, can be designed with error criterion to minimize

$$\bar{P}_{b,j} = \Pr\{\hat{u}_j \neq u_j\} = 1 - \sum_{u_j=0}^{1} \left[\sum_{\boldsymbol{v} \in \mathcal{D}_j} P_{\boldsymbol{y}/u_j}(\hat{u}_j = u_j, \boldsymbol{v}) \right] \cdot p_{u_j} \tag{1.24}$$

$$= 1 - \sum_{u_j=0}^{1} \left[\sum_{\boldsymbol{v} \in \mathcal{D}_j} p_{u_j|\boldsymbol{y}}(u_j, \boldsymbol{v}) \cdot p_{\boldsymbol{y}}(\boldsymbol{v}) \right] \quad . \tag{1.25}$$

This error probability is not necessarily equal for each bit, nor consequently equal to the minimized symbol-error probability, P_e, although generally speaking minimizing the symbol error probability will usually lead to better bit-error probability. Calculation of error probability will require the probability distribution of the vector of bits $p_{\boldsymbol{u}|\boldsymbol{y}}$ in place of $p_{m/\boldsymbol{y}}$. The notation $\boldsymbol{u} \setminus u_j$ means the vector $\boldsymbol{u}$ with the j^{th} bit removed. The bit-error optimum receiver will average other bits as explicitly indicated by writing $p_{u_j/\boldsymbol{y}}(u_j, \boldsymbol{v})$ for any received symbol $\boldsymbol{v}$ as the sum of 2^{b-1} terms:

$$p_{\boldsymbol{y}/u_j}(u_j, \boldsymbol{v}) = \sum_{\boldsymbol{u} \setminus u_j} p_{\boldsymbol{y}|\boldsymbol{u}}(\boldsymbol{u}, \boldsymbol{v}) \cdot p_{\boldsymbol{u}} \tag{1.26}$$

$$P_{e,u_j} = 1 - \sum_{u_j=0}^{1} \sum_{\boldsymbol{v} \in \mathcal{D}_j} \left\{ \sum_{\boldsymbol{u} \setminus u_j} \sum_{\boldsymbol{u} \setminus u_j} p_{\boldsymbol{y}|\boldsymbol{u}}(\boldsymbol{u}, \boldsymbol{v}) \cdot p_{\boldsymbol{u}} \right\} \quad . \tag{1.27}$$

Thus, the minimized probability of bit error for any bit can then be computed from the given conditional channel-probability distribution and the input bit-vector probabilities (uniform in ML case), albeit the calculations may be tedious. These calculations may be simplified for specific channels, as evident in Section 1.3 for the additive white Gaussian noise channel and in Section 1.4 for the binary symmetric channel and binary erasure channel.

The bit-decision process can sometimes be simplified through the use of log-likelihood ratios (LLRs).

Definition 1.1.5 [Log Likelihood Ratio (LLR)] *A log likelihood ratio for a bit u_j is the logarithm of probability ratio that bit takes the values 0 and 1. Often convention considers the bit value 0 as correct and the bit value 1 as incorrect, without loss of generality.*

$$\begin{aligned} LLR_{u_j}(\boldsymbol{v}) &\triangleq \ln\left(\frac{P_{u_j=0}(\boldsymbol{v})}{P_{u_j=1}(\boldsymbol{v})}\right) \\ &= \ln\left(\frac{\sum_{\boldsymbol{u}\setminus u_j} p_{\boldsymbol{y}|\boldsymbol{u}}(\boldsymbol{v},\boldsymbol{u}\mid_{u_j=0})\cdot p_{\boldsymbol{u}}(\boldsymbol{u}\mid_{u_j=0})}{\sum_{\boldsymbol{u}\setminus u_j} p_{\boldsymbol{y}|\boldsymbol{u}}(\boldsymbol{v},\boldsymbol{u}\mid_{u_j=1})\cdot p_{\boldsymbol{u}}(\boldsymbol{u}\mid_{u_j=1})}\right) \quad . \end{aligned}$$

When $u_j \to \hat{u}_j$ above for a decoder with average bit-error rate $\bar{P}_b$, then

$$LLR_{\hat{u}_j} = \ln\frac{1-\bar{P}_{b,j}}{\bar{P}_{b,j}} \tag{1.28}$$

A positive value of LLR_{u_j} leads to the decision $\hat{u}_j = 1$, while a negative value leads to decision $\hat{u}_j = 0$. This type of decoding avoids the use of decision regions and maybe useful in systems where many bits are simultaneously decided, and there are relationships (called codes) between the bits that allow iterative construction of all the bits' log-likelihood ratios that converges to a final set of values that lead to optimum decisions for each bit at the end of the iterative decoding process. LLRs are a form of **soft information** or **soft decisions** that are intermediate to a final decision on a quantity, in this case a bit.

1.1.7 P_e Calcuation and The Bhattacharrya Bound

The Bhattacharya Bound (or B-Bound) for error probability finds use in systems with coding (like those of Chapters 2, 8, and beyond). The messages will be presumed to be the indices themselves so that $m \in \{0, ..., M-1\}$ each with corresponding symbol value $\boldsymbol{x}_m$. The B-bound bounds the probability that a MAP detector chooses a specific symbol $\boldsymbol{x}_{\tilde{m}}$ instead of the correct message m. This **error event** is denoted $\epsilon_{m\tilde{m}}$ with probability $P\{\varepsilon_{m\tilde{m}}\} \triangleq Pr\{\boldsymbol{x}_m \to \boldsymbol{x}_{\tilde{m}}\}$.

Theorem 1.1.5 [Bhattacharya Bound] *The error probability, using a maximum-likelihood decoder, that corresponds to choosing message $\tilde{m}$ in place of the correct message m is generally bounded according to the following expression:*

$$P\{\varepsilon_{m\tilde{m}}\} \leq \sum_{\boldsymbol{v}} \sqrt{p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v},\boldsymbol{x}_{\tilde{m}})\cdot p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v},\boldsymbol{x}_m)} \tag{1.29}$$

For symmetric channels where $P\{\epsilon_{m\tilde{m}}\} = P\{\epsilon_{\tilde{m}m}\}$ and all inputs equally likely, this B-Bound tightens to

$$P\{\varepsilon_{m\tilde{m}}\} \leq \frac{1}{2}\cdot\sum_{\boldsymbol{v}} \sqrt{p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v},\boldsymbol{x}_{\tilde{m}})\cdot p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v},\boldsymbol{x}_m)} \quad . \tag{1.30}$$

Proof: First the general case without symmetry or equally likely inputs: Let $P\{\varepsilon_{m\tilde{m}}\}$ denote the probability that message m is erroneously decided by a maximum likelihood decoder to be message $\tilde{m}$. Then, the corresponding received symbol $\boldsymbol{y} = \boldsymbol{v}$ must be such that

$p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_{\tilde{m}}) \geq p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_m)$. The region of $\boldsymbol{v}$ over which this error could occur is denoted $\mathcal{D}_{m\tilde{m}}$:

$$\mathcal{D}_{m\tilde{m}}(\boldsymbol{v}) \triangleq \left\{ \boldsymbol{v} : \frac{p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_{\tilde{m}})}{p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_m)} \geq 1 \right\} \quad . \tag{1.31}$$

Then,

$$P\{\varepsilon_{m\tilde{m}}\} = \sum_{\boldsymbol{v} \in \mathcal{D}_{m\tilde{m}}(\boldsymbol{v})} p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_m) = \sum_{\boldsymbol{v}} f(\boldsymbol{v}) \cdot p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_m) \quad , \tag{1.32}$$

where

$$f(\boldsymbol{v}) \triangleq \begin{cases} 1 & \boldsymbol{v} \in \mathcal{D}_{m\tilde{m}}(\boldsymbol{v}) \\ 0 & \boldsymbol{v} \notin \mathcal{D}_{m\tilde{m}}(\boldsymbol{v}) \end{cases} \quad . \tag{1.33}$$

Further,

$$f(\boldsymbol{v}) \leq \left[\frac{p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_{\tilde{m}})}{p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_m)} \right]^{1/2} \quad , \tag{1.34}$$

and thus (1.32) becomes

$$P\{\varepsilon_{m\tilde{m}}\} \leq \sum_{\boldsymbol{v}} \sqrt{p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_{\tilde{m}}) \cdot p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_m)} \, . \tag{1.35}$$

Symmetric case tightening: Under symmetry, the indicator function $f(\boldsymbol{v})$ applies equally to $\mathcal{D}_{m\tilde{m}}$ and to $\mathcal{D}_{\tilde{m}m}$ in Equation (**??**) applies to the sum of terms, which can add and insert in the sum in (1.31), bounding both the events $\epsilon_{m\tilde{m}}$ and $\epsilon_{\tilde{m}m}$ in that same equation, so then

$$2 \cdot P\{\varepsilon_{m\tilde{m}}\} \leq \sum_{\boldsymbol{v}} \sqrt{p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_{\tilde{m}}) \cdot p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{x}_m)} \tag{1.36}$$

where m and $\tilde{m}$ are interchangeable. Thus, this bound tightens by a factor of 2. **QED**.

The tighter symmetric-case B-Bound was noted by [8] in 2007, who in turn cite a 1965 equivalent in hypothesis testing [9].[13] For asymmetric channels or non-uniform input distributions, the event $\epsilon_{m\tilde{m}}$ can be viewed as the sum of probabilities of two error events $m \to \tilde{m}$ and $\tilde{m} \to m$. Decomposition into unequal terms complicates the simple nature of the B-Bound, so is not usually helpful except in the symmetric case that often applies to binary linear codes later in Chapters 2 and 8.

A memoryless channel (see Section 1.4 for more on memoryless channels) has the property

$$p_{\boldsymbol{y}/\boldsymbol{x}} = \prod_{n=1}^{N} p_{y_n/x_n} \tag{1.37}$$

Effectively $\nu = 0$, the defaul,t on memoryless channels.

Memoryless channels essentially have independent dimensions. For such channels the B-Bound takes a simpler form through the repeated use of distribution of multiplication over addition:

$$P\{\varepsilon_{m\tilde{m}}\} \leq \sum_{\boldsymbol{v}} \prod_{n=1}^{N} \sqrt{p_{\boldsymbol{y}/\boldsymbol{x}}(y_n, x_{\tilde{m},n}) \cdot p_{\boldsymbol{y}/\boldsymbol{x}}(y_n, x_{m,n})} = \prod_{n=1}^{N} \sum_{y_n} \sqrt{p_{\boldsymbol{y}/\boldsymbol{x}}(y_n, x_{\tilde{m},n}) \cdot p_{\boldsymbol{y}/\boldsymbol{x}}(y_n, x_{m,n})} \quad . \tag{1.38}$$

The sum over each dimension's output symbol values can be much less complex than the vector summation in the general case. Specialization to the case of bit-error probability and the vector $\boldsymbol{u}$ being treated as a symbol vector itself creates a special form. In this case, it is often convenient to investigate

[13] The author notes and sandwiches himself specifically between reference [8], former PhD advisee (by me) and UCLA Prof Richard D. Wesel (07-15-1966 -) and former PhD Advisor (of me) reference [9] Thomas Kailath (1935 -), an Indian-born American electrical engineer and Stanford professor.

the messages m and $\tilde{m}$ differing in d_H bit positions. If the bit-error probability was the same on all dimensions and set equal to p, then the B-bound has form:

$$
\begin{aligned}
P\{\varepsilon_{m\tilde{m}}\} &\leq \prod_{n=1}^{N} \sum_{y_n} \sqrt{p_{\boldsymbol{y}/\boldsymbol{x}}(y_n, \tilde{u}_n) \cdot p_{\boldsymbol{y}/\boldsymbol{x}}(y_n, 0)} && (1.39)\\
&= \prod_{i=1}^{d_H} \sum_{y_n} \sqrt{p_{\boldsymbol{y}/\boldsymbol{x}}(y_n, 1) \cdot p_{\boldsymbol{y}/\boldsymbol{x}}(y_n, 0)} && (1.40)\\
&= \prod_{i=1}^{d_H} \sum_{y_n} \sqrt{p(1-p)} && (1.41)\\
&= [4p(1-p)]^{d_H/2} \quad . && (1.42)
\end{aligned}
$$

(1.42) reduces by a factor of 1/2 when the input bits are equally likely. Many binary code analyses ignore this extra factor with large d_H because the dominant effect is the exponential reduction with d_H.

1.2 Data Modulation and Demodulation for Continuous-Time Channels

Figure 1.13 generalizes Figure 1.2 to continuous time. **Continuous-time channels** occur in many practical situations where the channel accepts only a continuous-time or "analog" waveform, $x(t)$, called a **signal**. The corresponding **received signal**, $y(t)$, is also continuous time. Examples include virtually all wireless channels where electromagnetic waveforms are physically transmitted, not actually the symbols. Examples also include virtually all forms of wireline (transmission-line or waveguide, optical or metallic) connections. These continuous-time channels require that the transmit symbols correspond uniquely to a set of continuous-time signals, $\{x_i(t)\}_{i=0,\ldots,M-1}$.

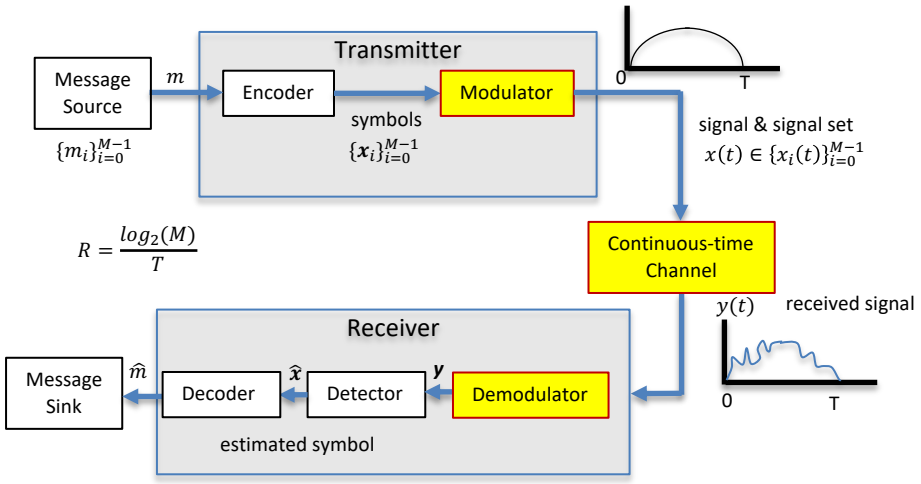


Figure 1.13: **Discrete data transmission with a continuous-time channel.**

As in Figure 1.13, the **modulator** converts the symbol vector $\boldsymbol{x}$ into the continuous-time signal that the transmitter sends into the continuous-time channel. Correspondingly, the **demodulator** converts continuous-time received signal $y(t)$ into the received-symbol vector $\boldsymbol{y}$, from which the detector tries to estimate $\boldsymbol{x}$, shown as $\hat{\boldsymbol{x}}$, and thus also into the message sent through the decoder. The estimated message, one from the message set associated with the message sender, then are provided by the receiver to the message recipient . A desirable property would be that the continuous-time channel can be completely represented by a discrete-time channel of Section 1.1. Indeed a very important practical channel, Section 1.3's **Additive White Gaussian Noise (AWGN)** channel, can be so converted without loss into a discrete-time equivalent channel exactly like that in Figure 1.13.

As an example, Binary Phase-Shift Keying (BPSK) is perhaps one of the simplest forms of modulation:

EXAMPLE 1.2.1 [Binary Phase-Shift Keying (BPSK)] Figure 1.14 repeats Figure 1.1 with a specific linear time-invariant channel that has the transfer function indicated. This channel essentially passes signals between 100 Hz and 200 Hz with 150 Hz having the largest gain. Binary logic familiar to most electrical engineers transmits some positive voltage level (say perhaps 1 volt) for a 1 and another voltage level (say 0 volts) for a 0 inside integrated circuits. Clearly such a constant 1/0 transmission on this "DC-blocking" channel would not pass through Figure 1.14's channel, leaving a received signal level of nearly 0 regardless of the channel-input signal's constant voltage level. This zero received-signal level would complicate receiver detection of the correct message. Instead the two modulated signals $x_0(t) = +cos(2\pi 150t)$ and $x_1(t) = -cos(2\pi 150t)$ easily pass through this channel and are more readily distinguishable by the receiver. This latter type of transmission is an example

of BPSK. If the symbol period is 1 second and if successive transmission is used, the data rate would be 1 bit per second (1 bps).[14] In more detail, the engineer could recognize the trivial vector encoder that converts the message bit of 0 or 1 into the real one-dimensional vectors $\boldsymbol{x}_0 = +1$ and $\boldsymbol{x}_1 = -1$. The modulator simply multiples this $\boldsymbol{x}_i$ value by the function $cos(2\pi t)$ to obtain BPSK.

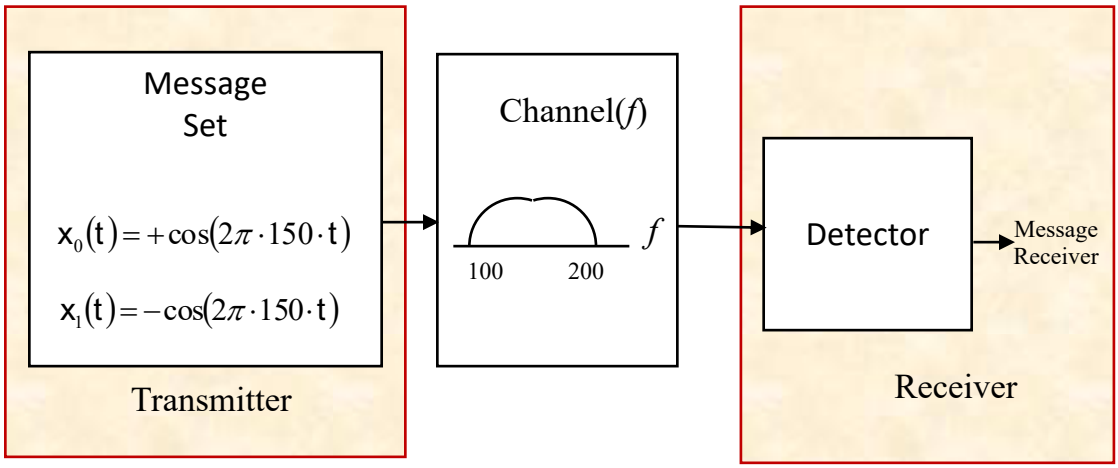


Figure 1.14: **Example of channel for which 1 volt and 0 volt binary transmission is inappropriate, but BPSK modulation matches well.**

Section 1.1's vector representation however is common and leads to a single modulation-independent performance analysis of the data transmission (or storage) system. This section describes such a discrete vector representation of any continuous-time signal set and the conversion between the vector symbols and the continuous-time signals. This symbol-based analysis approach was pioneered by Wozencraft and Jacobs. Each modulation method selects a set of basis functions that link a constellation $\{\boldsymbol{x}_i\}$ with the continuous signals $\{x_i(t)\}$. The modulation basis-function choice usually depends upon the channel. This section and Section 1.3 investigate and enumerate a number of different basis functions as well as the modulation-independent constellation designs that can be used with any modulation choice.

1.2.1 Signal Waveform Representation by Vectors

The reader should be familiar with the infinite-series decomposition of continuous-time signals from the basic electrical-engineering study of Fourier transforms. For the transmission and detection of a message during a symbol period, this text considers the set of real-valued functions $\{f(t)\}$ such that $\int_0^T f^2(t)dt < \infty$ (technically known as the Hilbert space of continuous-time functions and abbreviated as $L_2[0,T]$). This infinite dimensional vector space has an inner product that measures a distance-scaled angle between two different functions $f(t)$ and $g(t)$,

$$\langle f(t), g(t) \rangle = \int_0^T f(t) \cdot g(t) dt \quad .$$

Definition 1.2.2 more formally addresses the inner product.

[14] However, this chapter is mainly concerned with a single transmission. Each of this example's successive transmissions could be treated independently by ignoring transients at the beginning and/or end of any message transmission, because these transients would likely be negligible in time extent compared to a 1 second symbol period.

An **orthonormal basis function** allows formalization of the modulation concept:

Definition 1.2.1 [Orthonormal Basis Functions] *A set of N functions $\{\varphi_n(t)\}$ constitute an N-dimensional* **orthonormal basis** *if they satisfy the following property:*

$$\int_{-\infty}^{\infty} \varphi_m(t) \cdot \varphi_n(t) dt \quad = \quad \delta_{mn} \quad = \begin{cases} 1 & m = n \\ 0 & m \neq n \end{cases} \quad . \tag{1.43}$$

The discrete-time function δ_{mn} will be called the **discrete delta function**[15]. Any continuous-time function (or signal) $x(t) \in L_2(0,T)$ decomposes according to some set of N **orthonormal basis functions** $\{\varphi_i(t)\}_{n=1}^{N}$ as

$$x(t) = \sum_{n=1}^{N} x_n \cdot \varphi_n(t)$$

where $\varphi_n(t)$ satisfy $\langle \varphi_n(t), \varphi_m(t) \rangle = 1$ for $n = m$ and 0 otherwise, often written $\langle \varphi_n(t), \varphi_m(t) \rangle = \delta_{nm}$. The modulated signal $x(t)$ thus relates to the symbol vectof $\boldsymbol{x}$ though its dimensional components:

$$\boldsymbol{x} = \begin{bmatrix} x_N \\ \vdots \\ x_1 \end{bmatrix} \quad .$$

The number of basis functions that represent the signal set $\{x_i(t)\}$ for a particular modulation choice may be infinite, i.e. N may equal ∞, but are the same for each possible symbol value in the constellation and correspondingly for each possible signal in the signal set. Each signal $x_i(t)$ maps to a set of N real numbers $\{x_{in}\}$; these real-valued scalar coefficients assemble into the N-dimensional symbol vector $\boldsymbol{x}$.

Figure 1.15 illustrates the signal $x(t)$ graphically for $N = 3$-dimensional symbol with axes defined by the modulation basis functions $\{\varphi_n(t)\}$.

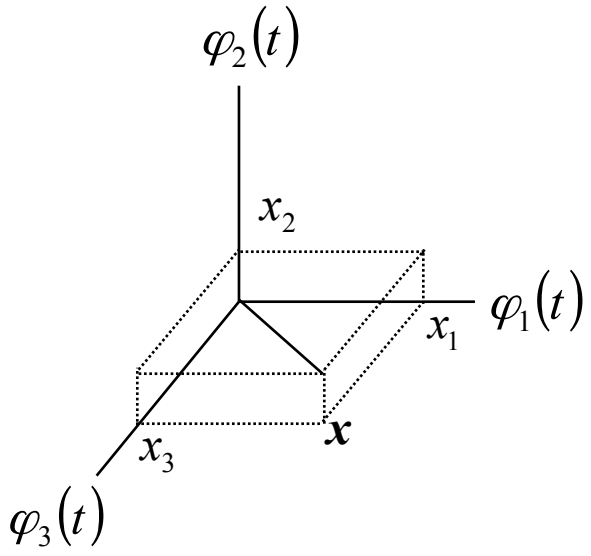


Figure 1.15: **Vector space.**

[15] δ_{mn} is also called a "Kronecker" delta.

Such a geometric viewpoint advantageously enables the visualization of the distance between continuous-time functions using distances between the associated symbol vectors in $\mathcal{R}^N$, the space of N-dimensional real vectors when $\boldsymbol{x}$ is a real vector. In fact, later developments show

$$\langle x_1(t), x_2(t) \rangle = \langle \boldsymbol{x_1}, \boldsymbol{x_2} \rangle \quad , \tag{1.44}$$

where Equation (1.44)'s right-hand side is the usual Euclidean inner product in $\mathcal{R}^N$ (discussed later in Definition 1.2.2). This continuous-time modulation representation formally extends to random processes using what is known as a "Karhunen-Loeve expansion," where the values x_n are considered random variables, and the functions $\varphi_n(t)$ are deterministic. Thus, the message index, i, usually does not appear, but the vector symbol value $\boldsymbol{x}$ is randomly chosen according to the message distribution from the symbol set in use. Thus, $\boldsymbol{x}_n$ refers to a random message component on the n^{th} modulator basis function, and not the "n^{th} message" as this text proceeds to avoid notational proliferation. The basis functions also extend for all time, i.e. on the infinite time interval $(-\infty, \infty)$, in which case the inner product becomes $\langle f(t), g(t) \rangle = \int_{-\infty}^{\infty} f(t) \cdot g(t) dt$. The modulator's composition of random processes is fundamental to demodulation and detection in the presence of noise. Modulation constructively assembles random signals for the communication system from a basis-function set $\{\varphi_n(t)\}$ and a set of symbol vectors $\{\boldsymbol{x_i}\}$. The chosen basis functions and symbol vectors typically satisfy system physical constraints and determine performance in the presence of noise.

Figure 1.16 explicitly shows construction of a **modulated waveform** $x(t)$, where again, each distinct symbol constellation vector point corresponds to a different modulated waveform, but all the waveforms share the same set of basis functions.

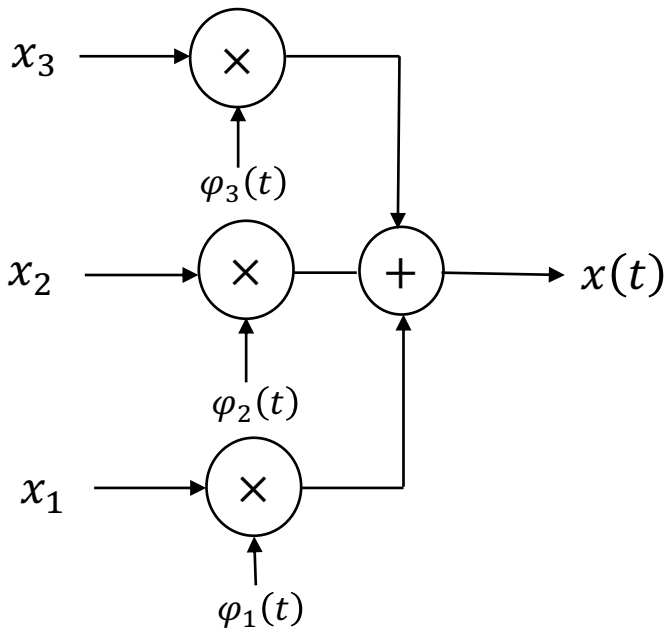


Figure 1.16: **The modulator.**

The power available in any physical communication system limits the average amount of energy required to transmit each successive data symbol. With inner productions, definition 1.1.2's average energy becomes

$$\mathcal{E}_{\boldsymbol{x}} = \mathbb{E}\left[\langle x(t), x(t) \rangle\right] = E\left[\langle \boldsymbol{x}, \boldsymbol{x} \rangle\right] = \mathbb{E}\left[\|\boldsymbol{x}\|^2\right] \tag{1.45}$$

or equivalently the average length of the constellation's symbol vectors. The minimization of $\mathcal{E}_{\boldsymbol{x}}$ intuitively places signal-constellation points near the origin; however, the distance between points shall relate

to the probability of correctly detecting the symbols in the presence of noise. The geometric problem of optimally arranging points in a vector space with minimum average energy while maintaining at least a minimum distance between each pair of points is the well-studied sphere-packing problem, said geometric viewpoint of communication formalized first in Shannon's 1948 seminal famous work, A Mathematical Theory of Communication (Bell Systems Technical Journal). Chapter 2 addresses directly this **coding** challenge through the use of symbol constellations (no matter the explicit modulation-type details).

1.2.2 Modulator Examples

Returning to Example 1.2.1, the next example illustrates the utility of the basis-function concept:

EXAMPLE 1.2.2 BPSK revisited] A more general form of BPSK's basis functions, which are parameterized by variable T, is $\varphi_1(t) = \sqrt{\frac{2}{T}} \cos\left[\frac{2\pi t}{T} + \frac{\pi}{4}\right]$ and $\varphi_2(t) = \sqrt{\frac{2}{T}} \cos\left[\frac{2\pi t}{T} - \frac{\pi}{4}\right]$ for $0 \leq t \leq T$ and 0 elsewhere. These two basis functions ($N = 2$), $\varphi_1(t)$ and $\varphi_2(t)$, are shown in Figure 1.17.

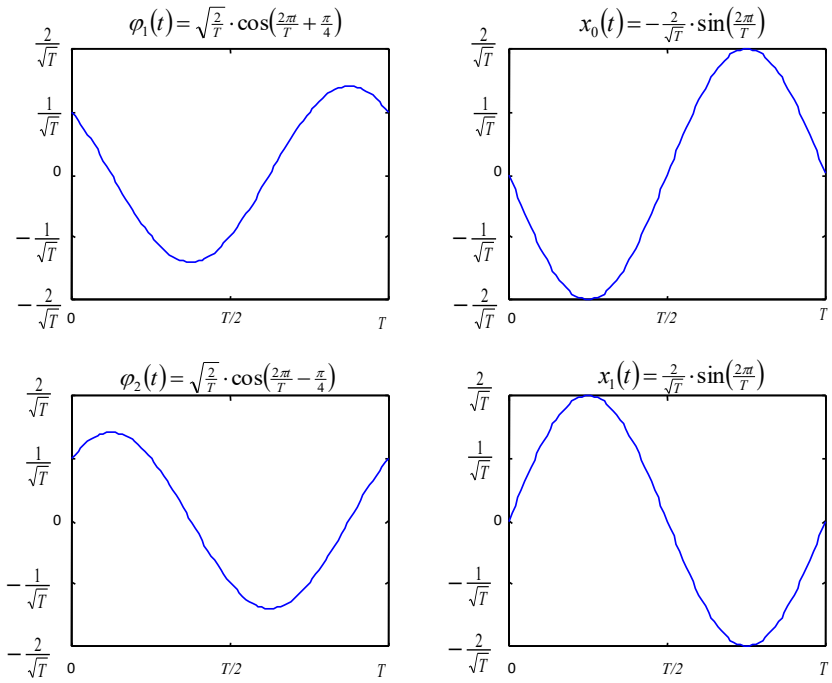


Figure 1.17: **BPSK basis functions and waveforms.**

The two basis functions are orthogonal to each other and both have unit energy, thus satisfying the orthonormality condition. The two possible modulated signals transmitted during the interval $[0, T]$ also appear in Figure 1.17, where $x_0(t) = \varphi_1(t) - \varphi_2(t)$ and $x_1(t) = \varphi_2(t) - \varphi_1(t)$. Thus, the data symbol vectors associated with the continuous-time signals are $\boldsymbol{x}_0 = [1 \ -1]'$ and $\boldsymbol{x}_1 = [-1 \ 1]'$ (a prime denotes transpose). The signal constellation appears in Figure 1.18.

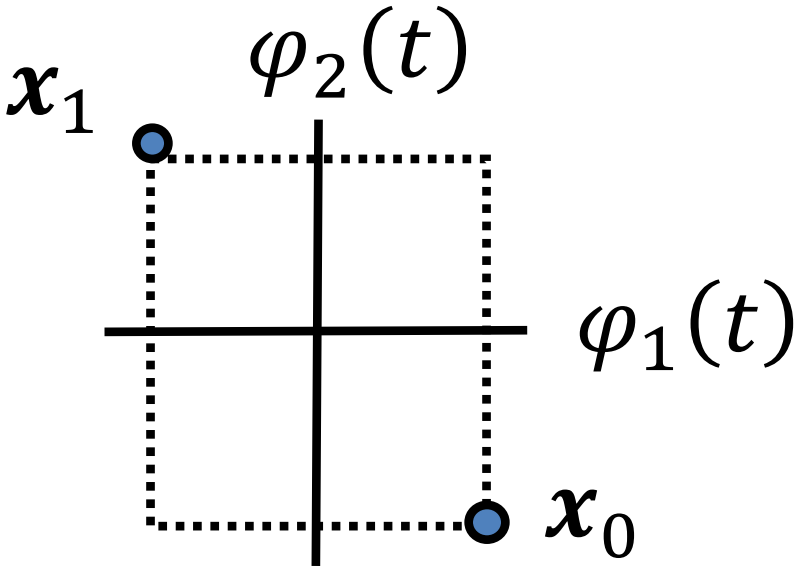


Figure 1.18: **BPSK and FM/Manchester signal constellation.**

The resulting waveforms are $x_0(t) = -\frac{2}{\sqrt{T}}\sin(\frac{2\pi t}{T})$ and $x_1(t) = \frac{2}{\sqrt{T}}\sin(\frac{2\pi t}{T})$. The name "binary phase-shift keying," because the two waveforms are shifted in phase from each other. Other basis functions (and rotated versions of the constellation) could thus also be called BPSK. Since only two possible waveforms are transmitted during each T second symbol period, the data rate is $R = \log_2(2) = 1$ bit per T seconds. Thus to transmit at 1 million bits per second, or abreviaed 1 Mbps, T must equal 10^{-6} seconds or 1 μs. (Additional scaling may adjust the BPSK transmit power/energy level to some desired value, and then applies uniformly to all possible constellation points and transmit signals.)

Another set of basis functions is known as "FM code" (FM is "Frequency Modulation") in the storage industry and also as "Manchester Encoding" in data communications. This method is used to write (modulate) in many commercial disk storage products. It was also used in a quite different area known as "Ethernet" (Ethernet is commonly used in local area networks short distance wired data transmssion). The basis functions are approximated in Figure 1.19 – in practice, the sharp edges are somewhat smoother depending on the specific implementation. The two basis functions again satisfy the orthonormality condition. The data rate equals one bit per T seconds; for a data transfer rate into the disk of 1 GByte/s or 8 Gbps, $T = 1/(8GHz) = 125$ps; by contrast at the different data rate of 10 Gbps in "10Gbase-SR Ethernet," $T = 100$ ps.[16] However, both modulation methods have the same signal constellation. Thus, for the FM/Manchester example, only two signal-constellation points are used, $\boldsymbol{x}_0 = [1 \ \ -1]'$ and $\boldsymbol{x}_1 = [-1 \ 1]'$, as shown in Figure 1.18, although the basis functions differ from the previous example. The resulting modulated waveforms appear in Figure 1.19 and correspond to the write currents that are applied to the head in the FM storage system.(Additional scaling may be used to adjust either the FM or Ethernet transmit power/energy level to some desired value, but this simply scales all possible constellation points and transmit signals by the same constant value.)

[16] In fact, Ethernet systems use 66/64 times higher symbol rate because of some overhead carried.

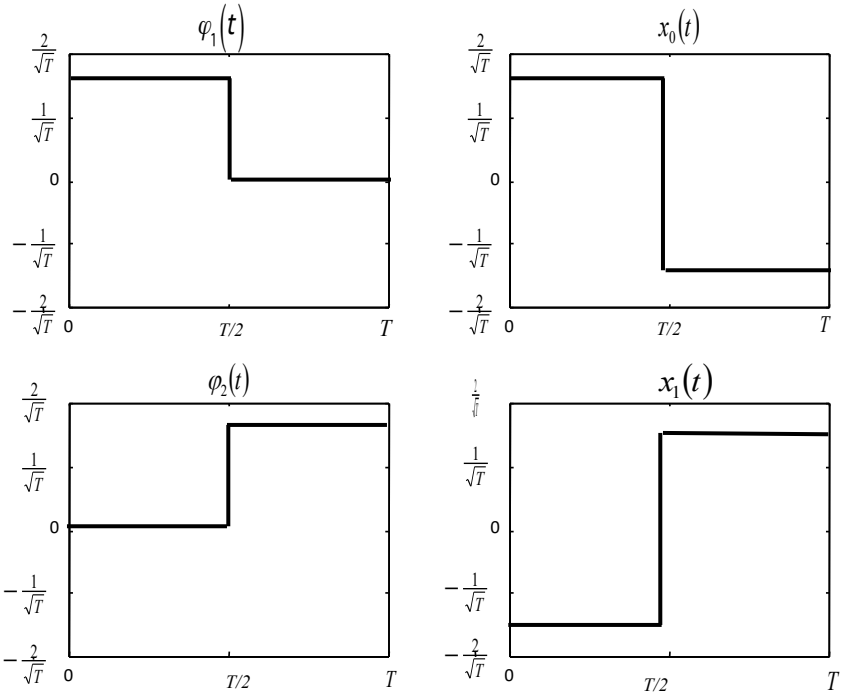


Figure 1.19: **Manchester/FM ("Ethernet") basis functions and waveforms.**

The common vector space representation (i.e. signal constellation) of the "Ethernet/FM" and "BPSK" examples allows the performance of a detector to be analyzed for either system in the same way, despite the gross differences in the overall systems.

In either of the systems in Example 1.2.2, a more compact representation of the signals with only one basis function is possible. (As an exercise, the reader should conjecture what this basis function could be and what the associated signal constellation would be.) Appendix A considers the construction of a minimal set of basis functions for a given set of modulated waveforms, which is often called "Gram-Schmidt" decomposition.

Two more examples briefly illustrate vector components x_n that are not necessarily binary-valued.

EXAMPLE 1.2.3 **[Short-Haul non-coherent Fiber Ethernet 802.3bm - 2B1Q]**[17]

This transmission system over fiber-optic cable uses $M = 4$ waveforms with one basis function $N = 1$. Thus, the system transmits $b = 2$ bits of information per T seconds of channel use. The basis function is roughly approximated[18] by $\varphi_1(t) = \sqrt{\frac{1}{T}}\text{sinc}(\frac{t}{T})$, where $1/T = 53.125$ GHz, and $\text{sinc}(x) \triangleq \frac{\sin(\pi x)}{\pi x}$. This basis function is not time limited to the interval $[0, T]$. The associated signal constellation appears in Figure 1.20. Longer-distance fiber transmission (up to 2 km) may transmit at 1/2 this symbol rate (26.5625 GHz), so at roughly 50 Gbps in other related IEEE 802.3 "Ethernet" standards. 2 bits are transmitted using one 4-level (or "quaternary") symbol every T seconds, hence the name "2B1Q."

[17] IEEE 802.3bm is a standard that contains specifications for short-length non-coherent transmission at (roughly) 100 Gbps on each of up to 8-16 parallel channels (8 wavelengths with each having two polarizations) on up to roughly 500m of fiber . IEEE 802.3 standards also use other constellations for alternatives on longer lengths of fiber.

[18] Actually $1/\sqrt{T}\text{sinc}(t/T)$, or some other "Nyquist" pulse shape is used, see Chapter 3 on Intersymbol Interference.

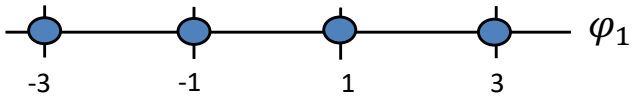


Figure 1.20: **2B1Q signal constellation.**

By contrast, telephone companies once long ago heavily transmitted the much lower data rate 1.544 Mbps "T1 Service" symmetrically on twisted pairs between the switches, or between switches and a small business (such a signal often carried twenty-four 64 kbps digital voice signals plus overhead signaling information of 8 kbps). A single very different basis function (with a much lower symbol rate) and the same constellation appear in these methods. A method, known as HDSL (High-bit-rate Digital Subscriber Lines), uses 2B1Q with $1/T=$ 392 kHz, and thus transmits a data rate of 784 kbps on each of two phone lines for a total of 1.568 Mbps (1.544 Mbps plus 24 kbps of additional HDSL management overhead). The range of this system is about 2 miles of twisted pair. Other versions of this modulation type with $M = 8, 16$, or 32 and corresponding T values to get 1.544 Mbps, and higher data rates, on a single twisted pair at different lengths that may be shorter than 2 miles. This is known as "Symmetric HDSL" or just "SDSL" or ITU standard G.991. The two very different transmission systems, fiber Ethernet and SDSL, use the same constellation and can be analyzed identically.

A second example uses two dimensions, similar to BPSK:

EXAMPLE 1.2.4 [32 Cross quadrature amplitude modulation] Consider a signal set with 32 waveforms ($M = 32$) and with 2 basis functions ($N = 2$) for transmission of 32 signals per symbol. The two BPSK-like basis functions for this "quadrature amplitude modulation" (see Section 1.3 for formal definition) are $\varphi_1(t) = \sqrt{\frac{2}{T}} \cdot \cos \frac{\pi t}{T}$ and $\varphi_2(t) = \sqrt{\frac{2}{T}} \cdot \sin \frac{\pi t}{T}$ for $0 \leq t \leq T$ and 0 elsewhere. A raw bit rate of 12.0Gbps[19] occurs with a symbol rate of $1/T = 2.4$ GHz. The signal constellation is shown in Figure 1.21; the 32 points are arranged in a rotated cross pattern, called 32 CR or 32 cross.

[19]The actual user information rate might actually 9.6 Gbps with the extra bits used for error-correction purposes as shown in Chapters 2 and 8.

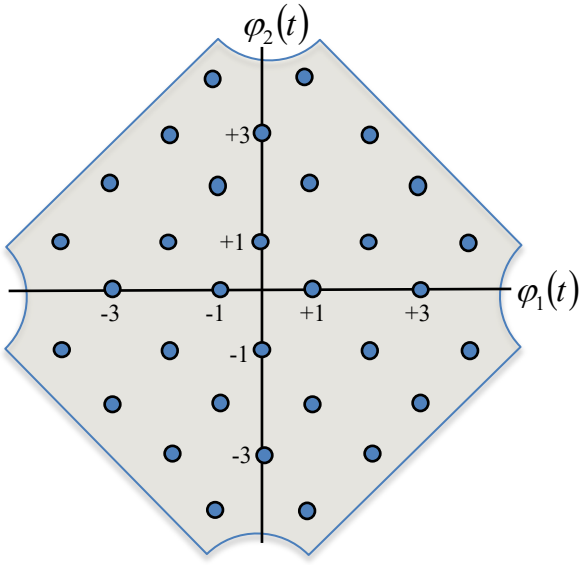


Figure 1.21: **32 Cross signal constellation.**

The last two examples also emphasize another tacit advantage of the vector representation, namely that the details of the rates and carrier frequencies in the basis-function modulation format are implicit in the normalization of the basis functions. Thus, these functions do not appear in the description of the signal constellation, allowing Section 1.1's results to apply across a wide range of data rates, symbol rates, and system bandwidths.

1.2.3 Vector-Space Interpretation of the Modulated Waveforms

This section more formally defines the inner product of two time functions and/or of two N-dimensional vectors:

Definition 1.2.2 [Inner Product] *The* **inner product** *of two (real) functions of time $u(t)$ and $v(t)$ is*

$$\langle u(t), v(t) \rangle \triangleq \int_{-\infty}^{\infty} u(t) \cdot v(t) dt \ . \tag{1.46}$$

The inner product of two (real) vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ is

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle \triangleq \boldsymbol{u}^* \boldsymbol{v} = \sum_{n=1}^{N} u_n \cdot v_n \ , \tag{1.47}$$

where $*$ denotes vector transpose (and conjugate vector transpose later when complex signals are introduced).

The two inner products in the above definition are equal under the conditions in the following theorem:

Theorem 1.2.1 **[Inner-product Invariance]** *If there exists a set of basis functions* $\varphi_n(t)$, $n = 1, ..., N$ *for some* N *such that* $u(t) = \sum_{n=1}^{N} u_n \cdot \varphi_n(t)$ *and* $v(t) = \sum_{n=1}^{N} v_n \cdot \varphi_n(t)$ *then*

$$\langle u(t), v(t) \rangle = \langle \boldsymbol{u}, \boldsymbol{v} \rangle \ . \tag{1.48}$$

where

$$\boldsymbol{u} \triangleq \begin{bmatrix} u_N \\ \vdots \\ u_1 \end{bmatrix} \ \text{and} \ \boldsymbol{v} \triangleq \begin{bmatrix} v_N \\ \vdots \\ v_1 \end{bmatrix} \quad . \tag{1.49}$$

The proof follows from

$$\begin{aligned} \langle u(t), v(t) \rangle &= \int_{-\infty}^{\infty} u(t) \cdot v(t) dt = \int_{-\infty}^{\infty} \sum_{n=1}^{N} \sum_{m=1}^{N} u_n \cdot v_m \cdot \varphi_n(t) \cdot \varphi_m(t) dt & (1.50) \\ &= \sum_{n=1}^{N} \sum_{m=1}^{N} u_n \cdot v_m \int_{-\infty}^{\infty} \varphi_n(t) \cdot \varphi_m(t) dt = \sum_{m=1}^{N} \sum_{n=1}^{N} u_n \cdot v_m \cdot \delta_{nm} = \sum_{n=1}^{N} u_n \cdot v_n & (1.51) \\ &= \langle \boldsymbol{u}, \boldsymbol{v} \rangle \ \textbf{QED}. & (1.52) \end{aligned}$$

Thus the inner product is "invariant" to the choice of basis functions and only depends on the components of the time functions along each of the basis functions. While the inner product is invariant to the choice of basis functions, the component values of the data symbols depend on basis functions. For example, for the 32CR example, one could recognize that the integral $\frac{2}{T} \int_0^T \left[2\cos\left(\frac{\pi t}{T}\right) + \sin\left(\frac{\pi t}{T}\right)\right] \cdot \left[\cos\left(\frac{\pi t}{T}\right) + 2\sin\left(\frac{\pi t}{T}\right)\right] dt = 2 \cdot 1 + 1 \cdot 2 = 4$.

Parseval's Identity is a special case (with $\boldsymbol{x} = \boldsymbol{u} = \boldsymbol{v}$) of inner-product invariance.

Theorem 1.2.2 **[Parseval's Identity]** *The following relation holds true for any modulated waveform*

$$\mathcal{E}_{\boldsymbol{x}} = \mathbb{E}\left[||\boldsymbol{x}||^2\right] = \mathbb{E}\left[\int_{-\infty}^{\infty} x^2(t) dt\right] \quad . \tag{1.53}$$

The proof follows from the previous Theorem 1.2.1 with $\boldsymbol{u} = \boldsymbol{v} = \boldsymbol{x}$

$$\begin{aligned} \mathbb{E}\left[\langle u(t), v(t) \rangle\right] &= \mathbb{E}\left[\langle \boldsymbol{x}, \boldsymbol{x} \rangle\right] & (1.54) \\ &= \mathbb{E}\left[\sum_{n=1}^{N} x_n \cdot x_n\right] & (1.55) \\ &= \mathbb{E}\left[\|\boldsymbol{x}\|^2\right] & (1.56) \\ &= \mathcal{E}_{\boldsymbol{x}} \ \textbf{QED}. & (1.57) \end{aligned}$$

Parseval's Identity implies that the average energy of a signal constellation is invariant to the basis-function choice, as long as they satisfy the orthonormality condition of Equation (1.43). As another 32CR example, the energy of the [2,1] point is $\frac{2}{T} \int_0^T \left[2\cos\left(\frac{2\pi t}{T}\right) + \sin\left(\frac{2\pi t}{T}\right)\right]^2 dt = 2 \cdot 2 + 1 \cdot 1 = 5$.

The individual basis functions themselves have a trivial vector representation; namely $\varphi_n(t)$ is represented by $\boldsymbol{\varphi_n} = [0\ 0\ , ...,\ 1\ , ...,\ 0]^*$, where the 1 occurs in the n^{th} position. Thus, the data symbol $\boldsymbol{x}_i$ has a representation in terms of the unit basis vectors $\boldsymbol{\varphi_n}$ that is

$$\boldsymbol{x}_i = \sum_{n=1}^{N} x_{in} \cdot \boldsymbol{\varphi_n} \quad . \tag{1.58}$$

The data-symbol component x_{in} can be determined as

$$x_{in} = \langle \boldsymbol{x}_i, \boldsymbol{\varphi_n} \rangle \ , \tag{1.59}$$

which, using inner-product invariance, becomes

$$x_{in} = \langle x_i(t), \varphi_n(t) \rangle = \int_{-\infty}^{\infty} x_i(t) \cdot \varphi_n(t) dt \quad n = 1, ..., N \ . \tag{1.60}$$

Thus any modulated-waveform set $\{x_i(t)\}$ can be interpreted as a vector signal constellation, with the components of any particular vector $\boldsymbol{x_i}$ given by Equation (1.60). In effect, x_{in} is the the i^{th} modulated waveform's projection on the n^{th} basis function. Appendix A's Gram-Schmidt procedure can be used to determine the minimum number of basis functions needed to represent any signal in the signal set.

1.2.4 Demodulation

As in Equation (1.60), the data symbol vector $\boldsymbol{x}$ can be recovered, component-by-component, by computing the inner product of $x(t)$ with each of the N basis functions. This recovery is called **correlative demodulation** because the modulated signal, $x(t)$, is "correlated" with each of the basis functions to determine $\boldsymbol{x}$, as Figure 1.22 illustrates.

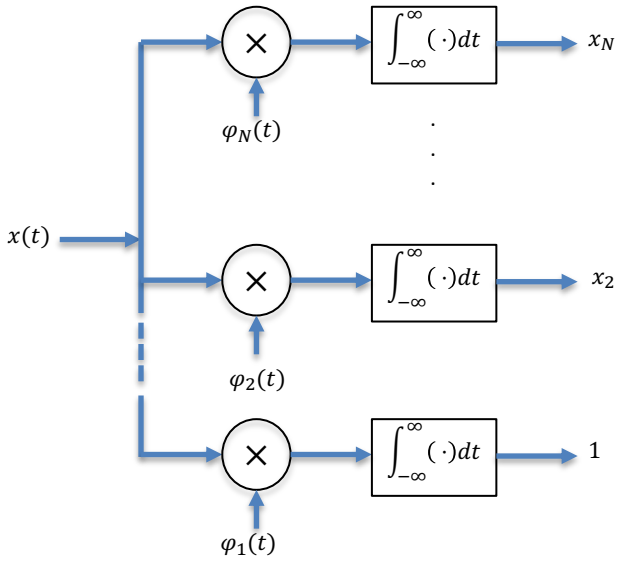


Figure 1.22: **The correlative demodulator.**

The modulated signal, $x(t)$, is first multiplied by each of the basis functions in parallel, and the multipliers' outputs then each pass to an integrator to produce a corresponding data-symbol component x_n. Practical realization of the multipliers and integrators may be difficult. Any physically implementable set of basis functions so far exists over the symbol period.[20] Then the computation of x_n alternately becomes

$$x_n = \int_0^T x(t) \cdot \varphi_n(t) dt \quad . \tag{1.61}$$

The computation in (1.61) is more easily implemented by noting that it is equal to

$$x(t) * \varphi_n(T - t)|_{t=T} \quad , \tag{1.62}$$

[20]This restriction to a finite time interval is later removed with the introduction of "Nyquist" Pulse shapes in Chapter 3, and the term "symbol period" will be correspondingly relaxed and expanded.

where $*$ indicates convolution. The signal's component x_n along the n^{th} basis function is equivalent to the convolution (filter) of the waveform $x(t)$ with a filter $\varphi_n(T-t)$ at output sample time T. Such a **matched-filter demodulator** "matches" the received signal to the corresponding modulator basis function. Figure 1.23 illustrates matched-filter demodulation.

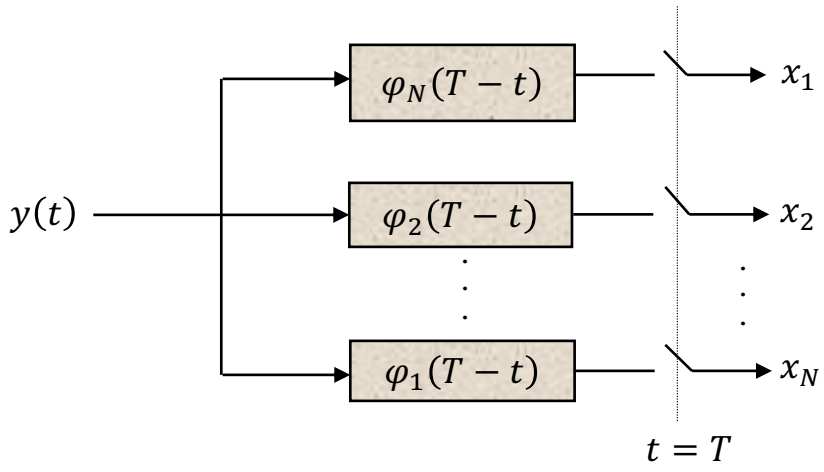


Figure 1.23: **The matched-filter demodulator.**

Figure 1.23 thus illustrates a conversion between the data symbol and the corresponding modulated waveform such that the modulated waveform can be represented by a finite (or countably infinite as $N \to \infty$) set of components along an orthonormal set of basis functions. Sections 1.1 used and 1.3 will use this concept to analyze the performance of some modulation schemes on the AWGN channel.

EXAMPLE 1.2.5 (BPSK Matched-Filter Demodulator) *Figure 1.24 illustrates the matched-filter demodulator for Example 1.2.2's BPSK basis functions.*

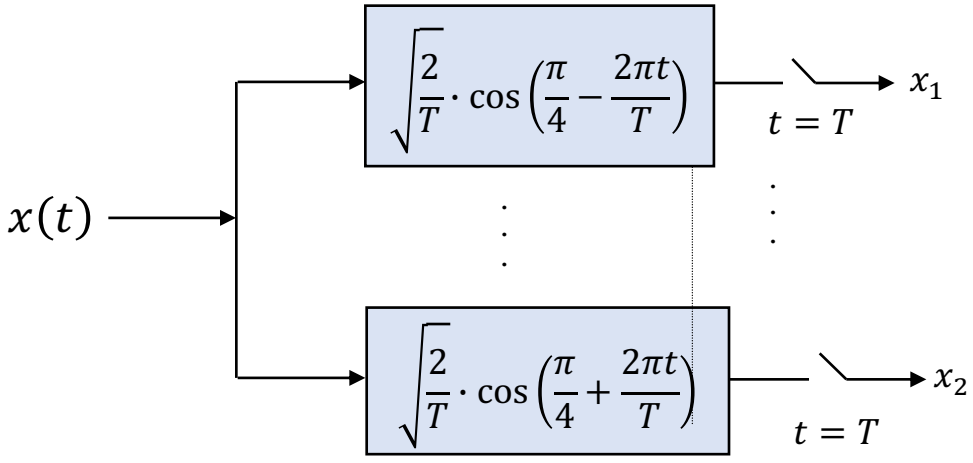


Figure 1.24: **BPSK's matched-fiilter demodulator for Example 1.2.2.**

1.2.5 MIMO Channel Basics

Multiple-Input-Multiple-Output (MIMO) channels also are vector channels. Figure 1.25 illustrates that MIMO cases have L_x spatial dimensions, and each dimension may be used temporally also. Consequently, multiple signals pass through several parallel spatial channels that presently remain free of crosstalk between the dimensions. In these simplest crosstalk-free MIMO cases, the spatially indexed MIMO basis functions need only be normalized, and need not be necessarily orthogonal to other spatial dimensions over Figure 1.25's) different parallel channels. The MIMO infrastructure ensures the orthogonality (as indicated by the parallel dashed lines through the MIMO channel). There are thus L_x "spatial" or "space-time" transmit dimensions and L_y output spatial dimensions, permitting up to $L \triangleq \min(L_x, L_y)$ MIMO channels in general. At this first introduction, $L_y = L_x$. At a basic mathematics level, "a dimension is a dimension" (space, time, frequency, or otherwise); however this text uses N as an temporal-dimension index that implies the dimensions arise from decomposing frequency-time at a single point in space while L will usually apply to dimensions generated at different points in space and time while using the same frequency. It is possible that there are N orthonormal basis functions on each of the L spatial channels, leading to an overall dimensionality of $N_{total} = L_x \cdot N$ and a more complex channel/system. More complete discussion of such larger dimensionality appears in Chapters 3 - 5, and beyond.

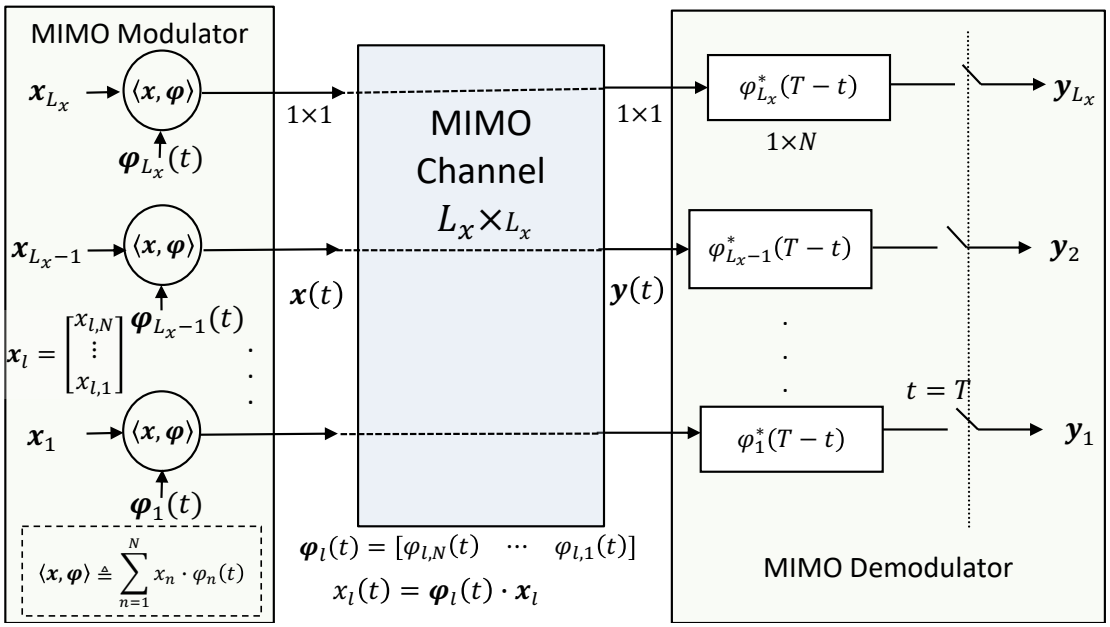


Figure 1.25: **The MIMO channel.**

The ℓ^{th} spatial component $x_\ell(t)$ has representation

$$x_\ell(t) = \sum_{n=1}^{N} \varphi_{\ell,n}(t) \cdot x_{\ell,n} = \boldsymbol{\varphi}_\ell(t) \cdot \boldsymbol{x}_\ell \;\; \forall \; \ell = 1, ..., L_x, \tag{1.63}$$

where $\boldsymbol{\varphi}_\ell$ is a $1 \times N$ temporal input basis-function vector, The vector $\boldsymbol{x}_\ell$ that may have nonzero components on all N temporal dimensions of the transmit vector signal component $\boldsymbol{x}_\ell(t)$. The overall $L_x \times 1$ transmit signal could also be written $\boldsymbol{x}(t) = \sum_{n=1}^{N} \boldsymbol{x}_n(t)$. An $L_x \times (N \cdot L_x)$ block-diagonal basis-function matrix becomes

$$\boldsymbol{\Phi}(t) = \begin{bmatrix} \varphi_{L_x}(t) & ... & \mathbf{0}^* \\ \vdots & \ddots & \vdots \\ \mathbf{0}^* & \mathbf{0}^* & \varphi_1(t) \end{bmatrix} . \tag{1.64}$$

($*$ is transpose here.) The $NL_x \times 1$ input symbol vector is

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_{L_x} \\ \vdots \\ \boldsymbol{x}_1 \end{bmatrix} \quad . \tag{1.65}$$

For the MIMO channel, the modulated signal then becomes

$$\boldsymbol{x}(t) = \boldsymbol{\Phi}(t) \cdot \boldsymbol{x} \quad . \tag{1.66}$$

The $\boldsymbol{\varphi}_\ell(t) = [\varphi_N(t) \ \ ... \ \ \varphi_1(t)]$ is often a common set of basis functions used on all the spatial dimensions[21]. The matrix of inner products is such that

$$\langle \boldsymbol{\varphi}_\ell^*(t), \boldsymbol{\varphi}_\ell(t) \rangle = I \ \ \forall \, \ell = 1, ..., L_x \quad , \tag{1.67}$$

which with presumed spatial independence essentially assures each MIMO spatial dimension's orthonormality in the case of (1.64)'s block-diagonal structure. Later chapters address the potential of crosstalk between the different spatial dimensions, where spatial independence is then no longer present.

In the simplest MIMO case,

$$\boldsymbol{\varphi}_1(t) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \varphi_1(t) \end{bmatrix} \quad , \tag{1.68}$$

while similarly all N single-nonzero component basis function vectors follow. This makes each MIMO dimension a separate channel free of crosstalk from other spatial dimensions.

An example could be a system that has L_x highly directional transmit antennas that each point at another set of L_x highly directional receive antennas. In effect, each transmit antenna has an input component $x_{\ell,n}$ of a transmit vector $\boldsymbol{x}_n$ with on the n^{th} normalized basis function vector $\boldsymbol{\phi}_\ell(t)$ that passes only to the corresponding ℓ^{th} output antenna. These types of systems are used in licensed wireless-band systems like cellular and Wi-Fi standards for wireless communication. The receivers as a set have corresponding components y_n, which can be aggregated into a channel-output vector $\boldsymbol{y}$. Similarly, L parallel wires could be used between common end points to increase speed. For instance, the IEEE 803.3z 1 Gbps Ethernet standard uses 4 parallel twisted pairs that each carry 250 Mbps of individual throughput (the actual data rate is 312.5 Mbps because an extra 25% is used for overhead that is not counted as user data in the quotation of 250 Mbps speed to user). These sets of $L = 4$ wires are often called ethernet cables, connecting with the familiar RJ45 connectors for ethernet (if one looks closely, 8 wires or 4 twisted pairs are in those connectors). Yet another example occurs in the above mentioned IEEE 802.3 ethernet fiber standards for 40 and 100 Gbps where 4 wavelengths on the same fiber (with no interference between them) each carry 1/4 of the overall data rate. Sometimes there is leakage between the channels, known as crosstalk, which is similar to the "intersymbol interference" addressed in Chapter 3, but crosstalk is better termed as "intra-symbol" interference. This topic is addressed in Section 1.5 and Chapters 4, 5, as well as later chapters. Important here is that the MIMO channel also fits into the vector-channel analysis that is common then to all forms of transmission in this book.

The inner product of (column) vector functions simply generalizes to (a superscript of * denotes transpose here)

$$\langle \boldsymbol{f}(t), \boldsymbol{g}(t) \rangle = \int_0^T \boldsymbol{f}^*(t) \cdot \boldsymbol{g}(t) dt \quad , \tag{1.69}$$

basically a sum of integrals instead of a single integral previously[22]. Inner products of the components on the (now) vector basis functions again equal the sum-of-integral inner products. This entire section could

[21] If different parallel channels had different functions, the total set would be the union of all sets as long as orthonormality is retained in creating the set of larger functions for modulation across all channels.

[22] It is sometimes convenient in vector functions also to write the inner product as $\langle \boldsymbol{f}(t), \boldsymbol{g}(t) \rangle = \text{trace} \left\{ \int_0^T \boldsymbol{f}(t) \cdot \boldsymbol{g}^*(t) dt \right\}$, which the astute reader may notice is the same for real functions. Indeed, any generalized norm can be used and the entire theory revisited for those familiar with vector spaces and norms/inner-products.

be reread with the basis vector functions replacing the scalar basis functions, and the modulated signal being a vector of transmitted time-domain waveforms $\boldsymbol{x}(t)$ that results in a vector of channel output waveforms $\boldsymbol{y}(t)$. For basis functions, MIMO orthogonality need not always apply across independent links (the independence assuring the effective equivalent of orthogonality), but usually the functions are normalized. Instead, (1.67) is a matrix of inner products. The trace of this matrix can be viewed as the inner product of the basis functions, which is more useful when later uses generalize $\boldsymbol{f}$ and/or $\boldsymbol{g}$ to themselves be matrices.

1.3 The Additive White Gaussian Noise (AWGN) Channel

Figure 1.26's AWGN is perhaps the most important, and certainly the most analyzed, continuous-time communication channel.

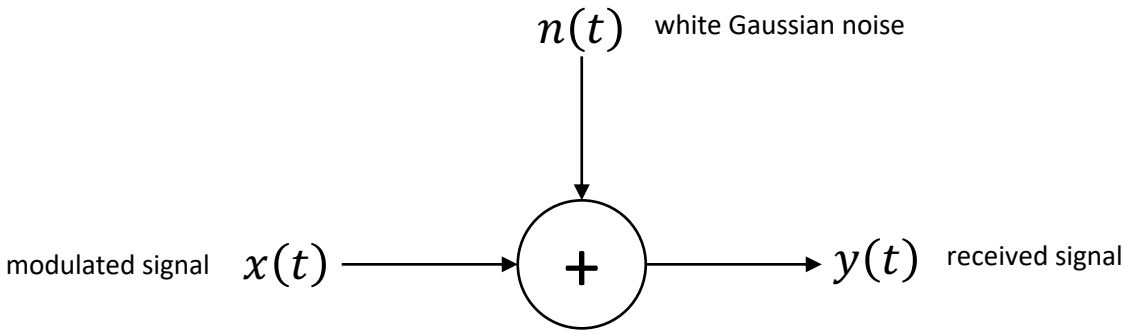


Figure 1.26: **The AWGN channel.**

The AWGN channel sums the modulated signal $x(t)$ with an uncorrelated Gaussian noise $n(t)$ to produce the received signal $y(t)$ (at the channel output or equivalently input to the receiver). The stationary[23] Gaussian noise is uncorrelated with itself (or "white") for any non-zero time offset τ, that is

$$\mathbb{E}\left[n(t) \cdot n(t-\tau)\right] = \frac{\mathcal{N}_0}{2} \cdot \delta(\tau) \quad , \tag{1.70}$$

and has zero mean, $\mathbb{E}[n(t)] = 0$. For the MIMO case, white noise generalizes to identically distributed, independent AWGNs added to each output dimension[24]. "Colored" noise is considered in Subsection 1.3.7.

The assumption of white Gaussian noise is valid in the very common situation where the noise is predominantly determined by analog front-end receiver's thermal noise. Such noise has a power spectral density given by the Boltzman equation:

$$S_n(f) = \frac{hf}{e^{\frac{hf}{kT}} - 1} \approx kT \text{ for "small" } f < 10^{12} \text{ 1 THz} \quad , \tag{1.71}$$

where Boltzman's constant is $k = 1.38 \times 10^{-23}$ Joules/degree Kelvin, Planck's constant is $h = 6.63 \times 10^{-34}$ Watt-s^2, and T is the temperature on the Kelvin (absolute) scale. This power spectral density is approximately -174 dBm/Hz ($10^{-17.4}$ mW/Hz) at room temperature (larger in practice). The Gaussian distribution assumption is a consequence of the addition of many small contributing noise sources, thus invoking the Central Limit Theorem[25].

This section's long AWGN development begins with Subsection 1.3.1 that shows that Section 1.2's modulation and demodulation process and consequent discrete vector-symbol transmission channel completely represents the AWGN; that is, there is no loss with respect to optimum performance even though continuous time is replaced by a discrete set of vector-symbol values. Subsection 1.3.1 also introduces the important concept of a **signal-to-noise ratio**, and its maximization, which is a recurring theme both in this text book and in good transmission design and analysis. Subsection 1.3.2 then progresses to develop many performance-analysis simplifications that are possible with the AWGN, particularly

[23] The Gaussian noise is strict sense stationary (See Appendix A for a discussion of stationarity types)

[24] All proofs in this section then generalize easily to the case where scalar x, y, and φ are generalized to vectors with the more general definition of inner product at the end of Subsection 1.2.5.

[25] The Central Limit Theorem is presumed known to the reader and basically says that the sum of many independent random variables tends towards a Gaussian distribution.

error probability bounds that are tight and depend only on distance between constellation symbol vectors and the number of nearest neighbors. These simplifications also recur throughout this book and in practical design. This leads to Subsection 1.3.3's discussion of fair comparison, a topic somewhat unique to this text and that reinforces a view of transmission that recognizes dimensionality in all its forms (often an area where area experts have different opinions because this fair comparison area is overlooked or misunderstood). Subsection 1.3.4 enumerates and evaluates many commonly encountered constellations and designs. Subsections 1.3.5 and 1.3.6 extend to complex channels. Complex symbol vectors thereafter replace previous subsections' real symbol vectors to simplify and extend analysis to channels where an exterior carrier is used to translate signals to and from an an appropriate frequency band. The use of complex arithmetic effectively makes the carrier superfluous to simplify analysis. The text will then proceed with complex signals, symbols, and various systems that process them with complex symbols replacing and/or generalizing real symbols. Subsection 1.3.7 then addresses bandlimited or filtered AWGN's and the closely related concept of "colored" (not white) additive Gaussian noise.

1.3.1 Continuous-Time AWGN Conversion to a Vector AWGN Channel

This subsection presumes the default $L_x = L_y = 1$ unless otherwise stated. In the absence of Figure 1.26's additive noise, $y(t) = x(t)$, and Subsection 1.2.4's demodulation process exactly recovera the transmitted signal. This subsection shows that for the AWGN Channel, this demodulation process provides sufficient information to determine optimally the transmitted signal. The demodulator's components $y_l \triangleq \langle y(t), \varphi_n(t) \rangle$, $n = 1, ..., N$ comprise a vector channel output, $\boldsymbol{y} = [y_1 \ , ..., \ y_N]'$ that is equivalent for detection purposes to $y(t)$. The analysis can thus convert the continuous channel $y(t) = x(t) + n(t)$ to a discrete vector channel model,

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n} \quad , \tag{1.72}$$

where $\boldsymbol{n} \triangleq [n_1 \ n_2 \ ... \ n_N]$ and $n_j \triangleq \langle n(t), \varphi_j(t) \rangle$. The received symbol vector at the demodulator output is the sum of the modulated signal's vector equivalent and a demodulated-noise vector. However, the exact noise sample function may not be reconstructed from $\boldsymbol{n}$,

$$n(t) \neq \sum_{j=1}^{N} n_j \cdot \varphi_j(t) \triangleq \hat{n}(t) \quad , \tag{1.73}$$

or equivalently,

$$y(t) \neq \sum_{n=1}^{N} y_n \cdot \varphi_n(t) \triangleq \hat{y}(t) \quad . \tag{1.74}$$

There may exist a component of $n(t)$ that is orthogonal to the space spanned by the basis functions $\{\varphi_1(t) \ ... \ \varphi_N(t)\}$. This unrepresented noise component is

$$\breve{n}(t) \triangleq n(t) - \hat{n}(t) = y(t) - \hat{y}(t) \quad , \tag{1.75}$$

from which a lemma quickly follows:

Lemma 1.3.1 [Uncorrelated noise samples] *The AWGN Noise samples in the demodulated noise vector are independent and of equal variance* $\frac{\mathcal{N}_0}{2}$.
Proof: *Write*

$$\mathbb{E}\left[n_k n_l\right] = \mathbb{E}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} n(t) \cdot n(s) \cdot \varphi_k(t) \cdot \varphi_l(s) dt \ ds\right] \tag{1.76}$$

$$= \frac{\mathcal{N}_0}{2}\int_{-\infty}^{\infty} \varphi_k(t) \cdot \varphi_l(t) dt \tag{1.77}$$

$$= \frac{\mathcal{N}_0}{2} \cdot \delta_{kl} \quad . \textbf{QED}. \tag{1.78}$$

Section 1.1's MAP-detector development could have replaced $\boldsymbol{y}$ by $y(t)$ everywhere and the development would have proceeded identically with the tacit inclusion of the time variable t in the probability densities (and also assuming stationarity of $y(t)$ as a random process). The Theorem of Irrelevance would hold with $[\boldsymbol{y}_1\ \boldsymbol{y}_2]$ replaced by $[\hat{y}(t)\ \check{n}(s)]$, as long as the relation (1.21) holds for any pair of time instants t and s. In a non-mathematical sense, the unrepresented noise is useless to the receiver, so there is nothing of value lost in the vector demodulator, even though some of the channel output noise is not represented. The following algebra demonstrates that $\check{n}(s)$ is irrelevant:

First,

$$\mathbb{E}\left[\check{n}(s)\cdot\hat{n}(t)\right]=\mathbb{E}\left[\check{n}(s)\cdot\sum_{l=1}^{N}n_l\cdot\varphi_l(t)\right]=\sum_{l=1}^{N}\varphi_l(t)\cdot\mathbb{E}\left[\check{n}(s)\cdot n_l\right]\quad . \tag{1.79}$$

and,

$$\begin{aligned}\mathbb{E}\left[\check{n}(s)\cdot n_l\right] &= \mathbb{E}\left[(n(s)-\hat{n}(s))\cdot n_l\right] &(1.80)\\ &= \mathbb{E}\left[\int_{-\infty}^{\infty}n(s)\cdot\varphi_l(\tau)\cdot n(\tau)d\tau\right]-\mathbb{E}\left[\sum_{k=1}^{N}n_k\cdot n_l\cdot\varphi_k(s)\right] &(1.81)\\ &= \frac{\mathcal{N}_0}{2}\int_{-\infty}^{\infty}\delta(s-\tau)\cdot\varphi_l(\tau)d\tau-\frac{\mathcal{N}_0}{2}\cdot\varphi_l(s) &(1.82)\\ &= \frac{\mathcal{N}_0}{2}\cdot[\varphi_l(s)-\varphi_l(s)]=0\quad . &(1.83)\end{aligned}$$

Second,

$$\begin{aligned}\mathrm{p}_{\boldsymbol{x}|\hat{y}(t),\check{n}(s)} &= \frac{p_{\boldsymbol{x},\hat{y}(t),\check{n}(s)}}{p_{\hat{y}(t),\check{n}(s)}} &(1.84)\\ &= \frac{p_{\boldsymbol{x},\hat{y}(t)}\cdot p_{\check{n}(s)}}{p_{\hat{y}(t)}\cdot p_{\check{n}(s)}} &(1.85)\\ &= \frac{p_{\boldsymbol{x},\hat{y}(t)}}{p_{\hat{y}(t)}} &(1.86)\\ &= p_{\boldsymbol{x}|\hat{y}(t)}\quad . &(1.87)\end{aligned}$$

Equation (1.87) satisfies the theorem of irrelevance, and thus the receiver need only base its decision on $\hat{y}(t)$, or equivalently, only on the received vector $\boldsymbol{y}$. The vector AWGN Channel is equivalent to the continuous-time AWGN channel.

Rule 1.3.1 **[The Vector AWGN Channel]** *The vector AWGN channel is given by*

$$\boldsymbol{y}=\boldsymbol{x}+\boldsymbol{n} \tag{1.88}$$

and is equivalent to the channel illustrated in Figure 1.26. The noise vector $\boldsymbol{n}$ is an N-dimensional Gaussian random vector with zero mean, equal-variance, uncorrelated components in each dimension. The noise distribution is

$$p_{\boldsymbol{n}}(\boldsymbol{u})=(\pi\mathcal{N}_0)^{-\frac{N}{2}}\cdot e^{-\frac{1}{\mathcal{N}_0}\|\boldsymbol{u}\|^2}=\left(2\pi\sigma^2\right)^{-\frac{N}{2}}\cdot e^{-\frac{1}{2\sigma^2}\|\boldsymbol{u}\|^2}\quad . \tag{1.89}$$

Application of $y(t)$ to either the correlative demodulator of Figure 1.22 or to the matched-filter demodulator of Figure 1.23, generates the desired vector channel output $\boldsymbol{y}$ at the demodulator output. The following section specifies the decision process that produces an estimate of the input message, given the output $\boldsymbol{y}$, for the AWGN Channel.

For the more general MIMO case, inner products between $\boldsymbol{y}(t)$ (or $\boldsymbol{n}(t)$) and input $L_x\times 1$ basis vectors or vector-signal components $\boldsymbol{x}_n(t)$ become $L_y\times L_x$ matrices, one for each n. Later chapters address these expanded signals, but setting $L_y=L_x=1$ for now simplifies early comprehension.

1.3.1.1 Optimum Detection with the AWGN Channel

For the vector AWGN Channel in (1.88),

$$p_{\boldsymbol{y}/\boldsymbol{x}}(\boldsymbol{v}|i) = p_{\boldsymbol{n}}(\boldsymbol{v} - \boldsymbol{x}_i) \quad , \tag{1.90}$$

where $p_{\boldsymbol{n}}$ is the vector noise distribution in (1.89). Thus for the AWGN Channel, the MAP Decision Rule becomes

$$\hat{m} \Rightarrow m_i \;\; if \;\; e^{-\frac{1}{\mathcal{N}_0}\|\boldsymbol{v}-\boldsymbol{x}_i\|^2} \cdot p_{\boldsymbol{x}}(i) \geq e^{-\frac{1}{\mathcal{N}_0}\|\boldsymbol{v}-\boldsymbol{x}_j\|^2} \cdot p_{\boldsymbol{x}}(j) \;\; \forall \;\; j \neq i \quad , \tag{1.91}$$

where the common factor of $(\pi\mathcal{N}_0)^{-\frac{N}{2}}$ has been canceled from each side of (1.91). As noted earlier, if equality holds in (1.91), then the decision can be assigned to any of the corresponding messages without change in minimized error probability. The ln of (1.91) is the preferred form of the MAP Decision Rule for the AWGN channel:

Rule 1.3.2 [AWGN MAP Detection Rule]

$$\hat{m} \Rightarrow m_i \;\; if \;\; \|\boldsymbol{v} - \boldsymbol{x}_i\|^2 - \mathcal{N}_0 \cdot \ln\{p_{\boldsymbol{x}}(i)\} \leq \|\boldsymbol{v} - \boldsymbol{x}_j\|^2 - \mathcal{N}_0 \cdot \ln\{p_{\boldsymbol{x}}(j)\} \;\; \forall \;\; j \neq i \tag{1.92}$$

If the channel input messages are equally likely, the natural-log terms on both sides of (1.92) cancel, yielding the AWGN ML Detection Rule:

Rule 1.3.3 [AWGN ML Detection Rule]

$$\hat{m} \Rightarrow m_i \;\; if \;\; \|\boldsymbol{v} - \boldsymbol{x}_i\|^2 \leq \|\boldsymbol{v} - \boldsymbol{x}_j\|^2 \;\; \forall \;\; j \neq i \quad . \tag{1.93}$$

The ML detector for the AWGN channel in (1.93) has the intuitively appealing physical interpretation that the decision $\hat{m} = m_i$ corresponds to choosing the data symbol $\boldsymbol{x}_i$ that is closest, in terms of the Euclidean distance, to the vector received symbol $\boldsymbol{y} = \boldsymbol{v}$. Without noise, the received vector is $\boldsymbol{y} = \boldsymbol{x_i}$ the transmitted symbol, but the additive Gaussian noise results in a received symbol most likely in the neighborhood of $\boldsymbol{x_i}$. The noise's Gaussian distribution implies the probability of a received point decreases as the distance from the transmitted point increases.

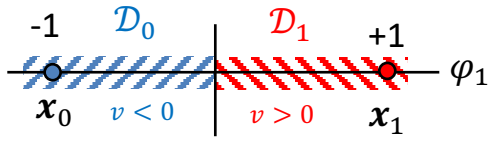


Figure 1.27: **Binary ML detector.**

As an example consider the decision regions for binary data transmission over the AWGN Channel illustrated in Figure 1.27. The ML Receiver decides $\boldsymbol{x}_1$ if $\boldsymbol{y} = \boldsymbol{v} \geq 0$ and $\boldsymbol{x}_0$ if $\boldsymbol{y} = \boldsymbol{v} < 0$. (One might have guessed this answer without need for theory.) With d defined as the distance $\|\boldsymbol{x}_1 - \boldsymbol{x}_0\|$, the decision regions are offset in the MAP detector by $\frac{\sigma^2}{d} \cdot \ln\{\frac{p_{\boldsymbol{x}}(j)}{p_{\boldsymbol{x}}(i)}\}$ with the decision boundary shifting towards the data symbol of lesser probability, as illustrated in Figure 1.28. Unlike the ML detector, the MAP detector accounts for the à priori message probabilities.

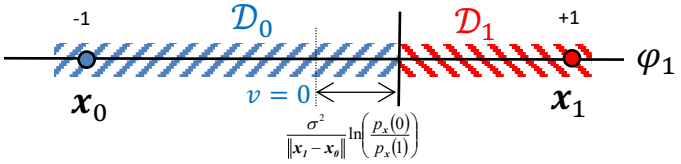


Figure 1.28: **Binary MAP detector.**

Figure 1.29 illustrates the decision region for a two-dimensional example of the QPSK[26] constellation, which uses the same basis functions as the V.32 example (Example 1.2.4), but with $M = 4$. The constellation's symbols are all assumed to be equally likely.

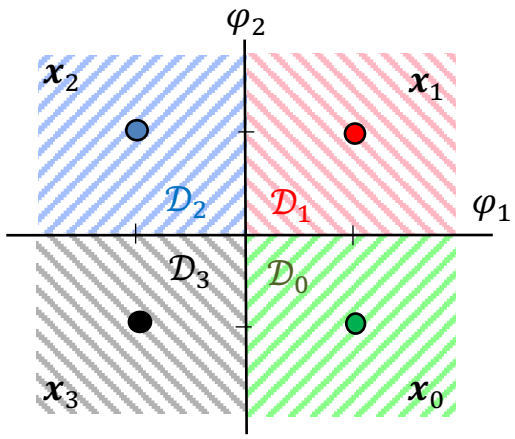


Figure 1.29: **QPSK decision regions.**

1.3.1.2 General Receiver Implementation

While the decision regions in the above examples appear simple to describe, an implementation may be more complex. This section investigates general receiver structures and the detector implementation. The MAP detector minimizes the quantity (the quantity $\boldsymbol{y}$ now replaces $\boldsymbol{v}$ averting strict mathematical notation, because probability density functions appear less often in the subsequent analysis):

$$\|\boldsymbol{y} - \boldsymbol{x}_i\|^2 - \mathcal{N}_0 \cdot \ln\{p_{\boldsymbol{x}}(i)\} \tag{1.94}$$

over the M possible messages, indexed by i. The quantity in (1.94) expands to

$$\|\boldsymbol{y}\|^2 - 2\langle \boldsymbol{y}, \boldsymbol{x}_i \rangle + \|\boldsymbol{x}_i\|^2 - \mathcal{N}_0 \cdot \ln\{p_{\boldsymbol{x}}(i)\} \quad . \tag{1.95}$$

Minimization of (1.95) can ignore the $\|\boldsymbol{y}\|^2$ term. The MAP decision rule then becomes

$$\hat{m} \Rightarrow m_i \;\; if \;\; \langle \boldsymbol{y}, \boldsymbol{x}_i \rangle + c_i \geq \langle \boldsymbol{y}, \boldsymbol{x}_j \rangle + c_j \; \forall j \; \neq i \quad , \tag{1.96}$$

where c_i is the constant (independent of $\boldsymbol{y}$)

$$c_i \triangleq \frac{\mathcal{N}_0}{2} \cdot \ln\{p_{\boldsymbol{x}}(i)\} - \frac{\|\boldsymbol{x}_i\|^2}{2} \quad . \tag{1.97}$$

[26] Quadrature Phase-Shift Keying

A system design can precompute the constants $\{c_i\}$ from the transmitted symbols $\{\boldsymbol{x_i}\}$ and their known probabilities $p_{\boldsymbol{x}}(i)$. The detector thus only needs to implement the M inner products, $\langle \boldsymbol{y}, \boldsymbol{x_i} \rangle$ $i = 0, \ldots, M-1$. When all the data symbols have the same energy ($\mathcal{E}_{\boldsymbol{x}} = \|\boldsymbol{x}_i\|^2 \ \forall \ \ i$) and are equally probable (i.e. MAP = ML), then the constant c_i is independent of i and can be eliminated from (1.96). The ML detector thus chooses the $\boldsymbol{x}_i$ that maximizes the inner product (or correlation) of the received value for $\boldsymbol{y} = \boldsymbol{v}$ with $\boldsymbol{x}_i$ over i.

There exist two common implementations of the MAP receiver in Equation (1.96). The first, shown in Figure 1.30, called a "basis detector," computes $\boldsymbol{y}$ using a matched filter demodulator. This MAP receiver computes the M inner products of (1.96) digitally (an $M \times N$ matrix multiply with $\boldsymbol{y}$), adds the constant c_i of (1.97), and picks the index i with maximum result. Finally, a decoder translates the index i into the desired message m_i. Often in practice, the signal constellation is such (see Section 1.3.6 for examples) that the max-and-decode functionality reduces to simple truncation of each received symbol-vector component.

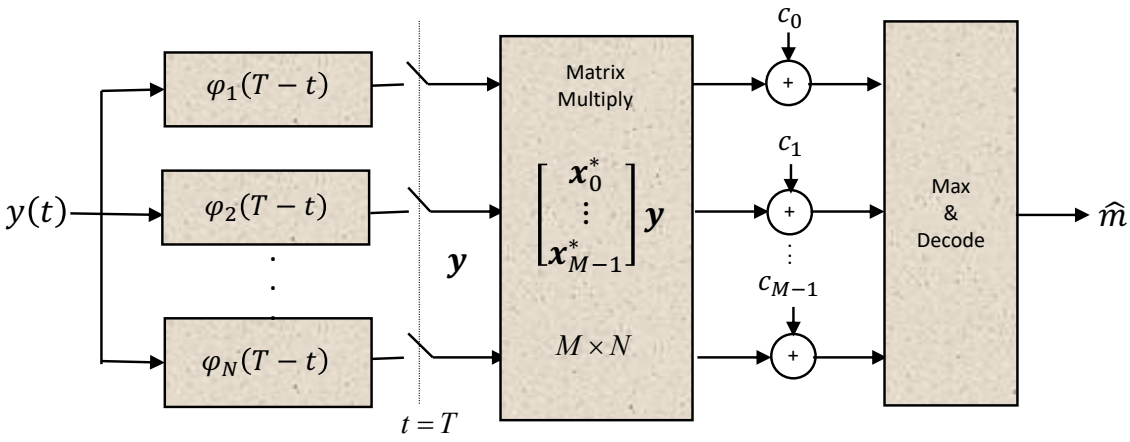


Figure 1.30: **Basis detector.**

The second demodulator form eliminates Figure 1.30's matrix multiply by exploiting directly the inner product equivalences between the discrete vectors $\boldsymbol{x_i}, \boldsymbol{y}$ and the continuous-time functions $x_i(t)$ and $y(t)$. That is

$$\langle \boldsymbol{y}, \boldsymbol{x}_i \rangle = \int_0^T y(t) \cdot x_i(t) dt = \langle y(t), x_i(t) \rangle \ . \tag{1.98}$$

Equivalently,

$$\langle \boldsymbol{y}, \boldsymbol{x_i} \rangle = y(t) * x_i(T-t)|_{t=T} \tag{1.99}$$

where $*$ indicates convolution. This type of detector is called a "signal detector" and appears in Figure 1.31.

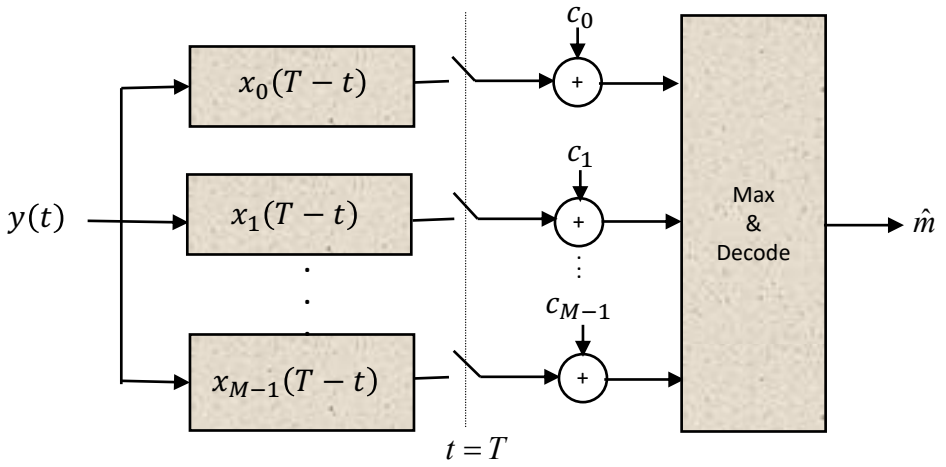


Figure 1.31: **Signal detector.**

EXAMPLE 1.3.1 **[pattern recognition as a signal detector]** Similar to Example 1.5, pattern recognition is a digital signal processing procedure that is used to detect whether a certain signal is present. A specific pattern-recognition application occurs when an aircraft/drone converts pictures of the ground into electrical signals, and these signals then permit analysis to determine the presence of certain objects in the pictures. The Gaussian noise would represent the various imperfections in the camera's accuracy and any conversion to electrical signals. This is a communication channel in disguise where the two inputs are the usual terrain of the ground and the terrain of the ground including the object to be detected. A signal detector consisting of two filters that are essentially the time reverse of each of these possible input signals, with a comparison of the outputs (after adding any necessary constants), allows detection of the presence of the object or pattern. There are many other examples of pattern recognition in voice/command recognition or authentication, facial recognition, written character scanning, and so on.

The above example/discussion illustrates that many digital-transmission-theory principles are common to other fields of digital signal processing and computer science, and Section 1.7 expands upon this theme.

1.3.1.3 Signal-to-Noise Ratio (SNR) Maximization with a Matched Filter

SNR measures system-performance as the ratio of signal power (message) to unwanted noise power. A discrete (continuous) channel's output SNR is defined as the ratio of the received signal's energy (power) to the mean-square noise value (power). The AWGN's SNR will be the same for both continuous- and discrete-time. Figure 1.32's matched filter satisfies the **SNR maximization property**, which the following theorem summarizes:

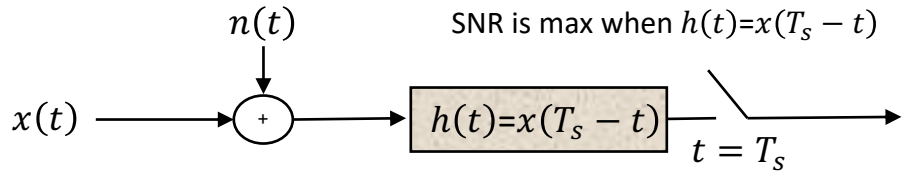


Figure 1.32: **SNR maximization by matched filter.**

Theorem 1.3.1 **[SNR Maximization]** *For the system shown in Figure 1.32, the filter*[27] $h(t)$ *that maximizes the signal-to-noise ratio at sample time* T_s *is given by the matched filter* $h(t) = x(T_s - t)$.

Proof: Compute the SNR at sample time $t = T_s$ as follows.

$$\text{Signal Energy} = [x(t) * h(t)|_{t=T_s}]^2 \tag{1.100}$$

$$= \left[\int_{-\infty}^{\infty} x(t) \cdot h(T_s - t)\, dt\right]^2 = [\langle x(t), h(T_s - t)\rangle]^2 \quad . \tag{1.101}$$

The sampled noise at the matched filter output has energy or mean-square

$$\text{Noise Energy} = E\left[\int_{-\infty}^{\infty} n(t) \cdot h(T_s - t)dt \int_{-\infty}^{\infty} n(s) \cdot h(T_s - s)ds\right] \tag{1.102}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \frac{N_0}{2} \cdot \delta(t - s) \cdot h(T_s - t) \cdot h(T_s - s)dtds \tag{1.103}$$

$$= \frac{N_0}{2}\int_{-\infty}^{\infty} h^2(T_s - t)dt \tag{1.104}$$

$$\tag{1.105}$$

$$= \frac{\mathcal{N}_0}{2}\|h\|^2 \quad . \tag{1.106}$$

The signal-to-noise ratio, defined as the ratio of the signal power in (1.101) to the noise power in (1.106), equals

$$SNR = \frac{2}{\mathcal{N}_0} \cdot \frac{[\langle x(t), h(T_s - t)\rangle]^2}{\|h\|^2} \quad . \tag{1.107}$$

The "Cauchy-Schwarz Inequality" states that

$$[\langle x(t), h(T_s - t)\rangle]^2 \leq \|x\|^2\|h\|^2 \tag{1.108}$$

with equality if and only if $x(t) = k \cdot h(T_s - t)$, where k is some arbitrary constant. Thus, by inspection, (1.107) is maximized over all choices for $h(t)$ when $h(t) = x(T_s - t)$. The filter $h(t)$ is "matched" to $x(t)$, and the corresponding maximum SNR (for any k) is

$$\text{SNR}_{max} = \frac{2}{\mathcal{N}_0} \cdot \|x\|^2 \quad . \tag{1.109}$$

An example use of the matched-filter's SNR-maximization property occurred earlier in Example 1.6's time-delay estimation, where a single matched filter processes the signal and a sampling device looks for the maximum same output.

1.3.2 Error Probability for the AWGN Channel

This section discusses average error-probability computation when the optimum detector incorrectly detects the transmitted message on an AWGN channel. From the previous section, the AWGN channel is equivalent to a vector channel with output given by

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n} \quad . \tag{1.110}$$

The computation of P_e often assumes that the inputs $\boldsymbol{x}_i$ are equally likely, or $p_{\boldsymbol{x}}(i) = \frac{1}{M}$. Under this assumption, the optimum detector is the ML detector, which has decision rule

$$\hat{m} \Rightarrow m_i \;\; if \;\; \|\boldsymbol{v} - \boldsymbol{x}_i\|^2 \leq \|\boldsymbol{v} - \boldsymbol{x}_j\|^2 \;\; \forall \;\; j \neq i \quad . \tag{1.111}$$

The P_e associated with this rule depends on the signal constellation $\{x_i\}$ and the noise variance $\frac{\mathcal{N}_0}{2}$. Two general invariance theorems in Subsection 1.3.2.1 facilitate the computation of P_e. The exact P_e,

$$P_e = \frac{1}{M} \cdot \sum_{i=0}^{M-1} P_{e/i} \tag{1.112}$$

$$= 1 - \frac{1}{M} \cdot \sum_{i=0}^{M-1} P_{c/i} \tag{1.113}$$

may be difficult to compute, so convenient and accurate bounding procedures in Subsections 1.3.2.2 through 1.3.2.4 can alternately approximate P_e.

1.3.2.1 AWGN Invariance to Rotation and Translation

The symbol constellation's orientation with respect to the coordinate axes and to the origin does not affect the P_e of the ML detector on the AWGN. This result follows because (1) the error depends only on relative distances between symbols in the symbol constellation, and (2) AWGN is spherically symmetric in all directions. First, the ML receiver's error probability is invariant to any rotation of the signal constellation, as summarized in the following theorem:

> **Theorem 1.3.2** [Rotational Invariance] *If all the data symbols in a symbol constellation are rotated by an orthogonal transformation, that is $\breve{\boldsymbol{x}}_i \leftarrow Q\boldsymbol{x}_i$ for all $i = 0, ..., M-1$ (where Q is an $N \times N$ matrix such that $QQ^* = Q^*Q = I$), then the ML receiver's error probability remains unchanged on an AWGN channel.*

Proof: First, an AWGN remains statistically equivalent after rotation by Q^*: A rotated Gaussian random vector is $\breve{\boldsymbol{n}} = Q^*\boldsymbol{n}$. $\breve{\boldsymbol{n}}$ is Gaussian since a linear combination of Gaussian random variables remains a Gaussian random variable. A Gaussian random vector is completely specified by its mean and covariance matrix: The mean is $\mathbb{E}[\breve{\boldsymbol{n}}] = 0$ since $\mathbb{E}[n_i] = 0, \; \forall \, i = 0, \ldots, N-1$. The covariance matrix is $\mathbb{E}[\breve{\boldsymbol{n}}\breve{\boldsymbol{n}}^*] = Q'\mathbb{E}[\boldsymbol{n}\boldsymbol{n}^*]Q = \frac{\mathcal{N}_0}{2} I$. Thus, $\breve{\boldsymbol{n}}$ is statistically equivalent to $\boldsymbol{n}$. The channel output for the rotated signal constellation is now $\breve{\boldsymbol{y}} = \breve{\boldsymbol{x}} + \boldsymbol{n}$ as illustrated in Figure 1.33. The corresponding decision rule is based on the distance from the received symbol vector $\breve{\boldsymbol{y}}$ to the rotated constellation symbol vector(s) $\breve{\boldsymbol{x}}_i$.

$$\|\breve{\boldsymbol{y}} - \breve{\boldsymbol{x}}_i\|^2 = (\breve{\boldsymbol{y}} - \breve{\boldsymbol{x}}_i)^* (\breve{\boldsymbol{y}} - \breve{\boldsymbol{x}}_i) \tag{1.114}$$

$$= (\boldsymbol{v} - \boldsymbol{x}_i)^* QQ * (\boldsymbol{y} - \boldsymbol{x}_i) \tag{1.115}$$

$$= \|\boldsymbol{y} - \boldsymbol{x}_i\|^2 \quad , \tag{1.116}$$

where $\boldsymbol{y} = \boldsymbol{x} + Q^*\boldsymbol{n}$. Since $\breve{\boldsymbol{n}} = Q^*\boldsymbol{n}$ has the same distribution as $\boldsymbol{n}$, and the distances measured in (1.116) are the same as in the original unrotated symbol constellation, the ML detector for the rotated constellation is the same as the ML detector for the original (unrotated) constellation in terms of all distances and noise variances. Thus, the error probability must be identical. **QED**.

An example of the QPSK constellation appears in Figure 1.29, where $N = 2$. With Q as a 45^o rotation matrix,

$$Q = \begin{bmatrix} \cos\frac{\pi}{4} & \sin\frac{\pi}{4} \\ -\sin\frac{\pi}{4} & \cos\frac{\pi}{4} \end{bmatrix} \quad , \tag{1.117}$$

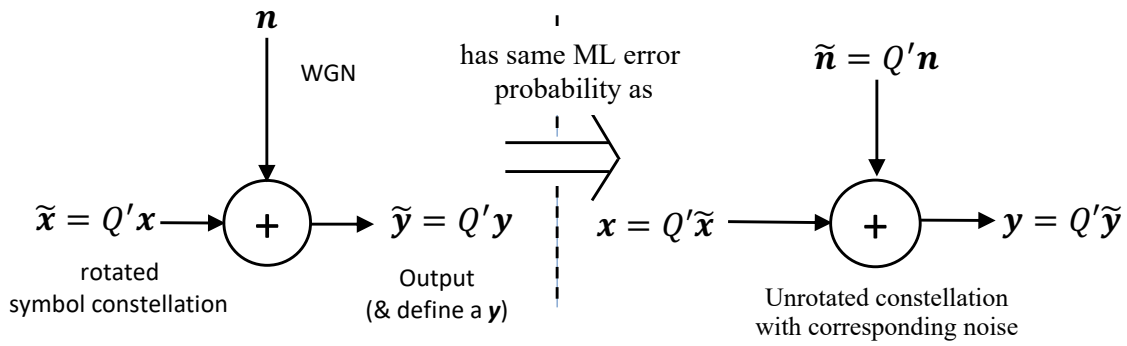


Figure 1.33: **Rotational invariance with AWGN.**

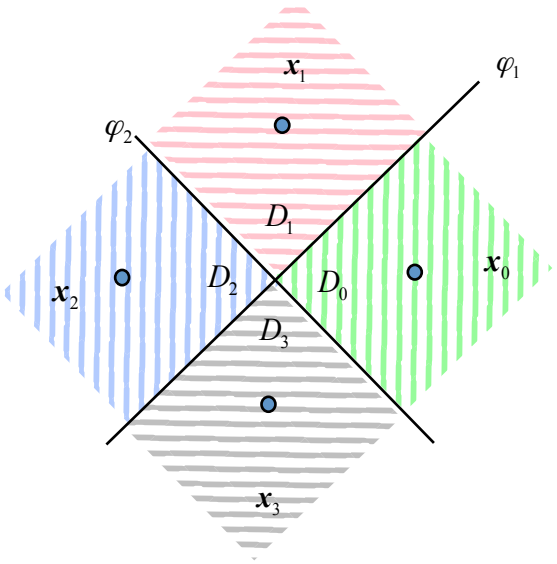


Figure 1.34: **QPSK rotated by 45^o.**

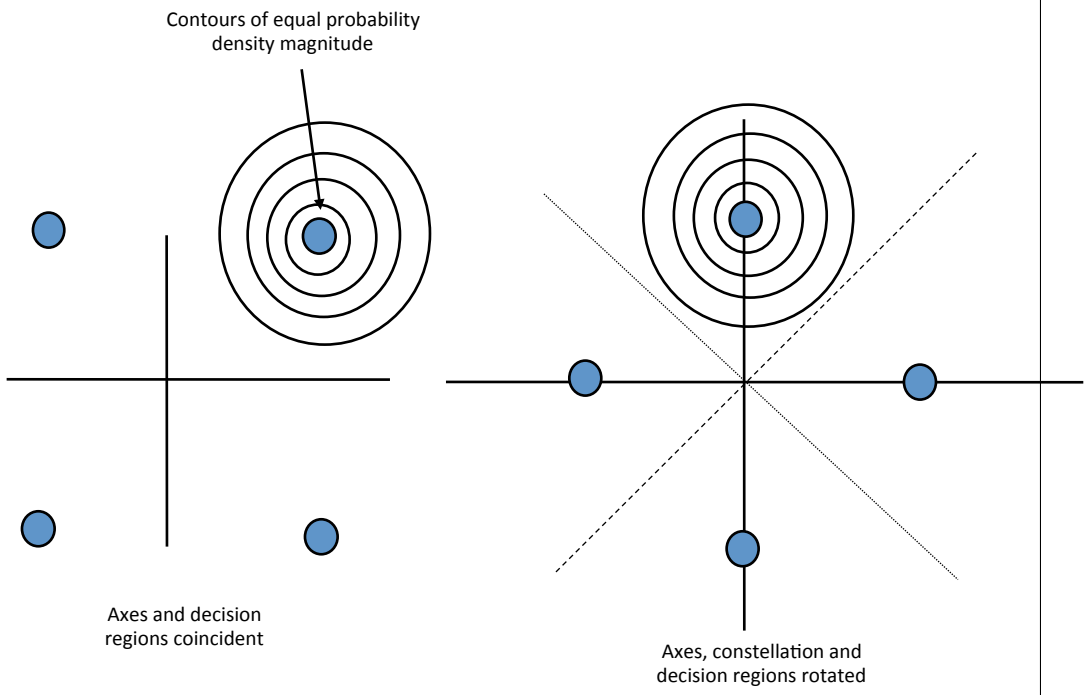


Figure 1.35: **Rotational invariance summary.**

then Figure 1.34 shows the rotated constellation and decision regions. From Figure 1.34, clearly the rotation has not changed the detection problem and has only changed the labeling of the axes, effectively giving another equivalent set of orthonormal basis functions. Since rotation does not change the squared length of any data symbols, the average energy also remains unchanged. The invariance depends on the noise components being uncorrelated with one another, and being of equal variance, as in (1.78); for other noise correlations (i.e., $n(t)$ not white, see Section 1.3.7), rotational invariance does not necessarily hold. Figure 1.35 summarizes rotational invariance. All three constellations in Figures 1.34 and 1.35 have identical P_e when used with identical AWGN.

Error probability is also invariant to translation by a constant vector amount for the AWGN, because again P_e depends only on relative distances, and the noise remains unchanged.

Theorem 1.3.3 [Translational Invariance] *If all the data symbols in a signal constellation are translated by a constant vector amount, that is* $\bar{\boldsymbol{x}}_i \leftarrow \boldsymbol{x}_i - \boldsymbol{a}$ *for all* $i = 0, ..., M-1$*, then the ML error probability remains unchanged on an AWGN channel.*

Proof: Note that the constant vector $\boldsymbol{a}$ is common to both $\boldsymbol{y}$ and to $\boldsymbol{x}$, and thus subtracts from $\|(\boldsymbol{v} - \boldsymbol{a}) - (\boldsymbol{x}_i - \boldsymbol{a})\|^2 = \|\boldsymbol{v} - \boldsymbol{x}_i\|^2$, so (1.111) remains unchanged. **QED.**

An important use of the Theorem of Translational Invariance is a constellation's **minimum energy translate**:

Definition 1.3.1 [Minimum-Energy Translate] *A constellation's* **minimum energy translate** *is obtained by subtracting the constant vector* $E[\boldsymbol{x}]$ *from each data symbol.*

To show that the minimum energy translate has the minimum energy among all possible translations of the signal constellation, the average energy of the translated signal constellation is written as

$$\begin{aligned}\mathcal{E}_{\boldsymbol{x}-\boldsymbol{a}} &= \sum_{i=0}^{M-1} \|\boldsymbol{x}_i - \boldsymbol{a}\|^2 \cdot p_{\boldsymbol{x}}(i) \qquad (1.118)\\ &= \sum_{i=0}^{M-1} \left[\|\boldsymbol{x}_i^2\| - 2\langle \boldsymbol{x_i}, \boldsymbol{a}\rangle + \|\boldsymbol{a}\|^2\right] \cdot p_{\boldsymbol{x}}(i)\\ &= \mathcal{E}_{\boldsymbol{x}} + \|\boldsymbol{a}\|^2 - 2\langle E\{\boldsymbol{x}\}, \boldsymbol{a}\rangle \qquad (1.119)\end{aligned}$$

From (1.119), the energy $\mathcal{E}_{\boldsymbol{x}-\boldsymbol{a}}$ is minimized over all possible translates $\boldsymbol{a}$ if and only if $\boldsymbol{a} = E\{\boldsymbol{x}\}$, so

$$\min_{\boldsymbol{a}} \mathcal{E}_{\boldsymbol{x}-\boldsymbol{a}} = \sum_{i=0}^{M-1} \left[\|\boldsymbol{x}_i - E\{\boldsymbol{x}\}\|^2 \cdot p_{\boldsymbol{x}}(i)\right] = \mathcal{E}_{\boldsymbol{x}} - [E(\boldsymbol{x})]^2 \quad . \qquad (1.120)$$

Thus, as transmitter energy (or power) is often a quantity to be preserved, the designer can always translate the signal constellation by $E\{\boldsymbol{x}\}$, to minimize the required energy without affecting performance. (However, there may be practical reasons, such as complexity and synchronization, where some designs avoid this translation.)

1.3.2.2 Union Bounding

Specific examples of calculating P_e appear in the next two subsections. This subsection calculates this error-probability upper bound for N-dimensional binary ($M = 2$) symbols.

Figure 1.27 illustrated binary-symbol decision regions for $N = 1$ dimension on an AWGN channel. If the system has two N-dimensional symbols, a the decision region still bisects the line between those two symbol values. Then ML-detector error probability is the probability that the noise vector $\boldsymbol{n}$'s component on the line connecting the two data symbols is greater than half the distance along this line. In this case, the noisy received vector $\boldsymbol{y}$ lies in the incorrect decision region, resulting in an error. Since the noise is white Gaussian, its projection in any dimension, in particular on the line segment that connects the two data symbols, has variance $\sigma^2 = \frac{\mathcal{N}_0}{2}$, as Theorem 1.3.2's proof. Thus,

$$P_e = P\{\langle \boldsymbol{n}, \boldsymbol{\phi}\rangle \geq \frac{d}{2}\} \quad , \qquad (1.121)$$

where $\boldsymbol{\phi}$ is a unit norm vector along the line between $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ and $d \triangleq \|\boldsymbol{x}_0 - \boldsymbol{x}_1\|$. This error probability is

$$\begin{aligned}P_e &= \int_{\frac{d}{2}}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{1}{2\sigma^2}u^2} du\\ &= \int_{\frac{d}{2\sigma}}^{\infty} \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{u^2}{2}} du \qquad (1.122)\\ &= Q\left[\frac{d}{2\sigma}\right] \quad . \qquad (1.123)\end{aligned}$$

The Q-function is defined in Appendix A, but basically computes the indefinite integral in Equation (1.122). As $\sigma^2 = \frac{\mathcal{N}_0}{2}$, (1.123) can also be written

$$P_e = Q\left[\frac{d}{\sqrt{2\mathcal{N}_0}}\right] \quad . \qquad (1.124)$$

Minimum Distance: Every signal constellation has an important characteristic known as the minimum distance:

Definition 1.3.2 [Minimum Distance, $d_{\min}$] *The* **minimum distance** *for a constellation with symbol vectors* $\boldsymbol{x} \triangleq \{\boldsymbol{x}_i\}_{i=0,\ldots,M-1}$, *is* $d_{\min}(\boldsymbol{x})$ *and measures the smallest separation between any two different constellation symbol (or "codeword") values. The argument* $(\boldsymbol{x})$ *is often dropped when the specific signal constellation is obvious from the context, thus leaving*

$$d_{\min} \triangleq \min_{i \neq j} \|\boldsymbol{x}_i - \boldsymbol{x}_j\| \quad \forall\, i, j \quad . \tag{1.125}$$

Equation (1.123) is useful in the following theorem's proof of ML-detector error-probability bound for any constellation with M data symbols (on the AWGN Channel):

Theorem 1.3.4 [Union Bound] *An error probability bound for the ML detector on the AWGN Channel, with an* M*-point constellation and minimum distance* $d_{\min}$, *is*

$$P_e \leq (M-1) \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.126}$$

Proof: The Union Bound defines an "error event" ε_{ij} as the event where the ML detector chooses $\hat{\boldsymbol{x}} = \boldsymbol{x}_j$ while $\boldsymbol{x}_i$ is the correct transmitted data symbol. The conditional error probability given that $\boldsymbol{x}_i$ was transmitted is then

$$P_{e/i} = P\{\varepsilon_{i0} \cup \varepsilon_{i1} \ldots \cup \varepsilon_{i,i-1} \cup \varepsilon_{i,i+1} \cup \ldots \cup \varepsilon_{i,M-1}\} = P\{\bigcup_{\substack{j=0 \\ (j \neq i)}}^{M-1} \varepsilon_{ij}\} \quad . \tag{1.127}$$

Because the error events in (1.127) are mutually exclusive (meaning if one occurs, the others cannot), the probability of the union is the sum of the probabilities, and also bounded by the sum of the noise-component error events (which are not necessarily mutually exclusive because the noise might be so large as to have components in multiple directions to be larger than half the distance)

$$P_{e/i} = \sum_{\substack{j=0 \\ (j \neq i)}}^{M-1} P\{\varepsilon_{ij}\} \leq \sum_{\substack{j=0 \\ (j \neq i)}}^{M-1} P_2(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad , \tag{1.128}$$

where

$$P_2(\boldsymbol{x}_i, \boldsymbol{x}_j) \triangleq P\{\ \boldsymbol{y} \text{ is closer to } \boldsymbol{x}_j \text{ than to } \boldsymbol{x}_i\ \} \quad , \tag{1.129}$$

because

$$P\{\varepsilon_{ij}\} \leq P_2(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad . \tag{1.130}$$

As illustrated in Figure 1.36, $P\{\varepsilon_{ij}\}$ is the probability the received vector $\boldsymbol{y}$ lies in the shaded decision region for $\boldsymbol{x}_j$ given the symbol $\boldsymbol{x}_i$ was transmitted.

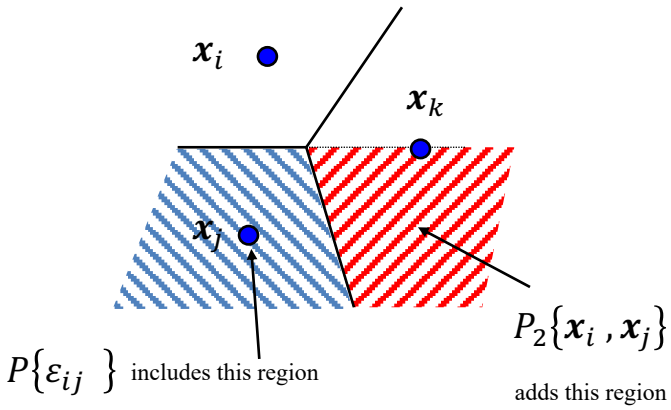


Figure 1.36: **Error probability regions.**

The incorrect decision region for the probability $P_2(\boldsymbol{x}_i, \boldsymbol{x}_j)$ includes part (shaded red in Figure 1.36) of the region for $P\{\varepsilon_{ik}\}$, which illustrates the inequality in Equation (1.130). Thus, the union bound overestimates $P_{e/i}$ by summing the results of integrating pairwise on possibly overlapping half-planes. Using the result in (1.123),

$$P_2(\boldsymbol{x}_i, \boldsymbol{x}_j) = Q\left[\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{2\sigma}\right] \quad . \tag{1.131}$$

Substitution of (1.131) into (1.128) results in

$$P_{e/i} \leq \sum_{\substack{j=0 \\ (j\neq i)}}^{M-1} Q\left[\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{2\sigma}\right] \quad , \tag{1.132}$$

and thus averaging over all transmitted symbols

$$P_e \leq \sum_{i=0}^{M-1} \sum_{\substack{j=0 \\ (j\neq i)}}^{M-1} Q\left[\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{2\sigma}\right] \cdot p_{\boldsymbol{x}}(i) \quad . \tag{1.133}$$

$Q(x)$ is monotonically decreasing in x, and thus since $d_{\min} \leq \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$,

$$Q\left[\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{2\sigma}\right] \leq Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.134}$$

Substitution of (1.134) into (1.133), and recognizing that $d_{\min}$ is not a function of the indices i or j, one finds the desired result

$$P_e \leq \sum_{i=0}^{M-1} (M-1) \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \cdot p_{\boldsymbol{x}}(i) = (M-1) \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.135}$$

QED.

Since the constellation contains M symbols, the factor $M-1$ equals the maximum number of neighboring symbols that can be at distance $d_{\min}$ from any particular symbol.

Examples The union bound can be tight (or exact) in some cases, but it is not always a good approximation to the actual P_e, especially when M is large. Two examples for $M = 8$ show situations where the union bound is a poor approximation to the actual probability of error. These two examples also naturally lead to the "nearest neighbor" bound of the next subsection.

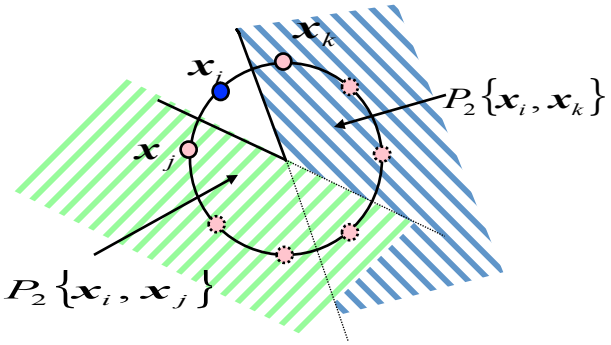


Figure 1.37: **8 Phase Shift Keying (8PSK).**

EXAMPLE 1.3.2 **[8PSK]** The constellation in Figure 1.37 is often called "eight phase" or "8PSK". For the maximum likelihood detector, the 8 decision regions correspond to sectors bounded by straight lines emanating from the origin that bisect the circle's arcs between symbols. The union bound for 8PSK is

$$P_e \leq 7Q\left[\frac{\sqrt{\mathcal{E}_{\boldsymbol{x}}}\cdot\sin(\frac{\pi}{8})}{\sigma}\right] \quad , \tag{1.136}$$

and $d_{\min} = 2\sqrt{\mathcal{E}_{\boldsymbol{x}}}\cdot\sin(\frac{\pi}{8})$.
Figure 1.38 magnifies the detection region for one of the 8 data symbols.

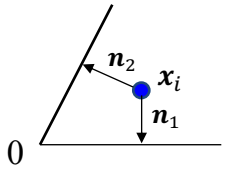


Figure 1.38: **8PSK P_e bounding.**

By symmetry the analysis would proceed identically, no matter which point is chosen, so $P_{e/i} = P_e$. An error can occur if the AWGN component along either of the two directions shown is greater than $d_{\min}/2$. These two events are not mutually exclusive, although the variance of the noise along either vector (with unit vectors along each defined as $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$) is σ^2. Thus,

$$\begin{aligned} P_e &= P\{(\| < \boldsymbol{n}, \boldsymbol{\phi}_1 > \| > \frac{d_{\min}}{2}) \bigcup (\| < \boldsymbol{n}, \boldsymbol{\phi}_2 > \| > \frac{d_{\min}}{2})\} && (1.137)\\ &\leq P\{(n_1 > \frac{d_{\min}}{2})\} + P\{(n_2 > \frac{d_{\min}}{2})\} && (1.138)\\ &= 2Q\left[\frac{d_{\min}}{2\sigma}\right] \quad , && (1.139) \end{aligned}$$

which is tighter than Equation (1.136)'s "union bound" on error probability. Also

$$P\{\|\boldsymbol{n}_1\| > \frac{d_{\min}}{2}\} \leq P_e \quad , \tag{1.140}$$

yielding a lower bound on P_e, thus the upper bound in (1.139) is tight. The bound in (1.139) overestimates the P_e by integrating the two half planes, which overlap as Figure 1.37 depicts. The lower bound of (1.140) only integrates over one half plane that does not completely cover the shaded region. The multiplier in front of the Q function in (1.139) equals the number of "nearest neighbors" for any one data symbol in the 8PSK constellation.

The following second example illustrates problems in applying the union bound to a 2-dimensional signal constellation with 8 or more symbols on a rectangular grid (or lattice):

EXAMPLE 1.3.3 [8AMPM] Figure 1.39 illustrates the 8-point "8AMPM" (amplitude-modulated phase modulation) constellation, or "8 Square" (8SQ) . The union bound for P_e yields

$$P_e \leq 7 \cdot Q\left[\frac{\sqrt{2}}{\sigma}\right] \quad . \tag{1.141}$$

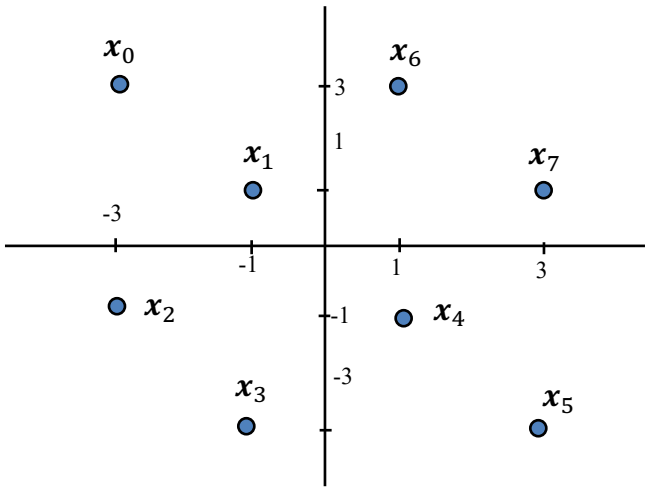


Figure 1.39: **8AMPM signal constellation.**

By rotational invariance, Figure 1.40's rotated 8AMPM constellation has the same P_e as Figure 1.39's unrotated constellation[28] The decision boundaries shown are pessimistic at the constellation's corners, so the P_e derived from them will be an upper bound. For notational brevity, let $Q \triangleq Q[d_{\min}/2\sigma]$. The probability of a correct decision for 8AMPM is

$$P_c \quad = \quad \sum_{i=0}^{7} P_{c/i} \cdot p_{\boldsymbol{x}}(i) = \sum_{i \neq 1,4} P_{c/i} \cdot \frac{1}{8} + \sum_{i=1,4} P_{c/i} \cdot \frac{1}{8} \tag{1.142}$$

[28]The bound is not supposed to be the tightest bound possible, but an accurate and easy to compute bound. By chopping the corners in Figure 1.40, some received values that ML would decode correctly are included as errors, but they have small contribution to the final result.

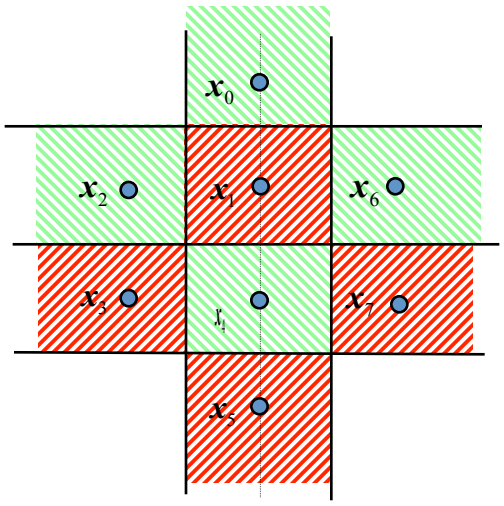


Figure 1.40: **8AMPM rotated by 45^o with decision regions.**

$$
\begin{aligned}
&> \frac{6}{8}(1-Q)(1-2Q) + \frac{2}{8}(1-2Q)^2 \text{easy to compute but not exact} && (1.143)\\
&= \frac{3}{4}\left(1-3Q+2Q^2\right) + \frac{1}{4}\left(1-4Q+4Q^2\right) && (1.144)\\
&= 1-3.25Q+2.5Q^2 \quad . && (1.145)
\end{aligned}
$$

Thus P_e is upper bounded by

$$P_e = 1 - P_c < 3.25 \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad , \tag{1.146}$$

which is tighter than the union bound in (1.141). As M increases for constellations like 8AMPM, the accuracy of the union bound degrades, since the union bound calculates P_e by pairwise error events and thus redundantly includes the probabilities of overlapping half-planes. It is desirable to produce a tighter bound. The multiplier on the Q-function in (1.146) is the average number of nearest neighbors (or decision boundaries) $= \frac{1}{4}(4+3+3+3) = 3.25$ for the constellation. The next subsection formalizes this rule of thumb as the Nearest-Neighbor Union bound (NNUB), which practicing designers often use.

1.3.2.3 The Nearest Neighbor Union Bound

The **Nearest Neighbor Union Bound** (NNUB) provides a tighter bound on a constellation's associated error probability by lowering the multiplier of the Q-function. The original union bound's factor $(M-1)$ is often too large for accurate performance prediction as in the preceding section's two examples. The NNUB requires more computation. However, it is easily approximated.

The NNUB's development uses the average number of nearest neighbors:

Definition 1.3.3 [Average Number of Nearest Neighbors] *A constellation's average number of nearest neighbors, N_e, is*

$$N_e \triangleq \sum_{i=0}^{M-1} N_i \cdot p_{\boldsymbol{x}}(i) \quad , \tag{1.147}$$

where N_i is the symbol $\boldsymbol{x}_i$'s number of neighboring constellation symbols, that is the number of other symbol vectors sharing a common decision region boundary[29] *with $\boldsymbol{x}_i$. Designers often approximate N_e as*

$$N_e \approx \sum_{i=0}^{M-1} \breve{N}_i \cdot p_{\boldsymbol{x}}(i) \quad , \tag{1.148}$$

where $\breve{N}_i$ is the number of symbols at minimum distance from $\boldsymbol{x}_i$, whence the often used name "nearest" neighbors. This approximation is often very tight and facilitates computation of N_e with complex signal constellations (i.e., coded systems - see Chapters 2, 8, and beyond).

Thus, N_e also measures the average number of decision-region boundaries surrounding any constellation symbol vector. These decision boundaries can be at different distances from any given symbol and thus might best not be called "nearest." N_e is used in the following theorem:

Theorem 1.3.5 **[Nearest Neighbor Union Bound]** *The ML detector's error probability for an M-point signal constellation on the AWGN channel with minimum distance $d_{\min}$ satisfies the bound*

$$P_e \leq N_e \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.149}$$

In the case that N_e is approximated by counting only "nearest" neighbors, then the NNUB approximates the symbol-error probability, and is not necessarily an upper bound.

Proof: For each symbol, the distance to each decision-region boundary must be at least $d_{\min}/2$. The error probability for point $\boldsymbol{x}_i$, $P_{e/i}$ satisfies the union bound

$$P_{e/i} \leq N_i \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.150}$$

Thus,

$$P_e = \sum_{i=0}^{M-1} P_{e/i} \cdot p_{\boldsymbol{x}}(i) \leq Q\left[\frac{d_{\min}}{2\sigma}\right] \sum_{i=0}^{M-1} N_i \cdot p_{\boldsymbol{x}}(i) = N_e \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.151}$$

QED.

The previous Examples 1.3.2 and 1.3.3 show that the Q-function multiplier in each case is exactly N_e for that constellation. Chapters 2 and 8 explore complex constellations with coding and larger N and then often approximate number of nearest neighbors as only those neighbors at minimum distance, so N_e is then approximately (1.148), no longer then a strict upper bound.

1.3.2.4 Alternative Performance Measures

The optimum receiver design minimizes the symbol error probability P_e. Other closely related performance measures also find use. An important useful measure is the **Bit Error Rate**. Most digital communication systems encode the message set $\{m_i\}$ into bits. Thus, their analysis computes the average number of bit errors. The average bit-error probability will depend on the constellation-symbols' specific binary labelings. The quantity $n_b(i,j)$ denotes the number of bit errors corresponding to a symbol error when the detector incorrectly chooses m_j instead of m_i, while $P\{\varepsilon_{ij}\}$ denotes the corresponding symbol-error probability.

The bit-error rate P_b obeys the following bound:

Definition 1.3.4 [Average Bit Error Rate] *The* **average bit error rate** *is*

$$P_b \triangleq \sum_{i=0}^{M-1} \sum_{\substack{j \\ j \neq i}} p_{\boldsymbol{x}}(i) \cdot P\{\varepsilon_{ij}\} \cdot n_b(i,j) \tag{1.152}$$

where $n_b(i,j)$ is the number of bit errors for the particular choice of encoder when symbol i is erroneously detected as symbol j. This quantity, despite the label using P, is not strictly a probability. It also related to the average total number of bit errors per error event.

The bit-error rate will always be approximately bounded for the AWGN in this text by:

$$\begin{aligned} P_b &\approx \sum_{i=0}^{M-1} \sum_{j=1}^{N_i} p_{\boldsymbol{x}}(i) \cdot P\{\varepsilon_{ij}\} \cdot n_b(i,j) && (1.153) \\ &\leq Q\left[\frac{d_{\min}}{2\sigma}\right] \cdot \sum_{i=0}^{M-1} p_{\boldsymbol{x}}(i) \sum_{j=1}^{N_i} n_b(i,j) \\ &\lesssim Q\left[\frac{d_{\min}}{2\sigma}\right] \cdot \sum_{i=0}^{M-1} p_{\boldsymbol{x}}(i) \cdot n_b(i) \\ &\lesssim N_b \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] && (1.154) \end{aligned}$$

where

$$n_b(i) \triangleq \sum_{j=1}^{N_i} n_b(i,j) \quad , \tag{1.155}$$

and the **Average Total Bit Errors per Error Event**, N_b, is:

$$N_b \triangleq \sum_{i=0}^{M-1} p_{\boldsymbol{x}}(i) \cdot n_b(i) \quad . \tag{1.156}$$

An expression similar to the NNUB for P_b is

$$P_b \approx N_b \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad , \tag{1.157}$$

where the approximation comes from Equation (1.153), which is an approximation because of the reduction in the sum's number of included terms over other symbols. This approximation's accuracy is good as long as those terms corresponding to distant neighbors (with distance $\geq d_{\min}$) have small value in comparison to nearest neighbors, which is a reasonable assumption for good constellation designs. The bit-error rate is sometimes a more uniform performance measure because it is independent of M and N. On the other hand, P_e is a symbol-error probability (with block length N) and can correspond to more than one bit in error (if $M > 2$) over N dimensions. Both P_e and P_b depend on the same distance-to-noise ratio (the argument of the Q function). While the notation for P_b commonly appears with a P, the bit-error rate is not a probability and could exceed unity in value in aberrant cases. A better measure that is a probability is to normalize the bit-error rate by the number of bits per symbol: Normalization of P_b produces a probability measure because it is the average number of bit errors divided by the number of bits over which those errors occur - this probability is the desired average bit-error probability:

Lemma 1.3.2 [Average bit-error probability $\bar{P}_b$.] *The* **average probability of bit error** *is defined by*

$$\bar{P}_b = \frac{P_b}{b} \quad . \tag{1.158}$$

The corresponding average total number of bit errors per bit is

$$\bar{N}_b \triangleq \frac{N_b}{b} \quad . \tag{1.159}$$

The average bit-error rate can exceed one, but the average bit-error probability never exceeds one.

Furthermore, P_e comparisons between systems with different dimensionality are not fair (for instance to compare a 2B1Q system operating at $P_e = 10^{-7}$ against a multi-dimensional design consisting of 10 successive 2B1Q dimensions decoded jointly as a single symbol also with $P_e = 10^{-7}$, the latter system really has 10^{-8} errors per dimension and so is better.) A more fair measure of symbol-error probability normalizes by the system's dimensionality (the number of symbol dimensions) to compare systems with different block lengths.

Definition 1.3.5 [Normalized Error Probability $\bar{P}_e$.] *The* **normalized error probability** *is*

$$\bar{P}_e \triangleq \frac{P_e}{N \cdot L_x} \quad . \tag{1.160}$$

The normalized average number of nearest neighbors is:

Definition 1.3.6 [Normalized Number of Nearest Neighbors] *The normalized number of nearest neighbors, $\bar{N}_e$, for a signal constellation is*

$$\bar{N}_e \triangleq \sum_{i=0}^{M-1} \frac{N_i}{N L_x} \cdot p_{\boldsymbol{x}}(i) = \frac{N_e}{N \cdot L_x} \quad . \tag{1.161}$$

Thus, the NNUB is

$$\bar{P}_e \leq \bar{N}_e \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.162}$$

EXAMPLE 1.3.4 [8AMPM] The average number of bit errors per error event for 8AMPM using the octal labeling indicated by the subscripts in Figure 1.39 is computed by

$$\begin{aligned} N_b &= \sum_{i=0}^{7} \frac{1}{8} \cdot n_b(i) \\ &= \frac{1}{8} \cdot [(1+1+2) + (3+1+2+2) + & (1.163) \\ &\quad (2+1+1) + (1+2+3) + (3+2+2+1) + & (1.164) \\ &\quad +(1+1+2) + (3+1+2) + (1+2+1)] & (1.165) \\ &= \frac{44}{8} = 5.5 \quad . & (1.166) \end{aligned}$$

Then

$$P_b \approx 5.5 \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.167}$$

Also,

$$\bar{N}_e = \frac{3.25}{2} = 1.625 \tag{1.168}$$

so that

$$\bar{P}_e \leq 1.625 \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad , \tag{1.169}$$

and

$$\bar{P}_b \approx \frac{5.5}{3} \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.170}$$

Thus the bit-error rate is somewhat higher than the normalized symbol-error probability. Careful assignment of bits to symbols can reduce the average bit-error rate slightly.

1.3.2.5 Block Error-Rate Measures

Higher-level communication-system design may use message-error counts within packets of several messages cascaded into a larger message. An entire packet may be somewhat useless if any part of it is in error. Thus, a symbol from this analysis perspective may be the entire message packet. The **"block-" or "packet-" error probability**, $P_{e,block}$, is truly identical to the symbol-error probability already analyzed as long as the entire packet is a single symbol. Generally packer-error-rate measures are examples of **Quality of Service (QoS)** measurement. QoS measures may include also include the actual data rate for a given performance level, system outages (total loss of connectivity for some time period), and the delay in modulation and encoding. This subsection focuses only on error measures.

If a packet contains B bits, each of which independently has an average bit-error probability $\bar{P}_b$, then the packet-error probability is approximately $B \cdot \bar{P}_b$. For example, if the packet-error probability is 10^{-7} and there are 125 bytes per packet, or 1000 bits per packet, then the average bit-error probability is as low as 10^{-10}. Low $P_{e,block}$ is thus a more stringent criterion on the detector performance than is $\bar{P}_b$. Nonetheless, analysis can proceed exactly as in this section. As B increases, the approximation above of $P_{e,block} = B \cdot \bar{P}_b$ can become inaccurate as per below.

Telecommunications systems often use an **erred second** to measure performance. An erred second is any second in which any bit error occurs. Obviously, fewer erred seconds is better. A given fixed number of erred seconds translates into increasingly lower average bit-error probability as the data rate increases. An **error-free second** is a second in which no error occurs. If a second contains B independent bits, then the exact probability of an error-free second is

$$P_{efs} = (1 - \bar{P}_b)^B \tag{1.171}$$

while the exact probability of an erred second is

$$P_{es} = 1 - P_{efs} = \sum_{i=1}^{B} \binom{B}{i} (1 - \bar{P}_b)^{B-i} \bar{P}_b^i \quad . \tag{1.172}$$

Dependency between bits and bit errors will change the above formulas' exact nature, but analysis often ignores any such dependency. More common in telecommunications is the derived concept of **percentage error-free seconds**, which is the percentage of seconds that are error free. Thus, if a detector has $\bar{P}_b = 10^{-7}$ and the data rate is 10 Mbps, then one might naively guess that almost every second contains errors according to $P_e = B \cdot \bar{P}_b$, and the percentage of error-free seconds is thus very low. To be exact, $P_{efs} = (1 - 10^{-7})^{10^7} = .368$, so that the link has 36.8% error-free seconds, so actually about 63% of the seconds have errors. Typically large telecommunications networks strive for **five nines** reliability, which translates into 99.999% error-free seconds. At 10 Mbps, this means that the detector has $\bar{P}_b = 1 - e^{10^{-7} \ln(.99999)} = 2.3 \cdot 10^{-12}$. At lower data rates, five nines is less stringent on the bit-error probability.

Advanced data networks, often designed for $\bar{P}_b > 10^{-12}$ operate with external "error detection and retransmission" protocols. Retransmission causes delay that may not be acceptable for the data network, at least at the physical layer. Later chapters' more sophisticated coding methods provide means to reduce error probability within delay limits through Chapter 2's redundancy, which really means somehow increasing the number of dimensions used but minimally so for system objectives. In any case, the average symbol- and bit-error probabilities are often fundamental to all other performance measures and can be used by the serious communication designer to evaluate system performance carefully.

1.3.3 General Classes of Constellations and Modulation

This subsection describes three constellation classes (and sometimes associated modulation) that abound in digital data transmission. This subsection applies throughout to any single spatial channel, or equivalently to 1 transmit antenna and 1 receive antenna in wireless; the same analysis will however identically apply for each spatial path in a MIMO system. Each of these three classes represent different geometric approaches to constellation construction. Three successive subsections examine the usual basis-function choice and constellation class and then develop corresponding general expressions for the average error probability P_e for the AWGN channel. Subsection 1.3.3.2 discusses cubic constellations (Section 1.3.6 also investigates some important extensions to the cubic constellations). Subsection 1.3.3.6 examines orthogonal constellations, while Subsection 1.3.3.7 studies circular constellations.

Constellation and modulation comparison requires measures. Modulation's cost depends upon transmitted power. Time units can translate to a number of dimensions, given a certain system bandwidth, so the energy per dimension is essentially a power measure. Given a wider bandwidth, the same time unit and power corresponds to proportionately more dimensions, but a lower power spectral density. While somewhat loosely defined, a system with symbol period T and bandwidth[30] W, has a number of temporal dimensions available for signal construction that is

$$N = 2 \cdot W \cdot T \quad \text{dimensions.} \tag{1.173}$$

Equation (1.173) holds for each MIMO spatial dimension of a signal $\boldsymbol{x}(t)$ or $\boldsymbol{y}(t)$ when $L_x \geq 1$ or $L_y \geq 1$ respectively. The reasons for this approximation will become increasingly apparent, but all this subsection's methods follow this simple rule with (reasonable and straightforward bandwidth definition). Field systems all follow Equation (1.173), or have fewer dimensions than this practical maximum, even though it may be possible to construct signal sets with slightly more dimensions theoretically. The number of dimensions in any case is a combined measure of the system's temporal resources of bandwidth and time - thus, fair comparisons normalize performance measures and energy by N (for any spatial dimension). The data rate concept thus generalizes to the **number of bits per dimension:**

Definition 1.3.7 [Average Number of Bits Per Dimension] *The* **average number of bits per dimension**, $\bar{b}$, *for a signal constellation* $\boldsymbol{x}$, *is*

$$\bar{b} \triangleq \frac{b}{N \cdot L_x} \quad . \tag{1.174}$$

The related previously defined quantity, **data rate**, *is*

$$R = \frac{b}{T} \quad . \tag{1.175}$$

Using (1.173), the temporal ($L_x = 1$)quantity

$$2 \cdot L_x \cdot \bar{b} = \frac{R}{W} \quad , \tag{1.176}$$

[30] It is theoretically not possible to have finite bandwidth and finite time extent, but in practice this can be approximated closely, as Chapter 3 will illustrate.

is the modulation and constellation method's **spectral efficiency** Transmission engineers use spectral efficiency to compare designs (how much data rate per unit of bandwidth). Spectral efficiency measures in the unit of "bits/second/Hz," which is really a measure of double the number of bits/dimension through Equation 1.176). Engineers often abbreviate the term bits/second/Hz to say bits/Hz, which is an (unfortunately) confusing term because the units are incorrect but instead abbreviate bits/s/Hz. Good designers automatically translate the verbal abbreviation bits/Hz to the correct units and interpretation, bits-per-second/Hz, or simply double the number of bits/dimension. An assumption in (1.176) is that $N = 2WT$, which is only true when $L = 1$ - that is there is no MIMO in use. When there are $L \leq \min(L_x, L_y)$ parallel spatial channels in use, the spectral efficiency will be the sum of the spectral efficiencies of all L subchannels, essentially meaning that free spatial dimensions improve spectral efficiency. Usually this MIMO number is just $L \cdot 2\bar{b}$, but $\bar{b}_{ave}$ can be defined as $1/L \cdot \sum_{L=1}^{L} \bar{b}_l$ across L spatial channels and then the MIMO spectral efficiency is $L\bar{B}_{ave}$.

As with data rate and $\bar{b}$, power $P = \mathcal{E}/T$ also generalizes to **energy per dimension**:

Definition 1.3.8 [Average Energy Per Dimension] *The* **average energy per dimension**, $\bar{\mathcal{E}}_{\boldsymbol{x}}$, *for a signal constellation* $\boldsymbol{x}$, *is*

$$\bar{\mathcal{E}}_{\boldsymbol{x}} \triangleq \frac{\mathcal{E}_{\boldsymbol{x}}}{N \cdot L_x} \quad . \tag{1.177}$$

A previously defined and related quantity is the **average power**,

$$P_{\boldsymbol{x}} = \frac{\mathcal{E}_{\boldsymbol{x}}}{T} \quad . \tag{1.178}$$

Clearly N (or $N \cdot L_x$ in the MIMO case) cannot exceed the actual number of constellation dimensions, but the constellation may require fewer dimensions for a complete representation. For example Figure 1.18's two-dimensional constellation reduces to only one one dimension by 45-degree rotation. The average power, which was also defined earlier, is a scaled quantity, but consistently defined for all constellations. In particular, the normalization of basis functions often absorbs gain into the signal constellation definition that may tacitly conceal complicated calculations based on transmission-channel impedance, load matching, and various non-trivially calculated analog effects. These effects can also be absorbed into band-limited channel models as is the case in Chapters 2, 3, 4, 10 and 11.

The energy per dimension allows the comparison of constellations with different dimensionality. The smaller the $\bar{\mathcal{E}}_{\boldsymbol{x}}$ for a given $\bar{P}_e$ and $\bar{b}$, the better the design. The concatenation of two successively transmitted $N \cdot L_x$-dimensional signals taken from the same $N \cdot L_x$-dimensional signal constellation as a single $2NL_x$-dimensional signal causes the resulting $2NL_x$-dimensional constellation, formed as a Cartesian product of the constituent NL_x-dimensional constellations, to have the same average energy per dimension as the NL_x-dimensional constellation. Thus, simple concatenation of a constellation with itself does not improve the design. However, Chapter 2 shows that careful packing of signals in increasingly larger dimensional signals sets can lead to a reduction in the required energy per dimension to transmit a given message set.

The average power is the usual measure of energy per unit time and is useful when sizing a modulator's power requirements or in determining scale constants for analog filter/driver circuits in the actual implementation. The power can be set equal to the square of the voltage over the load resistance.

The **noise energy per dimension** for an N-dimensional AWGN channel is

$$\overline{\sigma^2} = \frac{\sum_{l=1}^{N} \sigma^2}{N} = \sigma^2 = \frac{\mathcal{N}_0}{2} \quad . \tag{1.179}$$

AWGN is inherently infinite dimensional; but the theorem of irrelevance states that error-probability calculation need only consider those noise components in the same N dimensions as the constellation. (This paragraph also applies with $N \to N \cdot L_x$,)

The previously defined SNR can now also be written for the AWGN as

Definition 1.3.9 **[AWGN Channel SNR]** *The SNR is*

$$SNR = \frac{\bar{\mathcal{E}}_{\boldsymbol{x}}}{\sigma^2} \tag{1.180}$$

The UB and NNUB show that a constellation's AWGN performance depends on the minimum distance between any two of its symbol vectors . Increasing a particular constellations intra-symbol distance increases the constellation's average energy per dimension. The "Constellation Figure of Merit [3] combines the energy per dimension and the minimum-distance measures:

Definition 1.3.10 **[Constellation Figure of Merit - CFM]** *The constellation figure of merit, $\zeta_{\boldsymbol{x}}$ is*

$$\zeta_{\boldsymbol{x}} \triangleq \frac{\left(\frac{d_{\min}}{2}\right)^2}{\bar{\mathcal{E}}_{\boldsymbol{x}}} \quad , \tag{1.181}$$

a unit-less quantity, defined only when $\bar{b} \geq 1$.

The CFM $\zeta_{\boldsymbol{x}}$ measures constellation quality for AWGN-channel use. A higher CFM $\zeta_{\boldsymbol{x}}$ generally results in better performance. The CFM should only be used to compare systems with equal numbers of bits per dimension $\bar{b} = b/(NL_x)$, but can be used to compare systems of different dimensionality.

A different measure, known as the "energy per bit," measures performance in systems with low average bit rate of $\bar{b} \leq 1$ (see Chapter 10).

Definition 1.3.11 **[Energy Per Bit]** *A constellation's energy per bit, $\mathcal{E}_b$ is:*

$$\mathcal{E}_b = \frac{\mathcal{E}_{\boldsymbol{x}}}{b} = \frac{\bar{\mathcal{E}}_{\boldsymbol{x}}}{\bar{b}} \quad . \tag{1.182}$$

This measure is only defined when $\bar{b} \leq 1$ and has no meaning in other contexts. *Energy/bit may have intuitive appeal to sustainability interests, but it can hide abuse of other system resources like bandwidth or time, which may lead to overall contradiction; however it is a good measure when $\bar{b} \leq 1$.*

Definition 1.3.12 **[margin]** *A transmission system's* **margin** *at error probability P_e is the amount by which the the Q-function argument can reduce while retaining P_e.*

Transmission designers often quote margins to specify a confidence level against unusual (basically non-stationary) noise and/or signal-attenuation increases. Chapters 2 and 4 investigate margin further, while an example appears here.

EXAMPLE 1.3.5 **[Margin in DSL]** Digital Subscriber Line systems deliver 100's of kilobits to Gigabit data rates [31] over telephone lines and use sophisticated adaptive modulation systems as in Chapters 4 and 5. The two modems are located at the two telephone-line ends, edge (where fiber ends) and customer premise. DSLs error probabilities follow the NNUB

[31] Higher speeds of course occur on shorter line lengths generally.

expression $N_e \cdot Q(d_{\min}/2\sigma)$. Telephone-line intermittent noise-variance changes can be unpredictable because they sense everything from other phone lines' signals to radio signals to refrigerator doors and fluorescent and other lights as noise. DSL standards consequently mandate a 6dB margin at $P_e = 10^{-7}$ to help the customer enjoy stable service. This 6 dB essentially allows performance to be degraded by a combined factor of 4 in increased noise before some more significant corrective action need occur.

1.3.3.1 Fair Comparisons

When $L_x = 1$, two transmission systems' fair comparison requires consideration of the following 5 parameters:

1. data rate $R = b/T$,
2. power $\mathcal{E}_x/T$,
3. total bandwidth W,
4. total time or symbol period T, and
5. probability of error $\bar{P}_b$ (or P_e).

(For MIMO systems with $L_x 1$, the number of parallel channels should be held the same also - equivalently compare for each/any spatial dimension by itself.)

Any fair comparison thus holds 4 of the 5 parameters constant while varying the 5th. However, a simplification can be achieved with dimensionality as the normalizer instead of W and T. In this case, a fair comparison uses

1. bits per dimension $\bar{b}$,
2. energy per dimension $\bar{\mathcal{E}}_x$, and
3. error probability per dimension (or $\bar{P}_e$).

Any two of these 3 can be held constant and the 3rd compared. Transmission history is replete with examples of engineers (who should have known better) not keeping 4 of the 5 parameters, or the simpler 2 of the 3 normalized parameters, constant before comparing the last. The dimensional-normalization simplidiwa fair comparison. The earlier CFM $\zeta_{\boldsymbol{x}}$ presumes $\bar{b}$ fixed and then specifies the ratio of $d^2_{\min}$ to $\bar{\mathcal{E}}_{\boldsymbol{x}}$, essentially holding $\bar{\mathcal{E}}_{\boldsymbol{x}}$ fixed and looking at $\bar{P}_e$ (equivalent to $d_{\min}$ if nearest neighbors are ignored on the AWGN). The normalization essentially prevents an excess of symbol period or bandwidth from letting one modulation method appear better than another, tacitly including the third and fourth parameters (bandwidth and symbol period) from the parameter list in a fair comparison.

The CFM – when it is well defined – permits fair comparison if the $\bar{P}_e$ is held constant, because essentially this ratio is of a function of $\bar{b}$ and $\bar{\mathcal{E}}_{\boldsymbol{x}}$ (so in effect $\zeta_{\boldsymbol{x}}$ holds $\bar{b}$ constant relative to normalized energy). However, it is best in general to use the 3 quantities and directly hold 2 fixed while comparing the third.

1.3.3.2 Cubic Constellations

Cubic constellations find common use on simple data communication channels. Some cubic-constellation examples appear in Figure 1.41 for $N = 1$, 2, and 3. A cubic constellation maps $N = b$ bits sequentially into their corresponding basis-vector components in N successive dimensions. All constellation dimensions have the same scaling. Cubic constellations may have any translation or rotation in the N-dimensional space they occupy.

The simplest cubic constellation ($(L_x = 1)$appears in Figure 1.41, where $N = b = \bar{b} = 1$. This constellation is known as "binary signaling", since only two possible signals are transmitted using one basis function $\varphi_1(t)$. Several binary examples follow.

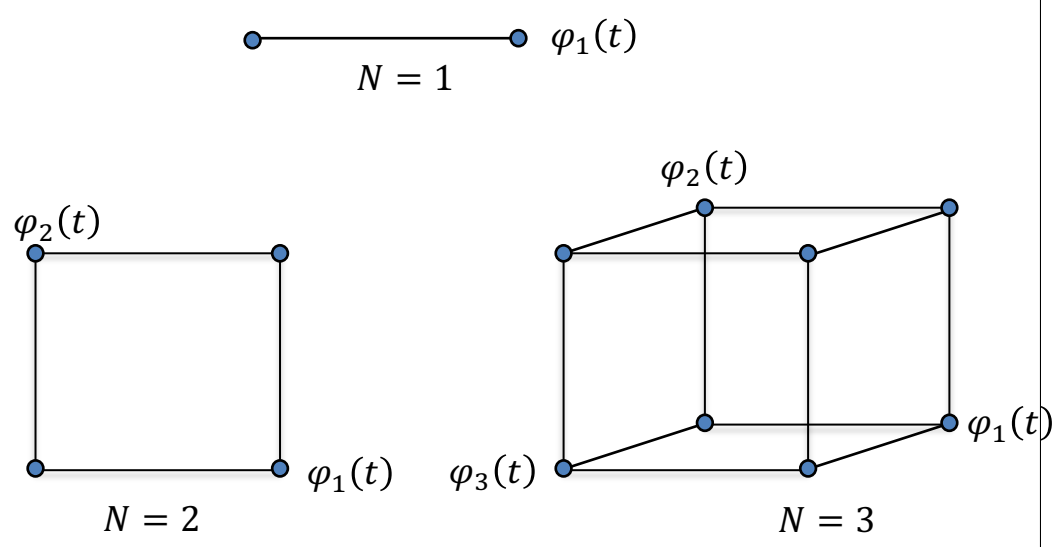


Figure 1.41: **Cubic constellations for $N = 1,\ 2,$ and 3.**

1.3.3.3 Binary Antipodal Constellations

Binary antipodal constellations have two possible $\boldsymbol{x} = x_1$ values that are equal in magnitude but opposite in sign, e.g. $x_1 = \pm\frac{d}{2}$. As for all binary signaling methods, the average error probability is

$$P_e = P_b = Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.183}$$

The CFM for binary antipodal signaling equals $\zeta_{\boldsymbol{x}} = (d/2)^2/[(d/2)^2] = 1$.

Particular binary-antipodal examples differ only in their basis-function choices, $\varphi_1(t)$, and have the same analysis. These basis functions may include "Nyquist" pulse-shaping waveforms to avoid intersymbol interference, as in Chapter 3. Besides the time-domain shaping, the basis function $\varphi_1(t)$ specifies the resultant modulated waveform's power spectral density. Thus, different basis functions may use different bandwidths, and so fair comparison rules need application.

Definition 1.3.13 [Binary Phase Shift Keying] Binary Phase Shift Keying (BPSK) uses a sinusoidal basis function to modulate the data-symbol sequence $\{\pm\sqrt{\mathcal{E}_{\boldsymbol{x}}}\}$.

$$\varphi_1(t) = \begin{cases} \sqrt{\frac{2}{T}}\sin\frac{2\pi t}{T} & 0 \le t \le T \\ 0 & \text{elsewhere} \end{cases} \tag{1.184}$$

This representation uses the minimum number of basis functions $N = 1$ to represent BPSK, rather than $N = 2$ as in Example 1.2.2.

Definition 1.3.14 **[Bipolar (NRZ) transmission]** Bipolar signaling, also known as "baseband binary" or "Non-Return-to-Zero (NRZ)" signaling, uses a square pulse to modulate the data symbols $\{\pm\sqrt{\mathcal{E}_{\boldsymbol{x}}}\}$.

$$\varphi_1(t) = \begin{cases} \frac{1}{\sqrt{T}} & 0 \le t \le T \\ 0 & \text{elsewhere} \end{cases} \tag{1.185}$$

NRZ transmission sequences may contain long runs of the same bit result in a constant output signal, with no transitions until the bit changes. Since Chapter 6's timing recovery circuits usually require some transitions, Manchester or BPL modulation function choice with binary-antipodal constellations guarantees a transition in the middle of each bit (or symbol) period T:

Definition 1.3.15 **[Manchester Coding (Bi-Phase Level)]**/ Manchester Coding, also known as "biphase level" (BPL) or, in magnetic and optical recording, as "frequency modulation (FM)," uses a sequence of two opposite phase square pulses to modulate each data symbol. The basis function is thus:

$$\varphi_1(t) = \begin{cases} \frac{1}{\sqrt{T}} & 0 \le t < T/2 \\ -\frac{1}{\sqrt{T}} & T/2 \le t < T \\ 0 & \text{elsewhere} \end{cases} \tag{1.186}$$

The modulated signal's power spectral density is proportional to the magnitude squared of the Fourier transform $\Phi_1(f)$ of the pulse $\varphi_1(t)$. The NRZ square pulse's Fourier transform is a sinc function with zero crossings spaced at $\frac{1}{T}$ Hz. Equation (1.186)'s BPL basis function requires approximately twice the bandwidth of Equation (1.185)'s NRZ basis function, because the BPL base function's Fourier transform is a sinc function with zero crossings spaced at $\frac{2}{T}$ Hz. Similarly BPSK requires double NRZ's bandwidth. Both BPSK and BPL are "rate 1/2" since $\bar{b} = \frac{1}{2}$, and thus BPSK's spectral efficiency is 1 bit/s/Hz. This means that BPSK and BPL for the same bandwidth, permit only half the data rate of NRZ, which has a spectral efficiency of 2 bits/s/Hz, or equivalently $\bar{b} = 1$.

For the AWGN channel using binary antipodal signaling, Subsection 1.1.7's Bhattacharya Bound for this memoryless channel is with N successive uses and input messages that differ in d_H positions:

$$P\{\varepsilon_{m\tilde{m}}\} \le \prod_{n=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} e^{-\frac{1}{4\sigma^2}(y_n - x_{\tilde{m},n})^2 - \frac{1}{4\sigma^2}(y_n - x_{m,n})^2} dy_n \tag{1.187}$$

$$= \prod_{n=1}^{d_H} \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}y_n^2} dy_n \cdot e^{-\frac{\bar{\varepsilon}_{\boldsymbol{x}}}{2\sigma^2}} \tag{1.188}$$

$$= e^{-d_H \frac{\bar{\varepsilon}_{\boldsymbol{x}}}{2\sigma^2}} \tag{1.189}$$

1.3.3.4 On-Off Keying (OOK)

Direct-detection optical data transmission uses **On-Off Keying**. So does most digital circuits in "gate-to-gate," uses the same basis function as NRZ.

$$\varphi_1(t) = \begin{cases} \frac{1}{\sqrt{T}} & 0 \le t \le T \\ 0 & \text{elsewhere} \end{cases} \tag{1.190}$$

Unlike bipolar transmission, however, one of the levels for x_1 is zero, while the other is nonzero ($\sqrt{2\mathcal{E}_{\boldsymbol{x}}}$). Because of the asymmetry, this method includes a DC offset, i.e. a nonzero mean value. The CFM is

$\zeta_{\boldsymbol{x}} = .5$, and thus OOK is 3 dB inferior to any type of binary antipodal transmission. The comparison between signal constellations is $10\log_{10}\left[\zeta_{\boldsymbol{x},OOK}/\zeta_{\boldsymbol{x},NRZ}\right] = 10\log_{10}(0.5) = -3$ dB.

As for any binary signaling method, OOK has

$$P_e = P_b = Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.191}$$

1.3.3.5 Vertices of a Hypercube (Block Binary)

Binary signaling in one dimension generalizes to the corners of a hypercube in N-dimensions, hence the name "cubic constellations." The hypercubic constellations all transmit $\bar{b} = 1$ bit per dimension. For two dimensions, the most common block-binary modulation is QPSK:

Quadrature Phase Shift Keying (QPSK) QPSK's two-dimensional basis functions are

$$\varphi_1(t) = \begin{cases} \sqrt{\frac{2}{T}} \cdot \cos\frac{2\pi t}{T} & 0 \le t \le T \\ 0 & \text{elsewhere} \end{cases} \quad \text{and} \tag{1.192}$$

$$\varphi_2(t) = \begin{cases} \sqrt{\frac{2}{T}} \cdot \sin\frac{2\pi t}{T} & 0 \le t \le T \\ 0 & \text{elsewhere} \end{cases} \quad . \tag{1.193}$$

The transmitted signal is a linear combination of both an inphase (cos) component and a quadrature (sin) component. The four possible data symbol vectors are

$$[x_1\ x_2] = \left\{ \begin{array}{l} \sqrt{\frac{\mathcal{E}_{\boldsymbol{x}}}{2}} \cdot [-1\ -1]' \\ \sqrt{\frac{\mathcal{E}_{\boldsymbol{x}}}{2}} \cdot [-1\ +1]' \\ \sqrt{\frac{\mathcal{E}_{\boldsymbol{x}}}{2}} \cdot [+1\ -1]' \\ \sqrt{\frac{\mathcal{E}_{\boldsymbol{x}}}{2}} \cdot [+1\ +1]' \end{array} \right\} \quad . \tag{1.194}$$

The additional basis function does not require any extra bandwidth with respect to BPSK, and the average energy $\mathcal{E}_{\boldsymbol{x}}$ is the same. While the squared minimum distance $d^2_{\min}$ has decreased by a factor of two, the number of dimensions has doubled, thus the CFM for QPSK is $\zeta_{\boldsymbol{x}} = 1$ again, as with BPSK. Thus a fair comparison finds QPSK better than BPSK. However, QPSK's $\bar{b} = 1$, so twice as much information per dimension as BPSK. If instead $\bar{\mathcal{E}}_{\boldsymbol{x}}$ is the same for BPSK and QPSK, then $\bar{P}_e$ will be the same (same $d^2_{\min}$), but $\bar{b}_{BPSK} = \frac{1}{2} < \bar{b}_{QPSK} = 1$, so again fair comparison finds QPSK better. QPSK uses resources better than BPSK, because BPSK essentially wastes a dimension by transmitting no energy on it.

Exact performance evaluation first computes the average probability of a correct decision P_c, and then $P_e = 1 - P_c$. Analysis here is for maximum likelihood detection on the AWGN channel with equally probable messages. By constellation symmetry, $P_{c|i}$ is identical $\forall\ i = 0, \ldots, 3$.

$$P_c = \sum_{i=0}^{3} P_{c/i} \cdot p_{\boldsymbol{x}}(i) = P_{c/i} \tag{1.195}$$

$$= \left(1 - Q\left[\frac{d_{\min}}{2\sigma}\right]\right)\left(1 - Q\left[\frac{d_{\min}}{2\sigma}\right]\right) \tag{1.196}$$

$$= 1 - 2Q\left[\frac{d_{\min}}{2\sigma}\right] + \left(Q\left[\frac{d_{\min}}{2\sigma}\right]\right)^2 \quad . \tag{1.197}$$

The noise's independence in the two dimensions allows progress from (1.195) to (1.196). The probability of a correct decision requires that both noise components fall within the decision region (see Figure 1.34),

which has probability equal to (1.196)'s product of probabilities. Thus

$$P_e = 1 - P_c \tag{1.198}$$

$$= 2Q\left[\frac{d_{\min}}{2\sigma}\right] - \left(Q\left[\frac{d_{\min}}{2\sigma}\right]\right)^2 < 2Q\left[\frac{d_{\min}}{2\sigma}\right] \quad , \tag{1.199}$$

where $d_{\min} = \sqrt{2\mathcal{E}_{\boldsymbol{x}}} = 2\bar{\mathcal{E}}_{\boldsymbol{x}}^{1/2}$. For reasonable error rates ($P_e < 10^{-2}$), the $\left(Q\left[\frac{d_{\min}}{2\sigma}\right]\right)^2$ term in (1.199) is negligible, and the bound on the right, which is also the NNUB, is tight. With a "reasonable" mapping of bits to data symbols (e.g. the Gray code $0 \to -1$ and $1 \to +1$), QPSK's bit-error probability is then $\bar{P}_b = \bar{P}_e$. QPSK's P_e is twice BPSK's P_e, but $\bar{P}_e$ is the same for both.

Block Binary For hypercubic signal constellations with $N \geq 3$, the symbols are the vertices of a hypercube centered on the origin. The error probability generalizes to

$$P_e = 1 - \left(1 - Q\left[\frac{d_{\min}}{2\sigma}\right]\right)^N < NQ\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.200}$$

where $d_{\min} = 2\bar{\mathcal{E}}_{\boldsymbol{x}}^{1/2}$. The basis functions usually satisfy $\varphi_n(t) = \varphi(t - nT')$, where $\varphi(t)$ is (1.185)'s square pulse with T' replacing T. One hypercube symbol's transmission has a symbol period $T = NT'$. Basis-function time-scaling can retain a symbol period with $T \to T'$, but the narrower pulse will require N times the bandwidth as the original width pulses. For this case again $\zeta_{\boldsymbol{x}} = 1$. As $N \to \infty$, $P_e \to 1$. While the probability of any single dimension being correct remains constant and less than one, as N increases, the probability of all dimensions being correct decreases, and thus P_e increases.

Ignoring the higher order terms $Q^i, i \geq 2$, the average normalized error probability is approximately $\bar{P}_e \approx Q(d_{\min}/(2\sigma))$, which equals $\bar{P}_e$ for binary antipodal signaling. This example illustrates that increasing dimensionality does not always reduce the error probability unless the signal constellation has been carefully designed. As block binary constellations simply concatenate several binary transmissions, the ML receiver decodes each separately. However, with a careful selection of a subset of the transmitted symbols, it is possible to drive the probability of both a message error P_e and a bit error P_b to zero with increasing dimensionality N. This "coded hypercube constellation" requires that the bits per dimension $\bar{b} < 1$ does not exceed a fundamental AWGN-variance-dependent rate known as the "capacity," $\bar{b} \leq \bar{\mathcal{C}}$ (Chapter 8).

1.3.3.6 Orthogonal Constellations

Orthogonal constellations have dimensionality that increases linearly with M, so then $M \propto N$. Orthogonal constellation thus decrease the number of bits per dimension with N, so $\bar{b} = \frac{\log_2(M)}{N \cdot L_x} = \frac{\log_2(\alpha N)}{N \cdot L_x} \to 0$ as $N \to \infty$.

Block Orthogonal **Block orthogonal** constellations have a dimension, or basis function, for each symbol vector. The block-orthogonal constellation thus contains $M = N$ orthogonal symbols $\boldsymbol{x}_{i=0,\ldots,M-1}$, that satisfy

$$\langle x_i(t), x_j(t) \rangle = \boldsymbol{x}_i^* \boldsymbol{x}_j = \mathcal{E}_{\boldsymbol{x}} \cdot \delta_{ij} \quad . \tag{1.201}$$

Figure 1.42 shows block orthogonal constellations for $N = 2$ and 3. The signal constellation vectors are, in general,

$$\boldsymbol{x}_i = \left[\, 0 \ldots 0 \ \sqrt{\mathcal{E}_{\boldsymbol{x}}} \ 0 \ldots 0 \,\right] = \sqrt{\mathcal{E}_{\boldsymbol{x}}} \cdot \boldsymbol{\phi}_{i+1} \quad . \tag{1.202}$$

The CFM should not be used on block orthogonal signal sets because $\bar{b} < 1$, but the fair comparison of 2/3 of $\bar{b}$, $\bar{\mathcal{E}}_{\boldsymbol{x}}$, and $\bar{P}_e$ can be used, and Block Orthogonal will not compare well.

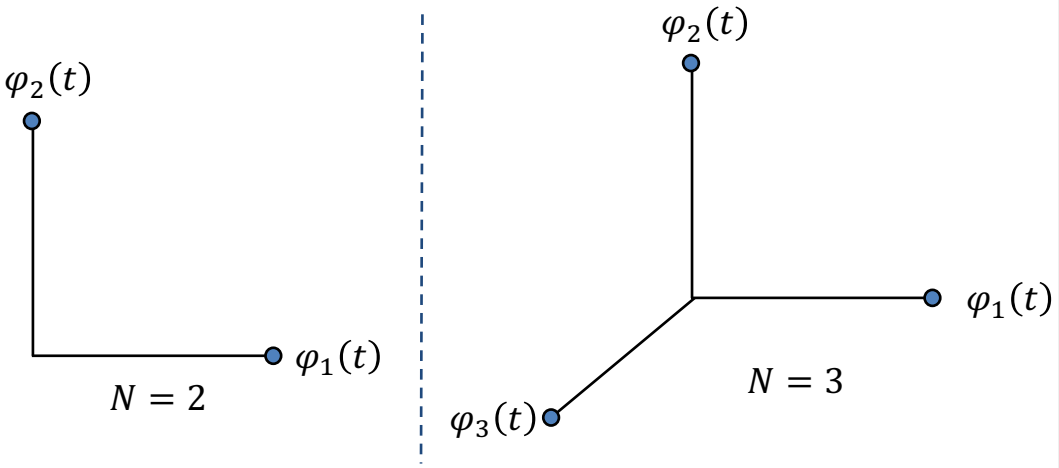


Figure 1.42: **Block orthogonal constellations for $N = 2$, and 3.**

Two block-orthogonal signaling examples follow:

Definition 1.3.16 **[Return to Zero (RZ) Signaling]** **RZ** uses the following two basis functions for Figure 1.42's two-dimensional constellation:

$$\varphi_1(t) = \begin{cases} \frac{1}{\sqrt{T}} & 0 \le t \le T \\ 0 & \text{elsewhere} \end{cases} \tag{1.203}$$

$$\varphi_2(t) = \begin{cases} \frac{1}{\sqrt{T}} & 0 \le t < T/2 \\ -\frac{1}{\sqrt{T}} & T/2 \le t < T \\ 0 & \text{elsewhere} \end{cases} \tag{1.204}$$

"Return to zero" indicates that the transmitted voltage (i.e. the signal waveform's real value) always returns to the same value at the beginning of any symbol period. Equivalently, for the same energy/dimension $\bar{\mathcal{E}}_{\boldsymbol{x}}$ and normalized error probability $\bar{P}_e$, RZ provides half the data rate of NRZ.

As for any binary signal constellation, and thus for RZ,

$$P_b = P_e = Q\left[\frac{d_{\min}}{2\sigma}\right] = Q\left[\sqrt{\frac{\mathcal{E}_{\boldsymbol{x}}}{2\sigma^2}}\right] \quad . \tag{1.205}$$

RZ is thus 3 dB inferior to binary antipodal signaling, or equivalently, uses twice NRZ's bandwidth. Thus, $\bar{b}_{RZ} = \frac{1}{2}$ while $\bar{b}_{NRZ} = 1$ for the same $\bar{P}_e$ and $\bar{\mathcal{E}}_{\boldsymbol{x}}$ in the 2-of-3 fair comparison.

Definition 1.3.17 **[Frequency Shift Keying (FSK)]** **Frequency shift keying** uses in Figure 1.42's two basis functions with $N = 2$.

$$\varphi_1(t) = \begin{cases} \sqrt{\frac{2}{T}} \sin \frac{\pi t}{T} & 0 \le t \le T \\ 0 & \text{elsewhere} \end{cases} \tag{1.206}$$

$$\varphi_2(t) = \begin{cases} \sqrt{\frac{2}{T}} \sin \frac{2\pi t}{T} & 0 \le t \le T \\ 0 & \text{elsewhere} \end{cases} \tag{1.207}$$

The term "frequency-shift" indicates that modulatotor input bit sequence of "1's" and "0's" shifts between two different respective frequencies, $1/(2T)$ and $1/T$.

As for any binary signal constellation,

$$P_b = P_e = Q\left[\frac{d_{\min}}{2\sigma}\right] = Q\left[\sqrt{\frac{\mathcal{E}_{\boldsymbol{x}}}{2\sigma^2}}\right] \quad . \tag{1.208}$$

FSK is also 3 dB inferior to binary antipodal signaling. FSK performs fundamentally the same as RZ, just with different basis functions.

FSK extends to higher-dimensional block-orthogonal constellations $N > 2$ by adding the following basis functions ($i \geq 3$):

$$\varphi_i(t) = \begin{cases} \sqrt{\frac{2}{T}} \sin \frac{i\pi t}{T} & 0 \leq t \leq T \\ 0 & \text{elsewhere} \end{cases} \tag{1.209}$$

The required bandwidth necessary to realize the additional basis functions grows linearly with N for this FSK extension. Such FSK systems do not fairly compare well with RZ or cubic constellations. They may be used for implementation simplicity when efficient dimensionality use is of less importance.

P_e Computation for Block Orthogonal Block orthogonal's P_e computation eturns to Figure 1.31's signal detector. Because all the signals are equally likely and of equal energy, this dector's constants c_i can be omitted (because they are all the same constant $c_i = c$). The MAP receiver then simplifies to

$$\hat{m} \Rightarrow m_i \;\; if \;\; \langle \boldsymbol{y}, \boldsymbol{x}_i \rangle \geq \langle \boldsymbol{y}, \boldsymbol{x}_j \rangle \; \forall \; j \neq i, \tag{1.210}$$

Block-orthogonal's constellation symmetry leads to $P_{e/i} = P_e$ or $P_{c/i} = P_c$ for all i. Analysis thus calculates only $P_c = P_{c|i=0}$, in which case the elements of $\boldsymbol{y}$ are

$$y_0 = \sqrt{\mathcal{E}_{\boldsymbol{x}}} + n_0 \tag{1.211}$$

$$y_i = n_i \quad \forall \, i \neq 0 \, . \tag{1.212}$$

For an ML decision to be message 0, then $\langle \boldsymbol{y}, \boldsymbol{x}_0 \rangle \geq \langle \boldsymbol{y}, \boldsymbol{x}_i \rangle$ or equivalently $y_0 \geq y_i \; \forall \; i \neq 0$. This decision's probability of being correct is

$$P_{c/0} = P\{y_0 \geq y_i \; \forall \; i \neq 0 | \text{ given 0 was sent}\} \quad . \tag{1.213}$$

If y_0 has a particular value v, then since $y_i = n_i \; \forall \; i \neq 0$ and since all the noise components are independent,

$$P_{c/0,y_0=v} = P\{n_i \leq v, \; \forall \;\; i \neq 0\} \tag{1.214}$$

$$= \prod_{i=1}^{N-1} P\{n_i \leq v\} \tag{1.215}$$

$$= \left[1 - Q(v/\sigma)\right]^{N-1} \quad . \tag{1.216}$$

The last equation uses the fact that the n_i are independent, identically distributed Gaussian random variables $\mathcal{N}(0, \sigma^2)$. Finally, recalling that y_0 is also a Gaussian random variable $\mathcal{N}(\sqrt{\mathcal{E}_{\boldsymbol{x}}}, \sigma^2)$.

$$P_c = P_{c/0} = \int_{-\infty}^{\infty} (\sqrt{2\pi\sigma^2})^{-1} \cdot e^{-\frac{1}{2\sigma^2}\left(v - \sqrt{\mathcal{E}_{\boldsymbol{x}}}\right)^2} \cdot \left[1 - Q(v/\sigma)\right]^{N-1} dv \quad , \tag{1.217}$$

yielding

$$P_e = 1 - \int_{-\infty}^{\infty} (\sqrt{2\pi\sigma^2})^{-1} \cdot e^{-\frac{1}{2\sigma^2}(v - \sqrt{\mathcal{E}_{\boldsymbol{x}}})^2} \cdot \left[1 - Q(v/\sigma)\right]^{N-1} dv \quad . \tag{1.218}$$

This function must be evaluated numerically using a computer, as in Figure 1.43.

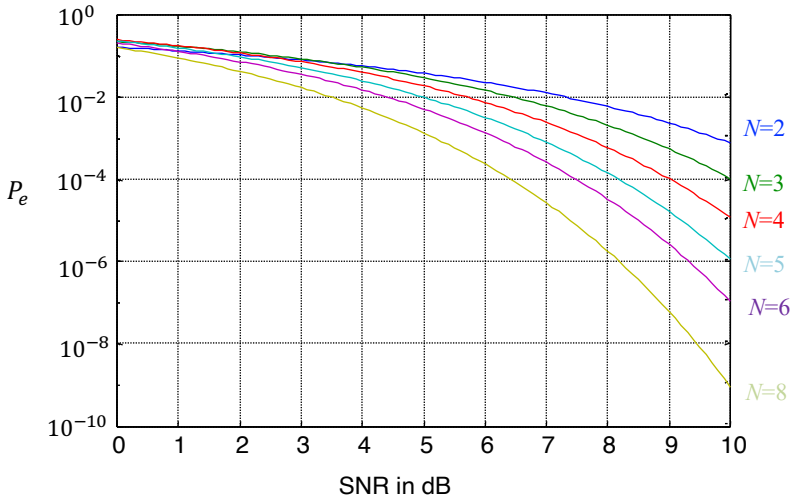


Figure 1.43: **Block-orthogonal symbol-error probability.**

A simpler calculation yields the NNUB, which also coincides with the union bound because the number of nearest neighbors $M-1$ equals the total number of neighbors to any point for block orthogonal constellations. The NNUB is given by

$$P_e \leq (M-1) \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] = (M-1) \cdot Q\left[\sqrt{\frac{\mathcal{E}_{\boldsymbol{x}}}{2\sigma^2}}\right] \quad . \tag{1.219}$$

Figure 1.43 shows that as N gets large, performance improves without increase of SNR, but at the expense of a lower $\bar{b}$. This illustrates the possibility of driving $P_e \to 0$ at the cost of diminishing data rate. Chapter 2 will show that for finite SNR, the data rate need not diminish as long as it is below a theoretically computed maximum called the capacity. Block orthogonal signaling is not necessarily a good method to obtain high reliability.

Simplex Constellation Block-orthogonal constellations have nonzero mean value, $E[\boldsymbol{x}] = (\sqrt{\mathcal{E}_{\boldsymbol{x}}}/M)[\,1\;1\;...\;1\,]$. Constellation translation by $-\mathbb{E}[\boldsymbol{x}]$ minimizes the average constellation energy without changing the average error probability. The translated constellation is the **simplex constellation**, is

$$\boldsymbol{x}_i^s = \left[\, -\frac{\sqrt{\mathcal{E}_{\boldsymbol{x}}}}{M},\; ...,\; -\frac{\sqrt{\mathcal{E}_{\boldsymbol{x}}}}{M},\; \sqrt{\mathcal{E}_{\boldsymbol{x}}} \cdot (1 - \frac{1}{M}),\; -\frac{\sqrt{\mathcal{E}_{\boldsymbol{x}}}}{M},\; ...\; -\frac{\sqrt{\mathcal{E}_{\boldsymbol{x}}}}{M} \,\right]' \quad , \tag{1.220}$$

where the term $\sqrt{\mathcal{E}_{\boldsymbol{x}}} \cdot (1 - \frac{1}{M})$ occurs in the i^{th} position. The superscript s distinguishes the simplex constellation $\{\boldsymbol{x}_i^s\}$ from the block orthogonal constellation $\{\boldsymbol{x}_i\}$ from which the simplex constellation is constructed. The simplex constellation's average energy is

$$\mathcal{E}_{\boldsymbol{x}}^s = \frac{M-1}{M}\mathcal{E}_{\boldsymbol{x}} \quad , \tag{1.221}$$

which provides significant energy savings for small M (over block orthogonal). The constellation's symbol vectors, however, are no longer orthogonal.

$$\langle \boldsymbol{x}_i^s, \boldsymbol{x}_j^s \rangle = (\boldsymbol{x}_i - E[\boldsymbol{x}])'\,(\boldsymbol{x}_j - E[\boldsymbol{x}]) \tag{1.222}$$

$$= \mathcal{E}_{\boldsymbol{x}} \cdot \delta_{ij} - \langle \mathbb{E}[\boldsymbol{x}], (\boldsymbol{x}_i + \boldsymbol{x}_j) \rangle + \frac{\mathcal{E}_{\boldsymbol{x}}}{M} \tag{1.223}$$

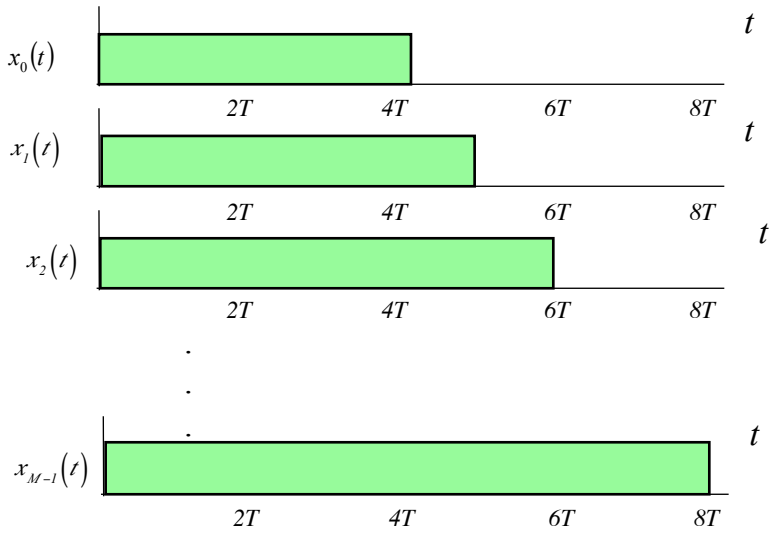


Figure 1.44: **Pulse Duration Modulation (PDM).**

$$= \mathcal{E}_{\boldsymbol{x}} \cdot \delta_{ij} - 2\frac{\mathcal{E}_{\boldsymbol{x}}}{M} + \frac{\mathcal{E}_{\boldsymbol{x}}}{M} \tag{1.224}$$

$$= \mathcal{E}_{\boldsymbol{x}} \cdot \delta_{ij} - \frac{\mathcal{E}_{\boldsymbol{x}}}{M} \quad . \tag{1.225}$$

By the Theorem of Translational Invariance 1.3.3, the simplex constellation's P_e equals the block-orthogonal constellation's P_e in (1.218), and bounded in (1.219), albeit the simplex constellations uses less energy. The simplex constellation for binary block orthogonal is binary antipodal. In general, by using a Gram-Schmidt decomposition, it is possible to reduce the simplex constellation's number of dimensions by 1 dimension.

Pulse Duration & Position Modulation Figure 1.44 shows **pulse-duration modulation's** modulated waveforms. The number of symbols M in the PDM constellation increases linearly with the number of dimensions N, as for block-orthogonal constellations. PDM finds use with some modifications in read-only optical data storage (i.e. compact disks). Such optical-disk data storage records messages by the length of a hole or "pit" burned into the storage medium. The minimum pit width ($4T$ in the figure) is much larger than the separation (T) between the different PDM signal waveforms. The PDM signal set is evidently not orthogonal.

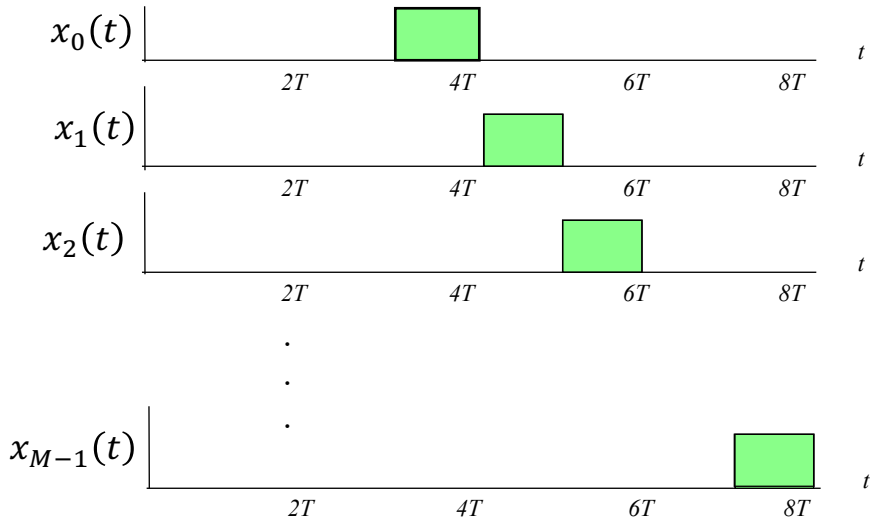


Figure 1.45: **Pulse-Position Modulation (PPM).**

A second performance-equivalent (to PDM, without regard for energy) modulated waveform set is Figure 1.45's **Pulse Position Modulation** (PPM). The PPM Constellation is a block-orthogonal constellation, which has the previously derived P_e. PDM's average energy of the PDM constellation clearly exceeds PPM's energy, which in turn exceeds that of a corresponding simplex constellation. Nevertheless, constellation energy minimization is usually not important when PDM is used; for the optical-storage example, the optical-channel physics mandate the minimum "pit" duration, and the resultant "energy" increase is not of primary concern.

Biorthogonal Constellations A variation on block-orthogonal modulation is **biorthogonal modulation**, which doubles the message-set size from $M = N$ to $M = 2N$ by including each symbol vector's negative in the constellation. From this perspective, QPSK has both a biorthogonal constellation and a cubic constellation.

The error-probability analysis for biorthogonal constellations parallels that for block-orthogonal constellations. As with orthogonal signaling, because all the signals are equally likely and of equal energy, the constants c_i in the signal detector in Figure 1.31 can be omitted, and the MAP receiver becomes

$$\hat{m} \Rightarrow m_i \;\; if \;\; < \boldsymbol{y}, \boldsymbol{x}_i > \; \geq \; < \boldsymbol{y}, \boldsymbol{x}_j > \; \forall \, j \neq i \quad . \tag{1.226}$$

By symmetry $P_{e/i} = P_e$ or $P_{c/i} = P_c$ for all i. Let $i = 0$. Then

$$y_0 = \sqrt{\mathcal{E}_{\boldsymbol{x}}} + n_0 \tag{1.227}$$

$$y_i = n_i \quad \forall \, i \neq 0 \quad . \tag{1.228}$$

If x_0 was sent, then a correct decision occurs if $\langle \boldsymbol{y}, \boldsymbol{x}_0 \rangle \geq \langle \boldsymbol{y}, \boldsymbol{x}_i \rangle$ or equivalently if $y_0 \geq |y_i| \;\; \forall \, i \neq 0$. Thus

$$P_{c/0} = P\{y_0 \geq |y_i|, \;\; \forall \, i \neq 0 \;/\; 0 \text{ was sent}\} \quad . \tag{1.229}$$

If y_0 takes on a particular value $v \in [0, \infty)$, then since the noise components n_i are i.i.d.

$$P_{c/0, y_0 = v} = \prod_{i=1}^{N-1} P\{|n_i| \leq v\} \tag{1.230}$$

$$= [1 - 2Q(v/\sigma)]^{N-1} \quad . \tag{1.231}$$

If $y_0 < 0$, then an incorrect decision ocurrs if symbol zero was sent. (The reader should visualize this constellation's decision regions). Thus

$$P_c = P_{c/o} = \int_0^\infty (\sqrt{2\pi\sigma^2})^{-1} \cdot e^{-\frac{1}{2\sigma^2}\cdot\left(v-\sqrt{\mathcal{E}_{\boldsymbol{x}}}\right)^2} [1-2Q(v/\sigma)]^{N-1}\, dv \quad , \tag{1.232}$$

yielding

$$P_e = 1 - \int_0^\infty (\sqrt{2\pi\sigma^2})^{-1} \cdot e^{-\frac{1}{2\sigma^2}\cdot\left(v-\sqrt{\mathcal{E}_{\boldsymbol{x}}}\right)^2} [1-2Q(v/\sigma)]^{N-1}\, dv \quad . \tag{1.233}$$

This function can be evaluated numerically using a computer.

Using the NNUB, which is slightly tighter than the union bound because the number of nearest neighbors is $M-2$ for biorthogonal signaling,

$$P_e \leq (M-2) \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] = 2(N-1) \cdot Q\left[\sqrt{\frac{\mathcal{E}_{\boldsymbol{x}}}{2\sigma^2}}\right] \quad . \tag{1.234}$$

1.3.3.7 Circular Constellations - M-ary Phase Shift Keying

Figures 1.29 and 1.37 provided PSK examples. In general, M-ary PSK places the data symbol vectors at equally spaced angles (or phases) around a two-dimensional circle of radius $\sqrt{\mathcal{E}_{\boldsymbol{x}}}$. PSK modulation changes only the signal phase, while the signal's amplitude or "envelope" remains constant, thus the name's origin.

PSK often finds use on channels with nonlinear amplitude distortion where modulated signals that include information content in the time varying amplitude could otherwise experience performance degradation from the nonlinear amplitude distortion. The minimum distance for M-ary PSK is given by

$$d_{\min} = 2\sqrt{\mathcal{E}_{\boldsymbol{x}}} \cdot \sin\frac{\pi}{M} = 2\sqrt{2\bar{\mathcal{E}}_{\boldsymbol{x}}} \cdot \sin\frac{\pi}{M} \quad . \tag{1.235}$$

The CFM is

$$\zeta_{\boldsymbol{x}} = 2\sin^2(\frac{\pi}{M}) \quad , \tag{1.236}$$

which is inferior to block binary signaling for any constellation with $M > 4$. The NNUB on error probability is tight and equal to

$$P_e < 2 \cdot Q\left[\frac{\sqrt{\mathcal{E}_{\boldsymbol{x}}}\sin\frac{\pi}{M}}{\sigma}\right] \quad , \tag{1.237}$$

for all M.

1.3.4 Rectangular (and Hexagonal) Signal Constellations

This subsection defaults to $L_x = 1$ on all developments. Data transmission very commonly uses rectangular constellations for transmission. Hexagonal constellations are in rare use, but help progress understanding toward Chapter 2's codes. These constellations use equally spaced symbols on translated one- or two-dimensional vector space known as a lattice. This study also introduces and re-uses some basic concepts, namely revisit of the previously defined SNR, and the new concepts of coding gain, its constituent fundamental and shaping gains, the continuous approximation, and the peak-to-average power ratio. This section's constellations are largely the foundation for all this text's subsequent developments.

Subsection 1.3.4.1 studies pulse amplitude modulation (PAM), while Subsection 1.3.4.2 studies quadrature amplitude modulation (QAM). Subsection 1.3.4.3 discusses lattice-based constellations and additional measures of constellation performance.

1.3.4.1 Pulse Amplitude Modulation (PAM)

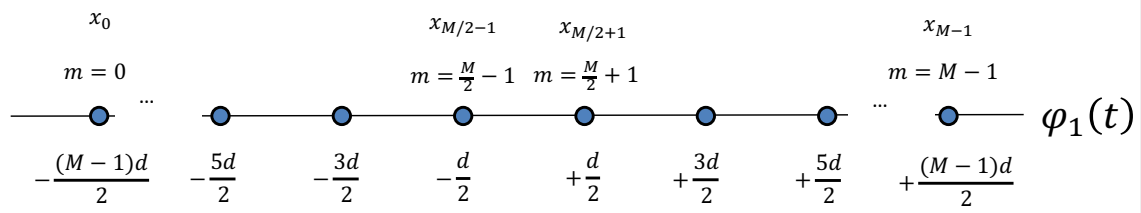


Figure 1.46: **PAM constellation.**

Pulse amplitude modulation, or amplitude shift keying (ASK), uses a one-dimensional constellation with $M = 2^b$ symbols with b as a positive integer. Figure 1.46 illustrates the PAM constellation, which is a subset of lowest-energy symbols from a scaled-by-d integer grid $\mathbb{Z}$, offset from 0 by $d/2$. The basis function can be any unit-energy function, but often $\varphi_1(t)$ is

$$\varphi_1(t) = \frac{1}{\sqrt{T}} \text{sinc}\left(\frac{t}{T}\right) \tag{1.238}$$

or another "Nyquist" pulse shape (see Chapter 3). The data-symbol amplitudes are

$$\boldsymbol{x} \in \left\{ \pm\frac{d}{2}, \pm\frac{3d}{2}, \pm\frac{5d}{2}, ..., \pm\frac{(M-1)d}{2} \right\} , \tag{1.239}$$

and all input levels are equally likely. The minimum distance between symbols in a PAM constellation abbreviates as

$$d_{\text{min}} = d \quad . \tag{1.240}$$

Both binary antipodal and "2B1Q" are examples of PAM signals.

PAM's average energy is

$$\begin{aligned}
\mathcal{E}_{\boldsymbol{x}} &= \bar{\mathcal{E}}_{\boldsymbol{x}} = \frac{1}{M}(2)\sum_{k=1}^{M/2}\left(\frac{2k-1}{2}\right)^2 \cdot d^2 && (1.241)\\
&= \frac{d^2}{2M}\sum_{k=1}^{M/2}(4k^2 - 4k + 1) && (1.242)\\
&= \frac{d^2}{2M}\cdot\left[4\left(\frac{(M/2)^3}{3} + \frac{(M/2)^2}{2} + \frac{(M/2)}{6}\right) - 4\left(\frac{(M/2)^2}{2} + \frac{(M/2)}{2}\right) + \frac{M}{2}\right] && (1.243)\\
&= \frac{d^2}{2M}\cdot\left[\frac{M^3}{6} - \frac{M}{6}\right] \quad . && (1.244)\\
&= \frac{d^2}{12}\cdot\left[M^2 - 1\right] \ . && (1.245)
\end{aligned}$$

The PAM minimum distance is a function of $\mathcal{E}_{\boldsymbol{x}}$ and M:

$$d = \sqrt{\frac{12\mathcal{E}_{\boldsymbol{x}}}{M^2 - 1}} \ . \tag{1.246}$$

Finally, given distance and average energy,

$$\bar{b} = \log_2 M = \frac{1}{2}\log\left(12\frac{\bar{\mathcal{E}}_{\boldsymbol{x}}}{d^2} + 1\right) \ . \tag{1.247}$$

Figure 1.46 shows that the decision region for any PAM constellation's interior point extends over a length-d interval centered on that point. The constellation's Voronoi boundary thus extends over an interval of Md over $[-\frac{Md}{2}, \frac{Md}{2}]$. A continuous approximation (later more completely defined) for PAM assumes a uniform distribution on this interval $[-\frac{Md}{2}, \frac{Md}{2}]$ with then approximate average energy

$$\mathcal{E}_{\boldsymbol{x}} = \bar{\mathcal{E}}_{\boldsymbol{x}} \approx \int_{-M/2}^{M/2} \frac{x^2}{2(M/2)} dx = \frac{(M/2)^2}{3} = \frac{M^2 d^2}{12} \ . \tag{1.248}$$

The continuous approximation for the average energy in (1.248) does not include (1.245)'s constant term $-\frac{d^2}{12}$, which becomes negligible as M becomes large.

Since $M = 2^{\bar{b}}$, then $M^2 = 4^{\bar{b}}$, leaving alternative relations ($\bar{b} = b$ for $N = 1$) for (1.245) and (1.246)

$$\mathcal{E}_{\boldsymbol{x}} = \bar{\mathcal{E}}_{\boldsymbol{x}} = \frac{d^2}{12} \cdot \left[4^b - 1\right] = \frac{d^2}{12} \cdot \left[4^{\bar{b}} - 1\right] \quad , \tag{1.249}$$

and

$$d = \sqrt{\frac{12\mathcal{E}_{\boldsymbol{x}}}{4^b - 1}} \quad . \tag{1.250}$$

The following recursion derives from increasing the number of bits, $b = \bar{b}$, in a PAM constellation while maintaining constant minimum distance between symbols:

$$\bar{\mathcal{E}}_{\boldsymbol{x}}(b+1) = 4 \cdot \bar{\mathcal{E}}_{\boldsymbol{x}}(b) + \frac{d^2}{4} \ . \tag{1.251}$$

Thus for moderately large b, the required signal energy increases by a factor of 4 for each additional bit in the messages corresponding to the constellation. This corresponds to an increase of 6dB per bit, a measure commonly quoted by communication engineers as the required SNR increase for a transmission scheme to support an additional bit-per-dimension of information (presuming PAM is in use) at this same $d_{\min}$.

The PAM probability of correct symbol detection is

$$\begin{aligned}
P_c &= \sum_{i=0}^{M-1} P_{c|i} \cdot p_{\boldsymbol{x}}(i) && (1.252)\\
&= \frac{M-2}{M} \cdot \left(1 - 2Q\left[\frac{d_{\min}}{2\sigma}\right]\right) + \frac{2}{M} \cdot \left(1 - Q\left[\frac{d_{\min}}{2\sigma}\right]\right) && (1.253)\\
&= 1 - \left(\frac{2M-4+2}{M}\right) \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] && (1.254)\\
&= 1 - 2\left(1 - \frac{1}{M}\right) \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] && (1.255)
\end{aligned}$$

Thus, the PAM (symbol) error probability is

$$P_e = \bar{P}_e = 2\left(1 - \frac{1}{2^{\bar{b}}}\right) \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] < 2Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . \tag{1.256}$$

The average number of nearest neighbors for the constellation is $2(1 - 1/M)$; thus, the NNUB is exact for PAM. Thus

$$P_e = 2\left(1 - \frac{1}{M}\right) \cdot Q\left(\sqrt{\frac{3}{M^2 - 1} SNR}\right) \tag{1.257}$$

$b = \bar{b}$	M	$\frac{d_{min}}{2\sigma}$ for $\bar{P}_e = 10^{-6} \approx 2Q\left(\frac{d_{min}}{2\sigma}\right)$	SNR $= \frac{(M^2-1)\cdot 10^{1.37}}{3}$	SNR increase $= \frac{M^2-1}{(M-1)^2-1}$
1	2	13.7dB	13.7dB	—
2	4	13.7dB	20.7dB	7dB
3	8	13.7dB	27.0dB	6.3dB
4	16	13.7dB	33.0dB	6.0dB
5	32	13.7dB	39.0dB	6.0dB

Table 1.2: **PAM constellation energies.**

For $P_e = 10^{-6}$, $\frac{d}{2\sigma} \approx 4.75$ (13.5dB). Table 1.2 relates $b = \bar{b}$, M, $\frac{d}{2\sigma}$, the SNR, and the required increase in SNR (or equivalently in $\bar{\mathcal{E}}_{\boldsymbol{x}}$) to transmit an additional bit at an error probability $P_e = 10^{-6}$. Table 1.2 shows that for $b = \bar{b} > 2$, the approximation of 6dB per bit is very accurate.

Pulse amplitude constellations with $b > 2$ are typically known as 3B1O - three bits per octal signal (for 8 PAM) and 4B1H (4 bits per hexadecimal signal), but are rare in use with respect to the yet more popular quadrature amplitude modulation of Section 1.3.4.2.

1.3.4.2 Quadrature Amplitude Modulation (QAM)

QAM is a two-dimensional generalization of PAM. The two basis functions are usually

$$\varphi_1(t) = \sqrt{\frac{2}{T}} \cdot \text{sinc}\left(\frac{t}{T}\right) \cdot \cos \omega_c t \quad , \tag{1.258}$$

$$\varphi_2(t) = -\sqrt{\frac{2}{T}} \cdot \text{sinc}\left(\frac{t}{T}\right) \cdot \sin \omega_c t \quad . \tag{1.259}$$

The $\text{sinc}(t/T)$ term may be replaced by any Nyquist pulse shape as discussed in Chapter 3. The ω_c is a radian carrier frequency that Subsections 1.3.5 and 1.3.6 discuss further; for now, $\omega_c \geq \pi/T$.

The QAM Square Constellation Figure 1.47 illustrates QAM Square Constellations. These constellations are the Cartesian products[32] of 2-PAM with itself and 4-PAM with itself, respectively.

[32] A Cartesian Product, a product of two sets, is the set of all ordered pairs of coordinates, the first coordinate taken from the first set in the Cartesian product, and the second coordinate taken from the second set in the Cartesian product.

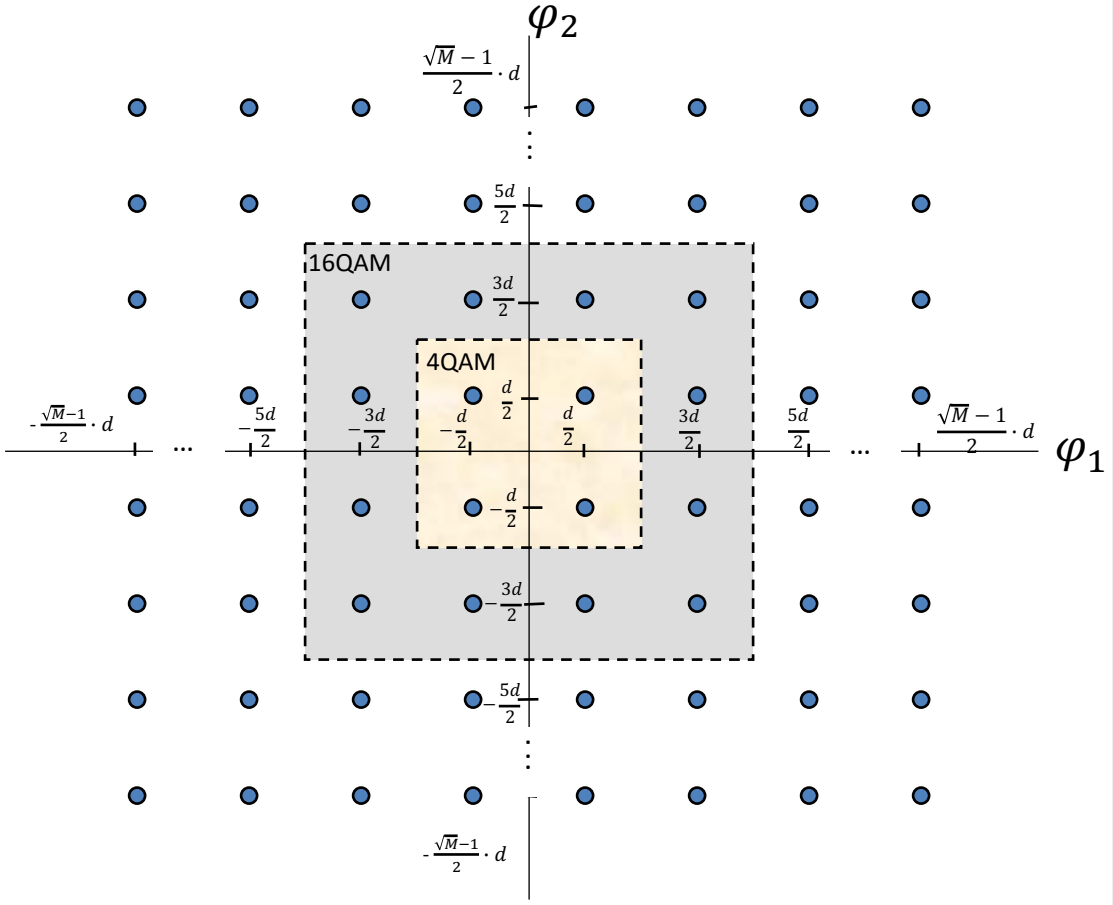


Figure 1.47: **QAM constellations.**

Generally, square M-QAM constellations derive from the Cartesian product of two $\sqrt{M}$-PAM constellations. For $\bar{b}$ bits per dimension, the $M = 4^{\bar{b}}$ symbols are at the coordinates

$$\left\{\pm\frac{d}{2},\ \pm\frac{3d}{2},\ \pm\frac{5d}{2},\ ...,\ \pm\frac{(\sqrt{M}-1)d}{2}\right\}$$

in each dimension. Square QAM constellation's have average energy

$$\begin{aligned}
\mathcal{E}_{M-\text{QAM}} = \mathcal{E}_{\boldsymbol{x}} = 2\bar{\mathcal{E}}_{\boldsymbol{x}} &= \frac{1}{M}\sum_{i,j=1}^{\sqrt{M}}\left(x_i^2 + x_j^2\right) && (1.260)\\
&= \frac{1}{M}\cdot\left[\sqrt{M}\sum_{i=1}^{\sqrt{M}}x_i^2 + \sqrt{M}\sum_{j=1}^{\sqrt{M}}x_j^2\right] && (1.261)\\
&= 2\frac{1}{\sqrt{M}}\cdot\sum_{i=1}^{\sqrt{M}}x_i^2 && (1.262)\\
&= 2\mathcal{E}_{\sqrt{M}-\text{PAM}} && (1.263)\\
&= d^2\left(\frac{M-1}{6}\right) && (1.264)
\end{aligned}$$

Thus, the M-QAM constellation's average energy per dimension is

$$\bar{\mathcal{E}}_{\boldsymbol{x}} = d^2 \left(\frac{M-1}{12} \right) \quad , \tag{1.265}$$

which expectedly equals the constituent $\sqrt{M}$-PAM constellation's $\bar{\mathcal{E}}_{\boldsymbol{x}}$. The minimum distance $d_{\min} = d$ derives from $\mathcal{E}_{\boldsymbol{x}}$ (or $\bar{\mathcal{E}}_{\boldsymbol{x}}$) and M as

$$d = \sqrt{\frac{6\mathcal{E}_{\boldsymbol{x}}}{M-1}} = \sqrt{\frac{12\bar{\mathcal{E}}_{\boldsymbol{x}}}{M-1}} \quad . \tag{1.266}$$

Since $M = 4^{\bar{b}}$, alternative relations for (1.265) and (1.266) in terms of the average bit rate $\bar{b}$ are

$$\bar{\mathcal{E}}_{\boldsymbol{x}} = \frac{\mathcal{E}_{\boldsymbol{x}}}{2} = \frac{d^2}{12} \left[4^{\bar{b}} - 1 \right] \quad , \tag{1.267}$$

and

$$d = \sqrt{\frac{12\bar{\mathcal{E}}_{\boldsymbol{x}}}{4^{\bar{b}} - 1}} \quad . \tag{1.268}$$

Finally,

$$\bar{b} = \frac{1}{2} \log_2 \left(\frac{6\mathcal{E}_{\boldsymbol{x}}}{d^2} + 1 \right) = \frac{1}{2} \log_2 \left(\frac{12\bar{\mathcal{E}}_{\boldsymbol{x}}}{d^2} + 1 \right) \quad , \tag{1.269}$$

the same as PAM.

For large M, $\bar{\mathcal{E}}_{\boldsymbol{x}} \approx \frac{d^2}{12} M = \frac{d^2}{12} 4^{\bar{b}}$, which is essentially the same as PAM's earlier continuous approximation: Square QAM's continuous approximation uses a uniform probability density over the square defined by $[\pm\frac{\sqrt{M}}{2}, \pm\frac{\sqrt{M}}{2}]$,

$$\mathcal{E}_{\boldsymbol{x}} \approx \int_{-\frac{\sqrt{M}}{2}}^{\frac{\sqrt{M}}{2}} \int_{-\frac{\sqrt{M}}{2}}^{\frac{\sqrt{M}}{2}} \frac{x^2 + y^2}{4L^2} dx \, dy = 2 \frac{(\frac{\sqrt{M}}{2})^2}{3} \quad , \tag{1.270}$$

or $\frac{\sqrt{M}}{2} = \sqrt{1.5\mathcal{E}_{\boldsymbol{x}}}$. Since each QAM constellation symbol's decision region has area d^2,

$$M \approx \frac{4 \cdot (\frac{\sqrt{M}}{2})^2}{d^2} = \frac{6\mathcal{E}_{\boldsymbol{x}}}{d^2} = \frac{12\bar{\mathcal{E}}_{\boldsymbol{x}}}{d^2} \quad . \tag{1.271}$$

This result agrees with Equation (1.265) for large M. The continuous approximation's energy-computation error rapidly becomes negligible as $M \to \infty$.

Increasing a QAM constellation's b while maintaining constant minimum distance leads to the following average-energy-increase recursion:

$$\mathcal{E}_{\boldsymbol{x}}(b+1) = 2 \cdot \mathcal{E}_{\boldsymbol{x}}(b) + \frac{d^2}{6} \, . \tag{1.272}$$

Asymptotically the average energy increases by 3dB for each bit added to a square QAM constellation.

Square QAM's error-probability calculation follows from 3 types of correct-decision conditional probabilities:

1. **corner** constellation symbol vectors (4 points with only 2 nearest neighbors)

$$P_{c|\text{corner}} = \left(1 - Q \left[\frac{d}{2\sigma} \right] \right)^2 \tag{1.273}$$

2. **inner** constellation symbol vectors $(\sqrt{M} - 2)^2$ points with 4 nearest neighbors)

$$P_{c|\text{inner}} = \left(1 - 2Q \left[\frac{d}{2\sigma} \right] \right)^2 \tag{1.274}$$

3. **edge** constellation symbol vectors $4(\sqrt{M}-2)$ points with 3 nearest neighbors)

$$P_{c|\text{edge}} = \left(1 - Q\left[\frac{d}{2\sigma}\right]\right)\left(1 - 2Q\left[\frac{d}{2\sigma}\right]\right) \quad . \tag{1.275}$$

The probability of being correct is then (abbreviating $Q \leftarrow Q\left[\frac{d}{2\sigma}\right]$)

$$\begin{aligned} P_c &= \sum_{i=0}^{M-1} P_{c/i} p_{\boldsymbol{x}}(i) && (1.276)\\ &= \frac{4}{M}(1-Q)^2 + \frac{(\sqrt{M}-2)^2}{M}(1-2Q)^2 + \frac{4(\sqrt{M}-2)}{M}(1-2Q)(1-Q) && (1.277)\\ &= \frac{1}{M}\Big[(4-8Q+4Q^2) + (4\sqrt{M}-8)(1-3Q+2Q^2) && (1.278)\\ &\quad +(M-4\sqrt{M}+4)(1-4Q+4Q^2)\Big] && (1.279)\\ &= \frac{1}{M}\left[M + (4\sqrt{M}-4M)Q + (4-8\sqrt{M}+4M)Q^2\right] && (1.280)\\ &= 1 + 4(\frac{1}{\sqrt{M}}-1)Q + 4(\frac{1}{\sqrt{M}}-1)^2Q^2 && (1.281) \end{aligned}$$

Thus, the (symbol) error probability is

$$P_e = 4\left(1-\frac{1}{\sqrt{M}}\right)\cdot Q\left[\frac{d}{2\sigma}\right] - 4\left(1-\frac{1}{\sqrt{M}}\right)^2 \cdot \left(Q\left[\frac{d}{2\sigma}\right]\right)^2 < 4\left(1-\frac{1}{\sqrt{M}}\right)\cdot Q\left[\frac{d}{2\sigma}\right] \quad . \tag{1.282}$$

The average number of nearest neighbors for the constellation equals $4(1-1/\sqrt{M})$, thus for QAM the NNUB is not exact, but usually tight. The corresponding normalized NNUB is

$$\bar{P}_e \le 2\left(1-\frac{1}{2^{\bar{b}}}\right)\cdot Q\left[\frac{d}{2\sigma}\right] = 2\left(1-\frac{1}{2^{\bar{b}}}\right)\cdot Q\left[\sqrt{\frac{3}{M-1}SNR}\right] \quad , \tag{1.283}$$

which equals the PAM result. For $P_e = 10^{-6}$, one determines that $\frac{d}{2\sigma} \approx 4.75$ (13.5dB). Table 1.3 relates $\bar{b}$, M, $\frac{d}{2\sigma}$, the SNR, and the required increase in SNR (or equivalently in $\bar{\mathcal{E}}_{\boldsymbol{x}}$) to transmit an additional bit of information. As with PAM for average bit rates of $\bar{b} > 2$, the approximation of 3dB per bit per two-dimensional QAM symbol for the average energy increase is accurate.

$b = 2\bar{b}$	M	$\frac{d}{2\sigma}$ for $\bar{P}_e = 10^{-6} \approx 2Q\left(\frac{d_{\min}}{2\sigma}\right)$	SNR $= \frac{(M-1)\cdot 10^{1.37}}{3}$	SNR increase $= \frac{M-1}{(M-1)-1}$	dB/bit
2	4	13.7dB	13.7dB	—	—
4	16	13.7dB	20.7dB	7.0dB	3.5dB
6	64	13.7dB	27.0dB	6.3dB	3.15dB
8	256	13.7dB	33.0dB	6.0dB	3.0dB
10	1024	13.7dB	39.0dB	6.0dB	3.0dB
12	4096	13.7dB	45.0dB	6.0dB	3.0dB
14	16,384	13.7dB	51.0dB	6.0dB	3.0dB

Table 1.3: **QAM constellation energies.**

The constellation figure of merit for square QAM is

$$\zeta_{\boldsymbol{x}} = \frac{3}{M-1} = \frac{3}{4^{\bar{b}}-1} = \frac{3}{2^b-1} \quad . \tag{1.284}$$

When b is odd, it is possible to define a SQ QAM constellation by taking every other point from a $b+1$ SQ QAM constellation. (See Problem 1.14.)

Two examples illustrate the wide use of QAM transmission.

EXAMPLE 1.3.6 [Cable Modem] Cable modems upgrade what was an existing cable-broadcast-TV systems' coaxial cables to two-way transmission[33] Cable modem conventions (i.e., DOCSIS). Early cable systems used simple QAM in both transmission directions. The downstream direction from cable TV end to customer is typically at a carrier frequency well above the used TV band, somewhere between 300 MHz and 2000 MHz. The upstream direction is most often below 50 MHz, typically between 5 and 40 MHz. The symbol rate is typically $1/T$=2MHz so the data rate is some multiple of 4 Mbps on any given carrier. Typically about 10 carriers were used (so a multiple of 40 Mbps upstream maximum) for a group of customers with consistent channel characteristics in an immediate neighborhood. Each group is thus shared leading to the famous "cable hogging" problem when one customer uses all his neighbors' bandwidth (Cable operators notoriously quote only the peak speed for the group when selling the service, which is misleading if multiple users simultaneously use the system.) As the cable industry advanced, transmission methods improved to Chapter 4's multi-carrier methods, although the cable-hogging problem persists well into the first decades of the 21st century.

EXAMPLE 1.3.7 [Satellite TV Broadcast] Satellite television uses 4QAM in downlink broadcast transmission at one of 20 carrier frequencies between 12.2 GHz to 12.7 GHz from satellite to customer receiver for some suppliers and satellites. Corresponding carriers between 17.3 and 17.8 GHz are used to send the uplink signals from the broadcaster to the satellite, again with QAM. The symbol rate is $1/T = 19.151$ MHz, so the aggregate data rate is a multiple of 38.302 Mbps on any of the 20 carriers. This is sufficient to carry multiple TV stations per carrier/QAM signal. (Some stations watched by many, for instance sports, may get a larger allocation of bandwidth and carry a higher-quality image than others that are not heavily watched. An ultra-high-definition TV channel requires 20-30 Mbps if sent with full fidelity. Each carrier is transmitted in a 24 MHz transponder channel on the satellite – these 24 MHz channels were originally used to broadcast a single analog TV channel, modulated via FM unlike terrestrial analog broadcast television (which uses only 6 MHz for analog TV). Thus, the migration to digital transmission increased the number of TV channels. Advanced satellite systems now may use M SQ QAM constellations to increase data rates and number of channels further.

[33] Cable TV provider sent personnel to the network's various unidirectional blocking points to install so-called bi-directional "diplex" filters to enable the upgrade from uni-directional broadcast.

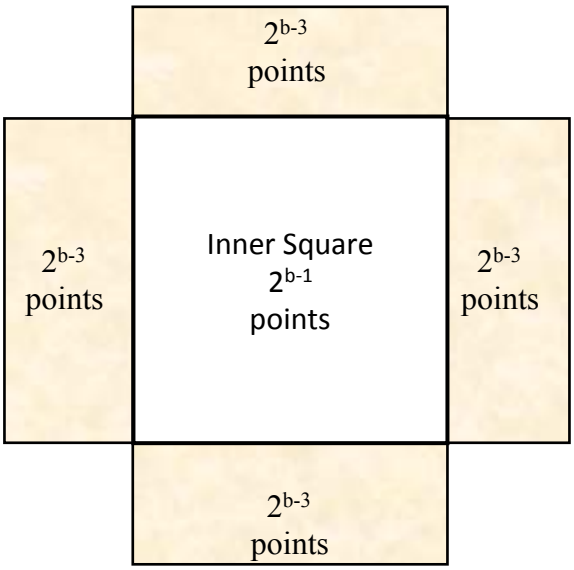


Figure 1.48: **Cross constellations.**

QAM Cross Constellations The QAM cross constellation also allows for odd numbers of bits per symbol in QAM data transmission. To construct a QAM cross constellation with b bits per symbol, the constellation augments a square 2^{b-1} QAM constellation by adding 2^{b-1} data symbols that extend this smaller QAM square's sides. The QAM CR constellation excludes the corners as in Figure 1.48. QAM CR constellations are better than QAM SQ constellations increasingly with b increase.

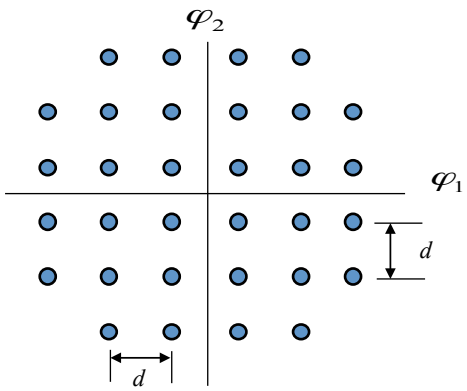


Figure 1.49: **32CR constellation.**

QAM CR average energy computation doubles the energy of the two large rectangles ($[2^{\frac{b-3}{2}}+2^{\frac{b-1}{2}}]\times 2^{\frac{b-1}{2}}$) and then subtracts the energy of the inner square ($2^{\frac{b-1}{2}}\times 2^{\frac{b-1}{2}}$). The energy of the inner square is

$$\mathcal{E}_{\boldsymbol{x}}(inner)=\frac{d^2}{6}(2^{b-1}-1) \quad . \tag{1.285}$$

The total sum of energies for all the data symbols in the inner-square-plus-two-side-rectangles is (looking only at one quadrant, and multiplying by 4 because of symmetry)

$$\begin{aligned}
\mathcal{E} &= \frac{d^2}{4}(4)\sum_{k=1}^{2^{\frac{b-3}{2}}}\sum_{l=1}^{3\cdot 2^{\frac{b-5}{2}}}\left[(2k-1)^2+(2l-1)^2\right] && (1.286)\\
&= \frac{d^2}{4}(4)\left[3\cdot 2^{\frac{b-5}{2}}\left(\frac{2^{\frac{3b-3}{2}}-2^{\frac{b-1}{2}}}{6}\right)+2^{\frac{b-3}{2}}\left(\frac{27\cdot 2^{\frac{3b-9}{2}}-3\cdot 2^{\frac{b-3}{2}}}{6}\right)\right] && (1.287)\\
&= \frac{d^2}{4}(4)\left[2^{\frac{b-7}{2}}\left(2^{\frac{3b-3}{2}}-2^{\frac{b-1}{2}}\right)+2^{\frac{b-5}{2}}\left(9\cdot 2^{\frac{3b-9}{2}}-2^{\frac{b-3}{2}}\right)\right] && (1.288)\\
&= \frac{d^2}{4}\left[2^{2b-3}-2^{b-2}+9\cdot 2^{2b-5}-2^{b-2}\right] && (1.289)\\
&= \frac{d^2}{4}\left[\frac{13}{32}2^{2b}-2^{b-1}\right] \quad . && (1.290)
\end{aligned}$$

Then

$$\begin{aligned}
\mathcal{E}_{\boldsymbol{x}} &= \frac{2\mathcal{E}-2^{b-1}\mathcal{E}_{\boldsymbol{x}}(inner)}{2^b} = \frac{d^2}{4}\left[\frac{26}{32}2^b-1-\frac{2}{3}2^{b-2}+\frac{2}{3}\frac{1}{2}\right] && (1.291)\\
&= \frac{d^2}{4}\left[\left(\frac{13}{16}-\frac{1}{6}\right)2^b-\frac{2}{3}\right] && (1.292)\\
&= \frac{d^2}{4}\left[\frac{31}{48}2^b-\frac{2}{3}\right] = \frac{d^2}{6}\left[\frac{31}{32}M-1\right] && (1.293)
\end{aligned}$$

The minimum distance $d_{\min}=d$ can be computed from $\mathcal{E}_{\boldsymbol{x}}$ (or $\bar{\mathcal{E}}_{\boldsymbol{x}}$) and M by

$$d=\sqrt{\frac{6\mathcal{E}_{\boldsymbol{x}}}{\frac{31}{32}M-1}}=\sqrt{\frac{12\bar{\mathcal{E}}_{\boldsymbol{x}}}{\frac{31}{32}M-1}}=\sqrt{\frac{12\bar{\mathcal{E}}_{\boldsymbol{x}}}{\frac{31}{32}4^{\bar{b}}-1}} \quad . \tag{1.294}$$

In (1.293), for large M, $\mathcal{E}_{\boldsymbol{x}}\approx\frac{31d^2}{192}M=\frac{31d^2}{192}4^{\bar{b}}$, the same as the continuous approximation.

The following recursion derives from increasing the number of bits, b, in a QAM cross constellation while maintaining constant minimum distance:.

$$\mathcal{E}_{\boldsymbol{x}}(b+1)=2\cdot\mathcal{E}_{\boldsymbol{x}}(b)+\frac{d^2}{6} \quad . \tag{1.295}$$

As with the square QAM constellation asymptotically the average energy increases by 3 dB for each added bit per two dimensional symbol.

QAM CR's error probability has a bound that derives from a lower bound on a correct decision's conditional probability being in one of two categories:

1. **inner** CR constellation symbols ($\left\{2^b-4\left(3\cdot 2^{\frac{b-3}{2}}-2\cdot 2^{\frac{b-5}{2}}\right)\right\}=\left\{2^b-4\left(2^{\frac{b-1}{2}}\right)\right\}$) with four nearest neighbors
$$P_{c/inner}=\left(1-2Q\left[\frac{d_{\min}}{2\sigma}\right]\right)^2 \tag{1.296}$$

2. **side** CR constellation symbols ($4\left(3\cdot 2^{\frac{b-3}{2}}-2\cdot 2^{\frac{b-5}{2}}\right)=4\left(2^{\frac{b-1}{2}}\right)$) with three nearest neighbors. (This calculation is only a bound because some of the side points have fewer than three neighbors at distance $d_{\min}$)
$$P_{c/outer}=\left(1-Q\left[\frac{d_{\min}}{2\sigma}\right]\right)\left(1-2Q\left[\frac{d_{\min}}{2\sigma}\right]\right) \quad . \tag{1.297}$$

The probability of a correct decision is then, abbreviating $Q = Q\left[\frac{d_{\min}}{2\sigma}\right]$,

$$\begin{aligned} P_c &\geq \frac{1}{M}\left[4\left(2^{\frac{b-1}{2}}\right)(1-Q)(1-2Q)\right] && (1.298)\\ &\quad + \frac{1}{M}\left[\left\{2^b - 4\left(2^{\frac{b-1}{2}}\right)\right\}(1-2Q)^2\right] && (1.299)\\ &= \frac{1}{M}\left[4\cdot 2^{\frac{b-1}{2}}(1-3Q+2Q^2) + \left[2^b - 2^{\frac{b+3}{2}}\right](1-4Q+4Q^2)\right] && (1.300)\\ &= 1 - \left[-2^{\frac{3-b}{2}}+4\right]Q + \left[2^{\frac{5-b}{2}} - 2\cdot 2^{\frac{5-b}{2}} + 4\right]Q^2 && (1.301) \end{aligned}$$

Thus, the symbol-error probability satisfies

$$\begin{aligned} P_e &\leq 4\left(1-\frac{1}{\sqrt{2M}}\right)Q\left[\frac{d_{\min}}{2\sigma}\right] - 4\left(1-\sqrt{\frac{2}{M}}\right)\left(Q\left[\frac{d_{\min}}{2\sigma}\right]\right)^2 && (1.302)\\ &< 4\left(1-\frac{1}{\sqrt{2M}}\right)Q\left[\frac{d_{\min}}{2\sigma}\right] < 4Q\left[\frac{d_{\min}}{2\sigma}\right] \quad . && (1.303) \end{aligned}$$

SQ CR's average number of nearest neighbors is $4(1-1/\sqrt{2M})$; thus the NNUB is again accurate. The normalized error probability is

$$\bar{P}_e \leq 2\left(1-\frac{1}{2^{\bar{b}+.5}}\right)Q\left[\frac{d_{\min}}{2\sigma}\right] \quad , \tag{1.304}$$

which agrees with the PAM result with an additional constellation bit, or equivalently an extra .5 bit per dimension. To evaluate (1.304), Equation 1.294 relates that

$$\left(\frac{d_{min}}{2\sigma}\right)^2 = \frac{3\ \mathrm{SNR}}{\frac{31}{32}M-1} \tag{1.305}$$

Table 1.4 lists the incremental energies and required SNR for QAM cross constellations in a manner similar to Table 1.3.

$b = 2\bar{b}$	M	$\frac{d}{2\sigma}$ for $\bar{P}_e = 10^{-6} \approx 2Q\left(\frac{d_{\min}}{2\sigma}\right)$	SNR $= \frac{([31/32]\cdot M-1)\cdot 10^{1.37}}{3}$	SNR increase $= \frac{[31/32]\cdot M-1}{[32/32]\cdot(M-1)-1}$	dB/bit
5	32	13.7dB	23.7 dB	—	—
7	128	13.7dB	29.8dB	6.1dB	3.05dB
9	512	13.7dB	35.8dB	6.0dB	3.0dB
11	2048	13.7dB	41.8dB	6.0dB	3.0dB
13	8192	13.7dB	47.8dB	6.0dB	3.0dB
15	32,768	13.7dB	53.8dB	6.0dB	3.0dB

Table 1.4: **QAM Cross constellation energies.**

Vestigial Sideband Modulation (VSB), CAP, and OQAM There are many equivalent basis function choices for QAM. These choices sometimes have value from an implementation perspective. From a performance perspective, they all perform the same according to this section's AWGN fundamentals. In successive one-shot transmission, the basis functions must be orthogonal to one another for all integer-symbol-period time translations. Then successive demodulator-output samples at integer multiples of T will be independent; then also, the AWGN-one-shot optimum receiver's repeated successive use detects successive messages optimally (see Chapter 3 successive transmission degradation in the presence of "intersymbol interference" on band-limited AWGN channels.) The PAM basis function always exhibits this desirable translation property on the AWGN, and so do the QAM basis functions as long as $\omega_c \geq \pi/T$. The QAM basis functions are not unique with respect to satisfaction of the translation property, with VSB/SSB, CAP, and OQAM all being variants:

VSB Vestigial sideband modulation (VSB) is an alternative modulation method that is equivalent to QAM. In QAM, typically the same unit-energy basis function ($\sqrt{1/T} \cdot \mathrm{sinc}(t/T)$) is "double-side-band modulated" independently by the same carrier's sine and cosine to generate the two QAM basis functions. In VSB, a double bandwidth sinc function and its Hilbert transform (see Section 1.3.5 for a discussion of Hilbert transforms) are "single-side-band modulated." The two VSB basis functions are[34]

$$\varphi_1(t) = \sqrt{\frac{1}{T}} \cdot \mathrm{sinc}\left(\frac{2t}{T}\right) \cdot \cos \omega_c t \quad , . \tag{1.306}$$

$$\varphi_2(t) = \sqrt{\frac{1}{T}} \cdot \mathrm{sinc}\left(\frac{t}{T}\right) \cdot \sin(\frac{\pi t}{T}) \cdot \sin \omega_c t \quad . \tag{1.307}$$

A natural symbol-rate choice for successive transmission with these two basis functions might appear to be $2/T$, twice the rate associated with QAM. However, these basis functions' successive translations by integer multiples of $T/2$ are not orthogonal – that is $< \varphi_i(t), \varphi_j(t - T/2) >\neq \delta_{ij}$; however, $< \varphi_i(t), \varphi_j(t - kT) >= \delta_{ij}$ for any integer k. Thus, the symbol rate for successive orthogonal transmissions is best $1/T$.

VSB designers often prefer to exploit the observation that $< \varphi_1(t), \varphi_2(t - kT/2) >= 0$ for all odd integers k to implement the VSB transmission system as a time-varying one-dimensional modulation at rate $2/T$ dimensions per second. Thus, the modulator uses a different basis function on adjacent symbol periods, alternating between the two. The optimum receiver consists of two matched filters to the two basis functions, which have their outputs each sampled at rate $1/T$ (staggered relative to one another by $T/2$). The detector interleaves these samples to form a single one-dimensional detected-symbol stream. Nonetheless, different VSB designers may call the VSB constellations by two-dimensional names: For instance, one may hear of 16 VSB or 64 VSB, which are equivalent to 16SQ QAM (or 4PAM) and 64SQ QAM (8PAM) respectively. VSB transmission was initially more convenient for upgrading existing analog systems that were already VSB (i.e., commercial broadcast television before transition to all-digital for instance) to digital systems that use the same bandwidths and carrier frequencies - that is where the carrier frequencies are not centered within the existing band. VSB otherwise has no fundamental performance advantages or differences from QAM.

CAP Carrierless Amplitude/Phase (CAP) transmission systems[4] are also very similar to QAM. The basis functions of QAM are time-varying when ω_c is arbitrary – that is, the basis functions on subsequent transmissions may differ. CAP is a method that can eliminate this time variation for any carrier-frequency choice, making the combined transmitter implementation appear "carrierless" and thus time-invariant. CAP has the same one-shot basis functions as QAM, but also has a time-varying encoder constellation when used for successive two-dimensional symbol transmission. The time-varying CAP encoder implements a sequence of additional two-dimensional constellation rotations that the receiver knows and easily removes after the demodulator and just before the detector. The rotation sequence has a phase angle that increases linearly with time. (and virtually omitted when differential encoding - see Subsection 1.3.6 - is implemented).

OQAM Offset QAM (OQAM) or "staggered" QAM uses the alternative basis functions

$$\varphi_1(t) = \sqrt{\frac{2}{T}} \cdot \mathrm{sinc}\left(\frac{t}{T}\right) \cdot \cos\left(\frac{\pi t}{T}\right) \tag{1.308}$$

$$\varphi_2(t) = -\sqrt{\frac{2}{T}} \cdot \mathrm{sinc}\left(\frac{t - T/2}{T}\right) \cdot \sin\left(\frac{\pi t}{T}\right) \tag{1.309}$$

effectively "offseting" the two dimensions by $T/2$. For one-shot transmission, such offset has no effect (the receiver matched filters effectively re-align the two dimensions) and OQAM and QAM are the same.

[34] This simple description is actually single-side-band (SSB), a special case of VSB. VSB uses practical realizable functions instead of the unrealizable sinc functions that simplify fundamental developments here in Chapter 1. Subsection 1.3.6.1 more completely addresses VSB.

For successive transmission, the derivative (rate of change) of $x(t)$ is less for OQAM than for QAM, effectively reducing transmitted signals' spurious bandwidth when the sinc functions cannot be perfectly implemented. OQAM signals will never take the value $x(t) = 0$, while this value is instantaneously possible with QAM – thus nonlinear transmitter/receiver amplifiers are not as stressed by OQAM. There is otherwise no fundamental performance difference between OQAM and QAM.

Forney's Gap The **gap**, Γ, is an approximation introduced by Forney[35] [3] for constellations with $\bar{b} \geq 1/2$. This gap is empirically evident in the PAM and QAM tables. Specifically, if one knows an AWGN channel's SNR, the number of bits that can be transmitted with PAM or QAM is

$$\bar{b} = \frac{1}{2} \cdot \log_2 \left(1 + \frac{\text{SNR}}{\Gamma} \right) \quad . \tag{1.310}$$

This expression follows Chapter 2-developed highest AWGN channel data rate where the "gap" is $\Gamma \geq 1$ (often specified in dB or 0 dB for $\Gamma = 1$); thus, $\Gamma > 1$ means the transmission encoding/modulation scheme performs at a rate below best. For Square QAM ($N = 2$), or equivalently PAM ($N = 1$) in terms of bits/dimension, Forney's Gap is also a function of a specified $\bar{P}_e$

$$\bar{P}_e \approx 2 \cdot \left(1 - \frac{1}{2^{\bar{b}}} \right) \cdot Q \left(\sqrt{\frac{3 \cdot SNR}{2^{N \cdot \bar{b}} - 1}} \right) \quad . \tag{1.311}$$

The SNR corresponding to the specified P_e is (with Q^{-1} as the argument of the $Q(x)$-function that produces the value Q^{-1}, or simply put the inverse function, **not** $1/Q$)

$$SNR(\bar{P}_e, \bar{b}) = \frac{2^{N\bar{b}} - 1}{3} \cdot \left[Q^{-1} \left(\frac{\bar{P}_e}{2(1 - 2^{-(N/2) \cdot \bar{b}})} \right) \right]^2 \quad . \tag{1.312}$$

The best achievable data rate has instead

$$SNR = 2^{N \cdot \bar{b}} - 1 \quad . \tag{1.313}$$

The gap between the PAM/QAM and the best data rate is then

$$\begin{aligned} \Gamma &= \frac{SNR(\bar{P}_e, \bar{b})}{SNR} && (1.314) \\ &= \frac{1}{3} \cdot \left[Q^{-1} \left(\frac{\bar{P}_e}{2(1 - 2^{-(N/2) \cdot \bar{b}})} \right) \right]^2 && (1.315) \\ &= 20 \cdot \log_{10} \left[Q^{-1} \left(\frac{\bar{P}_e}{2(1 - 2^{-(N/2) \cdot \bar{b}})} \right) \right] - 4.7712 \text{ dB}. && (1.316) \end{aligned}$$

With $2^{N\bar{b}} > 1$, this simplifies increasingly accurately to[36]

$$\Gamma \approx 20 \cdot \log_{10} \left[Q^{-1} \left(\frac{\bar{P}_e}{2} \right) \right] - 4.7712 \quad . \tag{1.317}$$

At error probability $\bar{P}_e = 10^{-6}$, $\Gamma = 9.0$ dB. For $\bar{P}_e = 10^{-7}$, $\Gamma = 9.7$ dB. If the designer knows the SNR and the desired performance level ($\bar{P}_e$) or equivalently the gap, then the number of bits per dimension (and thus the achievable data rate $R = b/T$) immediately follow through (1.310). Chapters 2, 8, and 10 introduce more sophisticated encoder designs where the gap can be reduced, ultimately to 0 dB, enabling a highest possible data rate of $.5 \log_2(1 + \text{SNR})$, sometimes known as the AWGN's "channel capacity." QAM and PAM are thus about 9 dB away in terms of efficient use of SNR from ultimate limits.

[35] After G. David "Dave" Forney, Jr., 1940 - , An American Engineering who pioneered many of the basic methods used in Modulation and Coding, while both at MIT and Motorola Codex Corp.

[36] Special Thanks to Stanford graduate student and course TA Ms. Yun Liao and visiting scholar Dr. Norman Swenson for this nice development of the gap.

1.3.4.3 Constellations based on lattices

Appendix B.2 reviews lattices. A **lattice** is a set of vectors in N-dimensional space that is closed under vector addition – that is, the sum of any two vectors is another vector in the set. Examples include the scalar integers $\mathbb{Z}$ and the two-dimensional integer lattice $\mathbb{Z}^2$ (ordered pairs of integers). PAM/QAM-based constellations (and later sequences of such PAM or QAM "subsymbols," $\mathbb{Z}^N$ see Chapter 2, which largely define most practical codes) select $|C|$ symbol values as finite subsets these lattices. Such constellation subsets pick adjacent (geometrically uniform) values within a low-energy region, usually resulting in $\mathbb{E}[\boldsymbol{x}\} \neq 0$. The translation $\boldsymbol{x} \rightarrow \boldsymbol{x} - \mathbb{E}[\boldsymbol{x}]$ does not affect performance (translational invariance) and zeros the mean while reducing the set's energy to a minimum value for the selected constellation symbol values. Such a lattice translation produces a **coset** of the lattice. The **fundamental volume** for a lattice measures the region around a point:

> **Definition 1.3.18** [Fundamental Volume] *A lattice Λ's fundamental volume $V(\Lambda)$ is the volume of the decision region, or Voronoi ewfion, for any single lattice point $\boldsymbol{x} \in \Lambda$ $\mathcal{V}(\Lambda)$, with volume $V(\Lambda) = |\mathcal{V}(\Lambda)|$. A lattice's Voronoi Region, $\mathcal{V}(\Lambda)$, differs from the constellation's Voronoi Boundary, $\mathcal{V}_{\boldsymbol{x}}$, with the latter being the union of $|C|$ of the former $V(\Lambda)$. $\mathcal{V}_{\boldsymbol{x}}$ may follow a different ("shaping") lattice Λ_s which itself may have a different Voronoi region, nor may it even be a scaled version of, Λ (which is then the coding lattice).*

For example with $M = |C|$, an M-QAM constellation $C - M - QAM$ as $M \rightarrow \infty$ is a translated subset (coset) of the two-dimensional rectangular lattice Z^2, so M-QAM is a translation of Z^2 as $M \rightarrow \infty$. Similarly as $M \rightarrow \infty$, the M-PAM constellation becomes a coset of the one dimensional lattice Z.

The Continuous Approximation For a geometrically uniform (basically the constellation's choice of symbol values has "no holes" or missing points that clearly should be included) constellation, C, the continuous approximation computes the constellation's average energy by approximating a discrete energy sum with a continuous integral. To use such continuous integration, a continuous uniform distribution approximates the constellation's discrete probability distribution over a region defined by the constellation's boundaries. In this constellation region, each symbol appears in the center of an identically shaped fundamental-volume region (or **Voronoi Region**), $\mathcal{V}(\Lambda)$, where as in Appendix B.2 Λ is a lattice[37] from which the symbols were selected. The Voronoi region is (for momentary purposes) the same as the decision region. The **Voronoi region's volume** is $V(\Lambda)$, so $V(\Lambda) = |\mathcal{V}(\Lambda)|$. This Voronoi region's centered symbol vector thus has a maximum number of nearest neighbors, or equivalently number of sides. The union of all symbols' Voronoi regions is the constellation's **Voronoi boundary** $\mathcal{V}_{\boldsymbol{x}} \triangleq \bigcup_{m=0}^{M-1} V_i(\Lambda)$ where $\mathcal{V}_i(\Lambda)$ corresponds to the i^{th} symbol's Voronoi region. The Voronoi boundary envelops a volume $|C| \cdot V(\Lambda)$ (or $M \cdot V(\Lambda)$ when $M = |C|$).

The continuous approximation replaces the constellation symbols' discrete distribution with a uniform probability density[38] region contained within the Voronoi boundary. That continuous probability density is $p_{\boldsymbol{x}}(\boldsymbol{u}) = \frac{1}{|C| \cdot V(\Lambda)} \ \forall \ \boldsymbol{u}$ in $\mathcal{V}_{\boldsymbol{x}}$. Chapter 2's coded systems allow $N > 2$, and thus the uniform probability density will be over that larger dimensionality where symbols are equally likely (uniform distribution) - which does not necessarily imply uniform distribution in 1 or 2 dimensional slices of that larger region, see Chapter 2 where the Voronoi regions may not be simple hypercubes and thus cause such effect.

[37] Or, more properly a coset thereof, see Appendix B for more on lattices and cosets, but for now view a lattice as a grid of regularly spaced symbols is sufficient

[38] This text will often use the terms probability density and distribution interchangeably even though strict terminology might only use distribution for discrete random variables and density for continuous random variables.

Definition 1.3.19 [Continuous Approximation] *The continuous approximation to a constellation's average energy equals*

$$\mathcal{E}_{\boldsymbol{x}} \approx \tilde{\mathcal{E}}_{\boldsymbol{x}} = \int_{\mathcal{V}_{\boldsymbol{x}}} \|\boldsymbol{u}\|^2 \cdot \frac{1}{|C| \cdot V(\Lambda)} d\boldsymbol{u} \quad , \tag{1.318}$$

where the N-dimensional integral covers the Voronoi boundary $\mathcal{V}_{\boldsymbol{x}}$ for the constellation C. The constant term $|C| \cdot V(\Lambda)$ is a normalizing scale-factor. It might correspond to any continuous region with the same volume with uniform density. Indeed, the number of constellation symbols $|C|$ is no longer relevant other than it helps "size" the region as a scale factor.

For large size $|C|$ with regular symbol-vector spacing, the continuous approximation's deviation from actual energy is small, as several examples will later demonstrate.

For many regions, mathematicians have tabulated the squared energy, or equivalently the region's second moment (scaled by the inverse volume $V^{-2/N}$). Problem 1.19 investigates a few simple regions. The energy/second-moment and volume can be computed from basic geometric parameters. For instance for a circle (2D), the Area is πr^2 with radius r and the second moment is $\frac{1}{2}\pi r^4$, making continuous approximation energy equal to $\frac{1}{2}r^2$. The ratio of this to volume is then $\frac{1}{2\pi}$ or equivalently the 2D volume (area) is 2π times the energy of a circle. The circle's radius grows as more symbols pack into this circle so a larger constellation (presumably each point with some fixed $\mathcal{V}_{\boldsymbol{x}}$), but the ratio of energy/volume will remain (with large M or effectively continuous uniform distribution) $1/2\pi$.

A hypersphere of increasingly large dimensionality as $N \to \infty$ has well-known limiting second moment for radius r, which follows [3] an N-dimensional sphere's volume for radius r is (when N is even)

$$V_N(r) = \frac{(\pi r^2)^{N/2}}{(\frac{N}{2}!)} \quad . \tag{1.319}$$

Then energy per dimension is

$$\bar{\mathcal{E}}_{\boldsymbol{x}} = \frac{r^2}{N+2} \ . \tag{1.320}$$

The two dimensional energy total is $2 \cdot \bar{\mathcal{E}}_{\boldsymbol{x}}$. Energy is a squared quantity, so volume should also be related to a 2D squared quantity when compared with energy. The appropriate ratio is thus

$$\frac{\bar{\mathcal{E}}_{\boldsymbol{x}}}{V^{2/N}} = \frac{r^2}{N+2} \cdot \frac{(\frac{N}{2}!)^{2/N}}{(\pi r^2)} = \frac{(\frac{N}{2}!)^{2/N}}{\pi \cdot (N+2)} \ . \tag{1.321}$$

Stirling's formula is that as $m \to \infty$, then $m! \to (m/e)^m$, so then the inverse of (1.321) becomes

$$\lim_{N \to \infty} \frac{V^{2/N}}{\bar{\mathcal{E}}_{\boldsymbol{x}}} = 2\pi e \quad . \tag{1.322}$$

Having introduced many commonly used constellations, several performance measures compare coded systems' use of these constellations.

Coding Gain Of fundamental importance to two systems' comparison, when they both transmit the same number of bits per dimension, is the **coding gain**, which specifies one constellation/code's improvement.

Definition 1.3.20 [Coding Gain] *A particular constellation's* **coding gain** *(or loss),* γ*, with data symbols* $\{\boldsymbol{x}_i\}_{i=0,\ldots,M-1}$ *with respect to another constellation with data symbols* $\{\breve{\boldsymbol{x}}_i\}_{i=0,\ldots,M-1}$ *is*

$$\gamma \triangleq \frac{\left(d^2_{\min}(\boldsymbol{x})/\bar{\mathcal{E}}_{\boldsymbol{x}}\right)}{\left(d^2_{\min}(\breve{\boldsymbol{x}})/\bar{\mathcal{E}}_{\breve{\boldsymbol{x}}}\right)} = \frac{\zeta_{\boldsymbol{x}}}{\zeta_{\breve{\boldsymbol{x}}}} \quad , \tag{1.323}$$

where both constellations transmit $\bar{b}$ *information bits per dimension.*

A coding gain of $\gamma = 1$ (0dB) implies that the two systems perform equally. A positive gain (in dB) means that the constellation with data symbols $\boldsymbol{x}$ outperforms the constellation with data symbols $\breve{\boldsymbol{x}}$. The coding gain effectively causes the $\bar{\mathcal{E}}_{\boldsymbol{x}}$ to be the same in both systems through normalization to it in both the numerator and denominator of (1.323). Thus it is a fair comparison when with the same $\bar{b}$ for both systems. An example compares the two constellations in Figures 1.37 and 1.39 and obtains

$$\gamma = \frac{\zeta_{\boldsymbol{x}}(\text{8AMPM})}{\zeta_{\boldsymbol{x}}(\text{8PSK})} = \frac{\frac{2}{10}}{\sin^2(\frac{\pi}{8})} \approx 1.37 \ (1.4dB) \ . \tag{1.324}$$

The coding gain, γ of one constellation based on $\boldsymbol{x}$ with lattice Λ and volume $\mathcal{V}(\Lambda)$ with respect to another constellation with $\breve{\boldsymbol{x}}$, $\breve{\Lambda}$, and $V(\breve{\Lambda})$ can be rewritten as

$$\gamma = \frac{\left(\frac{d^2_{\min}(\boldsymbol{x})}{V^{2/N}(\Lambda)}\right)}{\left(\frac{d^2_{\min}(\breve{\boldsymbol{x}})}{V^{2/N}(\breve{\Lambda})}\right)} \cdot \frac{\left(\frac{V^{2/N}(\Lambda)}{\bar{\mathcal{E}}_{\boldsymbol{x}}}\right)}{\left(\frac{V^{2/N}(\breve{\Lambda})}{\bar{\mathcal{E}}_{\breve{\boldsymbol{x}}}}\right)} \tag{1.325}$$

$$= \gamma_f + \gamma_s \quad (dB) \tag{1.326}$$

The two quantities on the right in (1.326) are called the **fundamental gain** γ_f and the **shaping gain** γ_s respectively.

Definition 1.3.21 [Fundamental Gain] *A lattice* Λ*'s fundamental gain* γ_f *for a constellation with symbols* $\boldsymbol{x} \in C_{\boldsymbol{x}} \subset \Lambda$*, relative to a constellation with symbols* $\breve{\boldsymbol{x}} \in C_{\breve{\boldsymbol{x}}} \subset \breve{\Lambda}$ *is*

$$\gamma_f \triangleq \frac{\left(\frac{d^2_{\min}(\boldsymbol{x})}{V^{2/N}(\Lambda)}\right)}{\left(\frac{d^2_{\min}(\breve{\boldsymbol{x}})}{\breve{V}^{2/N}(\breve{\Lambda})}\right)} \quad . \tag{1.327}$$

The fundamental gain measures the constellation lattice's symbol-spacing efficiency without regard to constellation boundary $\mathcal{V}_{\boldsymbol{x}}$ *or energy.*

Definition 1.3.22 [Shaping Gain] *The constellation boundary* $\mathcal{V}_{\boldsymbol{x}}$*'s shaping gain* γ_s *is*

$$\gamma_s \triangleq \frac{\left(\frac{V^{2/N}(\Lambda)}{\bar{\mathcal{E}}_{\boldsymbol{x}}}\right)}{\left(\frac{\breve{V}^{2/N}(\breve{\Lambda})}{\bar{\mathcal{E}}_{\breve{\boldsymbol{x}}}}\right)} \quad . \tag{1.328}$$

The shaping gain measures the constellation boundary $\mathcal{V}_{\boldsymbol{x}}$*'s efficiency for the average energy per dimension, without regard for the constellation's intra-symbol spacing.*

Using the continuous approximation, the designer can extend shaping gain to constellations with different numbers of symbols as

$$\gamma_s = \frac{\left(\frac{V^{2/N}(\Lambda)}{\bar{\mathcal{E}}_{\boldsymbol{x}}}\right) \cdot 2^{2\bar{b}(\boldsymbol{x})}}{\left(\frac{\breve{V}^{2/N}\breve{\Lambda}}{\bar{\mathcal{E}}_{\breve{\boldsymbol{x}}}}\right) \cdot 2^{2\bar{b}(\breve{\boldsymbol{x}})}} \quad . \tag{1.329}$$

When both constellations uses the same coding lattice, $\breve{\Lambda} = \Lambda$, the shaping gain is the ratio of energies, effectively measuring how constellation's symbol-vector packing per energy unit.

Equation (1.322) permits comparison of a hypersphere's shaping gain versus the hypercube composed of $N \to \infty$ uses of PAM, or SQ QAM, as

$$\gamma_s \to \frac{2\pi e}{\frac{1}{1/12}} = \frac{\pi e}{6} = 1.53 \text{ dB} \tag{1.330}$$

a best possible shaping-gain improvement.

Peak-to-Average Power Ratio (PAR) Design requirements may also limit the system's peak power. This peak-power constraint can manifest itself in several different ways: For example, a modulator's Digital-to-Analog Converter (or the demodulator's Analog-to-Digital Converter) has a finite number of bits (or finite dynamic range), so then the signal peaks can not be arbitrarily large without saturation or clipping. Some channels or modulator/demodulators may include amplifiers or repeaters that saturate at high peak signal voltages. Some design requirements have adjacent-channel energy leakage requirements, particularly where a large peak's crosstalk causes an impulsive noise "hit" and an unexpected error in the adjacent system. The Peak-to-Average Power Ratio (PAR) measures immunity to these important design constraints.

The peak energy is:

Definition 1.3.23 [Discrete Peak Energy] *A constellation's N-dimensional* **discrete peak energy** *is $\mathcal{E}_{peak}$.*

$$\mathcal{E}_{peak} \triangleq \max_i \sum_{n=1}^{N} x_{in}^2 \quad . \tag{1.331}$$

A modulated signal's **continuous-time peak energy** *is*

$$\mathcal{E}_{cont} \triangleq max_{i,t}|x_i(t)|^2 \geq \mathcal{E}_{peak} \quad . \tag{1.332}$$

$\mathcal{E}_{cont}$ is important in analog amplifier design or equivalently in the filters $\varphi_n(t)$'s implementations. The peak energy concept allows precise definition of the PAR:

Definition 1.3.24 [Peak-to-Average Power Ratio] *The $N-$dimensional* **Peak-to-Average Power Ratio**, *$PAR_{\boldsymbol{x}}$, for N-dimensional Constellation is*

$$PAR_{\boldsymbol{x}} = \frac{\mathcal{E}_{peak}}{\mathcal{E}_{\boldsymbol{x}}} \tag{1.333}$$

For example 16SQ QAM has a PAR of 1.8 in two dimensions. For each of the one-dimensional 4-PAM constellations that constitute a 16SQ QAM constellation, the one-dimensional PAR is also 1.8. These two ratios need not be equal, however, in general. For instance, for 32CR, the two-dimensional PAR is

$34/20 = 1.7$, while observation of a single dimension when 32CR is used gives a one-dimensional PAR of $25/(.75(5) + .25(25)) = 2.5$. Typically, the continuous-time peak squared signal energy is inevitably yet higher in QAM constellations and depends on the choice of $\varphi(t)$.

1.3.4.4 Hexagonal Signal Constellations

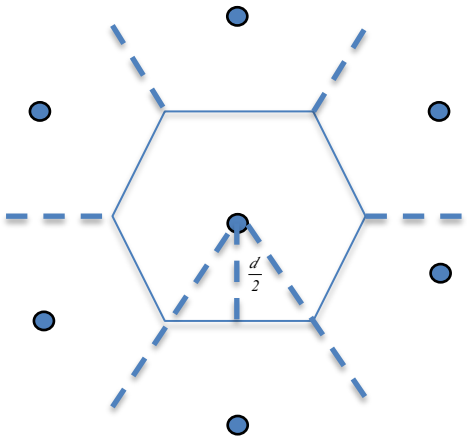


Figure 1.50: **Hexagonal (A_2) lattice.**

The most dense two-dimensional packing of regularly spaced points is Figure 1.50's hexagonal (A_2) lattice. The Voronoi region volume (area) is

$$V(A_2) = 6(\frac{1}{2})(\frac{d}{2})(\frac{d}{\sqrt{3}}) = d^2\frac{\sqrt{3}}{2} \quad . \tag{1.334}$$

If $d_{\min} = d$ in both constellations, then then A_2's fundamental gain over the QAM constellation's Z^2 lattice is

$$\gamma_f = \frac{d^2}{\frac{\sqrt{3}d^2}{2}} = \frac{2}{\sqrt{3}} = .625 \text{ dB} \quad . \tag{1.335}$$

Hexagonal constellation's encoder and detector may be more complex than those for QAM.

1.3.5 Baseband Modulation

This subsection's results apply to each and every spatial dimension of a MIMO system. **Baseband** modulation's basis functions have most energy at low frequencies. The majority of Sections 1.2's - 1.3's modulation methods are baseband, although a few (like PSK and QAM) use basis functions that have energy centered at or near a carrier or center frequency $\omega_c = 2\pi f_c$. These latter **passband** modulation methods are useful in many applications where transmission occurs over a limited narrow bandwidth, typically centered at or near this passband modulation's carrier frequency. Digital television transmission on the USA's Channel 2 has carrier frequency $f_c = 52$ MHz and non-negligible energy only between 50 to 56 MHz. (Chanels $i = 3$ through 60 typically are also 6 MHz wide using carrier frequencies of $f_{c,i} = 52 + (i-2) \cdot 6$ MHz.) Cellphones use carrier frequencies from $f_c = 600$ MHz to 70 GHz, but have nonzero energy over a narrow band that is typically from $W = 1$ MHz to 100 MHz wide. Cellular transmission systems effectively combine many narrow carriers, each of width typically 15 kHz wide as addressed in Chapter 4. Digital satellite transmission uses QAM and carriers 12 GHz $\leq f_c \leq$ 17 GHz bands with transponder bandwidths of about $W = 26$ MHz. There are numerous other examples. Signal energy is present only in these narrow "passbands" and consequently subject

to filtering. This section teaches a common analysis method for such systems without explicit need for the carrier frequency, nor its inclusion in the basis functions, nor even in the channel transfer function. This theory of **passband system analysis** allows a framework for later chapters' important suboptimal receivers for both baseband and passband modulation.

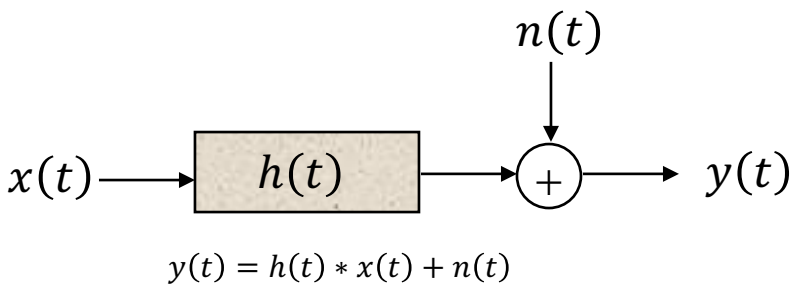


Figure 1.51: **The filtered AWGN.**

Passband transmission-system design typically addresses a band-limited channel that generalizes earlier sections' AWGN model by adding Figure 1.51's filter $h(t)$. When this filter's $h(t) \neq \delta(t)$, $h(t)$ models the physical channel's band-limiting effect, which may be caused by transmitter or receiver filters or by transmission lines', or wireless multiple-path's, physical finite-bandwidth constraints. The filter $h(t)$ distorts the transmitted modulated signal $x(t)$. Preferably, $x(t) * h(t) \approx x(t)$, but fixed modulator designs may not be able to ensure small distortion. All channels inevitably attenuation high frequencies, but many channels also attenuate low frequencies. Furthermore, different frequencies may have different attenuation levels in real channels. Whether modeling filters or actual physical effects, an imperfect channel impulse response $h(t) \neq \delta(t)$ affects tranmission performance.

This section develops new complex-baseband models that apply to either the real baseband case (like PAM), where trivially all imaginary components are zero, and to the passband case (like QAM) where all imaginary components are not necessarily zero, allowing a single complex-symbol-vector theory of receiver processing in the remainder of this text.

1.3.5.1 Passband Representations and Terminology

A passband modulated signal concentrates energy in the vicinity of a carrier frequency $\omega_c = 2\pi f_c$ for transmission through a passband channel that only passes energy in this same frequency band. Passband modulation usually includes a "lowpass" modulated signal's multiplication by a sinusoid to shift energy towards the frequencies near ω_c. Such passband modulation finds use on channels that do not pass DC or on channels that several signals simultaneously share in non-overlapping frequency bands (and thus have different carrier frequencies).

This subsection first investigates a number of equivalent representations of a passband signal, the most interesting of which is the baseband-equivalent signal in Subsection 1.3.5.1. The design replaces the original modulated passband signal with the baseband-equivalent signal in transmission analysis. Subsection 1.3.5.1's objective is the generation of such an equivalent signal from the original signal. Subsection 1.3.5.1 essentially shows baseband-equivalent signals' frequency content doubles and translates to DC. Since all baseband signals center transmitted energy at DC, Subsection 1.3.5.1 shows that a common baseband-processing method applies to any passband channel. Figure 1.57 summarizes this entire subsection.

Passband Signal Equivalents The real-valued signal $x(t)$ is a **passband signal** when its nonzero Fourier transform values are near ω_c, as in Figure 1.52. Passband signals never have DC content, so

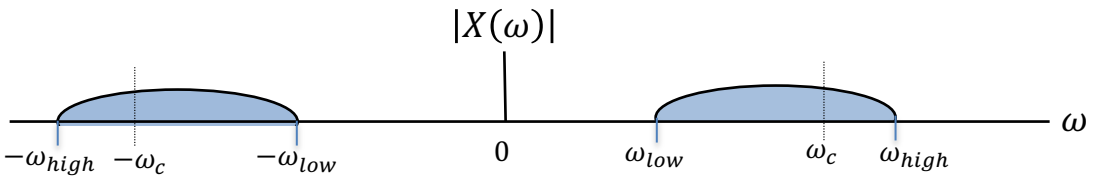


Figure 1.52: **Passband signal amplitude.**

$X(0) = 0$.

Definition 1.3.25 [Carrier-Modulated Signal] *A* **carrier-modulated signal** *is any passband signal that satisfies*

$$x(t) = a(t) \cdot \cos\left(\omega_c t + \theta(t)\right) \quad , \tag{1.336}$$

where $a(t)$ is the modulated signal's time-varying **amplitude** *or* **envelope** *and $\theta(t)$ is its time-varying* **phase**. *$\omega_c = 2\pi f_c$ is the* **carrier frequency** *(in radians/sec; f_c is in Hz).*

The carrier frequency ω_c is sufficiently large with respect to $a(t)$'s amplitude and phase variations that modulated signal's power spectral density has no significant energy at or near $\omega = 0$. See Figure 1.52, wherein the nonzero spectrum of $X(\omega)$ is in the passband $\omega_{low} < |\omega| < \omega_{high}$. In digital communication, $x(t)$ has an equivalent quadrature form using the trigonometric identity $\cos(u + v) = \cos(u)\cos(v) - \sin(u)\sin(v)$, leading to a quadrature decomposition:

Definition 1.3.26 [Quadrature Decomposition] *The* **quadrature decomposition** *of a carrier modulated signal is*

$$x(t) = x_I(t) \cdot \cos\left(\omega_c t\right) - x_Q(t) \cdot \sin\left(\omega_c t\right) \quad , \tag{1.337}$$

where $x_I(t) = a(t) \cdot \cos\left(\theta(t)\right)$ is the modulated signal's time-varying **inphase component**, *and $x_Q(t) = a(t) \cdot \sin\left(\theta(t)\right)$ is its time-varying* **quadrature component**.

Relationships determining $(a(t),\ \theta(t))$ from $(x_I(t),\ x_Q(t))$ are

$$a(t) = \sqrt{x_I^2(t) + x_Q^2(t)} \quad , \tag{1.338}$$

and

$$\theta(t) = \mathrm{Tan}^{-1}\left[\frac{x_Q(t)}{x_I(t)}\right] \quad . \tag{1.339}$$

In (1.339), the inverse tangent applies with known individual polarities of both numerator and denominator, so there is no quadrant ambiguity in computing $\theta(t)$.

Passband processing and analysis eliminate explicit carrier-frequency ω_c consideration and directly use only the inphase and quadrature components. These inphase and quadrature components combine into a two-dimensional vector, or into an equivalent complex signal. By convention, a graph of a quadrature-modulated signal plots the inphase component along the real axis and the quadrature component along the imaginary axis as Figure 1.53 shows.

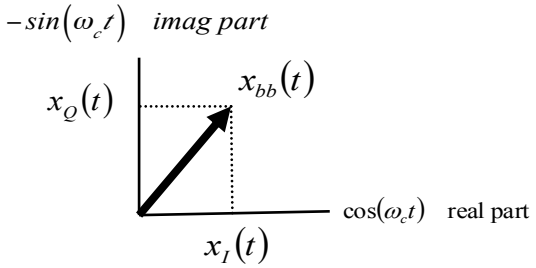


Figure 1.53: **Decomposition of baseband-equivalent signal.**

The resultant complex vector $x_{bb}(t)$ is the complex baseband-equivalent signal.

Definition 1.3.27 [Baseband-Equivalent Signal] *The complex* **baseband-equivalent signal** *for $x(t)$ in (1.336) is*

$$x_{bb}(t) \triangleq x_I(t) + \jmath x_Q(t) \quad , \tag{1.340}$$

where $\jmath = \sqrt{-1}$.

The baseband-equivalent signal expression no longer explicitly contains the carrier frequency ω_c. Another complex representation that explicitly uses ω_c is the analytic[39] equivalent signal for $x(t)$:

Definition 1.3.28 [Analytic-Equivalent Signal] *The* **analytic-equivalent signal** *for $x(t)$ in (1.336) is*

$$x_A(t) \triangleq x_{bb}(t) \cdot e^{\jmath\omega_c t} \quad . \tag{1.341}$$

The original real-valued passband signal $x(t)$ is the real part of the analytic-equivalent signal:

$$x(t) = \Re\{x_A(t)\} \quad . \tag{1.342}$$

The Hilbert Transform of $x(t)$, or $\check{x}(t)$, is the analytic signal's imaginary part

$$\check{x}(t) = \Im\{x_A(t)\} \quad . \tag{1.343}$$

(Appendix A provides more detail on the Hilbert Transform and a proof of (1.343).) Finally, the inphase component $x_I(t)$ and the quadrature component $x_Q(t)$ dervice from the signal $x(t)$ and its Hilbert transform $\check{x}(t)$ as (using $x_{bb}(t) = x_I(t) + \jmath x_Q(t) = x_A(t) \cdot e^{-\jmath\omega_c t}$):

$$x_I(t) = x(t) \cdot \cos(\omega_c t) + \check{x}(t) \cdot \sin(\omega_c t) \tag{1.344}$$

$$x_Q(t) = \check{x}(t) \cdot \cos(\omega_c t) - x(t) \cdot \sin(\omega_c t) \quad . \tag{1.345}$$

Thus, four equivalent forms for representing a real passband signal $x(t)$ with carrier frequency ω_c are:

1.	magnitude, phase	$a(t), \theta(t)$
2.	inphase, quadrature	$x_I(t), x_Q(t)$
3.	complex baseband	$x_{bb}(t)$
4.	analytic	$x_A(t)$

(1.346)

[39] This use of "analytic" should not be confused with the mathematician's definition of an analytic signal, which means the signal and all its derivatives are absolutely integrable over a specified domain - often used in Lapace and Z/D Transforms. In fact the "complex-analytic" signals developed here would not strictly satisfy this mathematical definition because of the zero-energy constraint at low frequencies.

EXAMPLE 1.3.8 [Translation between equivalent representations:] A passband QAM signal is

$$x(t) = \text{sinc}(10^6 t) \cdot \cos(2\pi 10^7 t) + 3 \cdot \text{sinc}(10^6 t) \cdot \sin(2\pi 10^7 t) \quad . \tag{1.347}$$

The carrier frequency is 10 MHz and the symbol period is 1 μs. The inphase and quadrature components are

$$\begin{aligned} x_I(t) &= \text{sinc}(10^6 t) && (1.348) \\ x_Q(t) &= -3 \cdot \text{sinc}(10^6 t) \quad , && (1.349) \end{aligned}$$

so

$$x_{bb}(t) = (1 - 3\jmath) \cdot \text{sinc}(10^6 t) \quad . \tag{1.350}$$

The baseband signal's amplitude and phasel are

$$\begin{aligned} a(t) &= \sqrt{10} \cdot \text{sinc}(10^6 t) && (1.351) \\ \theta(t) &= \text{Tan}^{-1}\left[\frac{-3}{1}\right] = -71.6^o \quad . && (1.352) \end{aligned}$$

Thus,

$$x(t) = \sqrt{10} \cdot \text{sinc}(10^6 t) \cdot \cos(\omega_c t - 71.6^o) \quad . \tag{1.353}$$

Finally,

$$x_A(t) = (1 - 3\jmath) \cdot \text{sinc}(10^6 t) \cdot e^{\jmath 2\pi 10^7 t} \quad . \tag{1.354}$$

Subsubection 1.3.5.1 next considers the Fourier Transform relationships of $x(t), x_{bb}(t),$ and $x_A(t)$.

Frequency Spectrum of Analytic- and Baseband-Equivalent Signals Figure 1.54 illustrates equations' use (1.342) and (1.343) to represent the analytic signal as

$$x_A(t) = x(t) + \jmath\check{x}(t) \quad . \tag{1.355}$$

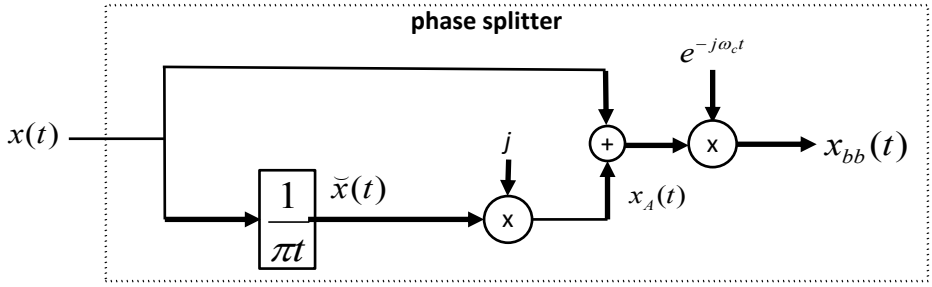


Figure 1.54: **Analytic signal composition.**

The Fourier Transform of (1.355) is[40]

$$\begin{aligned} X_A(\omega) &= \{1 + \text{sgn}(\omega)\} \cdot X(\omega) && (1.356) \\ &= \begin{cases} 2 \cdot X(\omega) & \omega > 0 \\ X(0) = 0 & \omega = 0 \\ 0 & \omega < 0 \end{cases} \quad . && (1.357) \end{aligned}$$

[40] If $\check{x}(t)$ is the Hilbert transform of $x(t)$, then the Fourier transform of $\check{x}(t)$ is $-\jmath\text{sgn}(\omega)X(\omega)$, where $X(\omega)$ is the Fourier Transform of $x(t)$, as shown in Appendix C.

The analytic equivalent signal, $x_A(t)$, has nonzero value only for $x(t)$'s positive frequencies and is identically zero for negative frequencies. The real signal $x(t)$'s Fourier transform $X(\omega)$ has two symmetry properties: The real part $\mathcal{R}\{X(\omega)\}$ is even in ω, while the imaginary part $\mathcal{I}\{X(\omega)\}$ is odd in ω. Knowledge of only the non-negative frequencies of $X(\omega)$, or equivalently the analytic signal, is sufficient for $X(\omega)$'s reconstruction. This confirms the analytic signal $x_A(t)$'s true "equivalence" with the original signal $x(t)$.

Using Equation (1.341), the baseband-equivalent signal's Fourier transform is simply the analytic signal's Fourier transform translated in frequency by ω_c. Thus

$$X_A(\omega) = X_{bb}(\omega - \omega_c) \tag{1.358}$$

and

$$X_{bb}(\omega) = X_A(\omega + \omega_c) \quad . \tag{1.359}$$

Use of (1.341) and (1.342) allows the passband signal $x(t)$'s reconstruction from the baseband-equivalent signal $x_{bb}(t)$ and the carrier frequency ω_c. The baseband-equivalent signal, in general, may be complex-valued, and thus Figure 1.55 shows an example spectrum for $x_{bb}(t)$ that is asymmetric about $\omega = 0$.

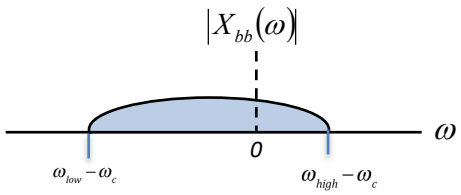


Figure 1.55: **Baseband signal spectrum.**

EXAMPLE 1.3.9 **[Continuing Example]** Figure 1.56 shows the original, baseband-, and analytic-equivalent spectra of the passband signal

$$x(t) = \text{sinc}(10^6 t) \cdot \cos(2\pi 10^7 t) + 3 \cdot \text{sinc}(10^6 t) \cdot \sin(2\pi 10^7 t) \quad . \tag{1.360}$$

The two complex signals' Fourier-transform magnitude doubles because these complex representations' nonzero signal components add together in a single positive-frequency band.

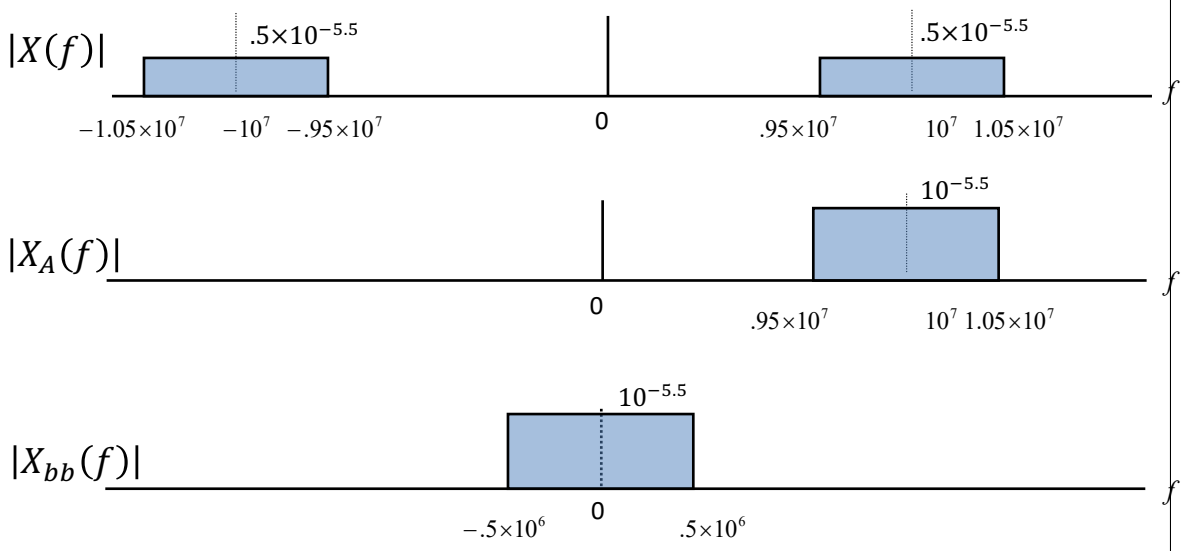


Figure 1.56: **Spectra equivalents for example waveform.**

Generation of the baseband equivalent Figure1.54's structure generates a signal's baseband equivalent, in which the second complex multiply simply is 4 real multiplies using Euler's formula $e^{\jmath\omega_c t} = \cos(\omega_c t) + \jmath\sin(\omega_c t)$. The first multiply by $\jmath$ alone is symbolic and simply means that the receiver processing views that path's signal as the imaginary part.

Passband Channels The filtered passband channel has a simple input/output relationship between the baseband equivalent signals. The output baseband equivalent is the multiplication of the baseband input spectra by the channel's Fourier transform translated from ω_c to DC, as this section shows. Filtering does affect the ML detector performance and complexity, which is discussed further in upcoming Subsection 1.3.7 and Chapter 3. In particular, if a passband linear channel processes a passband input signal, passband analysis directly determines the corresponding baseband- and analytic-equivalent filter-output representations for $y(t)$ in terms of the channel $h(t)$ and the input $x(t)$. Figure 1.57 summarizes the results that this section finds.

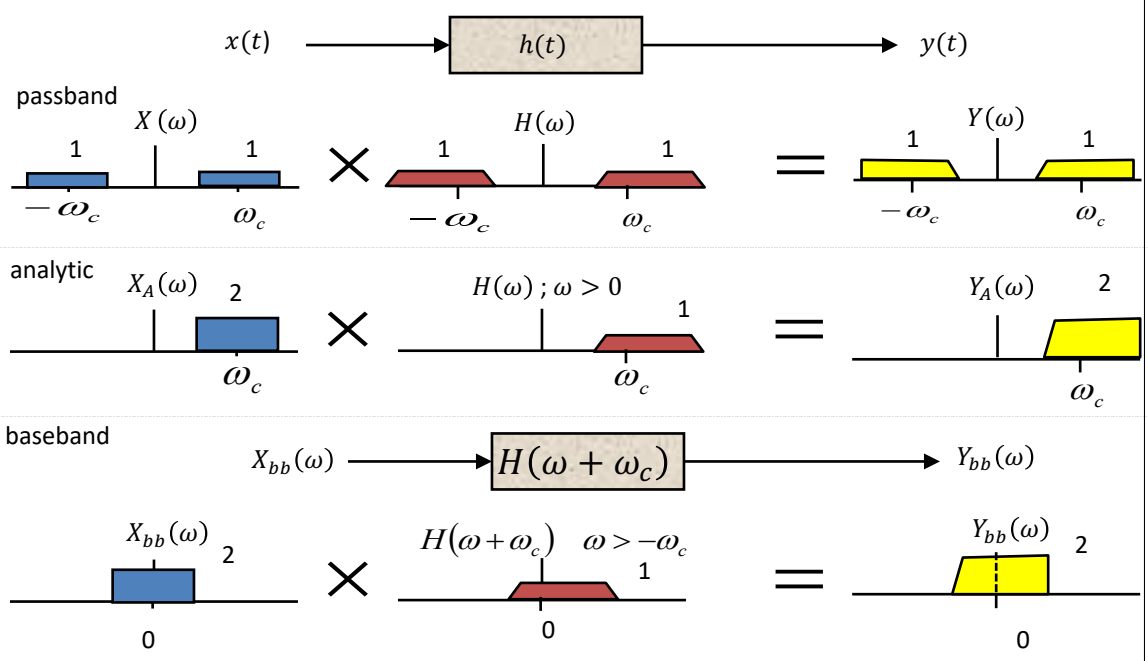


Figure 1.57: **Equivalent representations of the channel input/output relationship.**

Equivalent representations of the channel response. Any of the previous subsections four passband-signal representations $\{x_{bb}(t), x_A(t), x_I(t)$ and $x_Q(t)$ apply to the channel's impulse response and/or Fourier transform by substitution of h for x in the same equations. For instance, a linear time-invariant channels with real-valued impulse response $h(t)$ has **analytic-equivalent channel** $h_A(t)$ as

$$h_A(t) \triangleq h(t) + \jmath\check{h}(t) \quad . \tag{1.361}$$

Similarly in the frequency domain,

$$H_A(\omega) = \{1 + \mathrm{sgn}(\omega)\} \cdot H(\omega) \quad . \tag{1.362}$$

The **baseband-equivalent channel** follows the same equations as the baseband equivalent signal, except the carrier frequency ω_c is set equal to that of the input, and output, signals.

Definition 1.3.29 [Baseband Equivalent Channel (at carrier frequency ω_c)] *The* **baseband equivalent channel** *at any carrier frequency ω_c is*

$$h_{bb}(t) \triangleq h_A(t) \cdot e^{-\jmath\omega_c t} \quad . \tag{1.363}$$

For valid application of the term "baseband equivalent", the carrier frequency should be sufficiently large to guarantee that $h_{bb}(t)$ has no significant energy content at frequencies $|\omega| > \omega_c$, i.e., $|H_{bb}(\omega)| = 0, \ \forall\ |\omega| > \omega_c$.

The equivalent views of channel input/output relations Figure 1.57's passband-signal's frequency-domain representation (at the top) is

$$Y(\omega) = H(\omega) \cdot X(\omega) \quad . \tag{1.364}$$

Multiplying both sides of (1.364) by $1 + sgn(\omega)$ leads to (middle of Figure 1.57)

$$Y_A(\omega) = H(\omega) \cdot X_A(\omega) \tag{1.365}$$

$$\Rightarrow Y_A(\omega) = \left\{ H(\omega) \cdot \frac{1}{2} \cdot (1 + sgn(\omega)) \right\} \cdot X_A(\omega) = \left[\frac{1}{2} \cdot H_A(\omega) \right] \cdot X_A(\omega), \tag{1.366}$$

where (1.366) follows because the input has nonzero spectra only for positive frequencies. Thus, only the channel filtering at those same positive frequencies (recalling that the factor $(1/2) \cdot [1 + sgn(\omega)] = 0 \; \forall \; \omega < 0$) has interest. More importantly, since the linear time-invariant passband channel $h(t)$ only scales and phase shifts each frequency independently, the output $y(t)$ has its power spectral density concentrated in the same frequency region[41] as the input $x(t)$. Shift of the output spectrum $y(t)$ down by ω_c yields

$$Y_{bb}(\omega) = Y_A(\omega + \omega_c) = \left[\frac{1}{2} H_A(\omega + \omega_c) \right] X_A(\omega + \omega_c) \tag{1.367}$$

$$= \left[\frac{1}{2} H_{bb}(\omega) \right] X_{bb}(\omega) \tag{1.368}$$

$$= H(\omega + \omega_c) \cdot X_{bb}(\omega) \;\; \omega > -\omega_c \;\; , \tag{1.369}$$

which appears at the bottom of Figure 1.57. This leads to the definition of the **baseband equivalent system**

Definition 1.3.30 [Baseband Equivalent System] *The* **baseband equivalent system** *for a passband system described by $y(t) = x(t) * h(t)$, where $x(t)$ is a passband signal,*

$$y_{bb}(t) = \left(x_{bb}(t) * \frac{1}{2} h_{bb}(t) \right) \tag{1.370}$$

or

$$Y_{bb}(\omega) = H(\omega + \omega_c) \cdot X_{bb}(\omega) \quad . \tag{1.371}$$

Obtaining the baseband equivalent channel is easy! Simply slide the channel's Fourier transform response down to DC. Because the channel may be asymmetric with respect to ω_c, the baseband equivalent channel can be complex and usually is. The baseband-equivalent channel representation removes the notational complexity of cosines, sines, and carrier frequencies. A common analysis framework thus applies to any channel with any carrier frequency, which facilitates many digital-transmission analyses. This is why baseband-equivalent channels dominate in their use in digital-transmission analysis. The analysis convolves the baseband-equivalent input with the complex channel corresponding to $H(\omega+\omega_c)$ to get the baseband-equivalent output. A channel that is not passband, but rather initially real baseband, simply corresponds to the baseband equivalent input/output representation with all imaginary parts zeroed, and $H(\omega)$ used directly ($\omega_c = 0$).

The input/output relationships can thus be summarized as follows: For the passband signals and systems,

$$y(t) = x(t) * h(t) \tag{1.372}$$

$$Y(\omega) = X(\omega) \cdot H(\omega). \tag{1.373}$$

For the analytic-equivalent system,

$$y_A(t) = x_A(t) * \frac{1}{2} h_A(t) \tag{1.374}$$

$$Y_A(\omega) = X_A(\omega) \cdot H(\omega) \;\; \forall \omega \geq 0 \; . \tag{1.375}$$

[41] or a smaller region if the channel zeroes a band

For the baseband equivalent system,

$$y_{bb}(t) = x_{bb}(t) * \frac{1}{2}h_{bb}(t) \tag{1.376}$$

$$Y_{bb}(\omega) = X_{bb}(\omega) \cdot H(\omega + \omega_c) \; \forall \; \omega > -\omega_c. \tag{1.377}$$

Any of these three equivalent relations (and ω_c) fully describe the passband system.

EXAMPLE 1.3.10 [Bandpass channel for previous bandpass signals] A channel impulse response is $h(t) = 2 \times 10^6 \cdot \text{sinc}(10^6 t) \cdot \cos(2\pi 10^7 t)$ corresponds to

$$H(f) = \begin{cases} 1 & |f \pm 10^7| < .5 \times 10^6 \\ 0 & \text{elsewhere} \end{cases} \quad . \tag{1.378}$$

Then

$$h_I(t) = 2 \times 10^6 \cdot \text{sinc}(10^6 t)$$

and

$$h_Q(t) = 0 \quad ,$$

so that $h_{bb}(t) = h_I(t)$ or

$$H_{bb}(f) = \begin{cases} 2 & |f| \leq 500 \text{ kHz} \\ 0 & \text{elsewhere} \end{cases} \quad .$$

Using Examples 1.3.8 and 1.3.9's signal as the channel input, the corresponding channel output is

$$Y_{bb}(f) = H(f + f_c) \cdot X_{bb}(f) = 1 \cdot X_{bb}(f) \quad |f| < 500 \text{ kHz} \tag{1.379}$$

or

$$x_{bb}(t) * \frac{1}{2}h_{bb}(t) = x_{bb}(t) = (1 - 3\jmath) \cdot \text{sinc}(10^6 t) \quad . \tag{1.380}$$

Zero quadrature-channel component, or $h_Q(t) = 0$, does **not** mean that no quadrature signal components pass to the channel output – it means that inphase components remain inphase components at the channel output, and quadrature components similarly then remain quadrature components at the channel output, otherwise known as **zero phase distortion**. zero phase distortion also occurs when the phase is linear versus frequency, in which case the phase's negative slope corresponds only to each component's delay, but no phase crosstalk.

Appendix A extends this section's results to passband random processes (this section only considered deterministic signals) on passband deterministic channels. The next subsection considers additive random noise at the channel output.

1.3.5.2 Baseband-Equivalent AWGN Channel

This subsection investigates a passband filtered AWGN channel's action on transmitted signals, enabling a common analysis of real (e.g., PAM inputs) and complex (e.g., QAM) channels. Appendix A's passband-random-process results find use here, so familiarity with that appendix is helpful, although not completely necessary. Figure 1.58 summarizes a scaling factor that appears, explicitly or tacitly, in all passband-random-process developments . This simple scale factor maintains consistent analysis in all regards with those appearing earlier in this chapter.

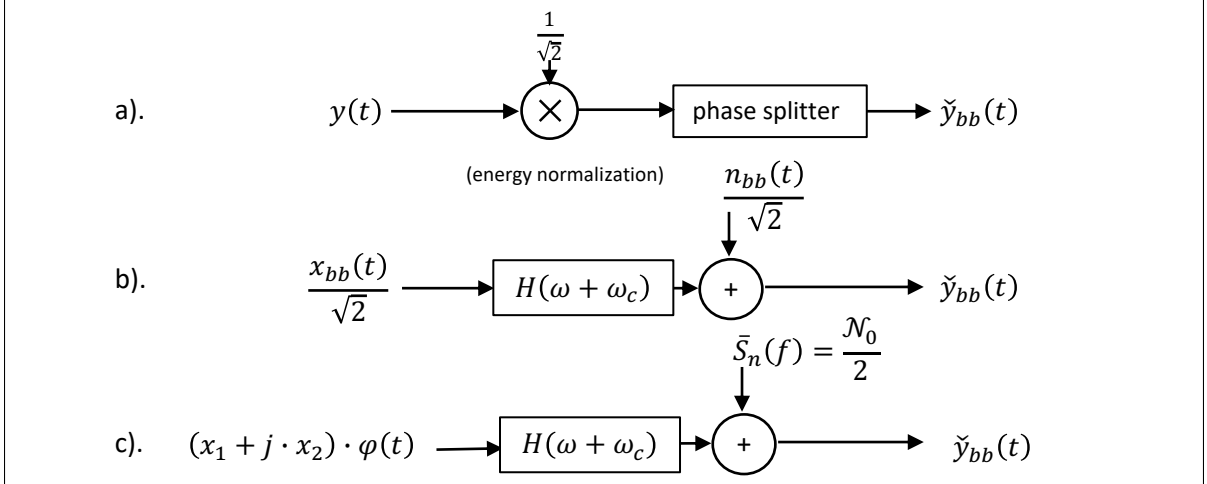


Figure 1.58: **Scaling for passband white Gaussian noise**

Figure 1.58a's receiver processes the channel output $y(t)$ by the cascade of a scale factor $\frac{1}{\sqrt{2}}$ and a phase splitter to generate a baseband equivalent signal $\breve{y}_{bb}(t)$. Communication engineers often drop the "cup" so that the splitter output is simply $y_{bb}(t)$, implying the scaling factor's inclusion to avoid notational proliferation. Generally speaking, the phase splitter adds a signal to j times its own Hilbert transform. A random processes' Hilbert transform has the same power as the original process (Appendix A). Thus, the phase splitter generally doubles power. The scale factor $1/\sqrt{2}$ that precedes Figure 1.58's phase splitter causes the power of $\breve{y}_b b(t)$ and $y(t)$ to be the same. The power-scaling occurs equally for both the noise and signal, since both are present in the channel output $y(t)$. So, performance remains the same (no matter what the scale factor is), and the ratio of minimum distance to noise standard deviation also remains invariant under any scaling. Nonetheless, Figure 1.58's particular scaling makes the ensuing analysis consistent with this chapter's earlier real-channel results when complex signals' use generalizes designs.

For analysis, the scale factor can be "pushed back" through the channel and to the noise, and the Figure 1.58b's baseband-equivalent system corresponds to Figure 1.58's system with the explicit baseband-input-signal scaling. The scale factor then occurs separately in each of the output's noise and signal components. The next two subsections investigate this scaling and illustrate its consistency with previous results. Since there is a factor of $1/\sqrt{2}$ in both inphase and quadrature QAM basis functions, this factor then is already present conceptually in Figure 1.58c's equivalent system, and thus the scaled noise $\frac{n_{bb}(t)}{\sqrt{2}}$ and the baseband channel $H(\omega+\omega_c)$ are the proper combination to represent the filtered AWGN with complex baseband input symbol (x_1+jx_2) and the QAM basis-function component $\varphi(t)$. The noise per dimension also remains $\frac{\mathcal{N}_0}{2}$.

Noise Scaling Appendix A shows that a random process' analytic-equivalent power spectral density is four times the original signal's power spectral density for $\omega \geq 0$ (and is zero for $\omega < 0$), which tacitly implies the random process power's doubling .[42]

Since the scaled WGN, $n_{bb}(t)/\sqrt{2}$, in Figure 1.58 has power spectral density.

$$\mathcal{S}_n(\omega) = \frac{\mathcal{N}_0}{4} \quad , \tag{1.381}$$

the power spectrum of the analytic equivalent of $n_{bb}(t)/\sqrt{2}$ is

$$S_A(\omega) = \begin{cases} \mathcal{N}_0 & \omega > 0 \\ \frac{\mathcal{N}_0}{2} & \omega = 0 \\ 0 & \omega < 0 \end{cases} \quad . \tag{1.382}$$

[42] The autocorrelation of the analytic equivalent noise is $r_A(\tau) = 2\left(r_n(\tau) + j\check{r}_n(\tau)\right)$. See Appendix B for more details.

The baseband-equivalent noise has power spectrum that simply translates $S_A(\omega)$ to baseband, or

$$S_{bb}(\omega) = \begin{cases} \mathcal{N}_0 & \omega > -\omega_c \\ \frac{\mathcal{N}_0}{2} & \omega = -\omega_c \\ 0 & \omega < -\omega_c \end{cases} \quad . \tag{1.383}$$

Strictly speaking, $S_A(\omega)$ and $S_{bb}(\omega)$ do not correspond to white noise. However, practical systems will always use a carrier frequency that is at least equal to the signal frequency, ω_{high}, that corresponds to the highest-frequency nonzero baseband signal component – that is, the design always modulates with a carrier frequency large enough to "get away" from DC. In this case, the baseband equivalent' s power spectrum appears as if it were "white" or flat at $\mathcal{N}_0$ for all frequencies of practical interest. This baseband demodulated noise signal is complex AWGN with power spectral density $\mathcal{N}_0$, and correspondingly power spectral density $\frac{\mathcal{N}_0}{2}$ for each real dimension.

For Figure 1.58's scaled phase-splitting arrangement, the baseband-equivalent WGN is:

Definition 1.3.31 [Baseband Equivalent WGN] Baseband Equivalent White Gaussian Noise *is a random process, $\check{n}_{bb}(t)$, that Figure 1.58's demodulator generates. The complex random process' autocorrelation, $r_{bb}(\tau)$ is thus*

$$r_{bb}(\tau) = \mathcal{N}_0 \cdot \delta(\tau) \quad , \tag{1.384}$$

and the corresponding power spectral density is thus

$$S_{bb}(f) = \mathcal{N}_0 \quad . \tag{1.385}$$

From Appendix A, the baseband autocorrelation is

$$r_{bb}(\tau) = 2r_I(\tau) = 2r_Q(\tau) = \mathcal{N}_0 \cdot \delta(\tau) \quad , \tag{1.386}$$

so that the inphase and quadrature noises each have power spectral density $\frac{\mathcal{N}_0}{2}$ and are each AWGN signals. Further, from Appendix A and (1.386),

$$r_{IQ}(\tau) = 0 \quad , \tag{1.387}$$

that is, the inphase and quadrature noises are uncorrelated for all time lags τ with baseband-equivalent WGN.

The complex baseband noise is two dimensional (two real dimensions), and the noise variance per dimension is thus $\frac{\mathcal{N}_0}{2}$, which is the reason for the Figure 1.58's scaling. This scaling makes the noise variance per dimension the same as earlier in this chapter.

Signal Scaling A brief review of Section 1.2's basis-function modulator assists understanding of scaling's effect: The two normalized QAM passband functions for one-shot AWGN transmission are again

$$\varphi_1(t) = \sqrt{2} \cdot \varphi(t) \cdot \cos(\omega_c t) \tag{1.388}$$

$$\varphi_2(t) = -\sqrt{2} \cdot \varphi(t) \cdot \sin(\omega_c t) \quad , \tag{1.389}$$

where for practical reasons, ω_c is high enough.[43] The modulated signal

$$x(t) = x_1 \cdot \varphi_1(t) + x_2 \cdot \varphi_2(t) \tag{1.390}$$

$$= \sqrt{2}\{x_1 \cdot \varphi(t) \cdot \cos(\omega_c t)\} - \sqrt{2}\{x_2 \cdot \varphi(t) \cdot \sin(\omega_c t)\} \quad , \tag{1.391}$$

[43] One verifies that these functions are indeed normalized – if $\varphi(t)$ is normalized – by investigating their power spectra under modulation.

has baseband equivalent signal

$$x_{bb}(t) = \sqrt{2} \cdot (x_1 + \jmath x_2) \cdot \varphi(t) \quad . \tag{1.392}$$

Figure 1.58c's scaling removes the extra factor of $\sqrt{2}$ that arose through the modulating basis-function's normalization. Figure 1.58c explicitly shows this removal so that the system appears as a complex baseband system with complex input

$$\breve{x}_{bb}(t) = (x_1 + \jmath x_2) \cdot \varphi(t) \quad . \tag{1.393}$$

Equation (1.393) becomes

$$\breve{x}_{bb}(t) = \boldsymbol{x}_{bb} \cdot \varphi(t) \tag{1.394}$$

where

$$\boldsymbol{x}_{bb} \stackrel{\Delta}{=} (x_1 + \jmath x_2) \quad . \tag{1.395}$$

Equations (1.394) and (1.395) constitute a single-dimension **complex** baseband representation of the QAM modulator with (now normalized) basis function $\varphi(t)$ that is entirely consistent in all regards with the two-real-dimensional representation. The complex signal constellation's average energy is

$$\mathcal{E}_{bb} = \mathcal{E}_{\boldsymbol{x}} = 2\bar{\mathcal{E}}_{\boldsymbol{x}} \quad , \tag{1.396}$$

which maintains the convention that a complex signal is equivalent to a two-dimensional real signal in defining $\bar{\mathcal{E}}_{\boldsymbol{x}}$.

The cups are necessary for introductory baseband analysis, but the literature generally drops them without comment. So complex channel designs equivalent to Figure 1.59 need not include them, and the signal/noise scaling remains consistent because of Figure 1.58 $1/\sqrt{2}$ factor[44]

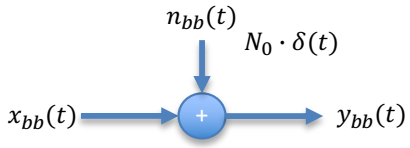


Figure 1.59: **Baseband AWGN channel**

Furthermore, Figure 1.59 often represents one-dimensional real systems where there are no passband modulation effects. In this case, the quadrature (imaginary) dimension is tacitly zeroed, while the real dimension carries the signal and has noise power spectral density $\frac{\mathcal{N}_0}{2}$, entirely consistent with this chapter's earlier developments.

1.3.5.3 Conversion to a baseband equivalent channel

This subsection applies baseband analysis to a QAM system with baseband-equivalent channel and noise. The system then looks like a PAM system except that inputs, outputs and internal quantities are all complex with the real dimension corresponding to the "cosine" modulated component and the imaginary dimension corresponding to the "sine" modulated component. After moving to a complex baseband equivalent, the effect of the carrier is absent in all subsequent analysis.

[44] This whole development may seem tedious for what appears a trivial final result. The author's experience in teaching these basics over decades though finds that they initially confuse many students new to the area. The development here almost universally eliminates that confusion.

Carrier demodulation for baseband-channel generation Figure 1.60's baseband-equivalent output-signal's generation has two equivalent forms: Figure 1.60(a) repeats Figure 1.54's "phase-splitter" analytic-equivalent signal generation to obtain $y_A(t)$. The noise power-spectral density and baseband input modulator each absorb the scale factor for convenient analysis, as in the last subsection and Figure 1.55. The carrier demodulator multiplies the analytic signal $y_A(t)$ by $e^{-j\omega_c t}$ to generate $y_{bb}(t)$. Figure 1.60(b) illustrates a more obvious form of generating $y_{bb}(t)$ that is sometimes used in practice. Figure 1.60(b)'s structure generates the inphase and quadrature components, $y_I(t)$ and $y_Q(t)$ by multiplying $y(t)$ by $2\cos(\omega_c(t))$ and $2\sin(\omega_c(t))$ in parallel. Then,

$$\begin{aligned} 2\cdot\cos(\omega_c(t))y(t) &= y_I(t)\cdot 2\cdot\cos(\omega_c(t))^2 - y_Q(t)\cdot 2\cdot\sin(\omega_c(t))\cos(\omega_c(t)) \\ &= y_I(t)\cdot(1+\cos(2\omega_c t)) - y_Q(t)\cdot\sin(2\omega_c(t)) \end{aligned}$$

and

$$\begin{aligned} 2\cdot\sin(\omega_c(t))y(t) &= y_I(t)\cdot 2\cdot\cos(\omega_c(t))\sin(\omega_c(t)) - y_Q(t)\cdot 2\cdot\sin(\omega_c(t))^2 \\ &= y_I(t)\cdot\sin(2\omega_c t) + y_Q(t)\cdot(\cos(2\omega_c(t)) - 1) \end{aligned}$$

Lowpass filtering of $2\cdot[\cos(\omega_c(t))]\cdot y(t)$ and $2\cdot[\sin(\omega_c(t))]\cdot y(t)$ removes the signal artifacts centered at $2\omega_c$. Figure 1.60 (b)'s two identical lowpass filters are usually easier to implement than Figure 1.60 (a)'s Hilbert filter. More sophisticated designs, especially those involving equalization (see Chapter 3), may prefer Figure 1.60 (a)'s implementation because it has advantages in carrier-phase locking (see Chapter 6).

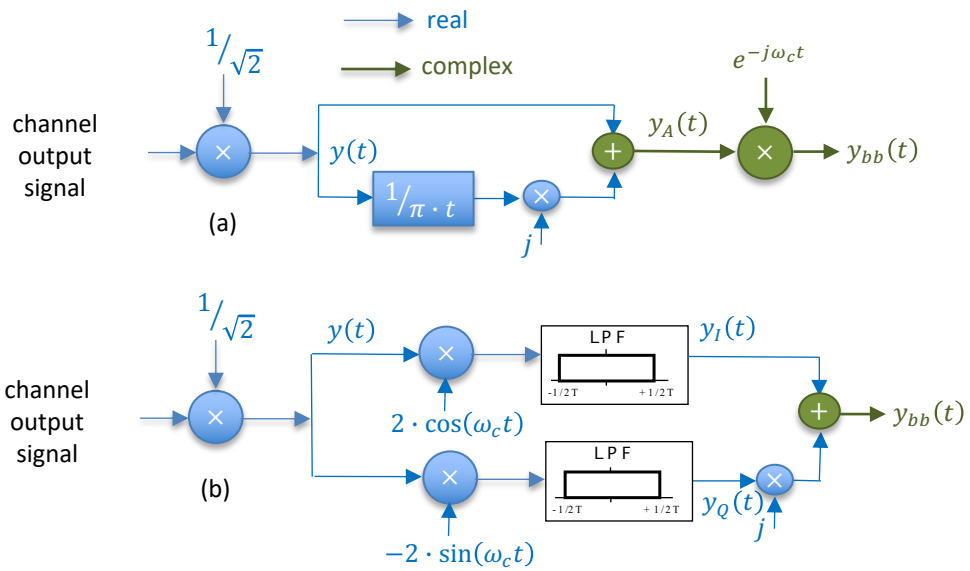


Figure 1.60: **Complex demodulator.**

EXAMPLE 1.3.11 [demodulation of a specific signal] As in Section 1.3.5.1's Example 1.3.8, a passband AWGN-channel output QAM signal is

$$z(t) = \text{sinc}(10^6 t)\cdot\cos(2\pi 10^7 t) + 3\cdot\text{sinc}(10^6 t)\cdot\sin(2\pi 10^7 t) + n(t) \quad . \tag{1.397}$$

The carrier frequency is 10 MHz and the symbol period is 1 μs. $z(t)$ is Figure 1.60(a)'s channel output signal that enters the complex demodulator. The scaled by $\frac{1}{\sqrt{2}}$ signal is

$$y(t) = \frac{z(t)}{\sqrt{2}} = \frac{1}{\sqrt{2}}\cdot\text{sinc}(10^6 t)\cdot\cos(2\pi 10^7 t) + \frac{3}{\sqrt{2}}\cdot\text{sinc}(10^6 t)\cdot\sin(2\pi 10^7 t) + \frac{n(t)}{\sqrt{2}} \quad , \tag{1.398}$$

which views the $1/\sqrt{2}$ scaling as performed in the channel. This scaled signal's Hilbert Transform in Figure 1.60(a)'s lower parallel path is

$$\check{y}(t) = \frac{1}{\sqrt{2}} \cdot \text{sinc}(10^6 t) \cdot \sin(2\pi 10^7 t) - \frac{3}{\sqrt{2}} \cdot \text{sinc}(10^6 t) \cdot \cos(2\pi 10^7 t) + \check{n}(t) \quad . \tag{1.399}$$

Multiplying the Hilbert Transform by $\jmath$ and adding to the upper unchanged component ($y(t)$) creates the analytic signal

$$y_A(t) = y(t) + \jmath \check{y}(t) \quad , \tag{1.400}$$

which after multiplication by the carrier-demodulating term $e^{-\jmath\omega_c t}$ provides a baseband equivalent signal $y_{bb}(t) = e^{-\jmath\omega_c t} \cdot y_A(t)$ or

$$y_{bb}(t) = \frac{1 - 3\jmath}{1000 \cdot \sqrt{2}} \cdot 1000 \cdot \text{sinc}(10^6 t) + n_{bb}(t) \quad . \tag{1.401}$$

This baseband has real signal component of $.001/\sqrt{2}$ and an imaginary component of $-.003/\sqrt{2}$. Each dimension;s noise component is $\frac{\mathcal{N}_0}{2}$. These are the same components that were associated with this signal earlier in this chapter as

$$x(t) = \text{sinc}(10^6 t) \cdot \cos(2\pi 10^7 t) + 3 \cdot \text{sinc}(10^6 t) \cdot \sin(2\pi 10^7 t) + n(t) = x_1 \cdot \varphi_1(t) + x_2 \cdot \varphi_2(t) \tag{1.402}$$

where the **normalized** basis functions are

$$\varphi_1(t) = \sqrt{2} \cdot 1000 \cdot \text{sinc}(10^6 t) \cdot \cos(\omega_c t) \tag{1.403}$$

$$\varphi_2(t) = -\sqrt{2} \cdot 1000 \cdot \text{sinc}(10^6 t) \cdot \sin(\omega_c t) \quad . \tag{1.404}$$

If the baseband signal $y_{bb}(t)$ is now passed through the matched filter $1000 \cdot \text{sinc}(10^6 t)$ and sampled at time 0, the two components $.001/\sqrt{2}$ and $-.003/\sqrt{2}$ are the real and imaginary output dimensions. Each dimension's relevant independent noise component is $\frac{\mathcal{N}_0}{2}$, or equivalently the original white noise's power-spectral density, which is equal to its variance per dimension with normalized basis functions.

The subsequent subsection now adds channel filtering with impulse response $h(t)$ to the system and investigates the filtered-AWGN channel's modeling as a complex baseband equivalent for QAM transmission.

Generating the baseband equivalent: some examples Transmission design considers a variety of transmission-channel information. This information is ooften not in a convenient initial form. The designer may spend considerable time and effort to understand and model the channel. Such effort often includes the baseband-equivalent channel model's construction. A first example explores a two-ray mathematical channel model. The second example starts converts a transmission-line model into an acceptable baseband response.

EXAMPLE 1.3.12 [Two-ray wireless channel] Wireless transmission sometimes use a "two-ray" model. This model has two paths between a transmit antenna and a receive antenna, one direct "line-of-sight" and the other delayed and attenuated through reflection. The reflected path usually represents a reflection from a building, mountain, or other physical entity. The second path has a delay (say by τ=1.1 μs) and signal attenuation (say 90% of the first path's amplitude) with respect to the first path. At the carrier frequency, the reflected signal also has 180 degrees of phase shift. The channel impulse response is then

$$h(t) = g \cdot [\delta(t) - .9 \cdot \delta(t - \tau)] \quad , \tag{1.405}$$

where g is an attenuation factor that models the path loss and antenna losses. The Fourier transform is

$$H(f) = g \cdot \left[1 - .9 \cdot e^{-\jmath 2\pi f \tau}\right] \quad . \tag{1.406}$$

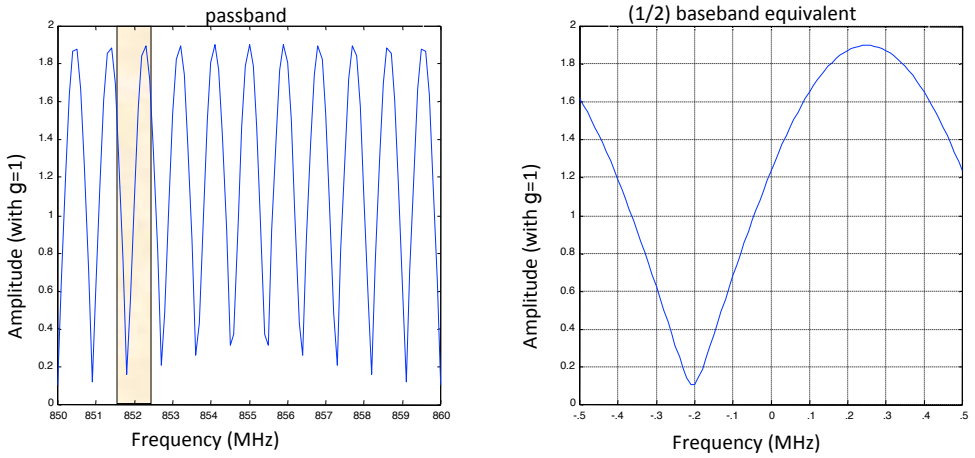


Figure 1.61: **Illustration of passband and (1/2 times) baseband-equivalent for wireless two-ray multipath channel.**

The noise is white and combines several compoents, natural and man-made with one-sided PSD -150 dBm/Hz. Wireless systems often use carrier frequencies between 800 and 900 MHz, so a design chooses a carrier for QAM modulation at 852 MHz and further choose 4QAM transmission with a symbol rate of $1/T$=1.0 MHz. Thus, frequencies between 852-.5 = 851.5 MHz and 852+.5=852.5 MHz are of interest, corresponding to a baseband-equivalent channel with frequencies between -500 kHz and + 500 kHz.

The complex-channel model for this transmission is then

$$\frac{1}{2}h_{bb}(t) = g \cdot [\delta(t) - .9 \cdot \delta(t-\tau)] \cdot e^{-\jmath 2\pi f_c t} \tag{1.407}$$

with Fourier transform

$$\frac{1}{2}H_{bb}(f) = H(f+f_c) = g \cdot \left[1 - .9 \cdot e^{-\jmath 2\pi (f+f_c)\tau}\right] \quad . \tag{1.408}$$

Figure 1.61 plots the passband channel magnitude from 850 MHz to 860 MHz, along with the complex baseband-equivalent channel's magnitude (with scaling factor of $1/\sqrt{2}$ included) from -500 kHz to 500 kHz. The channel clearly has "notching" effects because of the possibility of the second path adding out-of-phase (with phase π) at some frequencies. Wider QAM-transmission bandwidth more likely overlaps one (or more) of the transmission band "notches." Thus, this "multipath" distortion will lead to a non-flat or filtered-AWGN channel response (which means the techniques of Chapter 3 and later chapters are necessary for reliable recovery of messages). The baseband-equivalent is clearly not symmetric about frequency zero, meaning its baseband-equivalent impulse response is complex, as the formula above in (1.407) also implies. The baseband-equivalent response's real and imaginary parts appear in Figure 1.62. The baseband-equivalent noise for the model introduced in this Section is still white and has $\mathcal{N}_0$=-150 dBm/Hz, or equivalently $\mathcal{N}_0 = 10^{-18}$. For typical values of g in well designed transmission systems, this will be a few orders of magnitude below the signal levels. It's very simple in this case: Slide the Fourier transform in the band of interest down to DC, then set the complex noise level equal to the single-sided PSD.

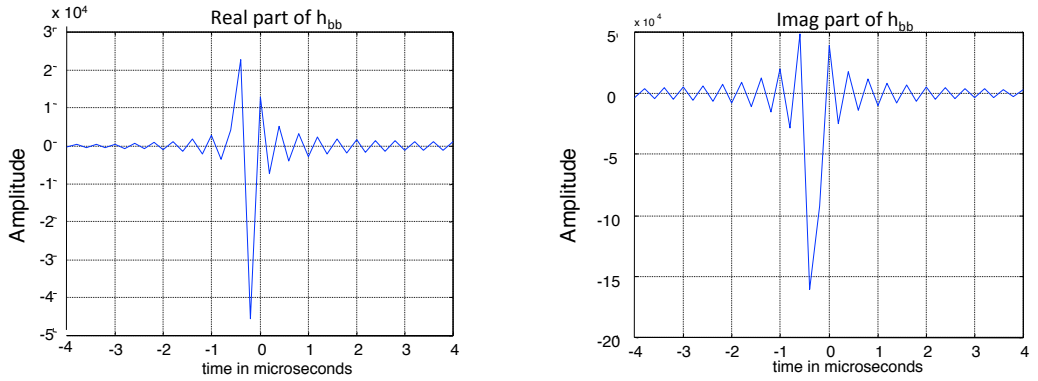


Figure 1.62: **Real and imaginary parts of (scaled) baseband equivalent channel response for two-ray example.**

The 2-ray model that easily led to a compact mathematical description is often inaccurate. More likely, the designer must measure the channel frequency attenuation in dB at several transmission-band frequencies, along with the measured channel signal delay at these same frequencies. The designer may also need to measure the noise power spectral density at these same frequencies (if not AWGN). The conversion to a complex baseband channel may be tedious, but follows the same steps as in the next example.

EXAMPLE 1.3.13 [Telephone Line Channel]

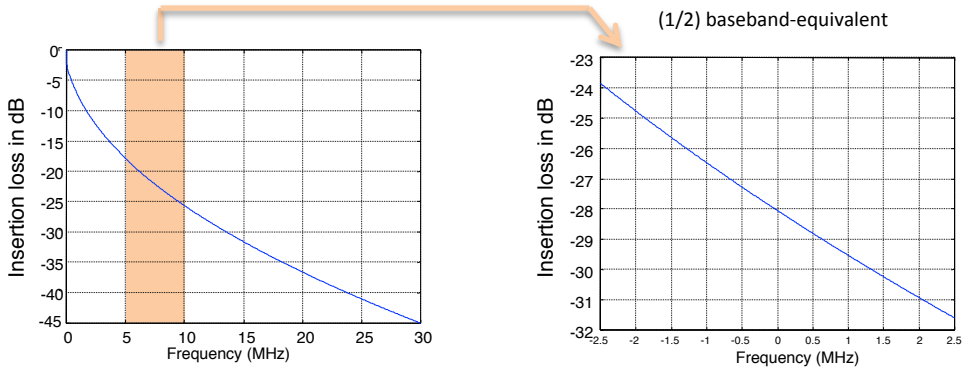


Figure 1.63: **Passband insertion loss and (1/2 times) baseband-equivalent transfer function for home-phone network example.**

Telephone lines today are sometimes used for data transmission within the home, and this example looks at a 10 Mbps data rate. The carrier frequency is 7.5 MHz and the symbol rate is 5 MHz for a 4 QAM signal. Telephone line attenuation versus frequency is often measured in terms of "insertion loss" in dB, a ratio of the voltage at the line output to the voltage at the same load point when the phone line is removed. For a well-matched system, it can be determined that this insertion loss is 6 dB above the transfer function from source to load, which is the desired function for digital transmission analysis. Figure 1.63 plots the insertion loss in dB for a 26-gauge phone line of length 300 meters. The baseband equivalent channel response is in the frequency range from 5 MHz to 10 MHz, which the designer "slides" so that 7.5 MHz now appears as DC, as also illustrated in Figure 1.63. Figure 1.63 increases

the baseband characteristic by 6 dB to get the transfer function. The slid baseband complex channel automatically includes the scale factor of 1/2. The designer would presumably obtain or measure the insertion loss at a sufficient number of frequencies between 5 and 10 MHz, store those values in a file, and then analyze them with this text's design methods. To use common digital signal processing operations like the inverse Discrete Fourier Transform, the measured values will need to be equally spaced in frequency between 5 MHz and 10 MHz. Perhaps 501 measurements with spacing 10 kHz have been taken. These 501 values form the channel transfer-function amplitudes (after conversion of dB back into linear-scale values) at the frequencies 5 MHz, 5.01 MHz, ... 10 MHz, or for baseband (increased by 6 dB to compute transfer function from insertion loss) equivalent from -2.5 MHz to 2.5 MHz.

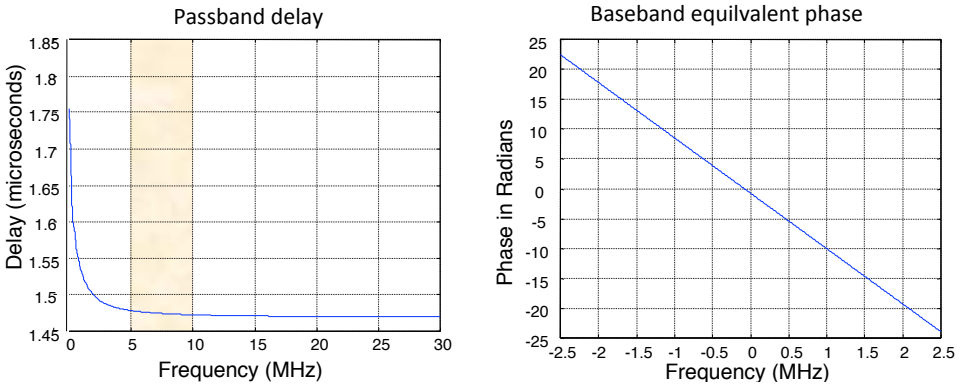


Figure 1.64: **Passband delay and baseband-equivalent phase for home-phone network example.**

Figure 1.64 plots channel delay measurements at all these same frequencies in microseconds. The index runs as $n = 0, ..., 500$ across the frequency band of interest. Since delay is the negative phase derivative, phase angle calcuation accumulates delay (with minus sign) from -2.5 MHz to each and every frequency index as

$$\angle H_{bb}(-2.5\text{MHz} + n \cdot .01\text{MHz}) = \theta_0 - \sum_{i=0}^{n} \text{Delay}\left[H_{bb}(i)\right] \quad . \tag{1.409}$$

θ_0 is an constant arbitrary phase reference that ultimately has no effect on transceiver performance, and thus usually taken to be 0. The baseband equivalent channel (scaled by 1/2) is the inverse DFT (IFFT command in Matlab) of vector of values $H_{bb}(n)$ $n = 0, ..., 500$. Because of the arbitrary phase, the time-domain response is usually not centered and has nonzero components at the reponse's beginning and end. Simple circular shift (already included in Figure 1.64) will provide a "centered" $h_{bb}(t)$ sampled at the symbol rate (which can be made causal by simple reindexing of the time axis). The transmit psd of the 4QAM signal is about -57 dBm/Hz, so that the power is then about 10 dBm (or 10 milliwatts). To interpolate the baseband response to finer time-resolution than the symbol rate, a band wider than 5-10 MHz must be measured, translated to DC, and then inverse transformed.

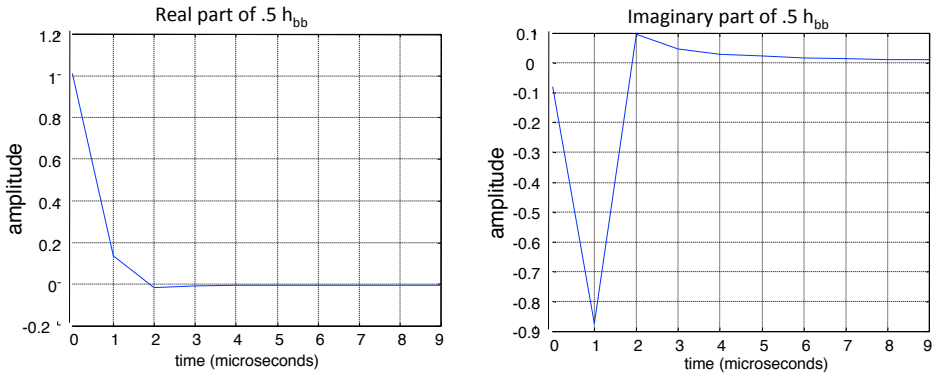


Figure 1.65: **Baseband equivalent complex channel for home-phone network example.**

An interesting effect in telephone-line transmission is that neighbors' data signals can be "heard" through electromagnetic coupling between phone lines in phone cables "upstream." This "crosstalk" then can return into other homes. This crosstalk can thus contribute to noise. Thus, the noise is not "white," and a simple model for the one-sided power spectral density of this noise has power spectral density:

$$-187 + 15\log_{10}(f) \text{ dBm/Hz} \quad . \tag{1.410}$$

This power-spectral density can be computed with f values from 5 MHz to 10 MHz, and then translated to baseband to obtain the baseband-equivalent power spectral density, as in Figure 1.66.

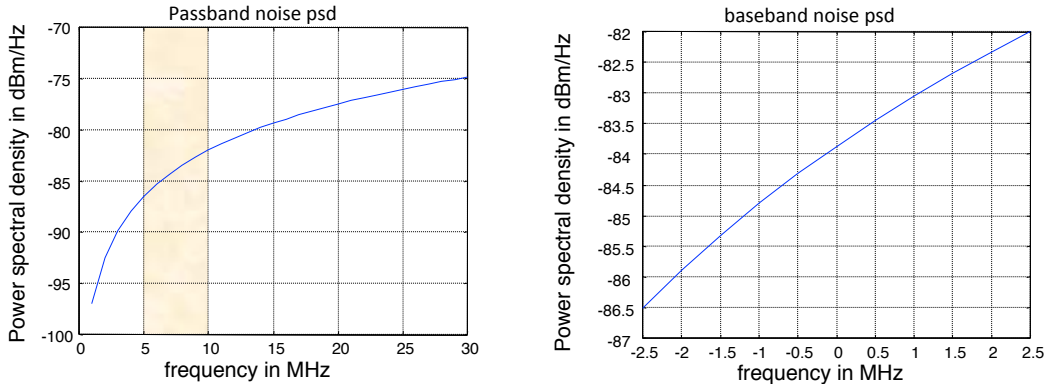


Figure 1.66: **Baseband-equivalent noise power spectrum for home-phone network example.**

To find Section 1.3.7's so-called "white-noise equivalent" for this complex baseband-equivalent channel, the inverse noise psd can be IFFT'd to the time-domain and factored using the roots command in matlab. Terms with roots of magnitude greater than 1 correspond to the minimum-phase factorization, said inverse can then be convolved with the channel $.5 \cdot h_{bb}(kT)$ to find Subsection 1.3.7's white-noise equivalent channel samples at the symbol rate.

This last example seems like much tedious work, but it is perhaps simple compared to what communication engineers do. The example emphasizes how important it is for communications designers

to know and well model their channel so that the theories and guidance learned from this text can be applied.

Complex generalization of inner products and analysis Optimum demodulation theory for complex signals with baseband-equivalent WGN , or more generally any complex channel (see later sections), is essentially the same as that for real signals earlier in this chapter. All previously presented detector analysis and structure holds with the following complex-arithmetic generalizations:

1. The inner product becomes

$$< \boldsymbol{x}, \boldsymbol{y} >= \boldsymbol{x}^* \boldsymbol{y} = \int_{-\infty}^{\infty} x^*(t) \cdot y(t) dt \quad , \tag{1.411}$$

   ($\boldsymbol{x}^*$ means conjugate transpose of $\boldsymbol{x}$).

2. The matched filter is conjugated, that is $\varphi(T-t) \rightarrow \varphi^*(T-t)$.

3. Energies of complex scalars are $\mathcal{E}_{\boldsymbol{x}} = E\left\{|x(t)|^2\right\}$, or the expected squared magnitude of the complex scalar, and $\bar{\mathcal{E}}_{\boldsymbol{x}} = \mathcal{E}_{\boldsymbol{x}}/2$.

For the MIMO case, a superscript of * means conjugate transpose of the matrix or vector. Further, Equation (1.411)'s integral becomes a sum of L_x integrals, the matched filter becomes L_x parallel matched filters. Energy per dimension will be divided by the total number of real dimensions, as always.

1.3.6 Passband Analysis for QAM alternatives

This subsection's results apply to each and every spatial dimension of a MIMO system. Passband analysis directly applies to QAM modulation in a way that simply requires computing a channel's baseband equivalent for convolution with the complex input $(x_1 + \jmath x_2) \cdot \varphi(t)$. Some transmission systems instead may use one of Section 1.3.6's three other implementations, VSB, CAP, or OQAM. This section addresses the specifics of how the passband analysis concepts discussed so far still apply to these other passband modulation types. In all cases, a complex-equivalent channel can be found easily from the given channel transfer function.

1.3.6.1 Passband VSB Analysis

This subsection starts with SSB (single-side-band) and then generalizes to VSB. With SSB, the transmitted signal has $x_I(t)$ and $x_Q(t)$ that are each other's Hilbert transforms and thus

$$x(t) = x_I(t) \cdot \cos(\omega_c t) - \check{x}_I(t) \cdot \sin(\omega_c t) \quad . \tag{1.412}$$

Such a signal only exhibits nonzero energy content for frequencies exceeding the carrier frequency (and for frequencies below the negative carrier frequency). The SSB baseband-equivalent signal is also therefore analytic, for with a new notation

$$x_{Ab}(t) = x_{bb}(t) = x_I(t) + \jmath \check{x}_I(t) \quad . \tag{1.413}$$

The subscript of Ab represents a new signal that is both analytic and baseband for SSB analysis. The channel output's baseband equivalent is consistently

$$y_{Ab}(t) = x_{Ab}(t) * \left(\frac{h_{Ab}(t)}{2}\right) \tag{1.414}$$

$$Y_{Ab}(\omega) = X_{Ab}(\omega) \cdot H(\omega + \omega_c) \;\; \omega > 0 \quad . \tag{1.415}$$

The previous passband-channel analysis applies for any carrier frequency and not just one centered within the passband. Thus, baseband-equivalent analysis directly applies to SSB also and the "Ab" notation has just made explicit the carrier-frequency position on the band's lower edge. However,

the input construction is such that $x_Q(t)$ is no longer independent of $x_I(t)$. Generally speaking, with this SSB constraint, twice as many dimensions per second are transmitted within $x_I(t)$ for SSB than would be the case for QAM with the same bandwidth. However, QAM can independently use the quadrature dimension whereas for SSB this quadrature dimension is completely determined from the inphase dimension (or vice versa). The analysis for lower sideband (instead of the assumed upper sideband) follows by simply negating the quadrature component and choosing the carrier frequency at the passband's upper edge, and then the baseband equivalent is nonzero only for negative frequencies, but again follows Sections 1.3.5.1 and 1.3.5.3's general analysis.

VSB transmission generalizes SSB transmission. VSB systems create $x_I(t)$ and $x_Q(t)$ so that they are almost Hilbert Transforms of one another. A VSB system may be easier to implement in practice and always uses an equivalent SSB signal. With this text's nomenclature, a VSB signal's baseband equivalent, $x_{Vb}(t)$, has "vestigial" symmetry about $f = 0$: that is $X_{Vb}(f) + X_{Vb}(-f) = X_{Ab}(f) \; \forall \; f > 0$ where $X_{Ab}(f)$ is for the (analytic) SSB signal in (1.413) upon which the VSB signal is based. The VSB signal iuses a carrier frequency that is not at the passband edge. This carrier frequency is the point around which the passband signal exhibits vestigial symmetry. This frequency determines the channel's baseband equivalent representation,

$$Y_{Vb}(\omega) = X_{Vb}(\omega) \cdot H(\omega + \omega_c) \;\; \omega > -\omega_c \quad . \tag{1.416}$$

US Terrestrial digital television broadcast uses VSB transmission with carrier frequencies at the nominal "TV channel" positions of 52 MHz + i·(6 MHz), effectively $\bar{b} = 2$ (constellation is coded so it is called 64 VSB, where extra levels are redundant for coding, see Chapter 2) and a symbol rate of roughly 5 MHz, for a data rate of 20 Mbps. The signals thus have non-zero energy from about 1.5 MHz below the carrier and to 3.5 MHz above the carrier using vestigial transmit symmetry with respect to that carrier.

1.3.6.2 Passband CAP Analysis

Analysis of CAP (carrierless amplitude phase) modulation essentially replaces baseband equivalents with analytic signal and channel equivalents. A CAP signal is generated according the J.J. Werner's [4] observation that follows:

$$\begin{aligned} x_A(t) &= \sum_k x_k \cdot \varphi(t - kT) \cdot e^{\jmath\omega_c t} && (1.417)\\ &= \sum_k x_k \cdot \varphi(t - kT) \cdot e^{\jmath\omega_c t} \cdot e^{-\jmath\omega_c kT} \cdot e^{+\jmath\omega_c kT} && (1.418)\\ &= \sum_k (x_k \cdot e^{+\jmath\omega_c kT}) \cdot \varphi(t - kT) \cdot e^{\jmath\omega_c (t-kT)} && (1.419)\\ &= \sum_k \breve{x}_k \cdot \varphi_A(t - kT) && (1.420) \end{aligned}$$

where the new quantities are defined as

$$\begin{aligned} \varphi_A(t) &= \varphi(t) \cdot e^{\jmath\omega_c t} \text{ and} && (1.421)\\ \breve{x}_k &= x_k \cdot e^{+\jmath\omega_c kT} && (1.422) \end{aligned}$$

(the latter is a rotated version of the input).[45] Thus, a CAP system simply rotates encoder outputs to create a symbol-time-invariant analytic modulation. The receiver also knows the rotation sequence and therefore need only detect $\breve{x}_k$, and rhen x_k can easily be determined by reversing the known rotation,

$$x_k = \breve{x}_k \cdot e^{-\jmath\omega_c kT} \quad .$$

In practice, designs ignore the rotations, and the sequence $\breve{x}_k$ itself directly carries the information, noting that the rotations at each end simply undo each other and have no bearing on performance nor functionality. They are necessary only for equivalence to a QAM signal.

[45] This $\breve{x}_k$ is notation used specific to the CAP situation here, and is not intended to be equivalent to any other temporary uses of a tilde on a quantity elsewhere in this textbook.

The channel output then follows

$$y_A(t) = x_A(t) * \left(\frac{h_A(t)}{2}\right) \tag{1.423}$$

$$Y_A(\omega) = X_A(\omega) \cdot H(\omega) \;\; \omega > 0 \quad . \tag{1.424}$$

CAP's focus only upon analytic signals, the complex channel is the analytic-equivalent channel with zero Fourier transform for negative frequencies (and for which the notation H_{CAP} is specific to analysis of CAP transmission over a channel with response generally denoted by $h(t)$.):

$$H_{CAP}(\omega) = H(\omega) \cdot \frac{1}{2}\left(1 + sgn(\omega)\right) \quad . \tag{1.425}$$

On a channel with narrow transmission band relative to the carrier frequency, intermediate-frequency (IF) demodulation may move (but not zero) the effective transmission band's center frequency closer to DC. Then CAP applies directly to the IF-demodulated signal. The IF demodulation treats the transmission signals as if they were analog signals and is separate from data transmission.

After conversion to complex equivalent channels, both QAM and CAP receiver processing can be generally described by the processing of a complex channel output, and such a complex model is this section's objective.

1.3.6.3 OQAM or "Staggered QAM"

The OQAM basis functions appear earlier in this subsection (see paragraph 1.3.6.3). This entire subsection could replace $\cos(\omega_c t)$ with $\text{sinc}(t/T) \cdot \cos(\omega_c t)$ and most importantly $\sin(\omega_c t)$ with $\text{sinc}([t - T/2]/T) \cdot \sin(\omega_c t)$ everywhere – and all results would still hold. However, there is an easier way to reuse previous results: Instead, OQAM's essential difference from QAM is OQAM quadrature component's delay by one-half symbol period with respect to the inphase component. OQAM analysis procedes with an equivalent channel input that doubles the symbol rate for a corresponding time-varying encoder that alternates between a nonzero inphase component (with zero quadrature component) and a nonzero quadrature component (with zero inphase component. OQAM then inputs this new time-varying double-speed symbol sequence to a conventional (double symbol rate) QAM modulator and thereby generates the OQAM sequence. All previous complex-analysis results then apply directly to this new equivalent system running at twice the symbol rate. The energy per dimension $\bar{\mathcal{E}}_{\boldsymbol{x}}$ will reduce by a factor of 2 if power is maintained constant, which is consistent also with the alternate zeroing of quadrature and inphase components

The channel output for OQAM input to an impulse response $h(t) = \Re\{h_{bb}(t) \cdot e^{j\omega_c t}\}$ convolves the baseband equivalent of the continuous-time $x(t)$ formed from the double-symbol-rate "interleaved" symbol sequence with $[h_{bb}(t)]/2$. Again a complex channel fits the analysis – the objective for this section.

The inphase and quadrature dimensions are not strictly independent since they have alternating zero values. This dependence or correlation effectively halves the bandwidth so that an OQAM system running with symbol rate $1/T$ and the basis functions in Section 1.6, even though analyzed as a QAM system running with interdependent symbols at rate $2/T$, occupies the same bandwidth as QAM. In fact as the function $\varphi_i(t)$ is generalized (See Chapter 3) so that $\varphi(t) \neq \sqrt{1/T}\text{sinc}(t/T)$, then OQAM typically requires less bandwidth in terms of the inevitable "non-brick-wall" energy roll-off associated with practical filter design.

1.3.6.4 The difference between complex and baseband equivalent channels

In digital-transmission literature and field of application, most practicing transmission specialists always use a complex channel to describe any channel, baseband or passband. When all complex quantities are real, the baseband case is a special case of the more general complex channel. Typically, a practicing engineer finds the complex equivalent channel from the real passband channel's provided magnitude/phase characteristics (or their equivalents). This process essentially slides the passband down to DC and then inverse transforms the result to produce a complex-equivalent channel. While precise bookkeeping

of factors-of-2 helps initial understanding, they eventual cancel under this section's rules. The energy-normalizing factor of $1/\sqrt{2}$ absorbs into the channel input and noise processes to retain $\frac{\mathcal{N}_0}{2}$ per dimension or as its (doubled-sided) passband power-spectral density. Since theoretical performance only depends on the ratio of signal power to noise, as long as this ratio is correct, scale factors on both signals are irrelevant from an analysis perspective. These factors of 2 may be crucial in determining the absolute dynamic range of an actual receiver design's channel-interfacing circuits, but otherwise do not affect analysis as long as the SNR is determined correctly, as in this section.

Thus, the literature on transmission almost always uses complex signals to represent channels and no factors of 2 appear, nor need to. However, designer carefully and correctly accounts for them when providing the SNR. Thus, when this text provides a complex channel (and not told specifically that it is a baseband equivalent channel derived as shown in this Chapter), the correct assumption is that convolution should occur without any additional factors of two. This subsection attempted to address these factors directly to assist field engineers who may well encounter them in calibrating their designs. It has been this author's personal experience that many professional and decorated engineers often trip on a factor of 2 interpretation[46] This section provided all the details in the hope of having a solid reference upon which any designer may return to check results when things somehow seem off (by 3 dB!).

The complex filter Another important distinction to mention is the complex filter. Once a receiver or transmitter has established a complex (two-dimensional) signal and is using complex filters to process that signal, there is no factor of 2 involved. Any factor of two would only be necessary if both of the convolved quantities did correspond to passband signals, and a strict equivalent need be retained. However if all signals are complex, then there is no need for passband equivalences and convolution proceeds correctly for any internal complex filters without any factors of two involved. Chapter 3 uses such filters within a receiver and does return to passband discussion, so convolution proceeds directly without factors of 2 and correctly represents the receiver's internal filtering of complex signals.

1.3.7 Additive Self-Correlated Noise

This subsection's results apply to each and every spatial dimension of a MIMO system. In practice, additive noise is often Gaussian, but its power spectral density may not be flat. Engineers often call such noise "self-correlated" or "colored". Colored noise remains independent of the message signals but correlates with itself from time instant to time instant. Colored noise's origins are many: Receiver filtering effects, noise generated by other communications systems ("crosstalk"), and electromagnetic interference all may introduce self-correlated noise. A narrow-band radio-signal transmission that somehow becomes noise for a different (unintended) channel is another common example of self-correlated noise and called "RF" noise (RF is an acronym for "radio frequency").. As many noises add, the central limit theorem applies to render the colored noise distribution nearly Gaussian (whatever its constituent component distributions)[47].

Self-correlated Gaussian noise can significantly alter the detector's performance with respect to a detector designed for white Gaussian noise. This section investigates the optimum detector for colored noise and also considers the performance loss when using a (consequently suboptimal) detector designed for AWGN.

This study first investigates Subsection 1.3.7's filtered "one-shot" AWGN channel. Subsection 1.3.7.2 then finds the optimum detector for additive self-correlated Gaussian noise, by adding a whitening filter that transforms the self-correlated noise channel into a filtered AWGN channel. Subsection 1.3.7.2 studies the vector channel, for which (for some unspecified reason) the noise has not been whitened and describes the optimum detector given this vector channel. Finally, Subsection 1.3.7.3 studies the

[46] Perhaps one of the most famous factor-of-2 errors was made by America's then finest industrial telecommunications research center and largest best-quality television supplier when they concluded that VSB was 3 dB better than QAM (it's not) after enormous money had been invested into the eventually red-faced-acknowledged mistake. Their VSB system was selected anyway until it was later replaced by Chapter 4's Coded-OFDM methods. Some famous text books still have the factor of 2 mistake as well.

[47] In practice, no noise will ever quite be exactly Gaussian, but designs based on this assumption have proven to be very robust throughout almost all transmission experience, to date anyway.

degradation that occurs when the noise correlation properties are unknown in receiver design, and the receiver uses instead an optimum AWGN (ML) detector.

1.3.7.1 The Filtered (One-Shot) AWGN Channel

Figure 1.67 illustrates the filtered AWGN channel. The modulated signal $x(t)$ undergoes filtering by $h(t)$ before the addition of the white Gaussian noise. When $h(t) \neq \delta(t)$, the filtered signal set $\{\breve{\boldsymbol{x}}_{\boldsymbol{i}}(t)\}$ may differ from the transmitted signal set $\{\boldsymbol{x}_i(t)\}$. This channel filtering usually changes the error probability as well as the ML-detector structure. This subsection (and chapter) still consider only one channel use with M possible messages. Transmission over a filtered channel can incur a significant penalty from intersymbol interference between successively transmitted data symbols. In the "one-shot" case, however, analysis need not consider this intersymbol interference. Chapters 3, 4, and 5 consider intersymbol interference.

For any channel input signal $x_i(t)$, the corresponding filtered output equals $\breve{x}_i(t) = h(t) * x_i(t)$. Decomposing $x_i(t)$ by an orthogonal basis set, $\breve{x}_i(t)$ becomes

$$\begin{aligned} \breve{x}_i(t) &= h(t) * x_i(t) && (1.426)\\ &= h(t) * \sum_{n=1}^{N} x_{in} \cdot \varphi_n(t) && (1.427)\\ &= \sum_{n=1}^{N} x_{in} \cdot \{h(t) * \varphi_n(t)\} && (1.428)\\ &= \sum_{n=1}^{N} x_{in} \cdot \phi_n(t) \quad , && (1.429) \end{aligned}$$

where

$$\phi_n(t) \triangleq h(t) * \varphi_n(t) \quad . \tag{1.430}$$

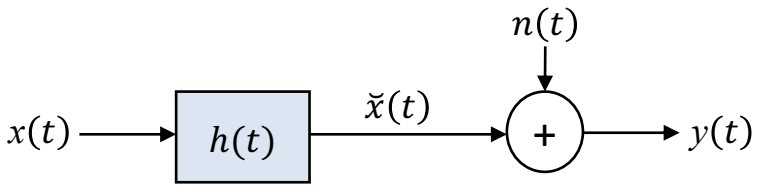


Figure 1.67: **Filtered AWGN channel.**

Observations include:

- The set of N functions $\{\phi_n(t)\}_{n=1,\ldots,N}$ is not necessarily orthonormal.
- For the channel to convey any and all constellations of M messages for the signal set $\{x_i(t)\}$, the new basis set $\{\phi_n(t)\}$ must be linearly independent.

The first observation can be easily proven by finding a counterexample. The second observation emphasizes that if filtering zeros some dimensions, signals in the original signal set that differ only along the lost dimension(s) would appear identical at the channel output. If $\phi_n(t)$ is such a dimension, then the two signals $\breve{x}_k(t)$ and $\breve{x}_j(t)$ that differ only on that dimension are not distinguishable:

$$\breve{x}_k(t) - \breve{x}_j(t) = \sum_{n=1}^{N} (x_{kn} - x_{jn}) \cdot \phi_n(t) = 0 \quad , \tag{1.431}$$

However, If the set $\{\phi_n(t)\}$ is linearly independent then the sum in (1.431) must be nonzero: a contradiction to (1.431). Conversely, if this function set is linearly dependent, then (1.431) can be satisfied, resulting in the possibility of ambiguous transmitted signals. Failure to meet the linear independence condition could mandate a redesign of the modulated signal set or a rate reduction (decrease of M). Chapters 4 and 5 investigate such dimensionality loss and provide redesigns of $\{x_i(t)\}_{i=0:M-1}$ to avoid this "singular channel." This chapter assumes such dimensionality loss does not occur.

If the set $\{\phi_n(t)\}$ is linearly independent, then the Gram-Schmidt procedure in Appendix A generates an orthonormal set of N basis functions $\{\psi_n(t)\}_{n=1,\ldots,N}$ from $\{\phi_n(t)\}_{n=1,\ldots,N}$. A new signal constellation $\{\breve{\boldsymbol{x}}_i\}_{i=0:M-1}$ follows from the filtered signal set $\{\breve{\boldsymbol{x}}_i(t)\}$ using the basis set $\{\psi_n(t)\}$.

$$\breve{\boldsymbol{x}}_{in} = \int_{-\infty}^{\infty} \breve{x}_i(t) \cdot \psi_n(t) dt = \langle \breve{x}_i(t), \psi_n(t) \rangle \ . \tag{1.432}$$

Using the previous AWGN analysis, a tight upper bound on message error probability remains

$$P_e \leq N_e Q \left[\frac{d_{\min}}{2\sigma} \right] \quad , \tag{1.433}$$

where $d_{\min}$ is the minimum Euclidean distance between any two points in the filtered signal constellation $\{\breve{\boldsymbol{x}}_i\}_{i=0:M-1}$. Figure 1.68's signal detector does require $\{\psi_n(t)\}_{n=1,\ldots,N}$'s determination, but it is tacit in analysis. (For reference, Figure 1.31 shows the detector for the unfiltered constellation).

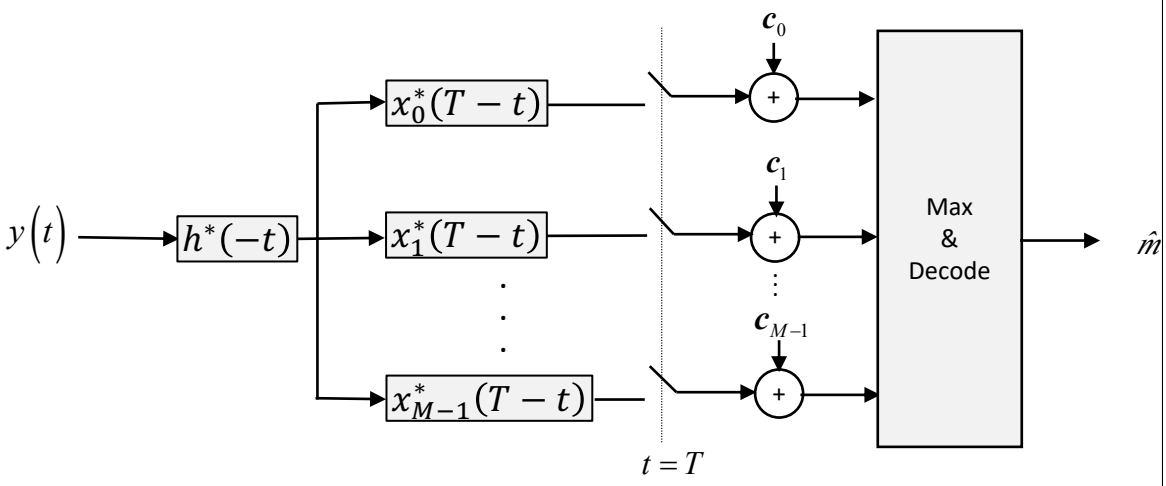


Figure 1.68: **Modified signal detector (each filter can become L_x filters for MIMO).**

Filtered AWGN analysis still measures the transmitted average energy $\mathcal{E}_{\boldsymbol{x}}$ at the channel input. Thus, $\mathcal{E}_{\breve{\boldsymbol{x}}}$ physical's significance is different from that of $\mathcal{E}_{\boldsymbol{x}}$. Nonetheless, for the original signal set satisfying the energy constraint, the altered constellation allows performance analysis.

1.3.7.2 Optimum Detection with Self-Correlated Noise

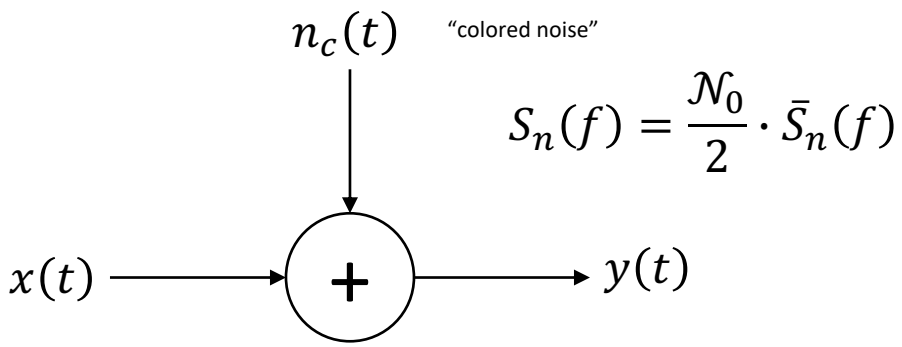


Figure 1.69: **ACGN channel.**

Figure 1.69 illustrates the Additive Self-Correlated Gaussian Noise (ACGN) channel. The only change with respect to Figure 1.26 is that the autocorrelation function of the additive noise $r_n(\tau)$, need not equal $\frac{\mathcal{N}_0}{2} \cdot \delta(\tau)$. Simplification of the ensuing development defines and uses a normalized noise autocorrelation function

$$\bar{r}_n(\tau) \triangleq \frac{r_n(\tau)}{\frac{\mathcal{N}_0}{2}} \quad . \tag{1.434}$$

The unnormalized noise's power spectral density is then

$$\mathcal{S}_n(f) = \frac{\mathcal{N}_0}{2} \cdot \bar{\mathcal{S}}_n(f) \quad , \tag{1.435}$$

where $\bar{\mathcal{S}}_n(f)$ is the Fourier Transform of $\bar{r}_n(\tau)$.

The Whitening filter ACGN channel design and analysis "whitens" the colored noise with filter $g(t)$, and then uses the previous section's filtered-AWGN results where the filter $h(t) = g(t)$. To ensure that $g(t)$ loses no information, $g(t)$ should be invertible. By the reversibility theorem, the receiver then can use an optimal detector for this newly generated filtered AWGN without performance loss. Actually, the condition on invertibility of $g(t)$ is sufficient but not necessary. For a particular signal set, a necessary condition is that the filter be invertible over that signal set's (non-zeroed) dimensions. For the filter to be invertible on any possible signal set, $g(t)$ must necessarily be invertible. This subtle point is often overlooked by most works on this subject.

For $g(t)$ to whiten the noise,

$$\left[\bar{\mathcal{S}}_n(f)\right]^{-1} = |G(f)|^2 \quad . \tag{1.436}$$

In general many filters $G(f)$, may satisfy Equation (1.436) but only some of the filters possess realizable inverses (the particular choice is the so-called minimum-phase choice that has all poles and zeros in the left-half plane, or on the $s = \jmath\omega$ axis with multiplicity 1 in that "marginally realizable" case - recognizing of course that is white noise so no whitening filter is needed, or the filter is trivially the Dirac delta function $\delta(t)$). Appendix A covers this area (for now, see Appendix A of Chapter 3).

To ensure a realizable inverse's existence, $\mathcal{S}_n(f)$ must satisfy the **Paley-Wiener Criterion**.

Theorem 1.3.6 [Paley-Wiener Criterion] *If*

$$\int_{-\infty}^{\infty} \frac{|\ln \mathcal{S}_n(f)|}{1+f^2} df < \infty \quad , \tag{1.437}$$

then there exists a $G(f)$ satisfying (1.436) with a realizable inverse. (Thus the filter $g(t)$ is a 1-to-1 mapping).

If the noise spectrum violates the Paley-Wiener criterion, then it is possible to design transmission systems with infinite data rate (that is when $\mathcal{S}_n(f) = 0$ over a given bandwidth, put the signals in that band). A full development of Paley Wiener is deferred until Appendix A of Chapter 3. This subsection's analysis always assumes Equation (1.437) is satisfied.[48] With a 1-to-1 $g(t)$ that satisfies (1.436), the ACGN channel converts into an equivalent filtered white Gaussian noise channel as Figure 1.67 shows by replacing $h(t)$ with $g(t)$. The performance analysis of ACGN is identical to that derived for the filtered AWGN channel in Subsection 1.3.7. A further refinement handles the filtered ACGN channel by whitening the noise and then analyzing the filtered AWGN with $h(t)$ replaced by $h(t) * g(t)$.

Appendix 3A develops "analytic continuation" for $\bar{\mathcal{S}}_n(s)$ to determine an invertible $g(t)$:

$$\bar{\mathcal{S}}_n(s) = \bar{\mathcal{S}}_n\left(f = \frac{s}{2\pi\jmath}\right) \quad , \tag{1.438}$$

where $\bar{\mathcal{S}}_n(s)$ can be canonically (and uniquely) factored into causal (and causally invertible) and anti-causal (and anticausally invertible) parts as

$$\bar{\mathcal{S}}_n(s) = \bar{\mathcal{S}}_n^+(s) \cdot \bar{\mathcal{S}}_n^-(s) \quad , \tag{1.439}$$

where

$$\bar{\mathcal{S}}_n^+(s) = \bar{\mathcal{S}}_n^-(-s) \quad . \tag{1.440}$$

If $\bar{\mathcal{S}}_n(s)$ is rational, then $\bar{\mathcal{S}}_n^+(s)$ is "minimum phase," i.e. all poles and zeros of $\bar{\mathcal{S}}_n^+(s)$ are in the left half plane. The filter $g(t)$ is then

$$g(t) = \mathcal{L}^{-1}\left\{\frac{1}{\bar{\mathcal{S}}_n^+(s)}\right\} \quad , \tag{1.441}$$

where $\mathcal{L}^{-1}$ is the inverse Laplace Transform. Figure 1.67's filtered AWGN can now replace filter $h(t) \to h(t) * g(t)$ and all results there hold. The matched filter $g^*(-t)$ is given by $g(-t) = \frac{1}{2\pi\jmath} \oint G(s) e^{-st} ds$, or equivalently by

$$g(-t) = \mathcal{L}^{-1}\left\{\frac{1}{\bar{\mathcal{S}}_n^-(s)}\right\} \quad . \tag{1.442}$$

$g(-t)$ is anticausal and cannot be realized. Thus any matched-filter detector like in Figure 1.68 now has $h^*(-t) \to h^*(-t) * g^*(-t)$. Practical receivers instead realize $g(T - t)$, where T is sufficiently large to ensure causality.

In general $g(t)$ may be difficult to implement by this method; however, the next subsection considers a discrete equivalent of whitening that is more straightforward to implement digitally in practice. When the noise is complex, Equation (1.440) generalizes to

$$\bar{\mathcal{S}}_n^+(s) = \left[\bar{\mathcal{S}}_n^-(-s^*)\right]^* \quad . \tag{1.443}$$

MIMO-channel noise whitening typically uses an inverted square-root matrix to remove any spatial correlation between different dimension's noises, then augments the spatial whitening by any necessary further scalar whitening in time-frequency. Practically, the same noise impinges upon the different antennas or wires and this is spatially removed by the square root with only the "spatially white" noise remaining. More generally, this whitening can be handled in the practical discrete-time cases as shown in Chapter 3, particularly Appendix A there, and also in Chapters 4 and 5.

Comment on filter $h(t)$ definition; In communication design, the combination of a modulation basis function and a channel filter often has the name **pulse response**, $p(t) \triangleq h(t) * \varphi(t)$ to distinguish it from the "impulse response" $h(t)$, particularly when $N = 1$ as in Chapter 3 to come more formally. When the modulation filter $\varphi(t)$ is fixed by design and not significantly dependent on the channel specifics[49], the pulse response characterizes the channel. Any anti-alias filtering for systems using digital signal

[48] Chapters 4 and 5 also expand to the correct form of transmission that should be used when (1.437) is not satisfied.

[49] All the basis functions n this chapter are independent of channel other than possibly the choice of a carrier frequency, for example.

processing and conversion devices is absorbed into the $h(t)$ definition always, as in Figure 1.70's anti-alias filters $lpf(t)$ that surround the continuous time channel $h_c(t)$; it may be in some simple modulators that indeed $\varphi(t) = lpf(t)$ in practice (for instance with $p = q$ and $\varphi_k = \delta_k$), but not always and usually not in sophisticated designs. This text's later chapters often adapt the basis function $\varphi(t)$, under various names to come, to the channel $h(t)$, in which case the pulse response has little or no use. These statements all apply equally well to an $L_y \times L_x$ MIMO matrix channel like $H(t)$ as later chapters also address.

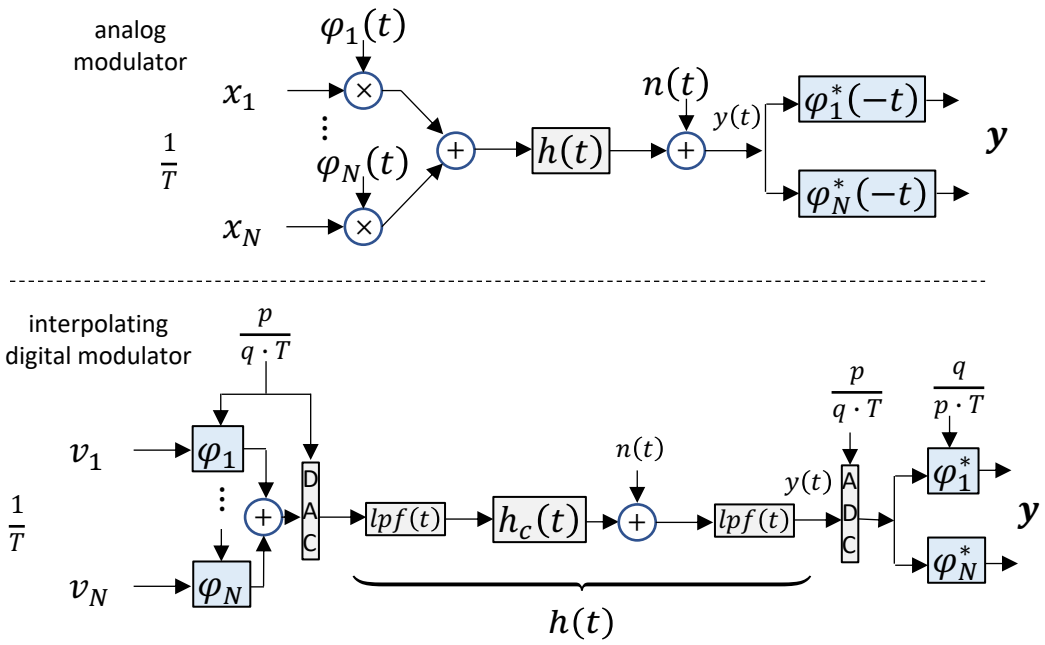


Figure 1.70: **Interpolated All-Digital Modulator**

Figure 1.70 shows the typical digital modulator defintion where the basis functions are actually implemented, typically at some rational fraction $\frac{p}{q \cdot T}$, wherein $p \in \mathbb{Z}^+ \geq q \in \mathbb{Z}^+$ (See Chapter 6 on system timing and clock generation). This makes the adaptive implementations of $]\varphi$ more realistic as an interpolated filter that attempts at the higher sampling rate to capture some good basis function well designed for the channel of interest. The channel output may after some decimating digitial signal processing return to the original symbol rate as in Figure 1.70. Both systems have the same symbol rate $1/T$, but are implemented differently. When the digital sophisticated version is in use, this text will use H for the channel model, and separately identify transmit modulation filters. When the basis is fixed and simply, the pulse response $p(t)$ is instead used. The two notations help identify situation. Most notably, Chapter 3's receiver equalized systems allow no optimization of transmitter, and simply accept the pulse response and do all possible at the receiver to improve performance metrics.

The Vector Self-Correlated Gaussian Noise Channel Following the digital theme, this subsection considers a discrete ACGN equivalent

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n} \quad , \tag{1.444}$$

where the noise vector $\boldsymbol{n}$'s autocorrelation matrix is

$$\mathbb{E}\left[\boldsymbol{n} \cdot \boldsymbol{n}^*\right] = \boldsymbol{R_n} = \overline{\boldsymbol{R}}_n \cdot \sigma^2 \quad . \tag{1.445}$$

Both $\boldsymbol{R_n}$ and $\bar{\boldsymbol{R}}_{\boldsymbol{n}}$ are positive-definite matrices. This discrete ACGN channel replaces the continuous ACGN channel. MIMO applies here directly with the self-correlated noise now introducing explicitly

correlated between dimensions. All analysis proceeds identically, whether the original channel was MIMO or simply a set of successive time samples from a channel output's noise. The discrete noise vector can be "whitened", transforming $\overline{\boldsymbol{R}_{\boldsymbol{n}}}$ into an identity matrix. The discrete equivalent to whitening $y(t)$ by $g(t)$ is a matrix multiplication. The $N \times N$ whitening matrix in the discrete-time case corresponds to the whitening filter $g(t)$ in the continuous case.

Cholesky factorization determines the invertible whitening transformation according (see Appendix A of Chapter 3):

$$\overline{\boldsymbol{R}}_n = \overline{\boldsymbol{R}}^{1/2} \cdot \overline{\boldsymbol{R}}^{*/2} \quad , \tag{1.446}$$

where $\overline{\boldsymbol{R}}^{1/2}$ is lower triangular and $\overline{\boldsymbol{R}}^{*/2}$ is upper triangular. These matrices constitute the matrix equivalent of a "square root", and both matrices are invertible. Noting the definitions,

$$\overline{\boldsymbol{R}}^{-/2} \triangleq \left[\overline{\boldsymbol{R}}^{1/2}\right]^{-1} \quad , \tag{1.447}$$

and

$$\overline{\boldsymbol{R}}^{-*/2} \triangleq \left[\overline{\boldsymbol{R}}^{*/2}\right]^{-1} \quad , \tag{1.448}$$

then to whiten $\boldsymbol{n}$, the receiver passes $\boldsymbol{y}$ through the matrix multiply $\overline{\boldsymbol{R}}^{-/2}$,

$$\breve{\boldsymbol{y}} \triangleq \overline{\boldsymbol{R}}^{-/2} \cdot \boldsymbol{y} = \overline{\boldsymbol{R}}^{-/2} \cdot \boldsymbol{x} + \overline{\boldsymbol{R}}^{-/2} \cdot \boldsymbol{n} = \breve{\boldsymbol{x}} + \breve{\boldsymbol{n}} \quad . \tag{1.449}$$

The autocorrelation matrix for $\breve{\boldsymbol{n}}$ is

$$\mathbb{E}\left[\breve{\boldsymbol{n}} \cdot \breve{\boldsymbol{n}}^*\right] = \overline{\boldsymbol{R}}^{-/2} \mathbb{E}\left[\boldsymbol{n}\boldsymbol{n}^*\right] \overline{\boldsymbol{R}}^{-*/2} = \overline{\boldsymbol{R}}^{-/2} \left(\overline{\boldsymbol{R}}^{1/2} \overline{\boldsymbol{R}}^{*/2} \cdot \sigma^2\right) \overline{\boldsymbol{R}}^{-*/2} = \sigma^2 \cdot I \; . \tag{1.450}$$

Thus, $\breve{\boldsymbol{n}}$'s covariance matrix is the same as the AWGN vector's covariance matrix. By the reversibility theorem (Theorem 1.1.4), no information is lost in such a transformation.

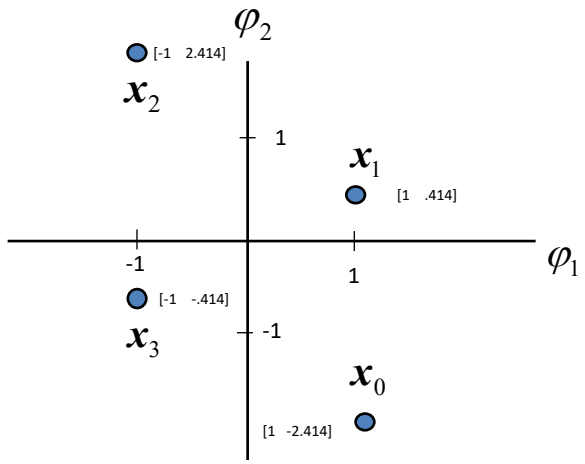


Figure 1.71: **Equivalent signal constellation for Figure 1.29.**

EXAMPLE 1.3.14 [QPSK with correlated noise] Figure 1.29's example now experiences colored noise with correlation matrix

$$\boldsymbol{R}_{\boldsymbol{n}} = \sigma^2 \begin{bmatrix} 1 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 1 \end{bmatrix} \tag{1.451}$$

Then

$$\overline{\boldsymbol{R}}^{1/2} = \begin{bmatrix} 1 & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \tag{1.452}$$

and

$$\overline{\boldsymbol{R}}^{*/2} = \begin{bmatrix} 1 & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \quad . \tag{1.453}$$

From (1.452),

$$\overline{\boldsymbol{R}}^{-/2} = \begin{bmatrix} 1 & 0 \\ -1 & \sqrt{2} \end{bmatrix} \tag{1.454}$$

and

$$\overline{\boldsymbol{R}}^{-*/2} = \begin{bmatrix} 1 & -1 \\ 0 & \sqrt{2} \end{bmatrix} \quad . \tag{1.455}$$

The signal constellation after the whitening filter becomes

$$\breve{\boldsymbol{x}}_0 = \overline{\boldsymbol{R}}^{-/2}\boldsymbol{x}_0 = \begin{bmatrix} 1 & 0 \\ -1 & \sqrt{2} \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ (-\sqrt{2}-1) \end{bmatrix} \quad , \tag{1.456}$$

and similarly

$$\breve{\boldsymbol{x}}_2 = \begin{bmatrix} -1 \\ \sqrt{2}+1 \end{bmatrix} , \tag{1.457}$$

$$\breve{\boldsymbol{x}}_1 = \begin{bmatrix} 1 \\ \sqrt{2}-1 \end{bmatrix} , \text{ and} \tag{1.458}$$

$$\breve{\boldsymbol{x}}_3 = \begin{bmatrix} -1 \\ -\sqrt{2}+1 \end{bmatrix} . \tag{1.459}$$

This new constellation forms Figure 1.71's parallelogram in two dimensions, where the minimum distance is now along the shorter diagonal (between $\breve{\boldsymbol{x}}_1$ and $\breve{\boldsymbol{x}}_3$), rather than along the sides, but this $d_{\min} = \sqrt{2^2 + (1-\sqrt{2})^2} = 2.164 > 2$. Thus, the optimum detector for this channel with self-correlated Gaussian noise has larger minimum distance than for the white noise case, illustrating the important fact that correlated noise is sometimes advantageous.

Example 1.3.14 shows that correlated noise may improve performance with respect to the same channel and signal constellation with white noise of the same average energy. Nevertheless, the noise autocorrelation matrix is often not known in implementation, or it may vary from channel use to channel use. Then, the detector is designed as if white noise were present anyway, and there is a performance loss with respect to the optimum detector. The next subsection calculates this performance loss.

1.3.7.3 Performance of Suboptimal Detection with Self-Correlated Noise

The AWGN channel's ML detector is suboptimum for the ACGN channel, but often in use anyway because the noise correlation properties may be hard to know in the design stage. In this case, the detector performance reduces with respect to optimum.

Performance reduction computation uses the **error-event** vectors

$$\boldsymbol{\epsilon}_{ij} \triangleq \frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|} \quad . \tag{1.460}$$

The noise vector's component along an error-event vector is $\langle \boldsymbol{n}, \boldsymbol{\epsilon}_{ij} \rangle$. The noise variance along this vector is $\sigma_{ij}^2 \triangleq E\left\{\langle \boldsymbol{n}, \boldsymbol{\epsilon}_{ij} \rangle^2\right\}$. Then, the NNUB becomes

$$P_e \leq N_e \cdot Q\left[\min_{i \neq j}\left\{\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{2\sigma_{ij}}\right\}\right] \quad . \tag{1.461}$$

With whitened noise, then (1.461) simplifies to

$$P_e \leq N_e \cdot Q\left[\min_{i\neq j}\left\{\frac{\|\breve{\boldsymbol{x}}_i - \breve{\boldsymbol{x}}_j\|}{2\sigma}\right\}\right] = P_e \leq N_e \cdot Q\left[\min_{i\neq j}\left\{\frac{\breve{d}_{min}}{2\sigma}\right\}\right] \tag{1.462}$$

because all dimensions have the same whitened noise component, but of course the rotated signals determine a new set of intra-symbol distances, with minimum $\breve{d}_{min} \triangleq \min_{i\neq j} \|\breve{\boldsymbol{x}}_i - \breve{\boldsymbol{x}}_j\|$

EXAMPLE 1.3.15 [Worst-Case Colored Noise]

This example continues Example 1.3.14. A second possible QPSK constellation input rotates the original set by 45 degrees, and is the set $\{\boldsymbol{z}_{i=0,1,2,3}\}$. The table below summarizes the constellation symbols for both the original case and the new case. The new minimum distance is the same for the entire constellation, so each table row repeats it.

i	$\boldsymbol{x}_i$	$\breve{\boldsymbol{x}}_i$	$\breve{d}_{min}$	$\boldsymbol{z}_i = e^{\jmath\pi/4}\cdot\boldsymbol{x}_i$	$\breve{\boldsymbol{z}}_i$	$\breve{d}_{min}$
0	$[1,1]^*$	$[1,-1+\sqrt{2}]^*$	2.165	$[0,\sqrt{2}]^*$	$[0,2]^*$	1.531
1	$[-1,1]^*$	$[-1,1+\sqrt{2}]^*$	2.165	$[-\sqrt{2},0]^*$	$[-\sqrt{2},\sqrt{2}]^*$	1.531
2	$[-1,-1]^*$	$[-1,1-\sqrt{2}]^*$	2.165	$[0,-\sqrt{2}]^*$	$[0,-2]^*$	1.531
3	$[-1,1]^*$	$[-1,-1-\sqrt{2}]^*$	2.165	$[\sqrt{2},0]^*$	$[\sqrt{2},-\sqrt{2}]^*$	1.531

While the original constellation had a larger $\breve{d}_{min}(\breve{\boldsymbol{x}})$, the rotated constellation instead has $\breve{d}_{min}(\breve{\boldsymbol{z}})/\breve{d}_{min}(\breve{\boldsymbol{x}}) = 2.165/1.531 = 3$ dB. If the receiver uses the AWGN ML detector, which again is suboptimum for this colored noise, the noise components along the two dimensions (horizontal and vertical) are the diagonal elements of $\overline{\boldsymbol{R}}_n$. Thus, the AWGN-ML detector's performance is $d_{\min}/2\sigma = 1/\sigma$ so the same P_e as if the noise were white - but again not the best. The original constellation had minimum distance 2, sp 2.165 is 0.69 dB better than if the channel were AWGN. The rotated constellation has instead 1.531/2, which is -2.31dB worse. This suggests the AWGN ML detector's use with unknown noise autocorrelation might be acceptable.

As Chapter 4 will show, these two rotations are indeed the best and worst cases for this noise autocorrelation matrix. Without precise noise knowledge, it is clear that using the AWGN ML detector is not too bad, unless the precise noise correlation is indeed known. Making a mistake by 45 degrees (which could be simple carrier-frequency constant phase offset) would result in a larger loss than simply using the AWGN-based detector.

1.3.8 The Matrix AWGN

Figure 1.72 illustrates the **matrix AWGN** with L_x-dimensional input symbol $\boldsymbol{x}$ and L_y-dimensional output after sampling and demodulation through some set of L_y anti-alias pre-sampling filters. The resultant channel has crosstalk between the L_x transmit-symbol dimensions to the L_y outputs. This is similar to the filtered AWGN channel, and indeed Chapters 2, 4, and 5 model the filtered AWGN's time samples with symbol blocks in such a fashion. When $H = I$, this is the MIMO channel created as in Figure 1.66. This matrix AWGN processes each message transmission without overlap with any preceding or succeeding symbol transmissions. This channel when $H \neq I$ will be further addressed in Section 1.5 and Chapters 2 - 5. The transmit energy constraint sums over all dimensions, or equivalently (see also Appendix D).

$$\mathcal{E}_{\boldsymbol{x}} = trace\{R_{\boldsymbol{x}\boldsymbol{x}}\} = \mathbb{E}\left[\|\boldsymbol{x}\|^2\right] \quad , \tag{1.463}$$

where $R_{\boldsymbol{x}\boldsymbol{x}} = \mathbb{E}\left[\boldsymbol{x}\cdot\boldsymbol{x}^*\right]$ further appears in Appendix D.

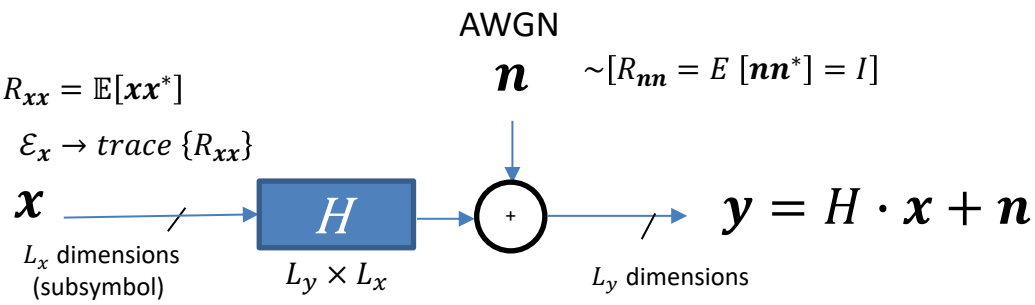


Figure 1.72: **The Matriix AWGN channel.**

1.4 Discrete Channels

Discrete channels have outputs restricted to a discrete set, $\boldsymbol{y} \in \{\boldsymbol{y}_j\}_{j=0,...,M'<\infty}$. Such discrete sets often arise from receiver restrictions. For instance, any channel that samples a continuous-time analog channel output with an analog-digital converter effectively creates a discrete channel, where $M' = 2^{b_{ADC}}$ for $b_{ADC} \in Z$ and $b_{ADC} \geq 1$ converter bits. More generally, the outputs can be any set of objects as can be the input, but both will map into non-negative integer indices, $j \in Z^+$. As always, the most general channel description is $p_{\boldsymbol{y}/\boldsymbol{x}}$, and Section 1.2's results apply.

When $M' = |C| = M$, some index orders may be convenient: for instance, the ML detector output that corresponds to message $\hat{m} = m$ might most simply also have label m, so typically $y_j \to j \in \{0, 1, ..., M'-1\}$ and $x_i \to i \in \{0, 1, ..., M-1\}$. For reasonable channels $M' \geq |C| = M$ so there is at least one output that a receiver can uniquely map to each possible input.

Subsection 1.4.1 begins with the discrete memoryless channel that retains independence between successive uses. Subsection 1.4.2 then refines the model to the very commonly encountered symmetric discrete memoryless channels.

1.4.1 Discrete Memoryless Channels

The discrete-memoryless channel ensures successive channel transmissions are independent, formally:

Definition 1.4.1 *[Discrete Memoryless Channel (DMC)] A* **discrete memoryless channel (DMC)** *has $M' \geq M = |C| < \infty$ with ordered transmitted message group $\boldsymbol{X} \triangleq \{\boldsymbol{x}_n,\ n = 1, ..., N\}$, with each message $\boldsymbol{x}_n \in \{i = 0, ..., M-1\}$, and with corresponding outputs $\boldsymbol{Y} \triangleq \{\boldsymbol{y}_n,\ n = 1, ..., N\}$ with each $\boldsymbol{y}_n \in \{j = 0, ..., M'-1\}$ that satisfy*

$$p_{\boldsymbol{Y}/\boldsymbol{X}}(j,i) = \prod_{n=1}^{N} p_{\boldsymbol{y}_n/\boldsymbol{x}_n}(j,i) \quad . \tag{1.464}$$

The integer n here is a dimensional index (typically reflecting successive time-based DMC uses, but not necessarily so). The indices j and i reflect instead particular (output, input) sample values from the discrete distribution. A **stationary DMC** *has $p_{\boldsymbol{y}_n/\boldsymbol{x}_n}(j,i) = p_{\boldsymbol{y}/\boldsymbol{x}}(j,i)\ \forall\ j, i$, or is thus independent of the dimensional index n.*

The DMC models situations where successive discrete outputs have no channel-related dependencies, that is, there is no channel-induced "intersymbol interference." If the inputs are also independent so that $p_{\boldsymbol{x}}(i) = \prod_{n=1}^{N} p_{\boldsymbol{x}_n}(i)$, then the DMC's ML and MAP detectors are a dimensionally indexed series of independent decisions, $\hat{\boldsymbol{x}}_n = \hat{i} \in \{0, 1, ..., M-1\},\ n = 1, ..., N$. This section presumes hereforth stationary DMCs.

An $M' \times M$ **probability transition matrix** often represents the DMC:

$$P_{\boldsymbol{y}/\boldsymbol{x}} \triangleq \left[p_{\boldsymbol{y}/\boldsymbol{x}}(j,i)\right]_{j=0,...,M-1;i=0,...,M'-1} = \begin{bmatrix} p_{M'-1/M-1} & \cdots & p_{M'/0} \\ \vdots & \ddots & \vdots \\ p_{0/M-1} & \cdots & p_{0/0} \end{bmatrix} \tag{1.465}$$

$$= \begin{bmatrix} \underbrace{\quad} & \cdots & \underbrace{\quad} \\ \triangleq & & \triangleq \end{bmatrix}$$

$$= \begin{bmatrix} \boldsymbol{p}_{\boldsymbol{y}/\boldsymbol{x}}(M-1) & ... & \boldsymbol{p}_{\boldsymbol{y}/\boldsymbol{x}}(0) \end{bmatrix} \quad . \tag{1.466}$$

The matrix $P_{\boldsymbol{y}/\boldsymbol{x}}$'s element names omit the subscripts $\boldsymbol{y}$ and $\boldsymbol{x}$ to simplify notation, but that subscript notation appears in the column vectors $\boldsymbol{p}_{\boldsymbol{y}/\boldsymbol{x}}(i)$ for the matrix, as well as the full matrix $P_{\boldsymbol{y}/\boldsymbol{x}}$ itself.

Various relationships[50] follow directly from the matrix $P_{\boldsymbol{y}/\boldsymbol{x}}$:

1. **unit column sum** - Each column sums to unity:

$$1 = \sum_{j=0}^{M'} p_{j/i} = [\mathbf{1}]^* \, \boldsymbol{p}_{\boldsymbol{y}/\boldsymbol{x}}(i) \; \forall \; i = 0, ..., M-1 \; . \tag{1.467}$$

2. **weighted row-sum is $p_{\boldsymbol{y}}(j)$** - Each row sums to the corresponding $\boldsymbol{y}$-value's probability:

$$p_{\boldsymbol{y}}(j) = \sum_{i=0}^{M} p_{j/i} \cdot p_{\boldsymbol{x}}(i) \; \forall \; j = 0, ..., M'-1 \; . \tag{1.468}$$

   Equivalently, If $\boldsymbol{p}_{\boldsymbol{y}}$ and $\boldsymbol{p}_{\boldsymbol{x}}$ are row vectors that stack $\boldsymbol{y}$ probability values $\boldsymbol{p}_{\boldsymbol{y}} \triangleq \left[p_{\boldsymbol{y}}(M'-1)...p_{\boldsymbol{y}}(0)\right]^*$ and $\boldsymbol{p}_{\boldsymbol{x}} = [p_{\boldsymbol{x}}(M-1)...p_{\boldsymbol{x}}(0)]^*$ respectively, then there is an input/output matrix-multiply relation

$$\boldsymbol{p}_{\boldsymbol{y}} = P_{\boldsymbol{y}/\boldsymbol{x}} \cdot \boldsymbol{p}_{\boldsymbol{x}} \; . \tag{1.469}$$

3. **Joint Probability Distribution** - The joint distribution is

$$P_{\boldsymbol{y},\boldsymbol{x}} = P_{\boldsymbol{y}/\boldsymbol{x}} \cdot \text{Diag}\left\{\boldsymbol{p}_{\boldsymbol{x}}\right\} \quad . \tag{1.470}$$

4. **Á Posteriori Distribution** - The à priori distribution is

$$P_{\boldsymbol{x}/\boldsymbol{y}} = \left[\text{Diag}\left\{\boldsymbol{p}_{\boldsymbol{y}}\right\}\right]^{-1} \cdot \underbrace{P_{\boldsymbol{y}/\boldsymbol{x}} \cdot \text{Diag}\left\{\boldsymbol{p}_{\boldsymbol{x}}\right\}}_{P_{\boldsymbol{y},\boldsymbol{x}}} \quad . \tag{1.471}$$

5. **ML Detector** - An ML detector selects for any specific received DMC channel output $\boldsymbol{y} = j$ or thus row j:

$$\hat{\boldsymbol{x}}_i = \hat{i} = \arg\left\{\max_{i \in \{0,...,M-1\}} \left[p_{j/i}\right]\right\} \quad , \tag{1.472}$$

   the index of row j's largest element. The ML decision region $\mathcal{D}_i$ is the set of all row indices $\{j\}$ for which element i maximizes those rows' probabilities in $P_{\boldsymbol{y}/\boldsymbol{x}}$.

6. **MAP Detector** - An MAP detector selects for any specific received DMC channel output $y = j$:

$$\hat{\boldsymbol{x}}_i = \hat{i} = \arg\left(\max_{i \in \{0,...,M-1\}} \left\{\left[P_{\boldsymbol{y}/\boldsymbol{x}} \cdot \text{Diag}\left(\boldsymbol{p}_{\boldsymbol{x}}\right)\right](j,i) \; \right\}\right) \quad . \tag{1.473}$$

   The MAP decision region $\mathcal{D}_i$ is the set of all row indices $\{j\}$ for which element i maximizes those rows' probabilities in $P_{\boldsymbol{y}/\boldsymbol{x}} \cdot \text{Diag}\left\{\boldsymbol{p}_{\boldsymbol{x}}\right\}$.

The decision regions correspond to rows' maximum elements. When $\boldsymbol{x} = i$ was transmitted and $\boldsymbol{y} = j$ received, then when element i contains maximum value for all row j's values, then element is also directly the correct-decision probability, $p_{j/i} \cdot p_{\boldsymbol{x}}(i)$. The decision is in error for any element $i' \neq i$ that instead contains the maximum value in row j . The DMC's probability of a correct decision, P_c, for a

[50] The notation "Diag $(\boldsymbol{x})$" means a square diagonal matrix formed from its vector argument $\boldsymbol{x}$ with vector elements along the diagonal.
The vector $[\mathbf{1}]$ has all elements equal to 1.

specific input i simply uses elements from these matrices' rows as

$$P_c(\hat{i} = i,\ j \in \mathcal{D}_i) = \max_i p_{j/i} \cdot p_x(i) \quad . \tag{1.474}$$

The average error probability for a decision $\hat{i} = i$ with a optimum-decision-region (or really any decision region corresponding to the a specific) rule $\mathcal{D}_i$ and corresponding $p_{j/i}$ would then be

$$P_{c,max} \triangleq \mathbb{E}\left[P_c(\hat{i} = i,\ j \in \mathcal{D}_i)\right] = \sum_{i=0}^{M-1} \left\{ \sum_{j \in \mathcal{D}_i} p_{j/i} \right\} \cdot p_{\boldsymbol{x}}(i) \quad . \tag{1.475}$$

Thus the minimum average P_e for the MAP detector can be computed as

$$P_{e,min} \triangleq 1 - P_{c,max} = 1 - \sum_{i=0}^{M-1} \left\{ \sum_{j \in \mathcal{D}_i} p_{j/i} \right\} \cdot p_{\boldsymbol{x}}(i) \quad . \tag{1.476}$$

The ML detector sets $p_{x_i} = 1/M$ in the above.

An example is Figure 1.73's **Asymmetric Binary Channel**:

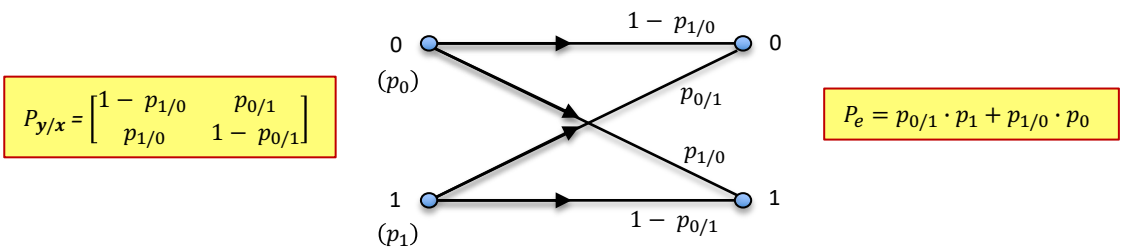


Figure 1.73: **The Asymmetric Binary Channel (ABC)**

The ABC has without loss of generality $\frac{1}{2} \geq p_{1/0} \geq p_{0/1} \geq 0$. The average ABC correct-decision probability (for both sub-optimum ML and MAP because they simplify to the same on this channel) is

$$P_c = (1 - p_{1/1}) \cdot p_1 + (1 - p_{0/0}) \cdot p_0 \tag{1.477}$$

or equivalently

$$P_e = p_{0/1} \cdot p_1 + p_{1/0} \cdot p_0 \quad . \tag{1.478}$$

The different inputs can be interchanged, and P_e also changes when $p_1 \neq p_{1/0} > p_{0/1} \neq p_0$, a characteristic asymmetry. As the message size increases, most channels become asymmetric.

1.4.2 Symmetric DMCs

A symmetric channel has P_e invariant to input labelling. That is, the design can relabel the channel inputs (without corresponding permutation of $P_{\boldsymbol{y}/\boldsymbol{x}}$'s columns) and maintain the same average error probability. The inputs thus are symmetric in their influence on performance. For instance, the AWGN channel with binary-antipodal input restriction is symmetric. The AWGN usually loses symmetry for $\bar{b} > 1$. This subsection focuses on symmetric DMC's.

Definition 1.4.2 [Symmetric Channel] *A* **symmetric channel** *has MAP-detector P_e that is independent of input distribution.*

The ML detector's function is always independent of input distribution, but the average error probability corresponding to the ML detector's use on a channel with non-equiprobable inputs is not necessarily input-distribution independent. However, simple review on Section 1.2's Minimax Theorem 1.1.2 proves that the ML detector's use on a symmetric channel also has average error probability independent of input distribution. Further, a **permutation** $\pi(\boldsymbol{y})$ reorders $\boldsymbol{y}$'s elements so $\breve{\boldsymbol{y}} = \pi(\boldsymbol{y})$.

Theorem 1.4.1 [Symmetric DMC Properties] *The following statements are equivalent:*

1. The DMC is symmetric.
2. The MAP and ML dectectors' error probability P_e is invariant to input distribution $\boldsymbol{p_x}$.
3. Any column of $P_{\boldsymbol{y/x}}$ is a permutation of another column.
4. For any 1-to-1 self-reversible permutation $\pi = \pi^{-1}$ on discrete $\boldsymbol{y}$, then $p_{\boldsymbol{y/x}}(i) = P_{\pi(\boldsymbol{y})/\boldsymbol{x}}(i')$ for some $i' \neq i$.

Proof:

- Statements 1 and 2 follow from symmetry's definition and the Minimax Theorem 1.1.2's identical precondition.
- Statement 3 implies that every column is also a permutation of all other columns. This means that the average error probability expression in (1.475) may interchange any two $P_{\boldsymbol{y/x}}$ columns $\boldsymbol{p}_{\boldsymbol{y/x}}(i)$ and $\boldsymbol{p}_{\boldsymbol{y/x}}(i')$ with respective corresponding input probabilities $\boldsymbol{p_x}(i)$ and $\boldsymbol{p_x}(i')$ and the resultant expression remains the same, which is the same as statement 1.
- Statement 4 is a mathematical way of rewording Statement 3 that often appears elsewhere in coding theory to define the symmetric channel.

QED.

An interesting symmetric DMC case occurs when $M' = M$ and thus the symmetric DMC can map inputs to finite field labels $\mathcal{GF}(\mathcal{M})$. This new channel description then has $\boldsymbol{y} = \boldsymbol{x} \oplus \boldsymbol{n}$ where $\boldsymbol{n}$ is independent of $\boldsymbol{x}$. The proof is in Exercise 1.57.

Three commonly encountered symmetric DMCs follow in the next 3 subsections:

1.4.2.1 The Binary Symmetric Channel (BSC)

Figure 1.74 illustrates the **binary symmetric channel (BSC)**. It is the ABC with $p_{1/0} = p_{0/1} = p < \frac{1}{2}$. The average error probability, and bit-error probability, is $\bar{P}_b = P_e = p$.

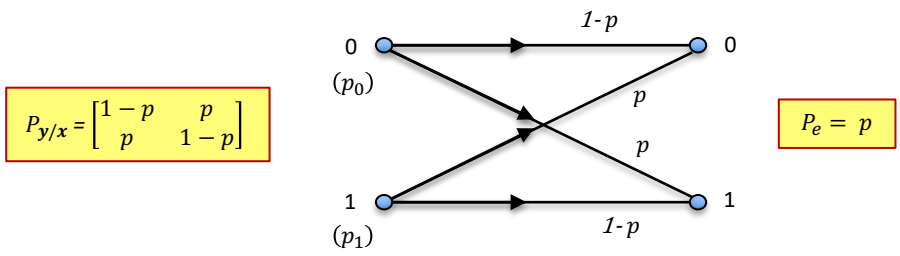


Figure 1.74: The Binary Symmetric Channel (BSC)

The BSC finds use in situations where a preliminary decision has already been made on a channel output with bit-error probability then $\bar{P}_b = p$. For an example of BSC creation, see Problem 1.58 Typically, Chapter 2's codes apply to $N > 1$ successive BSC uses so then symbols will be N-dimensional and binary. By itself with no codes, the BSC redundantly states the bit-error probability and provides no more insight.

1.4.2.2 The Binary Erasure Channel (BEC)

Figure 1.75's **binary erasure channel (BEC)** models detector uncertainty. When a channel like an AWGN has a $\boldsymbol{y} = \boldsymbol{v}$ value near a decision boundary, the probability of detection error is higher. Rather than decide definitively, a detector may instead choose to mark that decision as an **erasure**. By itself, the erasure is simply an error, but with coding over $N > 1$ dimensions or BEC uses, some codes can focus their corrective ability more on erased symbols (bits) than those for which the "inner" detector already has good confidence. For instance, a log-likelihood value near zero might be "erased."

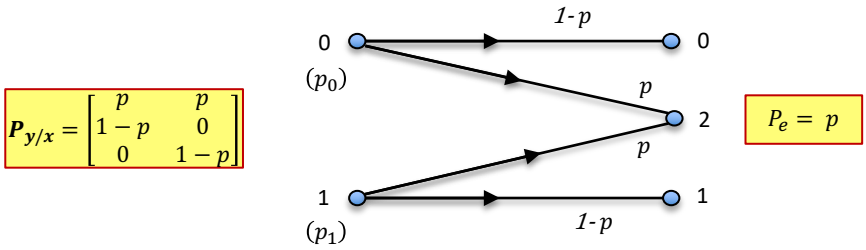


Figure 1.75: **The Binary Erasure Channel (BEC), 2= erase.**

Problem 1.59 addresses creation of a BEC.

1.4.2.3 The symmetric q'ary-DMC

The q-ary symmetric channel typically finds use with systems where blocks of $\log_2(q)$ bits pass through the DMC. All q inputs essentially are equivalent in terms of possible detector error $P_e = p$. Again, outer q-ary codes often apply to blocks of q-ary symbols. Such systems can with small redundancy or overhead (see Chapter 2) take an error rate like $p = 10^{-4}$ and drive it very close to zero to render the overall system highly reliable.

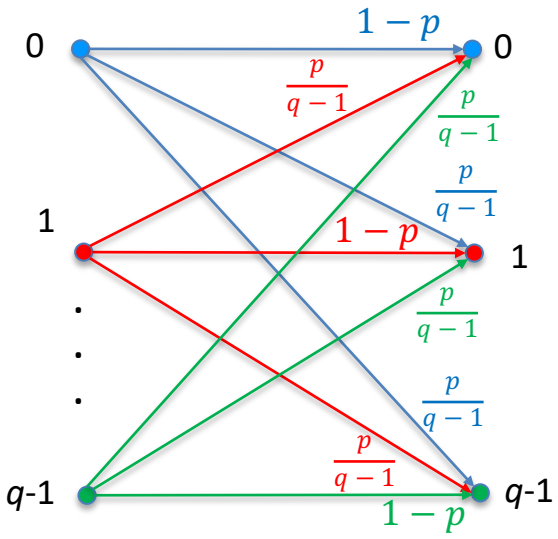


Figure 1.76: The q-ary Symmetric Channel (QEC)

1.5 Linear and Non-Linear Single-Message Channel Models

The discrete-time AWGN's linear-model generalization is

$$\boldsymbol{y} = H \cdot \boldsymbol{x} + \boldsymbol{n} \quad , \tag{1.479}$$

where H models inter-dimensional interference. One common example is MIMO wireless where the $((L_y \cdot N) \times (L_x \cdot N))$ H matrix models spatial crosstalk from L_x transmit antennas to each of L_y receive antennas, possibly with temporal inter-dimensional interference in time-frequency also on the N temporal dimensions. A single-dimensional $(L_y = L)_x = 1$wireline or wireless channel may also block N time-dimension uses into an $N \times N$ channel[51] H for each transmitted symbol $\boldsymbol{x}$ and received symbol $\boldsymbol{y}$. For this section the dimensionality is simply $N_y = L_y \cdot L_y$-dimensional outputs for $N_x = L_x \cdot N$-dimensional inputs.

Various communication applications best have nonlinear channel models, for which (1.479) generalizes to

$$\boldsymbol{y} = f(\boldsymbol{x}) + \boldsymbol{n} \quad , \tag{1.480}$$

as in Figure 1.77 where (1.479)'s linear model $(H\boldsymbol{x})$ is subset of a more general (and essentially arbitrary) vector function $f(\boldsymbol{x})$.

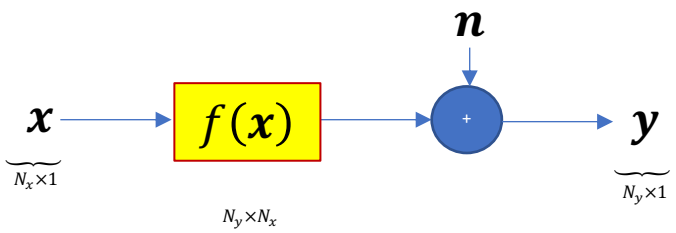


Figure 1.77: **General nonlinear channel in (1.480).**

Simple nonlinearities may involve limiting or saturation of amplifiers or even of conversion devices (digital-to-analog and analog-to-digital converters). More complex models can allow problems that initially might not appear to be communication problems be modeled as such. This section will continue with stationary channels, but those that can be modeled more generally as in (1.480) while Chapters 3 - 5 examine many widely used methods for (1.479)'s linear channel.

Equation (1.480)'s noise $\boldsymbol{n}$ remains AWGN[52] and so the maximum-likelihood detector remains the same in concept as "pick the closest" noise free channel vector and decode the corresponding channel input vector as the best estimate (that is minimizes the symbol-error probability):

$$\hat{\boldsymbol{x}} = \arg \left[\min_{\hat{\boldsymbol{x}}} \|\boldsymbol{y} - f(\hat{\boldsymbol{x}})\|^2 \right] \quad . \tag{1.481}$$

The minimization considers all possible channel-input symbols. The channel-output space has decision regions whose boundaries bisect the lines between the possible (noise-free) channel-output symbol vectors. Figure 1.78 provides an example. For a large number of output dimensions and/or input messages 2^b, this simple concept can become very complex to implement.

[51] Chapters 3 - 5 address handling transients' overlaps between symbols or **intersymbol interference**, but will preserve (1.479)'s linear model.

[52] Self-correlated noise can be whitened by a whitening square-root-inverse matrix and this alters the function f to a new function, but all proceeds identically.

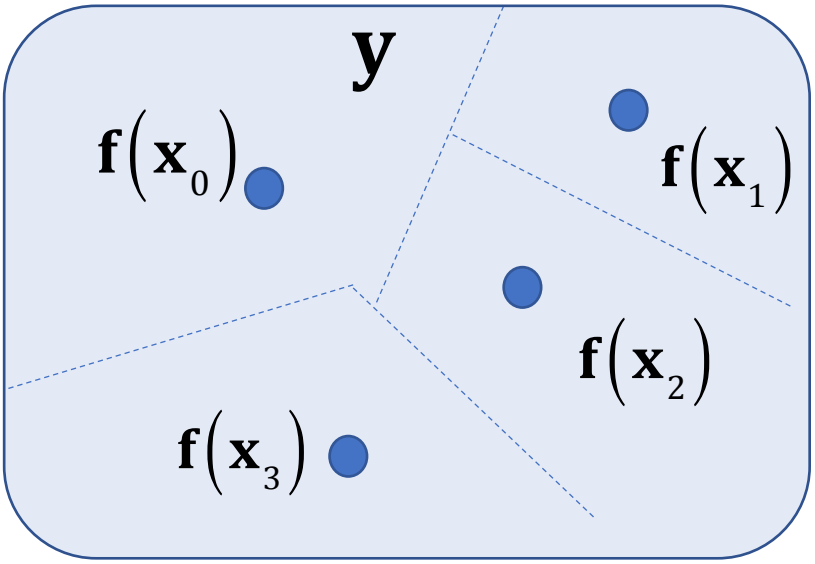


Figure 1.78: **Channel-identification basics and terminology.**

Chapters 2 - 5 for linear channels will find the best receivers essentially decompose the linear channel into an independent set of simple AWGNs; however one such **vector-coding** receiver employs singular-value-decomposition

$$H = F \cdot \Lambda \cdot M^* \tag{1.482}$$

where $F \cdot F^* = F^* \cdot F = I$, $M \cdot M^* = M^* \cdot M = I$ and Λ is (always real, even for complex H)) diagonal in its largest upper-left-most square portion, and zero elsewhere. The best receiver begins with F^* and the corresponding best transmitter ends with M so that the linear channel then transforms into

$$\boldsymbol{Y} = \Lambda \cdot \boldsymbol{X} + \boldsymbol{N} \quad , \tag{1.483}$$

which is a set of independent AWGN channels (with different gains, or energies, as $|\Lambda|^2 \cdot \mathcal{E}_{\boldsymbol{x}}$ diagonal elements. There are maximally $\min(N_y, N_x)$ channels with nonzero values along Λ's diagonal. These "subchannels" can be independently modulated and decoded with ML detectors. This section attempts to generalize this SVD-based concept to more general nonlinear channel models.

1.5.1 The Nonlinear Channel State and Consequent Decoders

A nonlinear channel model with a discrete set of $M = 2^b$ input symbols often divides into corresponding piecewise linear domains. There is then a set S of state indices, $s = 1, ..., |S|$, which correspond to these piecewise-linear input domains $\Theta_s(\boldsymbol{x})$. The integer quantity s denotes a **nonlinear-channel state index**, where $|S| \leq 2^b$. For each such state, the corresponding channel-output range may be written as $\mathcal{R}_s(\boldsymbol{y})$. Good constellation design would place domain boundaries on well-chosen decision-region boundaries. Clearly when $|S| = 2^b$, the ML decoder returns to decision regions and each domain corresponds to a decision region, and no simplification is possible. For each state index, the channel is linear within the domain:

$$\boldsymbol{y} = H_s \cdot \boldsymbol{x} + \boldsymbol{n} \; \forall \; \boldsymbol{x} \in \Theta_s(\boldsymbol{x}) \quad . \tag{1.484}$$

There are only 2^b input messages, so there can be maximally no more than $|S| = 2^b$ such models. In practice, many nonlinear channels will have $|S| << 2^b$. For instance, the linear channel has $|S| = 1$, and

the single domain is trivially as all complex vectors, $\Theta_1(\boldsymbol{x}) = \mathcal{C}^{N_x}$. The corresponding range associated with the linear channel is $\mathcal{R}_1(\boldsymbol{y}) = \mathcal{C}^{N_y}$. There are thus $|S|$ linear models, each with corresponding input domain and output range.

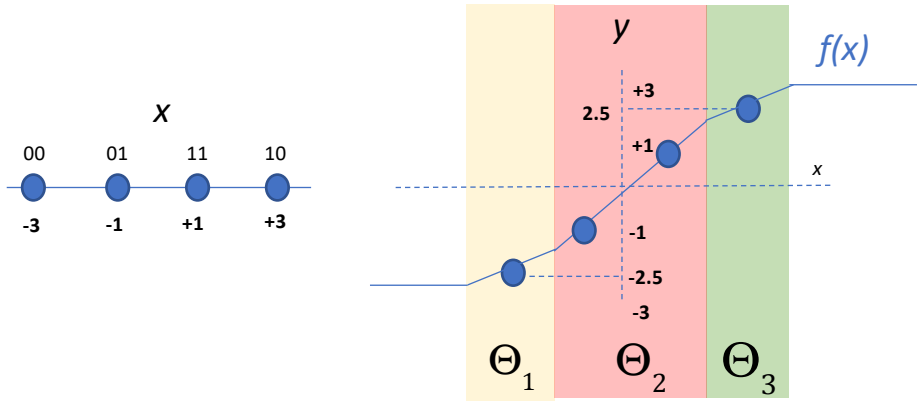


Figure 1.79: **Simple hard-limiter channel.**

A simple and very practical nonlinear-channel example is Figure 1.79's one-dimensional real channel. Such a channel model applies when transmit amplifiers may saturate near a limit. Figure 1.79's 4-level PAM input x passes through the channel function $f(x)$ to produce the hard limited output. The output model has 3 domains and corresponding ranges. The inner two channel-output symbol vectors are within a single region $\Theta_2(x)$. In this case the range of the function (presuming the slope of the line is 1) is $\mathcal{R}_2(y) = \Theta_2(y) = \{y | |y| \leq 2\}$. The point -3 is in domain $\Theta_1(x)$, while +3 is in $\Theta_3(x)$. The corresponding ranges $\mathcal{R}_1(y)$ and $\mathcal{R}_3(y)$ are linear, but with reduced slope relative to $\Theta_2(x)$.

This channel's ML decoder decides the first bit with a simple 1-bit ADC and decides the second bit as 0 if $|y| > 1.75$, and 1 otherwise. This is clearly simpler than finding the squared distance to all 4 points and comparing. The two domains $\Theta_1(x)$ and $\Theta_3(x)$ effectively reduce to $\Theta_{13}(\boldsymbol{x}) = \Theta_1(x) \bigcup \Theta_3(x)$ for this simple channel.

Of more interest in nonlinear channel modeling is Figure 1.80's soft-limiter. Here 8 PAM experiences different channel gain levels because the amplifier is not perfect, especially near the maximum values. There are 3 domains in this piecewise-linear nonlinear channel, and 3 corresponding ranges. Each range decodes separately, once the receiver decides in which region (range shown by the colored shades in Figure 1.80. There may be a bias (constant added/substracted) prior to the simple decoders. Such a system is simpler than computing all 8 squared distances. This system could assign input bits to reduce to two linear regions, given the symmetry of $\Theta_1(\boldsymbol{x})$ and $\Theta_3(\boldsymbol{x})$ - in general the positive and negative saturation may be different so 3 regions would be necessary.

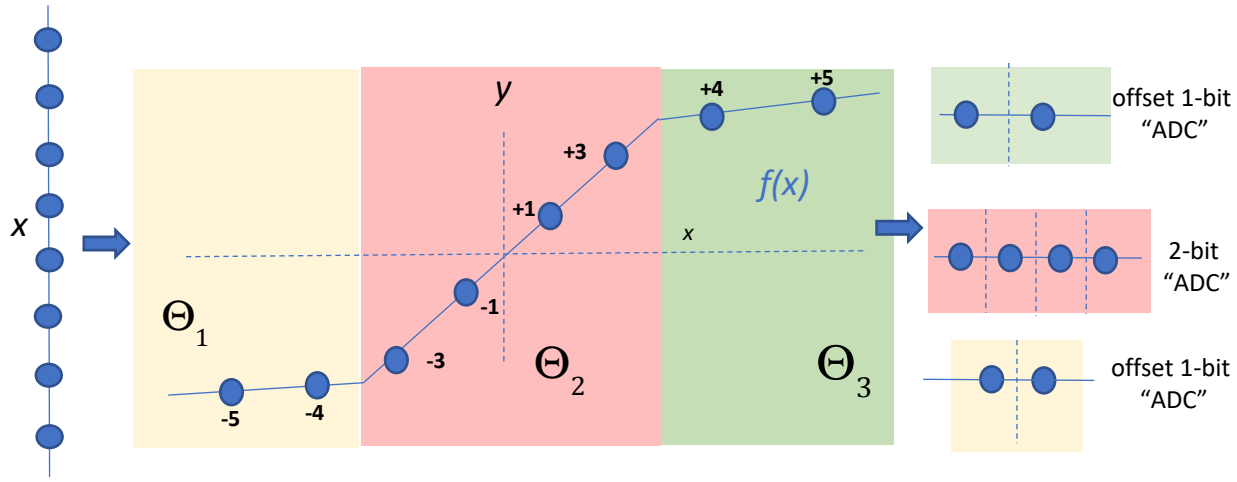


Figure 1.80: **Soft-limiter channel.**

The hard- and soft-limiter examples are one-dimensional, so the singular value decomposition is trivial with $M = F = 1$ for all domains and corresponding ranges; the ML decoders were obvious upon inspection. Figure 1.81's two-dimensional example helps illustrate the concept further,.

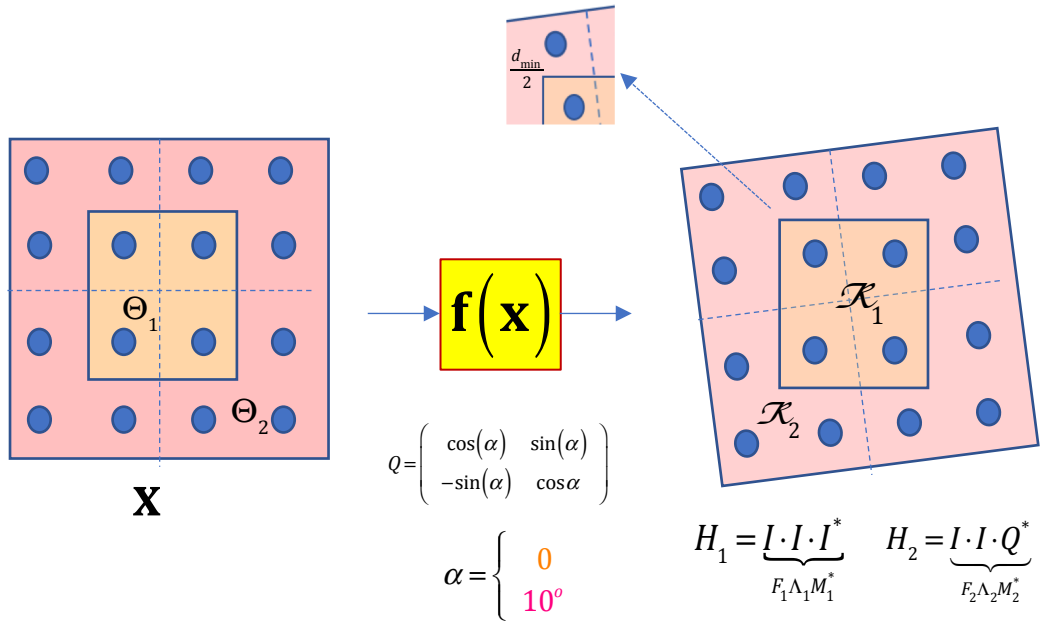


Figure 1.81: **Two-dimensional nonlinearity with amplitude-dependent rotation.**

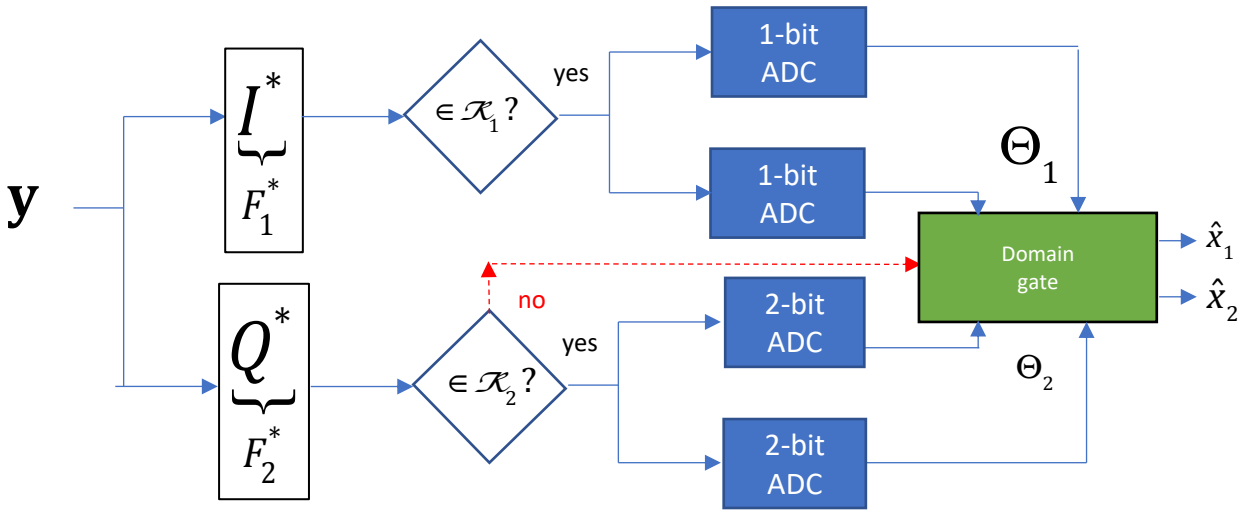


Figure 1.82: **Simplified decoder for the two-dimensional example (the I^* and Q^* are matrices, not "inphase" and "quadrature").**

Basically, low amplitude inner points in the orange-shaded region do not "twist" while larger amplitude points do twist. (A more complete model might rotate the $\pm 3, \pm 3$ points even more, but for illustration purposes here, 2 domains are sufficient. The decision on which range uses received signal amplitude.) In this case the SVD is shown in Figure 1.81 but now slightly more complicated and different for the two domains. The minimum distance is shown and half of it is from a point in $\mathcal{R}_2$ to the boundary $\mathcal{R}_1$. Rather than testing all 16 points for proximity to a received channel output, the decoder in Figure 1.82 shows the two decoder computation paths corresponding to the two ranges. Each range-related decoder path is fairly simple and follows the vector-coding model for that range's symbols. The minimum distance is slightly smaller as in Figure 1.81's magnified side bar, and so the receiver could twist erroneously an inner point or not twist an outer point. This probability is the same as the full ML decoder that looks at all 16 points because it depends on the same minimum distance. Thus, there is a simpler detector with essentially the ML detector's performance (the nearest neighbor count for the twist detector would actually be slightly higher if the twist occurs only based on received signal magnitude).

1.5.1.1 Multivariate Taylor Series and the Volterra Series

A function's **multi-variate Taylor Series** of a vector input $\boldsymbol{x}$ expands around the point $\boldsymbol{x}_0$ as:

$$f(\boldsymbol{x}) = f(\boldsymbol{x}_0) + \nabla f(\boldsymbol{x}_0) \cdot (\boldsymbol{x} - \boldsymbol{x}_0) + \frac{1}{2} \cdot (\boldsymbol{x} - \boldsymbol{x}_0)^* \cdot \mathcal{H}_f(\boldsymbol{x}_0) \cdot (\boldsymbol{x} - \boldsymbol{x}_0) + \text{ higher-order terms } , \quad (1.485)$$

where $\nabla f(\boldsymbol{x})$ is the function $f(\boldsymbol{x})$'s's gradient (a row vector) and $\mathcal{H}_f(\boldsymbol{x})$ is the square Hessian matrix of second partial derivatives. A region around $\boldsymbol{x}_0$ for any such point where only the first two terms in 1.485 are significant is the domain $\Theta_0(\boldsymbol{x})$. When more than one transmission symbol value is in a domain, piecewise linear models simplify the nonlinear-channel representation for the given symbol constellation. A multivariate Taylor Series can thus help identify the domains for a nonlinear function. For a vector function $\boldsymbol{f}(\boldsymbol{x})$, this argument to applies to each function's element, and input domains can apply to multiple elements of the vector function.

1.5.1.2 Continuous Models

There are other more smooth nonlinear models for saturation, for instance a model by Saleh is well-known and has QAM constellation points or more generally any signal with amplitude r and phase θ being distorted to

$$r \rightarrow \frac{\alpha_a \cdot r}{1 + \beta_a \cdot r^2} \tag{1.486}$$

$$\theta \rightarrow \frac{\alpha_\theta \cdot r^2}{1 + \beta_\theta \cdot r^2} \ , \tag{1.487}$$

where usually $\alpha_a = 2.1587$, $\beta_a = 1.1517$, $\alpha_\theta = 4.033$, and $\beta_\theta = 9.1040$. However, this model does not readily map into a piecewise linear model (even though it may be more accurate). It is the piecewise linear model that is allowing the "states" to occur corresponding to the input domains. Problem 1.61 explores this further.

More generally, the piecewise linear derives from up to $|S|$ singular-value decompositions

$$H_s = F_s \Lambda_s M_s^* \ . \tag{1.488}$$

The best transmitter for points in domain Θ_s is thus M_s and the corresponding receiver is F_s^*. These are orthogonal matrices that do not change the volume nor shape of the respective domains and ranges, but rather just rotate them. It is possible with nonlinear channels that some input points from different domains will be mapped into the same (or very close) range points by the channel nonlinearity - this can reduce minimum distance for the simplified decoder, as in Figure 1.81's example. Such rotated channel-output symbols from different domains/ranges that are too close should assign the same message such symbols to avoid the inevitable low performance (thus nonlinearity can reduce data rate or performance when this occurs). Constellation design shoud address such message mapping. Figure 1.83 generically depicts the corresponding decoder for the maximum number of domains; however simplification usually occurs when this number of domains is well below M.

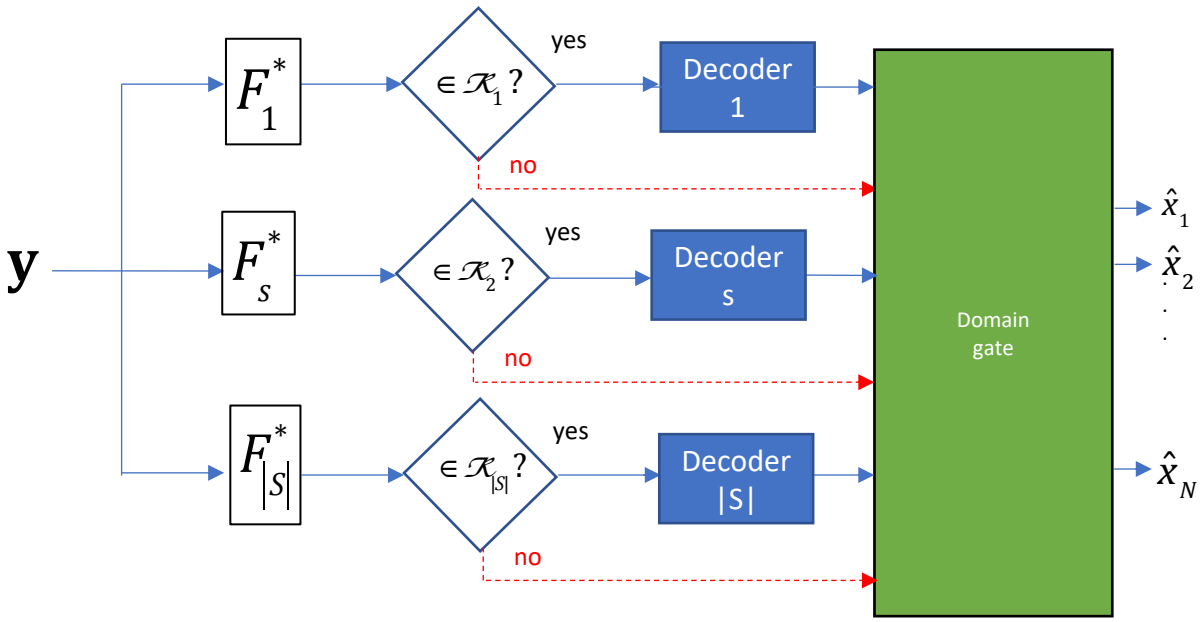


Figure 1.83: **General nonlinear decoder based on piecewise-linear mdoels.**

1.5.2 The relationship of the nonlinear model to neural-network modeling

There are a wide variety of approaches and methods to nonlinear modeling, including many approaches to neural networks for modeling correlated input/output relationships (including nonlinear functions).

One such neural network model in wide use is based on **rectified linear units (ReLU's)** as illustrated in Figure 1.84.

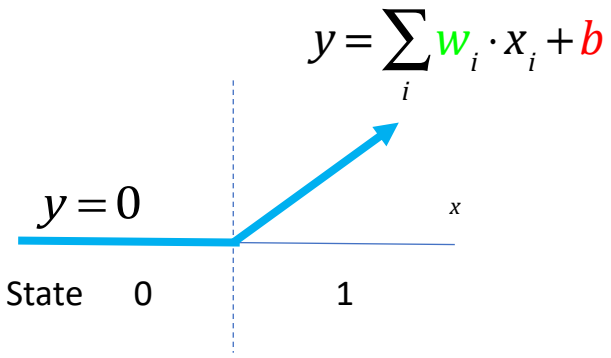


Figure 1.84: **Rectified Linear Unit (ReLU).**

ReLUs pass positive inputs and block negative inputs. A state s can be associated with a ReLU corresponding to the block (0) or pass (1) actions, so they depend on y's input polarity. A scalar bias b adds (not to be confused with the number of bits). Figure 1.84's mathematical equation presumes the input comes possibly from a previous stage of ReLUs (x_i)'s to create an output y. Essentially the bias b can move the output threshold from the cross-over point at the origin. Figure 1.85 illustrates ReLU's previous state that includes the linear combination.

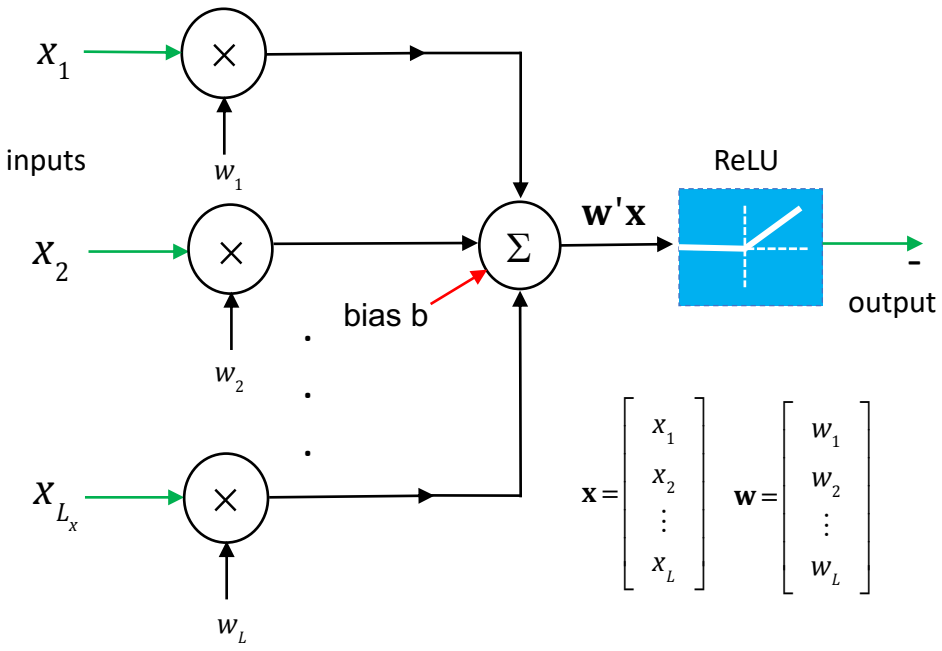


Figure 1.85: **The Neuron.**

Figure 1.86 shows multiple ReLU's use in a neural network to model f.

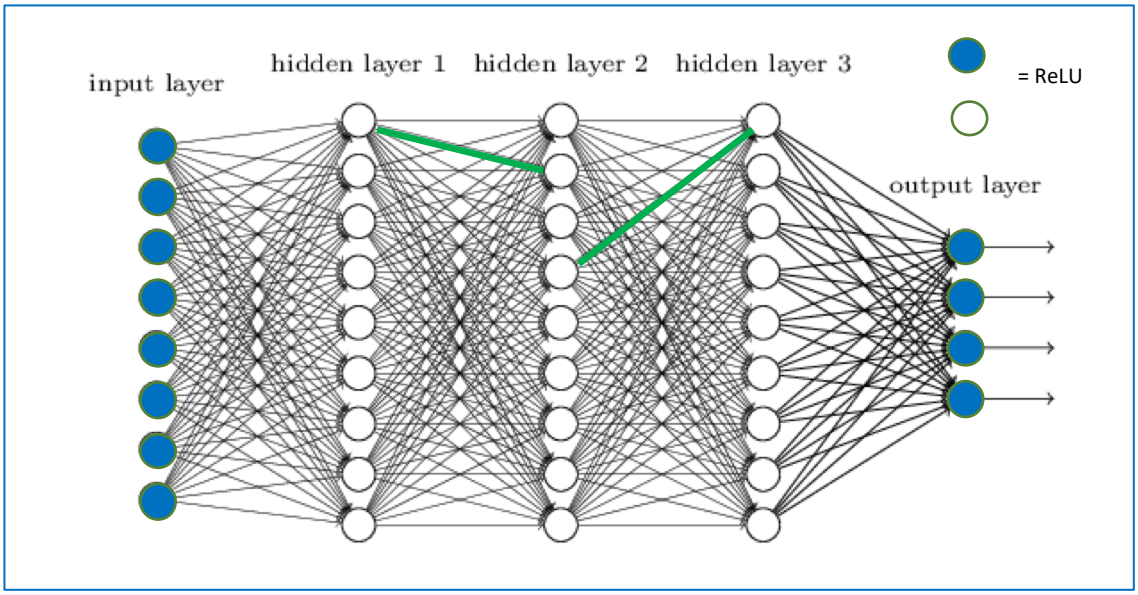


Figure 1.86: **Neural Network constructed using ReLU's.**

The state s of the piecewise-linear models of the previous subsection basically maps to the vector of states of all the constituent ReLU's in the neural net. For each of these states, there is a linear (really affine, as a vector $\boldsymbol{b}$ is added as a consequence of all the biases) transformation.

The sum of products that precedes the nonlinear rectification is linear and and array of such ReLU's feeding another array has a matrix transfer W_i for each stage. The number of such stages is the depth δ of the neural net. If the output of a particular sum of products is negative (after bias included), then the ReLU outputs zero. If not, the sum of products is passed. The pass/block actions can be represented by a matrix of 1's and 0's that is denoted $J_{s,i}$, which will have a 1 in a position corresponding to the preceding stages ReLU passing its information and a 0 in entries corresponding to a block. The overall transfer will then be

$$H_s = \prod_{i=1}^{\delta} W_i \cdot J_{s,i} \quad , \tag{1.489}$$

a linear matrix multiply that depends on the input state-index s (through $J_{s,i}$ but not through W_i). The overall output vector could then be written as

$$\boldsymbol{y} = H_s \boldsymbol{x} + \boldsymbol{b}_s + \boldsymbol{n} \quad . \tag{1.490}$$

Revisiting the hard limiter, Figure 1.87 now shows a cascade of two ReLUs with biases that model this channel. There would be 4 states nominally for 2 ReLU's, but two of them correspond to the interior region so just $|S| = 3$ states.

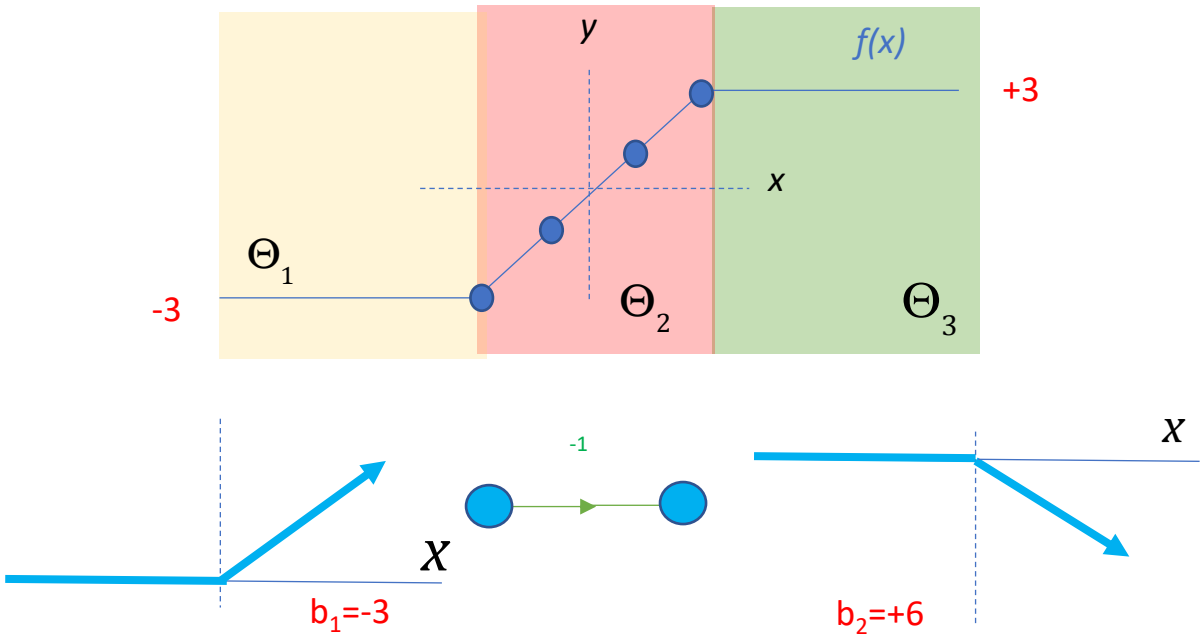


Figure 1.87: **Neural Network model of hard-limiting channel.**

Figure 1.88 shows how the ranges corresponding to different rotations could overlap. Common points in these ranges should decode to the same message by design of the code (see the blue points). This reduces the number of messages usually with respect to a linear channel, but eliminates ambiguity. Points that are very close should also be mapped to the same message, again to avoid performance degradation.

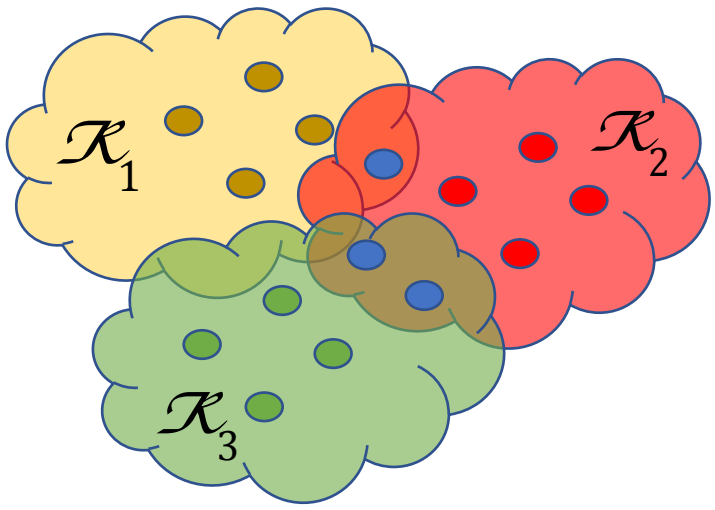


Figure 1.88: **Illustration of overlapping channel-output ranges with common points.**

Problem 1.60 explores this for the simple nonlinear rectifier channel.

1.6 Statistically parameterized Variable Channels

The most general channel-variation model is statistical, and is this section's subject. This model includes time variation as a special case.

Definition 1.6.1 [Statistically Parameterized Channel]

A **statistically parameterized channel**, $p_{\boldsymbol{y},a/\boldsymbol{x}}(\boldsymbol{v},\alpha,\boldsymbol{u})$, *has an additional random variable* a *with values in set* $\mathscr{A}$, *so* $a \in \mathscr{A}$. *The channel model varies*[53] *with the parameter* a. *A specific sample value of* $a = \alpha$ *corresponds to a set of 1 or more channel uses. Another* $a = \alpha' \neq \alpha$ *sample value similarly corresponds to another set of channel uses.*[54] *For each sample value* $a = \alpha$, *the channel has joint probability given the channel input:*

$$p_{\boldsymbol{y},a/\boldsymbol{x}}(\boldsymbol{v},\alpha,\boldsymbol{u}) = p_{\boldsymbol{y}/[\boldsymbol{x},a]}(\boldsymbol{v},\alpha,\boldsymbol{u}) \cdot p_{a/\boldsymbol{x}}(\alpha,\boldsymbol{u}) \quad . \tag{1.491}$$

For this text's statistically parameterized channels, the random variable a *and the channel-input symbol* $\boldsymbol{x}$ *are independent. Thus,* $p_{a/\boldsymbol{x}}(\alpha,\boldsymbol{u}) = p_a(\alpha)$ *and*

$$p_{\boldsymbol{y},a/\boldsymbol{x}}(\boldsymbol{v},\alpha,\boldsymbol{u}) \to p_{\boldsymbol{y}/[\boldsymbol{x},a]}(\boldsymbol{v},\alpha,\boldsymbol{u}) \cdot p_a(\alpha) \quad , \tag{1.492}$$

essentially a weighted (by $p_a(\alpha)$*) and indexed (by* α*) channel model.*

The independence of $\boldsymbol{x}$ and a merits note: An input-dependency of a would effectively augment the channel output $\boldsymbol{y} \to [\boldsymbol{y}, a]$, which reverts all to this chapter's previous analysis. Thus, the statistically parameterized channel addresses potential channel variation, not channel-output observation. Instead a parameterizes the receiver/detector's performance. A certain optimum (ML/MAP) performance occurs for each a value. Wireless system analyses often average over the possible a values. Usually ML/MAP detectors' analysis depends on the specific a value (although not always). This implies the receiver's design knows a for these ML/MAP detectors' correct implementation. In practice, the value a may not be directly observable itself and the receiver may adaptively estimate $\hat{a}$ in an adaptive ML-detector implementation. Thus, while the receiver knows a, the analysis uses only a's probability distribution/density. Chapter 7 investigates estimation of channel parameters with a single message-use group. Equation (1.492) suggests a channel "weighting" by $p_a(\alpha)$ for each random parameter value. This random parameter a may be discrete or continuous with probability distribution or density, respectively. Previous channel-parameter examples include the AWGN's $\frac{\mathcal{N}_0}{2}$ and the BSC's p. These were given constants and not random; equivalently they parameterize the channel distribution $p_{\boldsymbol{y}/\boldsymbol{x}}$. In this deterministic-parameter-channel case, equivalently the distributions are $p_{a,AWGN}(\alpha = \frac{\mathcal{N}_0}{2}) = 1$, or $p_{a,BSC}(\alpha = p) = 1$ respectively, and 0 otherwise; similarly the continuous probability-density examples are $p_{a,AWGN}(\alpha) = \delta(\alpha - \frac{\mathcal{N}_0}{2})$ or $p_{a,BSC}(\alpha) = \delta(\alpha - p)$.

Time-variation The random variable a may generalize to a random process $a(t)$ with the time variable t essentially indexing the previously mentioned channel-use groups, creating the most general form of a **time-varying channel**. The channel distribution may vary from message group to message group. Subsection 1.6.3 discusses further such variation. Presently, $p_a(\alpha)$ is a stationary distribution that characterizes the channel variation. This is important to emphasize again: The channel $p_{\boldsymbol{y},a/\boldsymbol{x}}$ itself is not truly stationary because a varies, but a stationary distribution characterizes that variation's parametrization. A time-varying channel can also be deterministic so that $a(t)$, or really then just t, indexes a channel parameter's known deterministic time-dependency.

For instance, a time-varying BSC might have time-varying error-probability parameter $p \to p(t)$. A statistically parameterized channel may have a continuous probability distribution that specifies values for $a = p \in \mathscr{A} = [0, 0.5)$. Any of these random p values might occur for any particular channel message group at specific time t for the statistically parameterized BSC. A specific known value occurs at each time t in the deterministic time-varying case. This deterministic time-varying case would easily have

a known ML detector with random corresponding time-varying error probability $P_e = p(t)$ for use at time t. Instead, the statistically parameterized channel has a random $P_e = p$ that invites statistical performance analysis.

Sample Averages and Ergodicity This text's convention admits a different **angle-bracket** notation to represent expectation over a and to distinguish this "average" from the E used for the noise and channel input expectations. Thus[55],

$$\langle a \rangle = \sum_{\alpha \in \mathscr{A}} \alpha \cdot p_a(\alpha) \ . \tag{1.493}$$

Similarly previous channel models might derive from the average

$$p_{\boldsymbol{y}/\boldsymbol{x}} = \langle p_{\boldsymbol{y}/[\boldsymbol{x},a]} \rangle = \sum_{\alpha \in \mathscr{A}} p_{\boldsymbol{y}/[\boldsymbol{x},a]} \cdot p_a(\alpha) \ . \tag{1.494}$$

Thus, for instance, the previous AWGN channel could be instead viewed as an averaged set of parameterized additive white Gaussian noise channels where each has noise as $n = a \cdot n'$, where a is independent of n', and thus $\frac{\mathcal{N}_0}{2} \to \langle a^2 \rangle \cdot \frac{\mathcal{N}_0}{2}'$. Or, if the BSC had different values of $p' = a$ for different use/dimension groups, then the previous BSC's $p \to < a >$.

Definition 1.6.2 [Sample Average] *Often, the angle-bracket average also represents a "sample average" that derives from a set of observations a_j for $j = 1, ..., J$:*

$$\langle a \rangle_J = \frac{1}{J} \cdot \sum_{j=1}^{J} a_j \ , \tag{1.495}$$

or more generally for any (deterministic) function $f(a)$ of random a,

$$\langle f(a) \rangle_J = \frac{1}{J} \cdot \sum_{j=1}^{J} f(a_j) \ . \tag{1.496}$$

The sample average, function of a's sample average, essentially models a **Monte Carlo** analysis that randomly selects a values from the distribution, computes deterministic performance thereof, and then averages over those performances. When the sample average tends to the angle-bracket average, then the random process implied by the indices j on a's sample observation values is **ergodic**, formally:

Definition 1.6.3 [Ergodic Sample Averages] *Ergodic sample averages satisfy*

$$\langle a \rangle = \lim_{J \to \infty} \langle a \rangle_J = \lim_{J \to \infty} \left\{ \frac{1}{J} \cdot \sum_{j=1}^{J} a_j \right\} \tag{1.497}$$

or (1.498)

$$\langle f(a) \rangle = \lim_{J \to \infty} \langle f(a) \rangle_J = \lim_{J \to \infty} \left\{ \frac{1}{J} \sum_{j=1}^{J} f(a_j) \right\} \tag{1.499}$$

over the sequence of increasingly large sets of size J.

[55] An integral replaces the sum when a has continuous distribution.

Ergodic-Average Performance Analysis A ergodic analysis will compute the average error probability (or more generally its distribution) over the random parameter a to represent the average[56] performance of some receiver on the statistically parameterized channel. There is an optimum MAP (or ML) receiver/detector for each value of a, presuming the receiver knows this specific value. These optimum receivers, or any sub-optimum receivers, have corresponding $P_{e,a}$ for a specific $a = \alpha \in \mathscr{A}$.

The consequent **ergodic-average error probability** over a is

$$\langle P_e \rangle = \sum_{\alpha \in \mathscr{A}} P_{e,\alpha} \cdot p_a(\alpha) \quad . \tag{1.500}$$

If the receiver knows (or learns) a, then the (adaptive) ML detector corresponds to this value a. If a is not known, then the receiver may be suboptimal. For instance BPSK ML detectors do not depend on noise variance (nor channel attenuation) so would still be ML for all $a > 0$ values if the noise were $a \cdot n(t)$. Other detectors might depend on a (like 16 QAM for instance). Nonetheless, whatever the receiver structure, the average over the possible error probabilities for different a values determines $\langle P_e \rangle$.

For wireless channels, a has value from the set $\mathscr{A}$ where different selection often represents channel time change. Statistics thus model time-change. In this way $100 \times p_a(\alpha)$ represents[57] the percentage of time that the channel has a certain parameter $a = \alpha$. However, a more generally may vary with time, frequency, space, or with other dimensional characterization in the case of Section 1.4's finite-output channels. Indeed, a may be instead $\boldsymbol{a}$ to characterize a vector parameter set that may change or even a matrix set (see Subsection 1.6.3.3) A as convenient.

While the ergodic-average error probability is informative, complete statistical characterization involves more than than simple averages. It is possible to compute error-probability distributions, moments, etc. One important metric in data transmission is the **outage probability**, which is essentially a confidence-interval indication of transmission-system performance.

Definition 1.6.4 [Outage Probability] *An* **outage** *occurs when the random error probability is above a constant threshold δ that characterizes the outage event,*

$$P_{e,a} > \delta \quad . \tag{1.501}$$

The **outage probability** *is the probability that the outage event occurs.*

$$P_{out}(\delta) = Pr\{P_{e,a} > \delta\} \quad . \tag{1.502}$$

With varying channels, the constant outage probability may essentially replace the random error probability (itself an average over inputs and channel disturbances/noise) to remove the dependence on the random parameter a. The quantity δ defines the outage event and is the largest acceptable loss of data/connection. It is comparable to the earlier stationary channel's P_e. Acceptable connection loss is a function of higher system-level actions that restore the connection, possibly through lost-packet retransmission, connection restart ("reboot"), or perhaps acceptable loss of voice/video quality, which is also true for P_e. The designer's target outage probability may be a function of applications expected to be in use. An outage probability of 5% may be acceptable for typical internet traffic like emails or web browsing because those applications have retransmission mechanisms that will resend lost packets and the system users may not notice the outage. Video and audio systems may require outages closer to 1%. Some high-quality networks require **five nines** reliability or $< P_{out} >= 10^{-5}$. Usually $\langle P_e \rangle < P_{out}(\delta)$ for reasonable distributions and choices of δ. A statistically parameterized channel will typically need

[56] More generally, this is a random parameter-sample average.

[57] Or, $100 \cdot p_a(\alpha) \cdot d\alpha$ for a continuous probability density.

Chapter 2's codes to ensure sufficient redundancy in a message sequence so that adequate recovery is possible from the outage.

So for instance, Section 1.4's BSC may well model another transmission system with outage probability $p = \frac{P_{out}}{b}$, again illustrating that the earlier channel models may themselves represent an aggregation, in this case over a random parameter's distribution, of various channel effects.

Spectral and spatial variation: This section mostly addresses time-variation, and later chapters (3 and 4 especially so) often use the index $n \to k$ to indicate time explicitly. Statistical variation can also occur when the index n refers to space as antenna index $n \to \ell$ or a spectral index n that enumerates independent carrier frequencies. In general the statistically variable parameter may have correspondingly different distributions corresponding to the type of dimensionality. Ergodicity can generalize then to mean that the distribution over the dimensions remains the same, but usually ergodicity refers to the channel being time indexed and the distribution from which the variable parameter arises remains constant over each time selection therefrom.

Section summary: Subsection 1.6.1 investigates the concept of memoryless channels in this expanded context of a statistically parameterized channel model. Subsection 1.6.2 investigates a very common AWGN channel that has random gain (and consequently random SNR). Various well-known channel-model distributions appear as examples for analysis. Subsection 1.6.3 instead expands the finite-output channel models of Section 1.4, which tend to have best use when modeled by a state machine, or Markov Model, that allows the discrete channel to have state-represented parameter and consequent state-dependent transition probabilities. Subsection 1.6.3 also introduces coherence concepts that model the situation where p_a rvaries from message group to message group, but still has long-term ergodic properties.

1.6.1 Memoryless Channels and Dimensionality

Memoryless channels have independent message transmissions: a series of one-shot uses. Memoryless channels therefore accept transmit symbols $\boldsymbol{X} \triangleq \{\boldsymbol{x}_k\}_{k=1,...,K}$ and produce corresponding received channel-output symbols[58] $\boldsymbol{Y} \triangleq \{\boldsymbol{y}_k\}_{k=1,...,K}$ such that

$$p_{\boldsymbol{Y}/\boldsymbol{X}} = \prod_{k=1}^{K} p_{\boldsymbol{y}_k/\boldsymbol{x}_k} \quad \forall\, k = 1, ..., K \; . \tag{1.503}$$

The previous AWGN and all the DMC's are memoryless channels. When the channel is stationary, $p_{\boldsymbol{y}_k/\boldsymbol{x}_k} = p_{\boldsymbol{y}/\boldsymbol{x}}$, so then $p_{\boldsymbol{Y}/\boldsymbol{X}} = \left(p_{\boldsymbol{y}/\boldsymbol{x}}\right)^K$, which has been the case prior to this section. The stationary channel has the same ML detector for each symbol. When the channel is time-variant (or more generally dimensionally variant) with k as use index, each symbol's ML detector remains independent of the other symbols' ML detectors, but those detectors may each be different. If the inputs are also mutually independent so that $p_{\boldsymbol{X}} = \prod_{k=1}^{K} p_{\boldsymbol{x}_k}$, then MAP detectors are independent also, and similarly potentially each different.

Memoryless channels simply generalize in the statistically parameterized case:

$$\begin{aligned} p_{\boldsymbol{Y},a/\boldsymbol{X}} &= \prod_{k=1}^{K} p_{\boldsymbol{y}_k,a/\boldsymbol{x}_k} \quad \forall\, k = 1, ..., K \;\wedge a \in \mathscr{A} && (1.504) \\ &= (p_a)^K \cdot \prod_{k=1}^{K} \left(p_{\boldsymbol{y}_k/[a,\boldsymbol{x}_k]}\right) && (1.505) \\ &= \left(p_{\boldsymbol{y}/\boldsymbol{x}}\right)^K \text{ when stationary.} && (1.506) \end{aligned}$$

[58] The index name k appears here instead of the more general dimensional index n as this text uses k for a discrete-time index.

The ML (and MAP for independent inputs) detectors remain independent, but can vary with the specific value of $a = \alpha$, possibly making the receiver adaptive.

Chapter 2 addresses codes that aggregate many symbols from the same constellation into larger symbols or codewords, where these codes attempt to reduce P_e, or more generally P_{out}, by introducing dependency between input symbols over a codeword length in a way that the receiver can recover lost symbols caused by poor a-parameter-value channels.

1.6.2 Statistically Variable AWGNs

The statistically parameterized AWGN follows the basic relation

$$y = \underbrace{h}_{\text{gain}} \cdot \underbrace{x}_{\text{input}} + \underbrace{n}_{\text{AWGN}} \quad , \tag{1.507}$$

so the parameter h represents an attenutation relative to the nominal signal level. More generally $y \to \boldsymbol{y}$ and $x \to \boldsymbol{x}$ to which Section 1.5's linear model applies and $h \to H$, a matrix of random parameters. Here, H will be considered diagonal (or diagonalized) and so each dimension may have random gain selected from some probability distribution/density p_h. Equation (1.507) uses no dimensional indexing because all dimensions: time, frequency, and space will follow this same model, but each with a possibly different random sample of the **channel transfer amplitude** h and noise. The a distribution can be different for different dimensions, as in Subsections 1.6.3.2 and 1.6.3.3. This random gain h is statistically characterized by its probability distribution. An important derived quantity, now consequently random also, is the **noise-referenced channel gain**

$$g = \frac{|a|^2}{\sigma^2} \quad , \tag{1.508}$$

where σ^2 is the constant WGN noise variance. A channel-output random SNR including the gain is $\gamma \triangleq \bar{\mathcal{E}}_{\boldsymbol{x}} \cdot g$. Most designers would recognize that the noise in the denominator of Equation (1.508) is only important in its level relative to the nominal input energy $\bar{\mathcal{E}}_{\boldsymbol{x}}$ so that g is the important quantity.

Outages correspond to sufficiently low g values. Gain g values less than a certain desired level are **fades**; equivalently the channel becomes a **fading channel** when there is a significant probability of undesirable g values. Subsection 1.6.3 addresses the speed of such variation.

For the ergodic channel, the ergodic-average error probability $\langle P_e \rangle$ for random fading as

$$< P_e >= \int_{v=0}^{\infty} P_e(v) \cdot p_g(v) \cdot dv \quad . \tag{1.509}$$

Similarly an **ergodic-average bit rate** is

$$< \bar{b} >= \int_{u=0}^{\infty} \frac{1}{2} \cdot \log_2 \left(1 + \frac{\mathcal{E}(v) \cdot v}{\Gamma} \right) \cdot p_g(v) \cdot dv \quad , \tag{1.510}$$

and the outage probability, as a function of channel-gain threshold g_0, is

$$P_{out}(g_0) = \int_0^{g_0} p_g(v) \cdot dv \quad . \tag{1.511}$$

1.6.2.1 Some channel-gain distributions/densities

The words "distribution" and "density are interchangeable in the sequel, consistent with the general literature; analysis uses sums or integrals correspondingly in computing averages or other statistical quantities.

Scattering refers to reflection and refraction of electromagnetic waves. There are loosely three scattering levels/effects that characterize a random g value with the AWGN according to size relative to the carrier's wavelength:

Definition 1.6.5 [Scattering]

Scattering *refers to the effects of various reflections and refractions of a electromagnetic wave from a transmitter to a receiver.*

microscopic scattering Microscopic scattering occurs at geometries well below the carrier-frequency wavelength, usually caused by atmospheric molecules that the radio wave polarizes and causes to oscillate, which in turn creates random channel-attenuation contributions. The channel transfer amplitude h multiplies the channel input, but represents the superposition of these many contributions to the radio path. The central limit theorem is presumed to apply to the sum of contributions, rendering both of the orthogonal inphase h_I and quadrature h_Q channel transfer attenutations Gaussian. However, the relative delays along the path are roughly the same, but for any given sample path could correspond to any random phase angle $\theta \in [0, 2\pi)$. Thus a reasonable model at this microscopic level is that the random amplitude of the scattered path gain is $\sqrt{a_I^2 + a_Q^2}$, a Rayleigh distribution (see below). A **line-of-sight path (LOS)** may have a large mean amplitude in the scattered path, which leads to a Ricean distribution. The following paragraphs discusses both. Microscopic scattering also has the name "flat fading."

multi-path scattering Multi-path scattering superimposes microscopic scattering on different paths that have significant delay separation (relative to the symbol period), meaning path reflections occur from larger objects (roughly wavelength size or larger). Each such path, of these multiple paths (or multi-path), essentially has Raleigh statistics centered at times reflected by a **power-delay profile.** Multi-path fading also has the name "frequency-selective fading," which Subsection 1.6.3.2 discusses further.

macroscopic scattering Macroscopic scattering occurs as radio waves refract around and/or pass through large objects (people, appliances, homes, trucks, walls, mountains) or reflect from them. In this case the geometries are much longer than a wavelength. There is a mean value of the attenuation along a line-of-sight path that is known as the **path loss**, and also a variance cause by refraction and reflection that is known as shadow fading that models as a product of many path segments along the transmission path. Taking the log of the products of each of the macro section's attenuations is a sum of logarithmic gains, which tend to model again by the central limit theorem as Gaussian, leading to the **Log Normal Distribution** in the following paragraphs, whose mean is the path loss.

Rayleigh Distribution: The Rayleigh[59] Distribution (see Problem 1.62) has itself a parameter that is the expected value of $|h|^2 = h_I^2 + h_Q^2$, thus $\mathcal{E}_h \triangleq \mathbb{E}\left[|h|^2\right]$ (if normalized to real dimensions, then $\overline{\mathcal{E}}_h = \mathcal{E}_h/2$):

$$p_h(u) = \frac{u}{\overline{\mathcal{E}}_h} \cdot e^{-\frac{u^2}{2 \cdot \overline{\mathcal{E}}_h}} \quad 0 \leq u < \infty \; . \tag{1.512}$$

The Rayleigh distribution appears on the left in Figure 1.89 with $\mathcal{E}_h = 1$ (so that $\overline{\mathcal{E}}_h = 0.5$). The peak occurs at $1/\sqrt{\overline{\mathcal{E}}_h}$ generally, so showing at .707 in Figure 1.89. Values to the left of the peak correspond to more difficult channels with higher random-error probability.

[59] After Lord Rayleigh, a British Physicist living from 1842-1919 who studied the scattering of electromagnetic waves in the atmosphere.

χ-Square Distribution: The Rayleigh-derived channel gain g also has a well-known probability distribution, the χ^2**-square** distribution with 2 degrees of freedom, also known as an **exponential distribution**. This χ^2 random variable has probability density

$$p_g(v) = \frac{1}{\bar{\mathcal{E}}_g} \cdot e^{-\frac{v}{\bar{\mathcal{E}}_g}} \quad . \tag{1.513}$$

The average noise-normalized gain is

$$\overline{\mathbb{E}[g]} = \bar{\mathcal{E}}_g = \bar{\mathcal{E}}_h/\sigma^2 \; . \tag{1.514}$$

Figure 1.89 also plots the χ-Square for the same unit value of $\mathcal{E}_h = 1 = \mathcal{E}_g$. Small values of the amplitude squared appear to have higher probability. Squaring a small value below 1 makes it smaller yet. Thus, the probability distribution shifts lower for the channel SNR parameter g that more directly characterizes performance. Low channel-gain values are now more evident in the exponential/χ-square distribution, and thus evidence that fading can cause significant performance reduction.

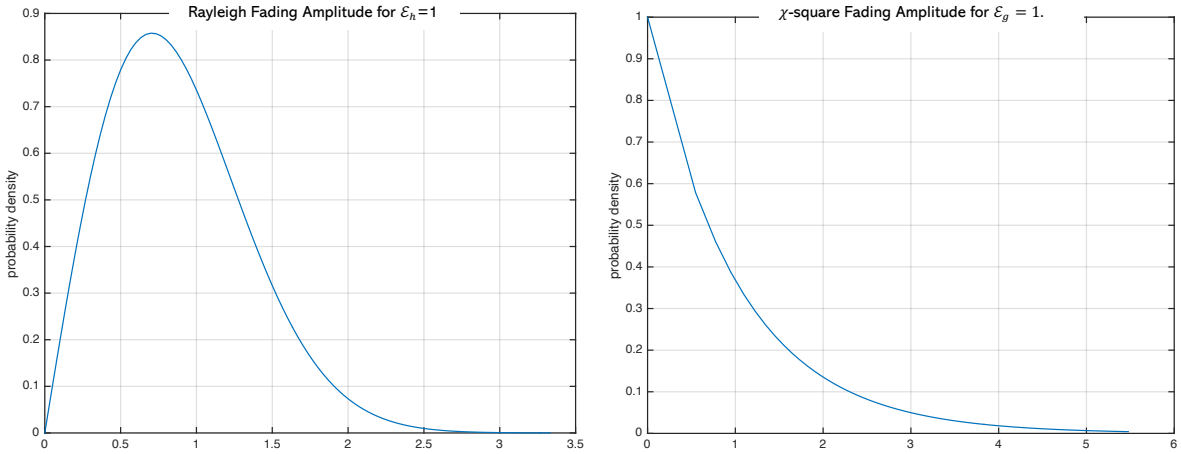


Figure 1.89: **Rayleigh and derived χ-square distributions.**

To complete the characterization of microscopic scattering, the phase has distribution

$$p(\theta) = \begin{cases} \frac{1}{2\pi} & 0 \le \theta < 2\pi \\ 0 & \text{elsewhere} \end{cases} \quad . \tag{1.515}$$

EXAMPLE 1.6.1 [Rayleigh+AWGN average error probability] For instance, the Rayleigh distribution has average $< P_e >$ averaged over the Gaussian noise (normalized through g to unit variance here and $\bar{\mathcal{E}}_{x,g} \triangleq \bar{\mathcal{E}}_x \cdot g$)

$$< P_e > = \int_{g=0}^{\infty} \underbrace{\left[\int_{u=\sqrt{g}}^{\infty} \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \cdot du \right]}_{Q(\sqrt{g})} \cdot \frac{e^{-\frac{g}{\bar{\mathcal{E}}_{x,g}}}}{\bar{\mathcal{E}}_{x,g}} \cdot dg \tag{1.516}$$

Reversing the integrations' order with the consequent limit changes, $< P_e >$ becomes

$$< P_e > = \int_{u=0}^{\infty} \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \left[\int_{g=0}^{u^2} \frac{e^{-\frac{g}{\bar{\mathcal{E}}_{x,g}}}}{\bar{\mathcal{E}}_{x,g}} \cdot dg \right] \cdot du \tag{1.517}$$

$$= \int_{u=0}^{\infty} -\frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \cdot \left[e^{-\frac{g}{\bar{\mathcal{E}}_{x,g}}} \Big|_{g=0}^{g=u^2} \right] \cdot du \tag{1.518}$$

$$= \int_{u=0}^{\infty} -\frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \cdot \left[e^{-\frac{u^2}{\bar{\mathcal{E}}_{x,g}}} - 1 \right] \cdot du \tag{1.519}$$

$$= \frac{1}{2} - \int_{u=0}^{\infty} \frac{e^{-\frac{u^2(1+1/\mathcal{E}_{x,g})}{2}}}{\sqrt{2\pi}} \cdot du \tag{1.520}$$

$$= \frac{1}{2} - \frac{1}{2}\sqrt{\frac{\bar{\mathcal{E}}_{x,g}}{\bar{\mathcal{E}}_{x,g}+1}} \tag{1.521}$$

$$\simeq \frac{1}{4 \cdot \kappa \cdot SNR} \text{ for large } SNR \tag{1.522}$$

(presuming $\bar{\mathcal{E}}_{x,g}$ is the channel-output SNR).

Rayleigh Fading's average error probability $< P_e >$ decays linearly with $\mathcal{E}_{x,g}$ and thus with SNR. This is a much slower decay with SNR than on stationary channels where the decay is exponential with SNR. Thus, (uncoded) Rayleigh fading channels perform worse, on average, than stationary channels at reasonable design levels, and a larger SNR is necessary to attain a certain level of ergodic-average error probability (or equivalently, the data rate is lower). However, coding can offset some of the performance difference as in later chapters. This average error probability is actually for 1 bit in one two-dimensional symbol

For $\sqrt{M}$ level-PAM, so then M-ary QAM SQ, and $M \geq 4$, recognizing that when $M = 4$ there is 3 dB less energy because 2 bits in two dimensions, the expression generalizes to

$$< P_e > = N_e \cdot \left\{ \frac{1}{2} - \frac{1}{2}\sqrt{\frac{\frac{3}{M-1} \cdot \mathcal{E}_{x,g}/2}{\frac{3}{M-1} \cdot (\mathcal{E}_{x,g}/2) + 1}} \right\} \tag{1.523}$$

$$\approx 2(1 - \frac{1}{\sqrt{M}}) \cdot \frac{M-1}{12 \cdot \mathcal{E}_{x,g}/2} = \frac{(\sqrt{M}-1) \cdot (M-1)}{3 \cdot \sqrt{M} \cdot \mathcal{E}_{x,g}} \text{ for large } \mathcal{E}_{x,g} \quad . \tag{1.524}$$

Similar factors could be inserted for other constellations on the energy term. Nearest neighbor counts also similarly apply.

Ricean Distribution: The Ricean[60] distribution adds the line-of-sight path attenuation (meaning its channel amplitude transfer is not random and is a real constant h_0) to the Gaussians in the Rayleigh distribution. The Ricean distribution (which also has random uniform phase in (1.515) presumed with it) is

$$p_h(u) = \frac{1}{\bar{\mathcal{E}}_h} \cdot e^{-\frac{u^2+h_0^2}{2\bar{\mathcal{E}}_h}} \cdot I_0\left(\frac{u \cdot h_0}{\bar{\mathcal{E}}_h}\right) \quad , \tag{1.525}$$

where $I_0(x)$ is the modified Bessel function of the first kind with order zero[61]. Of more interest in this text's approach is the related offset-χ-square distribution for the channel gain g:

$$p_g(v) = \frac{1}{\mathcal{E}_g} \cdot e^{-\frac{v+g_0}{\mathcal{E}_g}} \cdot I_0\left(\frac{\sqrt{2v}}{\mathcal{E}_g}\right) \quad . \tag{1.526}$$

[60] After Stephen O. Rice, liiving 1907-1986, an American electrical engineering professor at Columbia University.

[61] The modified Bessel function of the first kind with order zero is

$$I_0(x) = \sum_{m=0}^{\infty} \left(\frac{1}{m!}\right)^{-2} \cdot \left(\frac{x}{2}\right)^{2m} \quad .$$

Log-Normal Distribution: The **log normal** distribution that characterizes the path loss μ_h (in dB) and the shadow fading σ_h^2 has the LOS average attenuation parameter h_0 becoming a random variable itself according to

$$h_0 = e^{\mu_h + \sigma_h \cdot Z} \quad , \tag{1.527}$$

where Z is zero-mean Gaussian with unit variance. There are really two averages in the overall scattering-channel model: Macro on h_0, which then becomes a constant for the micro Ricean fading. The logarithms $\ln(h_0) = \frac{1}{2}\ln(h_0)$ then clearly are Gaussian with mean μ_h and variance σ_h^2. Conversely if h_0's mean is m_{h_0} and variance is $\sigma_{h_0}^2$ (without the log) , then

$$\mu_h = \ln\left(\frac{m_{h_0}}{\sqrt{1+\frac{\sigma_{h_0}^2}{m_{h_0}^2}}}\right) \text{ and} \tag{1.528}$$

$$\sigma_h^2 = \ln\left(1+\frac{\sigma_{h_0}^2}{m_{h_0}^2}\right) . \tag{1.529}$$

Typically μ_Z and σ_Z are specified in dB, presuming that ln is replaced by $20\log_{10}$ for h_0 or $10\log_{10}$ for g_0 in (1.528) and (1.529).

rich scattering: A multitude of multi-path scattered paths or rays chatracterizes **rich scattering**. Rich scattering has many rays that converge to any spatial point where a receiver's antenna might be located. A MIMO (Vector OFDM) design may be able to exploit these multiple paths (Section 1.5's channel matrix H, which will have many non-zero singular values with rich scattering).

1.6.2.2 Outage Probability

The AWGN outage probability becomes channel gain's probability is below a certain threshold call it g_{out}:

$$P_{out} \triangleq Pr\{g < g_{out}\} \quad . \tag{1.530}$$

The outage probability can be associated for reasonable monotonic fading models with probability density $p_g(v)$ as

$$P_{out} = \int_{v=0}^{g_{out}} p_g(v) \cdot dv \quad . \tag{1.531}$$

In all but trivial cases, the integral in (1.532) will need numerical evaluation. However, its form allows also the simple upper bound **for the Rayleigh case when** $g_{out} < \infty$:

$$\underbrace{P_{out} < \langle P_e \rangle}_{\text{always}} < \underbrace{\frac{1}{4 \cdot \mathcal{E}_g}}_{\text{Rayleigh}} \quad . \tag{1.532}$$

The Q-function density largely concentrates at low channel-gain values, and the design could target low $< P_e >$ that ensures good outage probability as well. Clearly forcing a low ergodic-average error probability will also force a better outage probability. When the "tails" of the outage (small values) are more probable, actual outage probability can dominate $\angle P_e\rangle$ in design consideration. Also, the outage probability will help design C-OFDM loading strategies in Chapter 4. Basically, P_{out} also worst-case reduces only linearly with SNR for Rayleigh fading in the absence of coding over the fading dimensions.

1.6.2.3 The Wireless Designer's Paradox

There are yet more distributions that correspond to different wireless situations. Ergodic analysis essentially finds $\langle P_e \rangle$ and P_{out} over these distributions on g and g_{out}. This may require numerical integration, or perhaps the designer simply samples the distributions, evaluates performance for each sample value,

and then averages (or computes outage probability by numerical integration for each sample) based on generation of samples from a random number generator. This is sometimes called **Monte Carlo** simulation. The flaw in such analytical or simulated performance projection is the distribution assumption/accuracy. Thus, a designer can ergodicly analyze the wireless system, but results depend upon the distribution assumption. When wireless systems do not meet designer's expectations, indeed this distribution assumption needs further validation or investigation.

A better approach may well instead measure g values (or the implied h or H for a MIMO matrix of h values) across channel use. A computer then processes the values into an estimated distribution $\hat{p}_g$ that matches the data. This distribution then becomes the p_g that can be used for subsequent analysis. Good designs then adjust transmit data rates accordingly so that reliability targets are met. Chapter 7 revisits this rich topic, both theoretically and practically.

1.6.3 Channels with Parameter Memory

Chapters 3 - 5 address channels with memory of previous symbols; such channels are not memoryless. This subsection instead addresses channels with **parameter memory**. Parameter memory broadly means that the current channel model uses an a value in some dimension that depends an a value in other dimensions.

One AWGN-channel parameter memory measure is the **coherence** between "adjacent" channel uses' g values.

Definition 1.6.6 [Coherence] Coherence *measures parameter memory according to dimension type:*

coherence time , T_Δ, measures the time over which adjacent channel-output time-indexed dimensions' (sample) gains may be significantly correlated at the same frequency and location. Coherence time is of particular interest in flat fading, although not limited to such. It typically means either (both) of transmitter and receiver are moving, which means the receiver's local clock appears in error.

coherence bandwidth , W_Δ, measures the frequency bandwidth over which adjacent channel-output (Modulation formats that use many parallel carriers appear in Chapters 2 - 5.) frequency-indexed dimensions' gains may be significantly correlated at the same time and location. Coherence bandwidth is of particular interest in frequency-selective fading, although not limited to such.

spatial coherence , Λ_Δ, measures the correlation between various spatial dimensions' (likely antennas') gains at the same time and frequency. Typically the spatial coherence is related to the ratio of the used wavelengths to the spacing between antennas[62]. It can be spatially indexed, although such indexing has meaning only if the antenna-array geometry has an order reflecting adjacent channels' spatial position. Spatial coherence can be of interest in shadow fading, but not limited to such.

Coherence's implied correlation also allows interpolation or prediction of adjacent dimensions' subchannel gains. Previous sections' stationary channels (with deterministic parameters) would have complete coherence because different channel uses experience the same parameter value.

1.6.3.1 Coherence Time

Coherence time's most meaningful interpretation is relative to the symbol period. If the coherence is less than a symbol time, $T_\Delta < T$, then a receiver re-estimates the parameter g each message period. Such rapid variation reduces transmission performance because the receiver must accommodate error in estimating g. By contrast when $T_\Delta >> T$, it is possible to estimate the random parameter g more

accurately. Coherence time is thus important in tracking the time-variation of g. Coherence often uses the measure correlation coefficient between two jointly random vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ as:

$$\rho_{\boldsymbol{x},\boldsymbol{y}} \triangleq \frac{\mathbb{E}\left[\boldsymbol{x}^*\boldsymbol{y}\right]}{\sigma_x \cdot \sigma_y} \quad . \tag{1.533}$$

EXAMPLE 1.6.2 [Mobile Wireless Coherence-Time Example] Typical mobile cellular use might occur in a vehicle moving at $v = 30$ m/s (roughly 100 kph or 60 mph) with passband modulated QAM signals roughly at carrier $f_c = 2$ GHz. Such a vehicle speed maximally offsets the carrier frequency if the vehicle motion is colinear with a line between antenna and vehicle. Basic geometric computes that maximum **Doppler Frequency** carrier offset $f_d = \left(\frac{v}{c_0}\right) \cdot f_c = 200$ Hz, where $c_0 = 3 \times 10^8$ m/s is the speed of light.

A rough correlation-coefficient estimate between signals at two moving-vehicle sample points being 1/2 maximum value is when the sinusoidal Doppler-frequency phase shift is at 45 degrees (or cos = sin = $\frac{1}{\sqrt{2}}$). This is 1/8 of a Doppler cycle or would then estimate $T_\Delta \cdot f_d = 0.125$ for QPSK signals to have correlation already reduced by this factor 1/2 with respect to a stationary maximum. Far more sophisticated analysis by Rappaport [6] (pp. 165-166) evaluates various fading models and effects with many assumptions and approximations to find this same correlation-coefficient level to occur at $T_\Delta \cdot f_d = 0.179$. This example uses of more conservative model with 0.125.

More generally, this equation should tighten by 6 dB for each additional QAM-constellation bit in $\bar{b}$, so a refined model is

$$T_\Delta \leq \frac{0.125}{f_d \cdot 2^{2(\bar{b}-1)}} = 2^{2(1-\bar{b})} \cdot [6.25 \text{ ms}] \quad . \tag{1.534}$$

For $\bar{b} = 1$, this is then $1/T_\Delta = 1600$ Hz, so then $1/T >> 1600$ Hz. Example cellular systems have symbol rates of 12 kHz or 15 kHz (see Chapter 4) and easily satisfy (1.534) up to $\bar{b} = 2$ (16 QAM). If the speed lowers by 1/2 to 50 kph (or 30 mph), then 64QAM probably is within limit (but close). Stationary or walking speed allows large $\bar{b}$ from a doppler perspective, namely 256, 1024, and 4096 QAM.

If vehicle speed increases to 300 kph (fast train perhaps) and the carrier frequency were to increase to 60 GHz, then this coherence time's inverse increases to roughly 16 kHz. Cellular systems with such high carrier frequencies have symbol rates of 120 kHz (instead of 12 or 15 kHz used at lower carrier frequencies), basically to ensure reasonable coherence time.

Example 1.6.2 illustrates what might initially be counter-intuitive: more rapid-variation-channel leads to designs that transmit "faster." In effect, faster message sending[63] means a fixed coherence time appears relatively small. The receiver's estimation of g can occur over more symbol periods, thus increasing its accuracy, and consequently also then improving the accuracy of ML detectors that are a function of g. This basic result drove wireless transmission's progression through generations 1G, 2G, 3G, 4G, 5G, and beyond to wider and wider bandwidths, making relative time variation appear less rapid. This of course requires the wider bandwidth be possible (or allowed by regulated spectrum bands/channels)[64]. The wider bandwidth per user then requires a variety of orthogonal basis function

[63] Faster message sending uses a smaller symbol period T and thus a higher $W = 1/T$ that often specifies the corresponding modulation basis functions' bandwidth.

[64] Indeed, 3G systems increased bandwidth substantially by almost an order of magnitude over earlier systems and met with heavy sarcasm from an embedded wireless interest that opposed Qualcomm's (same Irwin Jacobs on the first reference in this chapter and his co-founder Andrew Viterbi) "wideband CDMA" that eventually was indeed accepted after years of standards' politics.

sets that occupy the same bandwidth with different simultaneous users, which is feasible in many ways as future chapters indicate.

1.6.3.2 Coherence Bandwidth

The coherence bandwidth depends on channel-filtering's dispersion as mathematically represented in Subsection 1.3.7.1's filter $h(t)$ (see also Figure 1.67). The filtered channel exhibits a delay spread between fastest and slowest channel frequencies. The filtered channel has Fourier Transform $H(f) = |H(f)|e^{\jmath\theta(f)}$, where the delay at any frequency f is $\tau(f) = -\frac{d\theta_h}{d\theta}$. The delay $\tau(f)$ will often have a constant linear **group-delay** component τ_0 and a **phase-delay** variation $\tau(f)$ about that linear component, so $\theta(f) \approx 2\pi\tau_0 \cdot f - \int \tau(f) \cdot df$. The group delay simply corresponds to a causal time delay in transmission of τ_0, so that $h(t-\tau_0) \rightarrow H(f) \cdot e^{2\pi\jmath\tau_0 \cdot f} \cdot e^{-2\pi\jmath \int \tau(f) \cdot df}$, and the middle term does not lead to dispersion. However, the last component with $\tau(f)$ does produce dispersion or "spread." Typically this **root mean-square delay-spread** value follows directly in the time domain by using the squared magnitude $|h(t)|^2 \cdot dt$ like a time-indexed probability density so that

$$\tau_{rms} = \sqrt{\frac{\int_{-\infty}^{\infty} (t-\tau_0)^2 \cdot |h(t)|^2 \cdot dt}{\int_{-\infty}^{\infty} |h(t)|^2 \cdot dt}} \quad . \tag{1.535}$$

For the random case, $h(t)$ can be viewed as the sum $\varphi(t) * \sum_k \sqrt{g_k} \cdot \delta(t-\tau_k)$ where $\varphi(t)$ is a deterministic normalized transmit basis function, and g_k is the random channel-gain variable associated with various channel paths each delayed by τ_k. This is the multi-path situation where each random gain g_k is independent of $g_{j\neq k}$. The expected square is then the power-delay profile that reflects the average power received from an input impulse at each time t:

$$P_h(t) = \sum_k \mathcal{E}_{g,k} \cdot \varphi^2(t) \quad , \tag{1.536}$$

which then replaces $|h(t)|^2$ in Equation (1.535). The coherence bandwidth is then

$$W_\Delta = \frac{1}{\tau_{rms}} \quad . \tag{1.537}$$

The coherence bandwidth, unlike the coherence time, finds use in two ways: First, with a constant QAM constellation's use on all frequency dimensions (see Chapter 4 on loading for variable constellation's use), then the coherence bandwidth estimates the number of such frequency dimensions or tones that have similar gain. A design that uses the constellation on a bandwidth wider than the coherence bandwidth will then need either to use a worst-case constellation or to apply a code with sufficient redundancy to recover those dimensions where error probability is higher because the random gain is too low for reliable transmission on that dimension alone. The designs of Chapters 4 using codes as in Chapter 2 can neutralize coherence bandwidth to have less impact on performance. The coherence bandwidth's second use allows design of the frequency spacing between known embedded (non-message) pilot training sequences in wireless transmission, as in Chapter 7, where the channel gains parameters between known pilots' measured values are interpolated.

EXAMPLE 1.6.3 [Indoor Wireless Channel Model] An indoor wireless channel for a home or office has model (with $\frac{\mathcal{N}_0}{2} = 1$ assumed for the additive noise; and also u_k as a zero-mean, unit-variance, white Gaussian complex random process that is independent of noise and signal)

$$h = \sum_k h_k \tag{1.538}$$

$$h_k = 10^{-L(d)/20} \cdot \sqrt{P_{h,k}} \cdot \left[\underbrace{\sqrt{\frac{K}{K+1}}}_{LOS} \cdot \delta_k + \underbrace{\sqrt{\frac{1}{K \cdot \delta_k + 1}}}_{Rayleigh} \cdot u_k \right] \quad , \tag{1.539}$$

where $\delta_k = 1$ if $k = 0$ and $\delta_k = 0\ \forall k \in \mathbb{Z} \setminus 0$ and where u_k is a unit-variance zero- mean Gaussian random variable. The power-delay profile coefficients for two clusters of energy are:

k	delay (ns)	cluster 1 $P_{1,h,k}$ (dB)	cluster 2 (dB) $P_{2,h,k}$ (dB)
0	0	0	–
1	10	-5.4	-
2	20	-10.8	-3.2
3	30	-16.2	-6.3
4	40	-21.7	-9.4
5	50	–	-12.5
6	60	–	-15.6
7	70	–	-18.7
8	80	–	-21.8

The parameter value of $K = 1$ usually applies for smaller homes, but can be as large as $K = 4$ for larger homes. It is active only for the LOS path, so Ricean fading occurs in Cluster 1's first power-delay profile term and Rayleigh fading in the remaining terms and all Cluster 2 terms. When $K = 1$, half the energy is in the LOS path and the other half in the NLOS path. Large K thus implies a longer path where the increased number of reflections and diffractions cause the LOS energy fraction to be relatively greater. For the longer reflective ray paths, there is no LOS and so $K = 0$ implies all is Rayleigh fading. These paths correspond to the nonzero-relative-delay paths in the power-delay profile. Standards groups may define more clusters for larger residences and a variety of other effects may be included. All these are relative to some nominal level.

That nominal level is the fixed LOS path loss, along with the random shadowing loss, which is the log-normal term

$$L(d) \quad = L_{path}(d) + L_{shadow}(d) \qquad \text{dB} \quad d \le d_{bp} \tag{1.540}$$

$$= L_{path}(d_{bp}) + L_{shadow}(d_{bp}) + 35 \log_{10}\left(\frac{d}{d_{bp}}\right) \text{dB} \quad d > d_{bp} \tag{1.541}$$

where the break-point distance is $d_{bp} = 5$m for smaller homes and

$$L_{path} = 20 \cdot \log_{10}(d) + 20 \cdot \log_{10}(f) - 147.5 \text{ dB} \tag{1.542}$$

and L_{shadow} is a random log-normal variable selected from (in dB)

$$\frac{1}{\sqrt{2\pi\sigma_Z^2}} \cdot e^{-[L_{shadow}(d)]^2/(2\sigma_Z^2)} \tag{1.543}$$

with $\sigma_Z = 3$ dB before the breakpoint and σ_Z=4 dB after the break point.

This model simplifies the IEEE 802.11 committee's Model B. There are other models (C-F) with more clusters in the power-delay-profile, larger break-point distances, etc. Chapter 4 investigates further modulation design for these channel types.

1.6.3.3 Spatial Coherence

The increase of dimensionality by smaller T and thus wider bandwidth magnifies through use of multiple antenna MIMO methods that use modulated signals across space. MIMO allows multiple spatial-dimensional uses within a given symbol period. Each receiver antenna may have its own Rayleigh distribution when the inter-antenna distance exceeds half the carrier wavelength. Thus, the a value has best MIMO meaning as a matrix $A = H$ with $L_y \times L_x$ random entries. Each of these entries may follow models like Example 1.6.3. These entries often have a fixed gain and phase relationship over time at any particular frequency. There is crosstalk between the spatial dimensions, and so Section 1.5's MIMO processing may be necessary for each such symbol. Coherence time measures the crosstalk matrix' change

over time (at any frequency), while the coherence bandwidth measures the crosstalk matrix' change over frequency. Spatial coherence roughly measures if the input has excited only one singular mode of H, sometimes known as "single-mode" transmission. Less coherence means more modes carry energy. As Chapter 4 and beyond show with "vector coding" methods, less coherence is typically good and is equivalent to the "rich scattering" mentioned earlier.

1.6.3.4 Discretization of probability densities

One coherence model uses Markov-state machines with discrete a values within a finite set $\alpha \in \mathscr{A}$ with $|\mathscr{A}|$. If a has index k in some order (like time) then a_k's choice depends on the previous α_{k-1} value. There are $|\mathscr{A}|$ states for each time index k.

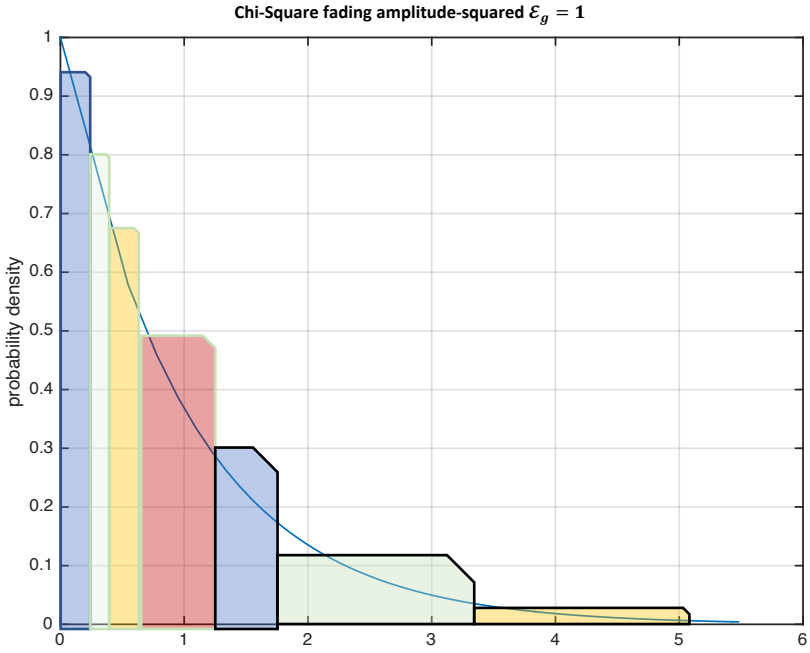


Figure 1.90: **Sampled χ-square distributions.**

As an example for the AWGN with distribution on the gain g, the continuous probability density function $p_g(v)$ can always be sampled and approximated by a discrete probability distribution where

$$p_g(v) \cdot dv \rightarrow p_{g,k} \quad , \tag{1.544}$$

where the sampled distribution is

$$p_{g,k} = \frac{p_g(k \cdot \Delta) \cdot \Delta}{\sum_k p_g(k \cdot \Delta) \cdot \Delta} \quad , \tag{1.545}$$

which is a sampling of the probability density at regular intervals Δ with appropriate normalization. As $\Delta \rightarrow dv$, (1.545)'s normalization becomes unnecessary. It is possible to apply sampling theory to the rate $1/\Delta$ being greater than twice the highest frequency of the probability density function's non-zero-amplitude Fourier Transform, but that is unnecessary since all channel-SNR probability distributions are going to be an approximation in any case. Figure 1.90 shows the χ-square distribution possibly sampled, noting specifically that more fine resolution may be wanted at the low values since these low values cause errors. When the channel is ergodic, the distribution can be learned by binning[65] channel SNR estimates.

[65] Binning counts the number of observed values in a discrete set of ranges and divides by the total number of observations to estimate the discrete probability distribution.

The discrete distribution $p_{g,k}$ can be simplified notationally to just $p_g(\alpha)$ where $\alpha \in \mathscr{A}$ where $\mathscr{A}$ is the correspond finite set of elements (ranges of probability for which a single probability p_g for the value g is assigned - g can be viewed as the center point of the range). The size of the set is $|\mathscr{A}|$. Values of g for which the probability is zero should not be included in the set. Such a zero value of g is useless for transmission, so need not be considered.

1.6.3.5 The Ergodic Markov model

The discrete (special case of a) Markov model has an $|\mathscr{A}| \times |\mathscr{A}|$ probability-transition matrix with entries $p_{k/j}$ representing the probability that the next sample value α_k will occur when the current/last value is α_j.

$$P_a = \begin{bmatrix} p_{|\mathscr{A}|-1/|\mathscr{A}|-2} & p_{|\mathscr{A}|-1/|\mathscr{A}|-3} & \cdots & p_{|\mathscr{A}|-1/0} \\ \vdots & \vdots & \ddots & \vdots \\ p_{0/|\mathscr{A}|-2} & p_{1/|\mathscr{A}|-3} & \cdots & p_{0/0} \end{bmatrix} \; . \; . \tag{1.546}$$

The α values at the next channel use follow the possible $|\mathscr{A}|$ state values of the current channel use, as in Figure 1.91. Often channel uses order in time, but of course order could also most generally include any path though sets of used dimensions in space, time, and/or frequency. Every non-zero conditional-probability entry in P_a corresponds to a transition possibility in Figure 1.91's directed graph of the state machine. Zero probability entries have no transition shown. Effectively, the randomness of g is somewhat constrained by the model's "coherence" in that values of a that are close together may have therefore probabilities associated with their transitions that are higher. Large a deviation from message group to message group may have small transition-probability values (or zero, which means no transition).

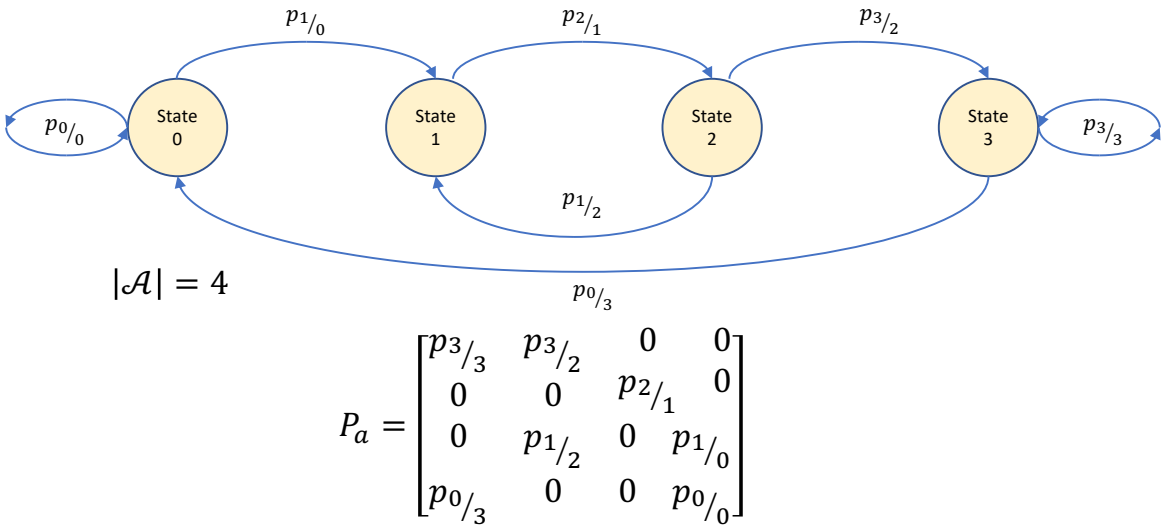


Figure 1.91: **Markov State-Machine Model, with non-zero transition probabilities shown.**

If at any use index k, the $|\mathscr{A}|$ values of the probability distribution have vector representation $\boldsymbol{p}_{a,k}$, then

$$\boldsymbol{p}_{a,k+1} = P_a \cdot \boldsymbol{p}_{a,k} \tag{1.547}$$

Similarly the distribution at time $k + L$ would be

$$\boldsymbol{p}_{a,k+L} = [P_a]^L \cdot \boldsymbol{p}_{a,k} \quad , \tag{1.548}$$

so the coherence over L uses could be thus determined (or effectively a new Markov model state machine with a new transition matrix $[P_a]^L$ now exists). The matrix P_a transforms one probability distribution into another, and its columns are themselves probability distributions, so in effect the input probability distribution weights the combination of column probability distributions to produce a new output distribution. In order for the state machine to avoid (colloquially) "getting stuck" in what mathematicians call "absorbing states," there must be a non-zero-probability path from any state to any other state in a finite number of uses. When this happens, the model is an **ergodic Markov** channel. Such a condition allows the famous Perron-Frobenious Theorem[7] on positive matrices, which simplifies in this case to state that an ergodic Markov channel has a unique **stationary distribution** $\boldsymbol{\pi}$ such that

$$\boldsymbol{\pi} = P_a \cdot \boldsymbol{\pi}.; \quad . \tag{1.549}$$

The distribution $\boldsymbol{\pi}$ is clearly an eigenvector corresponding to an eigenvalue of 1. All other eigvenvalues have magnitudes less than one and effectively have modes that decay to zero in Equation 1.548 as $L \to \infty$. $\boldsymbol{\pi}$ of course should be normalized so that it has unit sum (not unit squared norm) so $\sum_{k=1}^{|\mathscr{A}|} \pi_k = 1$. This ergodic distribution can be used in all previous "angle-bracket" averages. Thus, the channel parametrization leads to an ergodic stationary distribution for the channel, even though the parameters have memory.

Figure 1.92 shows the Gilbert-Elliot channel model. This simple model reflects a "good" state where the ML detector's error probability $P_{e,g}$ is low, and another "bad" state with higher error probability $P_{e,b} > P_{e,g}$. The values for a are thus g and b.

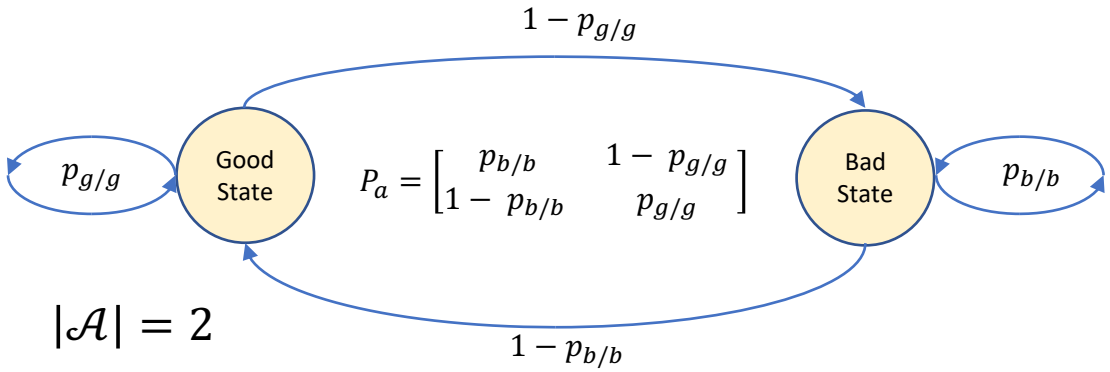


Figure 1.92: **Markov State-Machine Model, with non-zero transition probabilities shown.**

The stationary distribution can easily be found as

$$\boldsymbol{\pi} = \begin{bmatrix} p_{good} \\ - - - \\ p_{bad} \end{bmatrix} = \begin{bmatrix} \frac{1-p_{g/g}}{1-p_{g/g}-p_{b/b}} \\ - - - \\ \frac{1-p_{b/b}}{1-p_{g/g}-p_{b/b}} \end{bmatrix} \quad , \tag{1.550}$$

and thus the ergodic-average error probability is clearly

$$< P_e >= P_{e,g} \cdot p_{good} + P_{e,b} \cdot p_{bad} = \frac{P_{e,g} \cdot (1 - p_{g/g}) + P_{e,b} \cdot (1 - p_{b/b})}{1 - p_{g/g} - p_{b/b}} \quad . \tag{1.551}$$

This simple model would have $P_{out} = p_{bad}$.

1.7 Disguised Channels

Digital transmission's detectors have wide applicability beyond obvious communication-channel problems. This section attempts to illustrate that breadth with some examples that might not be viewed initially as communication problems. Subsection 1.7.1 revisits Example 1.1.3's facial recognition as an initial example, while Subsection 1.7.2 similarly revisits Example 1.1.4's lidar distance measurement. Two other examples in voice recognition and deep-packet inspection (or eavesdropping) also appear in Subsections 1.7.3 and 1.7.4 respectively. Finally Subsection 1.7.5 revisits Example 1.1.5's search engine as an example of use of Ergodic Markov Channels.

1.7.1 Facial Recognition

Figure 1.5 has 8 possible persons, one of whom passed in front of the camera and the others did not. Facial recognition determines which of the 8 did so. The vector $\boldsymbol{y}$ has roughly 36 million dimensions. Theoretically, $p_{\boldsymbol{y}/\boldsymbol{x}}$ has an enormous number of output possibilities if each such dimension might have perhaps a 8-bit quantized light intensity value for each of the colors red, green, and blue. Straightforward implementation of an ML detector would thus be a challenge. This is a two-dimensional recognition system. Adding a third dimension increases further the number of dimensions.

Practical facial-recognition systems will compress the dimensionality and number of intensity values through processing that extracts certain key facial features. Thus, the receiver compresses $\boldsymbol{y}$ to a much smaller N, essentially changing $p_{\boldsymbol{y}/\boldsymbol{x}}$ to reduced dimensionality for $\boldsymbol{y}$. (Of course $\boldsymbol{x}$ usually has $M >> 8$ also.). The compression of $\boldsymbol{y}$ typically includes a set of measurements like:

1. The distance between the center of the pupils, which of course means the pupils' positions themselves might first need to be identified (or even before that whether a face is even present).
2. The width of the mouth, which means the mouth itself needs to be identified.
3. The width of the eyes.
4. The position of the inside eye corner relative to the pupil center.
5. The position of the outside eye corner relative to the pupil center.
6. The skin color and texture.

Facial recognition essentially partitions in to several ML problems. The first is simple face presence (or not). The next set is **feature extraction** that attempts to rotate, maneuver the face to detect positions and types of various facial features as above. The final stage is **classification** that attempts to "classify" the image (who in this case).

This example illustrates an important concept, channel modeling. Communication engineers often spend the largest portion of design time finding ways to model and simplify the channel so that the essential received signal needed to make good decisions occurs. Thus, while $p_{\boldsymbol{y}/\boldsymbol{x}}$ is a nice mathematician/information-theorist's way of modeling the channel, it hides the most important part of most systems – obtaining that model in the first place. Nonetheless, at its core, facial recognition is a basic instance of a communication channel and detection problem, even if disguised at first as apparently something different.

1.7.2 Distance Measurement

Figure 1.6 illustrates the basic distance measurement problem in terms of a self-driving vehicle approaching an object. The delay (which translates into distance by simple geometry) is ML-selected from a set of waveforms corresponding to the delay of a single waveform. This is the principle behind radar and lidar. The channel model here may more simply be the AWGN that well models wireless transmission, perhaps with some of Subsection 1.6's channel-variation modeling also used because of the channel's wireless character.

The message set is a set of M delayed waveforms $x(t-T_i),\ i = 0, ..., M-1$ where T_i are the possible delays. The designer decides the T_i that are of interest. There will be a trade-off with noise if the

inter separations of $T_{min} = \min_{i' \neq i} T_i - T_{i'}$, which on the AWGN will correspond to $d_{\min}$ because a signal-detector implementation is simply a matched filter $X(t-T)$ whose output is sampled at times $t = T - T_i$. The largest is the ML decision and also determines the delay, which corresponds to (twice) the desired distance of the reflecting object from the transmitter.

The astute designer might recognize that a moving transmitter creates a Doppler shift that should be offset from each of the T_i corresponding to that Doppler delay change (which varies with i).

1.7.3 Speaker Recognition

Speaker recognition is similar to facial recognition. The largest challenge will be organizing the base of potential speakers (and/or words/phrases) from which the detector must select. Again, like facial recognition, dimensionality could rapidly render straightforward ML detection too complex. Instead segments of speech, often 10ms in duration each, are transformed to the frequency domain and then quantized according to their spectral content. Different speakers tend to have different such frequency-domain patterns, particularly when viewed as a sequence of transitions between different states from segment to segment. Those frequency-domain patterns and the underlying state-machine itself can be viewed as the dimensions characterizing each speaker with $\boldsymbol{x}_i$. Channel outputs are then transformed, and compared against the possible state machines progressions. The "closest" (which maximizes $p_{\boldsymbol{y}/\boldsymbol{x}}$, assuming all speakers are equally likely) is then the ML estimate of the speaker's indentify.

A similar process can follow for each speaker that matches such transform progressions to possible words/phrases for that speaker (as also learned from training using the speaker's previous speaking of a training document) to determine what they said as well.

1.7.4 Deep-Packet Inspection / Eavesdropping

Deep-Packet Inspection (or DPI) searches a network's stream of packets for certain information that identifies the flowing traffic as corresponding to a specific application type and/or user. A DPI monitors (non-invasively) flows that pass by/through it. These are channel outputs. The DPI system attempts to detect whether certain users' packets or certain applications are present. These applications and users may not always use identical patterns, but there may be consistent patters. Like facial and speaker recognition, the DPI system will decide which such pattern or user (one of the message options will be "not present") is in the data stream. The channel in this case is not the underlying communication channel (which presumably passes packets, although some of those may be in error and thus also contribute to the DPI searches uncertainty in the channel model $p_{\boldsymbol{y}/\boldsymbol{x}}$ for the application and user types.

1.7.5 Search and the PageRank Algorithm as Ergodic Markov Channels

Figure 1.1.5 suggested that a search engine is an example of a MAP detector. A search's first step finds all the web pages that contain the search phrase, which is a finite set indexed by $i = 0, ..., M-1$ here[66]. A search engine then decides which to display first (as well as second, third, fourth, etc.). The web pages' probabilities are not necessarily equal, and their probabilities are unknown from the perspective search phrase (or searcher). What a search-service provider can record and store over all common-search-phrase web sites is the probability that previous searches have seen searchers link to the other pages in the same set, $j \in \{0, ..., M-1\}$. This is essentially the percentage of time that each of the other web pages (or "states" as in Section 1.6.3)'s ergodic Markov channels. Thus, there is a probability-transition matrix with stationary distribution $\boldsymbol{\pi}$.

$$\boldsymbol{\pi}_{searchphrase} = P_{search} \cdot pivec_{searchphrase} \tag{1.552}$$

where P_{search} has the entries $p_{j/i}$ on the linkages' use between the same-search-phrase web pages (or messages). The channel output (or search phrase) now has a decision region that corresponds to all rows of P_{search}, or essentially all indices j correspond to a page with the search phrase on it. The tie is broken

[66] Use of $\boldsymbol{x}_i$ simply uses an alternative naming convention for this set of web pages.

by selecting the one with highest probability in $\boldsymbol{\pi}_{search}$. The second page listed is the second most likely, and so on. This **PageRank Algorithm** was the foundation of search engine's initial success[67]

Modern search engines add many rules that may bias or qualify the matrix P_{search}, but the basic operation of finding the stationary distribution and then ranking the web pages according to the most probable pages in the stationary distribution remains. Unfortunately from a communication engineer's perspective some of the biasing corresponds to those who pay for it, resulting in certain "advertisement" pages appearing first.

[67] It is a play on works in that its' inventor's last name, former Stanford graduate student Lawrence Page, coincidentally is also what many call a web site or "web page." The rank part is of course technical and refers to the stationary distribution's ranking of the pages.

Chapter 1 Excercises

1.1 *Our First Constellation - 10 pts*

a. Show that the following two basis functions are orthonormal. (2 pts)

$$\varphi_1(t) = \begin{cases} \sqrt{2}\,(\cos(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$\varphi_2(t) = \begin{cases} \sqrt{2}\,(\sin(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

b. Consider the following modulated waveforms.

$$x_0(t) = \begin{cases} \sqrt{2}\,(\cos(2\pi t) + \sin(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$x_1(t) = \begin{cases} \sqrt{2}\,(\cos(2\pi t) + 3\sin(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$x_2(t) = \begin{cases} \sqrt{2}\,(3\cos(2\pi t) + \sin(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$x_3(t) = \begin{cases} \sqrt{2}\,(3\cos(2\pi t) + 3\sin(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$x_4(t) = \begin{cases} \sqrt{2}\,(\cos(2\pi t) - \sin(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$x_5(t) = \begin{cases} \sqrt{2}\,(\cos(2\pi t) - 3\sin(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$x_6(t) = \begin{cases} \sqrt{2}\,(3\cos(2\pi t) - \sin(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$x_7(t) = \begin{cases} \sqrt{2}\,(3\cos(2\pi t) - 3\sin(2\pi t)) & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$x_{i+8}(t) = -x_i(t)\ i = 0, \cdots, 7$$

Draw the constellation points for these waveforms using the basis functions of (a). (2 pts)

c. Compute $\mathcal{E}_{\boldsymbol{x}}$ and $\bar{\mathcal{E}}_{\boldsymbol{x}}$ ($\bar{\mathcal{E}}_{\boldsymbol{x}} = \mathcal{E}_{\boldsymbol{x}}/N$) where N is the number of dimensions

(i) for the case where all signals are equally likely. (2 pts)

(ii) for the case where (2 pts)

$$p(x_0) = p(x_4) = p(x_8) = p(x_{12}) = \frac{1}{8}$$

and

$$p(x_i) = \frac{1}{24} \quad i = 1, 2, 3, 5, 6, 7, 9, 10, 11, 13, 14, 15$$

d. Let

$$y_i(t) = x_i(t) + 4\varphi_3(t)$$

where

$$\varphi_3(t) = \begin{cases} 1 & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases}$$

Compute $\mathcal{E}_{\boldsymbol{y}}$ for the case where all signals are equally likely. (2 pts)

1.2 *Inner Products - 10 pts*

Consider the following signals:

$$x_0(t) = \begin{cases} \frac{2}{\sqrt{T}}\cos(\frac{2\pi t}{T} + \frac{\pi}{6}) & \text{if } t\epsilon\,[0,T] \\ 0 & \text{otherwise} \end{cases}$$

$$x_1(t) = \begin{cases} \frac{2}{\sqrt{T}}\cos(\frac{2\pi t}{T} + \frac{5\pi}{6}) & \text{if } t\epsilon\,[0,T] \\ 0 & \text{otherwise} \end{cases}$$

$$x_2(t) = \begin{cases} \frac{2}{\sqrt{T}}\cos(\frac{2\pi t}{T} + \frac{3\pi}{2}) & \text{if } t\epsilon\,[0,T] \\ 0 & \text{otherwise} \end{cases}$$

a. Find a set of orthonormal basis functions for this signal set. Show that they are orthonormal. *Hint:* Use the identity for $\cos(a+b) = \cos(a)\cos(b) - \sin(a)\sin(b)$. (4 pts)

b. Find the data symbols corresponding to the signals above for the basis functions you found in (a). (3 pts)

c. Find the following inner products: (3 pts)

(i) $< x_0(t), x_0(t) >$

(ii) $< x_0(t), x_1(t) >$

(iii) $< x_0(t), x_2(t) >$

1.3 *Multiple sets of basis functions - 5 pts*

Consider the following two orthonormal basis functions:

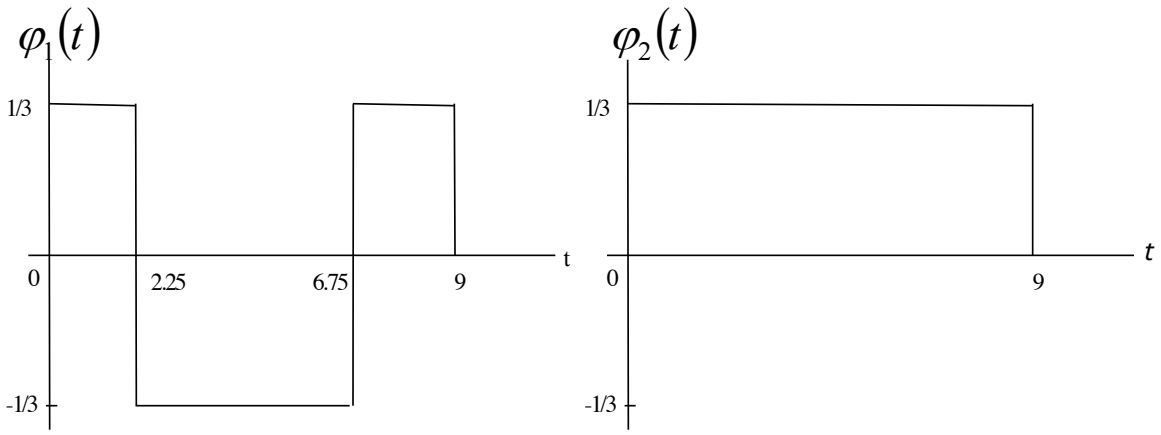


Figure 1.93: Basis functions.

a. Use the basis functions given in Figure 1.93 to find the modulated waveforms $u(t)$ and $v(t)$ given the data symbols $\mathbf{u} = [1\ 1]$ and $\mathbf{v} = [2\ 1]$. It is sufficient to draw $u(t)$ and $v(t)$. (2 pts)

b. For the same $u(t)$ and $v(t)$, a different set of two orthonormal basis functions is employed for which $\mathbf{u} = [\sqrt{2}\ 0]$ produces $u(t)$. Draw the new basis functions and find the $\mathbf{v}$ that produces $v(t)$. (3 pts)

1.4 *Minimal orthonormalization with MATLAB 5 pts*

Each column of the matrix A given below is a data symbol that is used to construct its corresponding modulated waveform from the set of orthonormal basis functions $\{\phi_1(t), \phi_2(t), \ldots, \phi_6(t)\}$. The set of modulated waveforms described by the columns of A can be represented with a smaller number of basis functions.

$$A = [\boldsymbol{a}_0\ \boldsymbol{a}_1\ ...\ \boldsymbol{a}_7] \tag{1.553}$$

$$= \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 3 & 5 & 7 & 9 & 11 & 13 & 15 \\ 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 2 & 0 & 2 & 0 & 2 & 0 & 2 \\ 2 & 4 & 2 & 4 & 2 & 4 & 2 & 4 \end{bmatrix} \tag{1.554}$$

The transmitted signals $a_i(t)$ are represented (with a superscript of * meaning matrix or vector transpose) as

$$a_i(t) = \boldsymbol{a}_i^* \begin{bmatrix} \phi_1(t) \\ \phi_2(t) \\ \vdots \\ \phi_6(t) \end{bmatrix} \tag{1.555}$$

$$A(t) = A^* \boldsymbol{\phi}(t) \quad . \tag{1.556}$$

Thus, each row of $A(t)$ is a possible transmitted signal.

a. Use MATLAB to find an orthonormal basis for the columns of A. Record the matrix of basis vectors. The MATLAB commands **help** and **orth** will be useful. In particular, execution of $Q = \text{orth}(A)$ in matlab produces a 6×3 orthogonal matrix Q such that $Q^*Q = I$ and $A^* = [A^*Q]\,Q^*$. The columns of Q can be thought of as a new basis – thus try writing $A(t)$ and interpreting to get a new set of basis functions and description of the 8 possible transmit waveforms. The Matlab command of **help orth** will give a summary of the **orth** command. To enter the matrix B in matlab (for example) shown below, simply type B=[1 2; 3 4]; (2 pts)

$$B = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \tag{1.557}$$

b. How many basis functions are actually needed to represent our signal set ? What are the new basis functions in terms of $\{\phi_1(t), \phi_2(t), \ldots, \phi_6(t)\}$?(2 pts)

c. Find the new matrix $\hat{A}$ which gives the data symbol representation for the original modulated waveforms using the smaller set of basis functions found in (b). $\hat{A}$ will have 8 columns, one for each data symbol. The number of rows in $\hat{A}$ will be the number of basis functions you found in (b). (1 pts)

1.5 *Decision rules for binary channels - 10 pts*

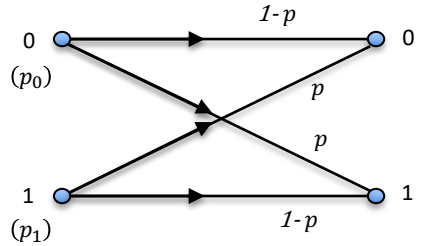


Figure 1.94: Binary Symmetric Channel(BSC).

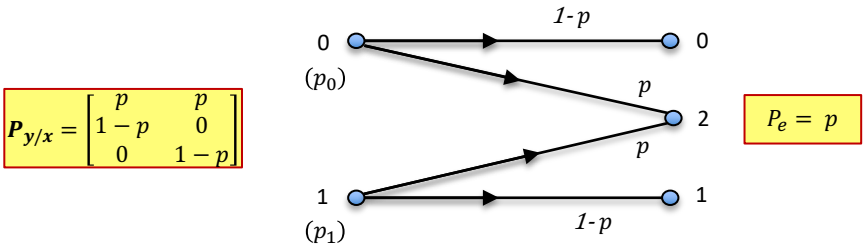


Figure 1.95: Binary Erasure Channel (BEC).

a. Figure 1.94's **Binary Symmetric Channel (BSC)** has binary (0 or 1) inputs and outputs. It outputs each bit correctly with probability $1-p$ and incorrectly with probability p. Assume 0 and 1 are equally likely inputs. State the MAP and ML decision rules for the BSC when $p < \frac{1}{2}$. How are the decision rules different when $p > \frac{1}{2}$? (5 pts)

b. Figure 1.95's **Binary Erasure Channel (BEC)** has binary inputs as with the BSC. However there are three possible outputs. Given an input of 0, the output is 0 with probability $1-p_1$ and 2 with probability p_1. Given an input of 1, the output is 1 with probability $1-p_2$ and 2 with probability p_2. Assume 0 and 1 are equally likely inputs. State the MAP and ML decision rules for the BEC when $p_1 < p_2 < \frac{1}{2}$. How are the decision rules different when $p_2 < p_1 < \frac{1}{2}$? (5 pts)

1.6 *Minimax [Wesel] - 5 pts*

This exercise considers a 1-dimensional vector channel

$$\boldsymbol{y} = \boldsymbol{x} + \mathbf{n} \quad ,$$

where $\boldsymbol{x} = \pm 1$, and $\mathbf{n}$ is Gaussian noise with $\sigma^2 = 1$. The Maximum-Likelihood (ML) Receiver that is minimax, has decision regions:

$$D_{ML,1} = [0, \infty)$$

and

$$D_{ML,-1} = (-\infty, 0)$$

So if $\boldsymbol{y}$ is in $D_{ML,1}$ then an ML receiver decodes $\boldsymbol{y}$ as $+1$; and $\boldsymbol{y}$ in $D_{ML,-1}$ decodes as -1.

This exercise considers another receiver, R, where the decision regions are:

$$D_{R,1} = [\frac{1}{2}, \infty)$$

and

$$D_{R,-1} = (-\infty, \frac{1}{2})$$

a. Find $P_{e,ML}$ and $P_{e,R}$ as a function of $p_{\boldsymbol{x}}(1) = p$ for values of p in the interval $[0, 1]$. On the same graph, plot $P_{e,ML}$ vs. p and $P_{e,R}$ vs. p. (2 pts)

b. Find $\max_p P_{e,ML}$ and $\max_p P_{e,R}$. Are your results consistent with the Minimax Theorem? (2 pts)

c. For what value of p is D_R the MAP decision rule? (1 pt)

Note: For this problem you will need to use the $Q(\cdot)$ function discussed in Appendix B. Here are some relevant values of $Q(\cdot)$.

x	Q(x)
0.5	0.3085
1.0	0.1587
1.5	0.0668

1.7 *Irrelevancy/Decision Regions. (From Wozencraft and Jacobs) - 7 pts*

a. Consider the channel in Figure 1.96 where x, n_1, and n_2 are independent binary random variables. All the additions shown below are modulo two. (Equivalently, the additions may be considered xor's.)

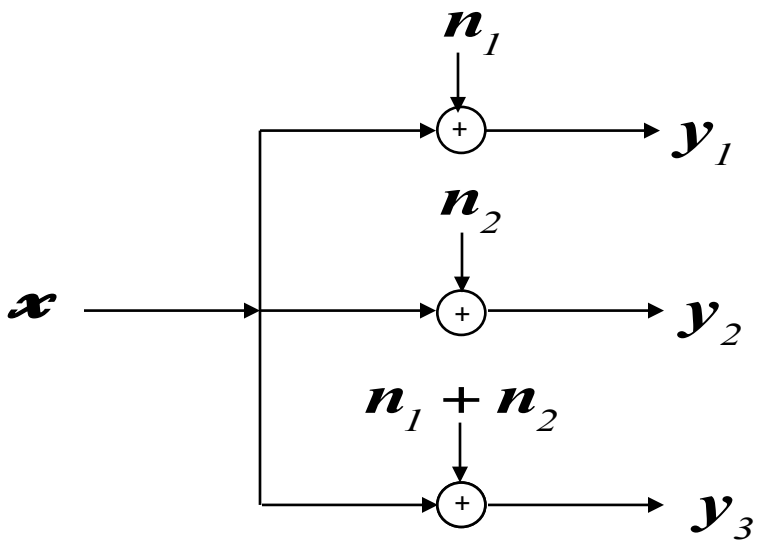


Figure 1.96: 1^{st} Channel for Irrelevancy/Decision Regions.

- Given only y_1, is y_3 relevant? (1 pt)
- Given y_1 and y_2, is y_3 relevant? (1 pt)

For the rest of the problem, consider the second channel in Figure 1.97,

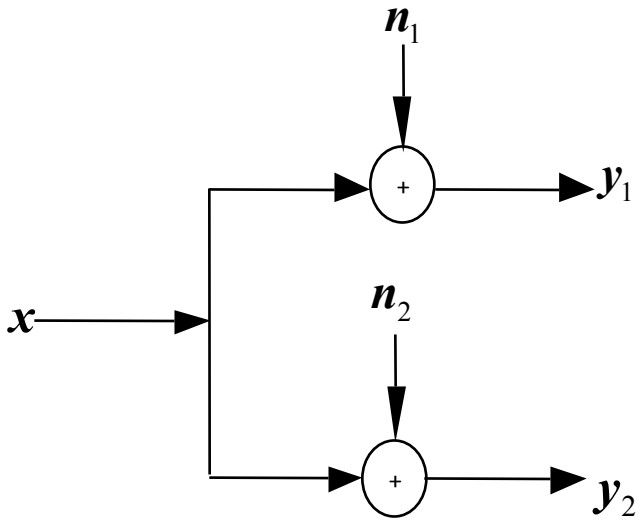


Figure 1.97: 2^{nd} channel for Irrelevancy/Decision Regions

One of the two signals $x_0 = -1$ or $x_1 = 1$ is transmitted over this channel. The noise random variables n_1 and n_2 are statistically independent of the transmitted signal x and of each other.

Their density functions are,

$$p_{n_1}(n) = p_{n_2}(n) = \frac{1}{2}e^{-|n|} \tag{1.558}$$

b. Given y_1 only, is y_2 relevant ? (1 pt)

c. Prove that the optimum decision regions for equally likely messages are shown in Figure 1.98, (3 pts)

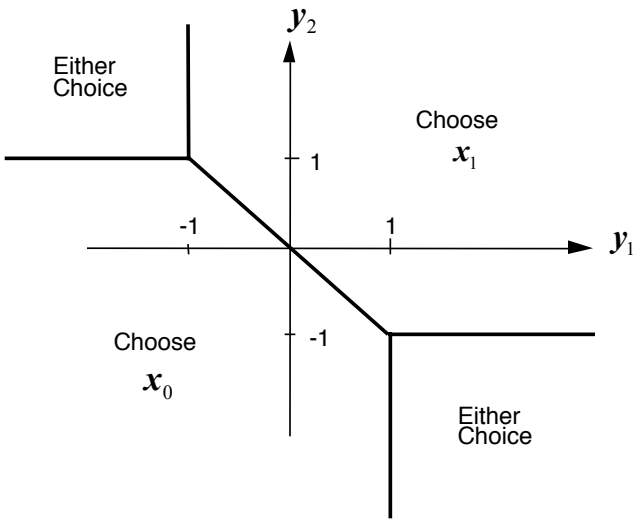


Figure 1.98: regions2

d. A receiver chooses x_1 if and only if $(y_1+y_2) > 0$. Is this receiver optimum for equally likely messages ? What is the probability of error ? (Hint: $P_e = P\{y_1+y_2 > 0/x = -1\} \cdot p_{\boldsymbol{x}}(-1) + P\{y_1+y_2/x = 1\} \cdot p_{\boldsymbol{x}}(1)$ and use symmetry. Recall the probability density function of the sum of 2 random variables is the convolution of their individual probability density functions) (4 pts)

e. Prove that the optimum decision regions are modified as indicated in Figure 1.99 when $Pr\{X = x_1\} > 1/2$. (2 pts)

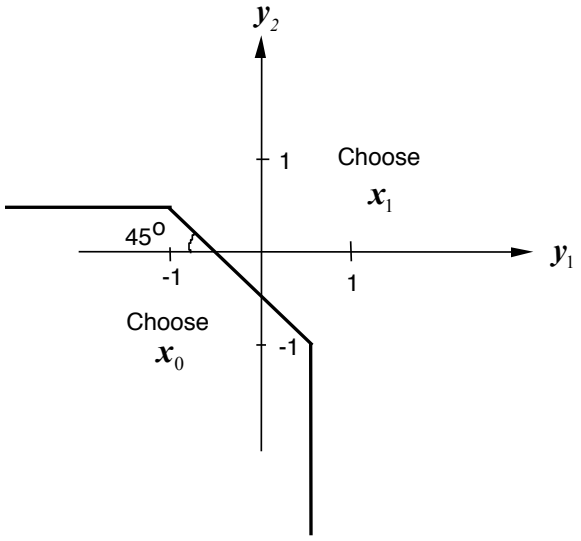


Figure 1.99: region3

1.8 *Optimum Receiver. (From Wozencraft and Jacobs) - 6 pts*

Suppose one of M equiprobable signals $x_i(t),\ i = 0, \ldots, M-1$ is to be transmitted during a period of time T over an AWGN channel. Moreover, each signal is identical to all others in the subinterval $[t_1, t_2]$, where $0 < t_1 < t_2 < T$.

a. Show that the optimum receiver may ignore the subinterval $[t_1, t_2]$. (2 pts)

b. Equivalently, show that if $\boldsymbol{x}_0, \ldots, \boldsymbol{x}_{M-1}$ all have the same projection in one dimension, then this dimension may be ignored.(2 pts)

c. Does this result necessarily hold true if the noise is Gaussian but not white ? Explain.(2 pts)

1.9 *Receiver Noise (use MATLAB for all necessary calculations - courtesy S. Li, 2005.) - 13 pts*

Each column of A given below is a data symbol that is used to construct its corresponding modulated waveform from a set of orthonormal basis functions (assume all messages are equally likely):

$$\Phi(t) = \left[\begin{array}{cccccc} \phi_1(t) & \phi_2(t) & \phi_3(t) & \phi_4(t) & \phi_5(t) & \phi_6(t) \end{array} \right] \quad .$$

The matrix A is given by

$$A = \left[\begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 \\ 5 & 6 & 7 & 8 & 5 & 6 & 7 & 8 \end{array} \right] \tag{1.559}$$

so that

$$x(t) = \Phi(t)A = [x_0(t)\ x_1(t)\ ...\ x_7(t)] \quad . \tag{1.560}$$

A noise vector $\boldsymbol{n} = \left[\begin{array}{c} n1 \\ n2 \\ n3 \\ n4 \\ n5 \\ n6 \end{array} \right]$ is added to the symbol vector $\boldsymbol{x}$, such that

$$y(t) = \Phi(t) \cdot (\boldsymbol{x} + \boldsymbol{n})$$

where $n_1 ... n_6$ are independent, with $n_k = \pm 1$ with equal probability.

The transmitted waveform $y(t)$ is demodulated using an ML detector. This problem examines the signal-to-noise ratio of the demodulated vector $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n}$ with $\sigma^2 \triangleq E(n^2)$

a. Find $\bar{\mathcal{E}}_{\boldsymbol{x}}$, σ^2, and SNR, $\bar{\mathcal{E}}_{\boldsymbol{x}}/\sigma^2$ if all messages are equally likely. (2 pts)

b. Find the minimal number of basis vectors and new matrix $\hat{A}$ as in Problem 1.4, and calculate the new $\bar{\varepsilon_x}$, σ^2, and SNR. (4 points)

c. Let the new vector be $\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{x}} + \tilde{\boldsymbol{n}}$, and discuss if the conversion from $\boldsymbol{y}$ to $\tilde{\boldsymbol{y}}$ is invariant (namely, if P_e is affected by the conversion matrix). Compare the detectors for parts a and b. (1 points)

d. Compare $\bar{b}$, $\bar{\varepsilon_x}$ with the previous system. Is the new system superior? Why or why not? (2 pts)

e. The new system now has three unused dimensions, and the source would like to send 8 more messages by constructing a big matrix $\bar{A}$, as follows:

$$\bar{A} = \begin{bmatrix} \hat{A} & \mathbf{0} \\ \mathbf{0} & \hat{A} \end{bmatrix}$$

Compare $\bar{b}$, $\bar{\varepsilon_x}$ with the original 6-dimensional system, and the 3-dimensional system in b). (4 pts)

1.10 *Tilt - 10 pts*

Consider the signal set shown in Figure 1.100 with an AWGN channel and let $\sigma^2 = 0.1$.

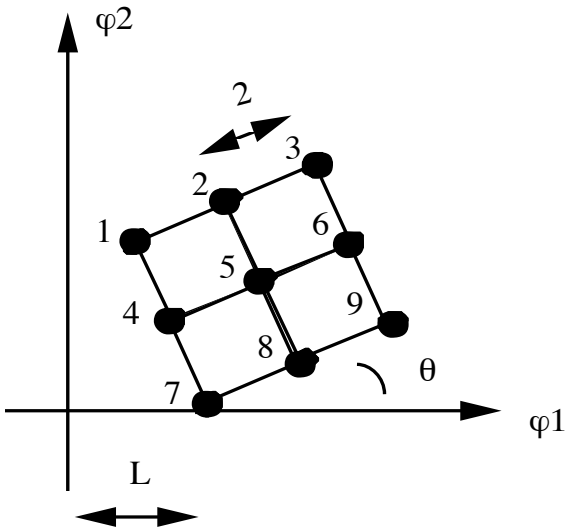


Figure 1.100: A Signal Constellation

a. Does P_e depend on L and θ? (1 pt)

b. Find the nearest neighbor union bound on P_e for the ML detector assuming $p_{\boldsymbol{x}}(i) = \frac{1}{9} \ \ \forall i$. (2 pts)

c. Find P_e exactly using the assumptions of the previous part. How close was the NNUB? (5 pts)

d. Suppose there is a minimum energy constraint on the signal constellation. How would this problem's constellation be altered without changing the P_e? How does θ affect the constellation energy? (2 pts)

1.11 *Parseval - 5 pts* Consider binary signaling on an AWGN $\sigma^2 = 0.04$ with ML detection for the following signal set. (Hint: consider various ways of computing $d_{\min}$.)

$$\begin{aligned} x_0(t) &= \text{sinc}^2(t) \\ x_1(t) &= \sqrt{2} \cdot \text{sinc}^2(t) \cdot \cos(4\pi t) \end{aligned}$$

Determine the exact P_e assuming that the two input signals are equally likely. (5 pts)

1.12 *Disk storage channel - 10 pts*

Binary data storage with a thin-film disk can be approximated by an input-dependent additive white Gaussian noise channel where the noise n has a variance dependent on the transmitted (stored) input. The noise has the following input dependent density:

$$p(n) = \begin{cases} \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{n^2}{2\sigma_1^2}} & \text{if } x = 1 \\ \frac{1}{\sqrt{2\pi\sigma_0^2}} e^{-\frac{n^2}{2\sigma_0^2}} & \text{if } x = 0 \end{cases}$$

and $\sigma_1^2 = 31\sigma_0^2$. The channel inputs are equally-likely.

a. For either input, the output can take on any real value. On the same graph, plot the two possible output probability density functions (pdf's). i.e. Plot the output pdf for $x = 0$ and the output pdf for $x = 1$. Indicate (qualitatively) the decision regions on your graph. (2 pts)

b. Determine the optimal receiver in terms of σ_1 and σ_0. (3 pts)

c. Find σ_0^2 and σ_1^2 if the SNR is 15 dB. SNR is defined as $\frac{\mathcal{E}_{\boldsymbol{x}}}{\frac{1}{2}(\sigma_0^2+\sigma_1^2)} = \frac{1}{\sigma_0^2+\sigma_1^2}$. (1 pt)

d. Determine P_e when SNR = 15 dB. (3 pts)

e. What happens as $\frac{\sigma_0^2}{\sigma_1^2} \longrightarrow 0$? You may restrict your attention to the physically reasonable case where σ_1 is a fixed finite value and $\sigma_0 \longrightarrow 0$. (1 pt)

1.13 *Rotation with correlated noise - 7 pts*

A two dimensional vector channel $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n}$ has *correlated* gaussian noise (that is the noise is not white and so not independent in each dimension) such that $E[n_1] = E[n_2] = 0$, $E[{n_1}^2] = E[{n_2}^2] = 0.1$, and $E[n_1 n_2] = 0.05$. n_1 is along the horizontal axis, and n_2 is along the vertical axis.

a. Suppose the transmitter uses the constellation in Figure 1.101 with $\theta = 45°$ and $d = \sqrt{2}$. (i.e. $x_1 = (1, 1)$ and $x_2 = (-1, 1)$). Find the mean and mean square values of the noise projected on the line connecting the two constellation points. This value more generally is a function of θ when noise is not white. (2 pts)

b. The noise projected on the line in the previous part is Gaussian. Find P_e for the ML detector. Assume the detector is designed for uncorrelated noise. (2 pts)

c. Fixing $d = \sqrt{2}$, find θ to minimize the ML detector P_e and give the corresponding P_e. You may continue to assume that the receiver is designed for uncorrelated noise. (2 pts)

d. Could your detector in part a be improved by taking advantage of the fact that the noise is correlated? (1 pt)

1.14 *32SQ QAM - 10 pts*

Consider the 64 QAM constellation with $d = 2$ (see Figure 1.102.): The 32 Square QAM ($\times$) selects every other point in a 64SQ QAM constellation. This problem investigates the properties of this 32SQ Constellation. This problem assumes that all points are equally likely and that the channel is an AWGN. Answers may be given by a formula that includes the unspecified σ for an AWGN channel.

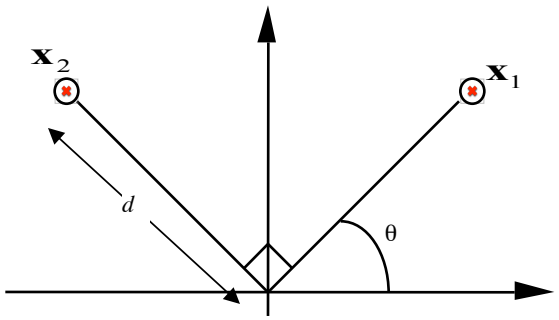


Figure 1.101: Constellation

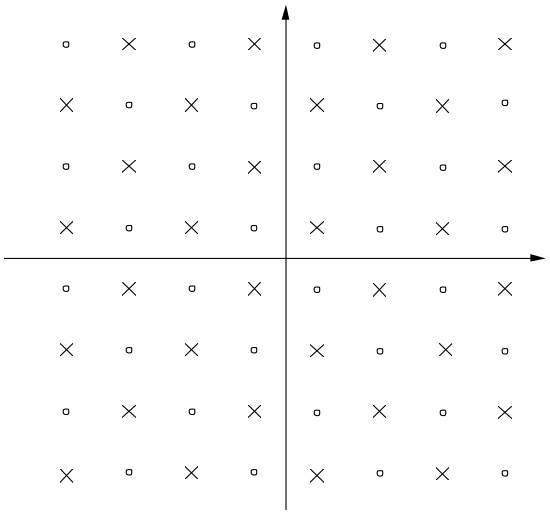


Figure 1.102: 32 SQ embedded in 64 SQ QAM Constellation

a. Compute the energy $\mathcal{E}_{\boldsymbol{x}}$ of the 64 QAM and the 32 hybrid QAM constellations. (2 pts)

b. What is d_{min} for a 32SQ QAM constellation having the same energy as 64SQ? (1 pt)

c. Find the NNUB for the symbol-error probabilities of the 64 QAM and 32 hybrid QAM constellations. (3 pts)

d. Find the NNUB error probability for the 32CR QAM constellation. Compare with the 32SQ QAM constellation. Which one performs better ? Why ? (2 pts)

e. Suppose the design would accommodate variable constellations and any power limitations apply only averaged over many symbol transmissions. Can you devise a simple alternating scheme that gets the same data rate? Is it better? (2 pts)

1.15 *Ternary Amplitude Modulation - 9 pts*

Consider the general case of the 3-D TAM constellation for which the data symbols are,

$$(x_l, x_m, x_n) = \left(\frac{d}{2}(2l - 1 - M^{\frac{1}{3}}), \frac{d}{2}(2m - 1 - M^{\frac{1}{3}}), \frac{d}{2}(2n - 1 - M^{\frac{1}{3}})\right)$$

with $l = 1, 2, \ldots M^{\frac{1}{3}}, m = 1, 2, \ldots M^{\frac{1}{3}}, n = 1, 2, \ldots M^{\frac{1}{3}}$. Assume that $M^{\frac{1}{3}}$ is an even integer.

a. Show that the energy of this constellation is (2 pts)

$$\mathcal{E}_{\boldsymbol{x}} = \frac{1}{M}\left[3M^{\frac{2}{3}}\sum_{l=1}^{M^{\frac{1}{3}}} x_l^2\right] \tag{1.561}$$

b. Now show that (3 pts)

$$\mathcal{E}_{\boldsymbol{x}} = \frac{d^2}{4}(M^{\frac{2}{3}} - 1)$$

c. Find the NNUB P_e and $\overline{P}_e$ for an AWGN channel with variance σ^2. (3 pts)

d. Find b and $\bar{b}$. (1 pt)

e. Find $\overline{\mathcal{E}}_{\boldsymbol{x}}$ and the energy per bit $\mathcal{E}_b$. (1 pt)

f. For an equal number of bits per dimension $\overline{b} = \frac{b}{N}$, find the constellation figure of merit for PAM, QAM and TAM constellations with appropriate sizes of M. Compare your results. (2 pts)

1.16 *Equivalency of rectangular-lattice constellations - 9 pts*

Consider an AWGN system with a SNR $= \frac{\bar{\mathcal{E}}_{\boldsymbol{x}}}{\sigma^2}$ of 22 dB, a target probability of error $P_e = 10^{-6}$, and a symbol rate $\frac{1}{T} = 8$ KHz. The transmit power is 20 dBm.

a. Find the maximum data rate $R = \frac{b}{T}$ that can be transmitted for (2 pts total this sub part)

(i) PAM ($\frac{1}{2}$ pt)

(ii) QAM ($\frac{1}{2}$ pt)

(iii) TAM (1 pt) - see Problem 1.15

b. What is the NNUB normalized probability of error $\overline{P}_e$ for the systems used in (a).(2 pts)

c. The remainder of this problem only considers QAM systems. Suppose that the desired data rate is 40 Kbps. What is the new transmit power needed to maintain the same probability of error? (The SNR is no longer 22 dB.) (2 pts)

d. With a yet newer SNR of 28 dB, what is the highest data rate that can be reliably sent at the same probability of error 10^{-6}? (1 pt)

1.17 *Frequency separation in FSK.* (Adapted from *Wozencraft & Jacobs) - 5 pts*

Consider the following two signals used in a Frequency Shift Key communications system over an AWGN channel.

$$x_0(t) = \begin{cases} \sqrt{\frac{2\mathcal{E}_{\boldsymbol{x}}}{T}} \cdot \cos(2\pi f_0(t)) & \text{if } t \in [0,T] \\ 0 & \text{otherwise} \end{cases}$$

$$x_1(t) = \begin{cases} \sqrt{\frac{2\mathcal{E}_{\boldsymbol{x}}}{T}} \cdot \cos(2\pi (f_0+\Delta)t) & \text{if } t \in [0,T] \\ 0 & \text{otherwise} \end{cases}$$

$$T = 100\mu s \quad f_0 = 10^5 \text{Hz} \quad \sigma^2 = 0.01 \quad \mathcal{E}_{\boldsymbol{x}} = 0.32$$

a. Find P_e if $\Delta = 10^4$. (2 pts)

b. Find the smallest $|\Delta|$ such that the same P_e found in part (a) is maintained. What type of constellation is this ? (3 pts)

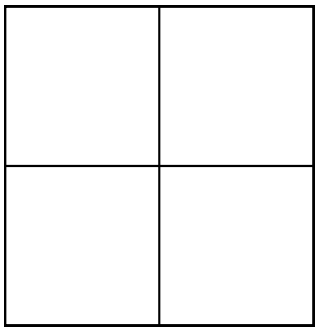


Figure 1.103: Sample pattern

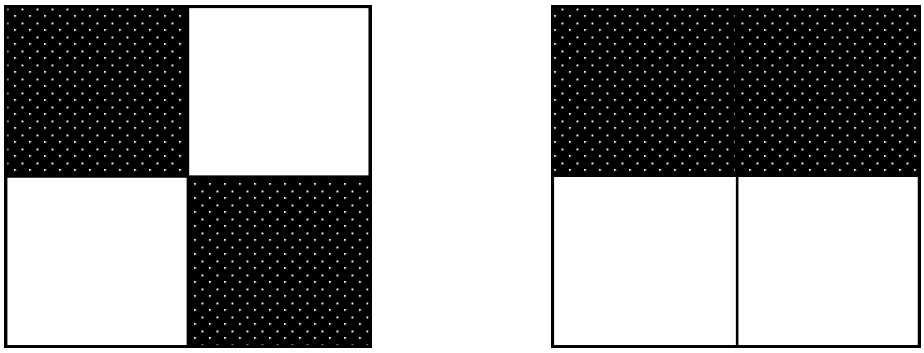


Figure 1.104: Examples of patterns considered

1.18 *Pattern Recognition - 8 pts*

In this problem a simple pattern recognition scheme, based on optimum detectors is investigated. The patterns considered consist of a square divided into four smaller squares, as shown in Figure 1.103,

Each square may have two possible intensities, black or white. The class of patterns studied will consist of those having two black squares, and two white squares. For example, some of these patterns are as shown in Figure 1.104,

Each pattern can be encoded into a vector $\boldsymbol{x} = [x_1\ x_2\ x_3\ x_4]$ where each component indicates the 'intensity' of a small square according to the following rule,

$$\begin{aligned} \text{Black square} &\Leftrightarrow x_i = 1 \\ \text{White square} &\Leftrightarrow x_i = -1 \end{aligned}$$

For a given pattern, a set of four sensors take measurements at the center of each small square and outputs $\boldsymbol{y} = [y_1\ y_2\ y_3\ y_4]$,

$$\boldsymbol{y} = \boldsymbol{x} + \mathbf{n} \tag{1.562}$$

Where $\mathbf{n} = [n_1\ n_2\ n_3\ n_4]$ is thermal noise (White Gaussian Noise) introduced by the sensors. The goal of the problem is to minimize the probability of error for this particular case of pattern recognition.

a. What is the total number of possible patterns ? (1 pt)

b. Write the optimum decision rule for deciding which pattern is being observed. Draw the corresponding signal detector. Assume each pattern is equally likely. (3 pts)

c. Find the union bound for the probability of error P_e. (2 pts)

d. Assuming that nearest neighbours are at minimum distance, find the NNUB for the probability of error P_e. (2 pts)

1.19 *Shaping Gain - 8 pts*

Find the shaping gain for the following two dimensional voronoi regions (decision regions) relative to the square voronoi region. Do this using the continuous approximation for a continuous uniform distribution of energy through the region.

a. equilateral triangle (2 pts)

b. regular hexagon (2 pts)

c. circle (2 pts)

d. Compare these different regions gains and explain the values qualitatively. (2 pts)

HINT: The following geometric identities may be helpful with a as a side or r as radius. The second moments listed are about the centroid, so $\int_{\mathcal{V}}((x-\mathbb{E}[x])^2+(y-E[y])^2)\cdot dx\cdot dy)$ and area $(\int_{\mathcal{V}} dx\cdot dy)$ may be helpful.

	equilateral triangle	circle	regular hexagon	square
Area	$\frac{\sqrt{3}}{4}a^2$	$\pi\cdot r^2$	$\frac{3\sqrt{3}}{2}\cdot a^2$	a^2
2nd Moment about centroid	$\frac{1}{16\sqrt{3}}\cdot a^4$	$\frac{\pi\cdot r^4}{2}$	$\frac{5\sqrt{3}}{8}a^4$	$\frac{1}{6}\cdot a^4$

1.20 *Recognize the Constellation (From Wozencraft and Jacobs) - 5 pts*

On an additive white Gaussian noise channel, determine P_e for the following signal set with ML detection. The answer will be in terms of σ^2.
(Hint: Plot the signals and then the signal vectors.)

$$
\begin{aligned}
x_1(t) &= \begin{cases} 1 & \text{if } t\epsilon\,[0,1] \\ 0 & \text{otherwise} \end{cases} \\
x_2(t) &= \begin{cases} 1 & \text{if } t\epsilon\,[1,2] \\ 0 & \text{otherwise} \end{cases} \\
x_3(t) &= \begin{cases} 1 & \text{if } t\epsilon\,[0,2] \\ 0 & \text{otherwise} \end{cases} \\
x_4(t) &= \begin{cases} 1 & \text{if } t\epsilon\,[2,3] \\ 0 & \text{otherwise} \end{cases} \\
x_5(t) &= \begin{cases} 1 & \text{if } t\epsilon\,[0,1] \\ 1 & \text{if } t\epsilon\,[2,3] \\ 0 & \text{otherwise} \end{cases} \\
x_6(t) &= \begin{cases} 1 & \text{if } t\epsilon\,[1,3] \\ 0 & \text{otherwise} \end{cases} \\
x_7(t) &= \begin{cases} 1 & \text{if } t\epsilon\,[0,3] \\ 0 & \text{otherwise} \end{cases} \\
x_8(t) &= 0
\end{aligned}
$$

1.21 *Comparing bounds - 6 pts*

Consider the following signal constellation in use on an AWGN channel.

$$
\begin{aligned}
\boldsymbol{x}_0 &= (-1,-1) \\
\boldsymbol{x}_1 &= (1,-1) \\
\boldsymbol{x}_2 &= (-1,1) \\
\boldsymbol{x}_3 &= (1,1) \\
\boldsymbol{x}_4 &= (0,3)
\end{aligned}
$$

Leave answers for parts a and b in terms of σ.

a. Find the union bound on P_e for the ML detector on this signal constellation. (2 pts)

b. Find the Nearest Neighbor Union Bound on P_e for the ML detector on this signal constellation. (2 pts)

c. Let the SNR = 14 dB and determine a numerical value for P_e using the NNUB. (2 pts)

1.22 *Basic QAM Design - 8 pts*

Either square or cross QAM can be used on an AWGN channel with SNR = 30.2 dB and symbol rate $1/T = 10^6$.

a. Select a QAM constellation and specify a corresponding integer number of bits per symbol, b, for a modem with the highest data rate such that $P_e < 10^{-6}$. (3 pts)

b. Compute the data rate for part a. (1 pt)

c. Repeat part a if $P_e < 2 \times 10^{-7}$ is the new probability of error constraint. (3 pts)

d. Compute the data rate for part c. (1 pt)

1.23 *Basic Detection - One shot or Two? - 10 pts*

A 2B1Q signal with $d = 2$ is sent two times in immediate succession through an AWGN channel with transmit filter $p(t)$, which is a scaled version of the basis function. All other symbol times, a symbol value of zero is sent. The symbol period for one of the 2B1Q transmissions is $T = 1$, and the transmit filter is $p(t) = 1$ for $0 < t < 2$ and $p(t) = 0$ elsewhere. At both symbol periods, any one of the 4 messages is equally likely, and the two successive messages are independent. The WGN has power spectral density $\frac{\mathcal{N}_0}{2} = .5$.

a. Draw an optimum (ML) basis detector and enumerate a signal constellation. (Hint: use basis functions.) (3 pts)

b. Find $d_{\min}$. (2 pts)

c. Compute $\tilde{N}_e$ counting only those neighbors that are $d_{\min}$ away. (2pts)

d. Approximate P_e for your detector. (3 pts)

1.24 *Discrete Memoryless Channel - 10 pts*

Given a channel with $p_{\boldsymbol{y}/\boldsymbol{x}}$ as shown in Figure 1.105: ($y \in \{0, 1, 2\}$ and $x \in \{0, 1, 2\}$)

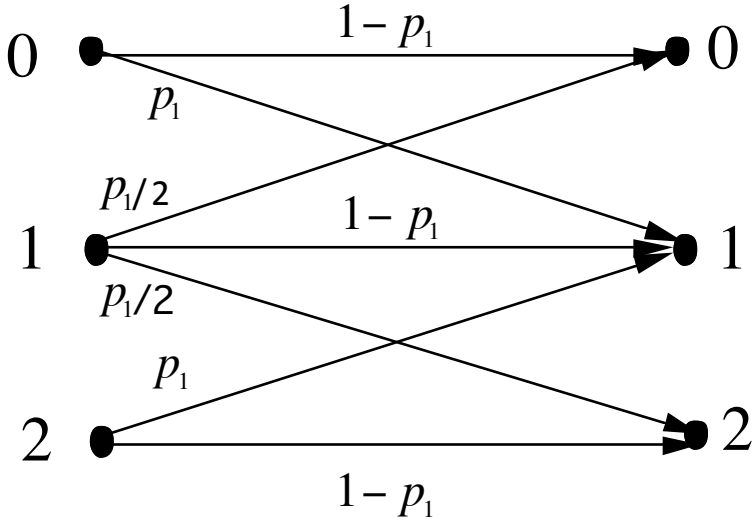


Figure 1.105: Discrete Memoryless Channel

Let $p_1 = .05$

a. For $p_{\boldsymbol{x}}(i) = 1/3$, find the optimum detection rule. (3 pts)

b. Find P_e for part a. (3 pts)

c. Find P_e for the MAP detector if $p_{\boldsymbol{x}}(0) = p_{\boldsymbol{x}}(1) = 1/6$ and $p_{\boldsymbol{x}}(2) = 2/3$. (4 pts)

1.25 *Detection with Uniform Noise - 9 pts*

A one-dimensional additive noise channel, $y = x + n$, has uniform noise distribution

$$p_n(v) = \begin{cases} \frac{1}{L} & |v| \leq \frac{L}{2} \\ 0 & |v| > \frac{L}{2} \end{cases}$$

where $L/2$ is the maximum noise magnitude. The input x has binary antipodal constellation with equally likely input values $x = \pm 1$. The noise is independent of x.

a. Design an optimum detector (showing decision regions is sufficient.) (2 pts)

b. For what value of L is $P_e < 10^{-6}$? (1 pt)

c. Find the SNR (function of L). (2 pts)

d. Find the minimum SNR that ensures error-free transmission. (2 pts)

e. Repeat part d if 4-level PAM is used instead. (2 pts.)

1.26 *Can you design or just use formulae? 8 pts*

32 CR QAM modulation is used for transmission on an AWGN with $\frac{\mathcal{N}_0}{2} = .001$. The symbol rate is $1/T = 400kHz$.

a. Find the data rate R. (1 pt)

b. What SNR is required for $P_e < 10^{-7}$? (ignore N_e). (2 pts)

c. In actual transmitter design, the analog filter rarely is normalized and has some gain/attenuation, unlike a basis function. Thus, the average power in the constellation is calibrated to the actual power measured at the analog input to the channel. Suppose $\bar{\mathcal{E}}_{\boldsymbol{x}} = 1$ corresponds to 0 dBm (1 milliwatt), then what is the power of the signals entering the transmission channel for the 32CR in this problem with $P_e < 10^{-7}$? (1 pt)

d. *The engineer under stress.* Without increasing transmit power or changing $\frac{\mathcal{N}_0}{2} = .001$, design a QAM system that achieves the same P_e at 3.2 Mbps on this same AWGN. (4 pts)

1.27 *QAM Design - 10 pts*

A QAM system with symbol rate $1/T$=10 MHz operates on an AWGN channel. The SNR is 24.5 dB and a $P_e < 10^{-6}$ is desired.

a. Find the largest constellation with integer b for which $P_e < 10^{-6}$. (2 pts)

b. What is the data rate for your design in part a? (2 pts)

c. How much more transmit power is required (with fixed symbol rate at 10 MHz) in dB for the data rate to be increased to 60 Mbps? ($P_e < 10^{-6}$) (2 pts)

d. With SNR = 24 dB, an reduced-rate alternative mode is enabled to accommodate up to 9 dB margin or temporary increases in the white noise amplitude. What is the data rate in this alternative 9dB-margin mode at the same $P_e < 10^{-6}$? (2 pts)

e. What is the largest QAM (with integer b) data rate that can be achieved with the same power, $\mathcal{E}_x/T$, as in part d, but with $1/T$ possibly altered? (2 pts)

1.28 *Basic Detection 12 pts*

A vector equivalent to a channel leads to the one-dimensional real system with $y = x + n$ where n is exponentially distributed with probability density function

$$p_n(u) = \frac{1}{\sigma\sqrt{2}} \cdot e^{-\sqrt{2}\frac{|u|}{\sigma}} \text{ for all } u \tag{1.563}$$

with zero mean and variance σ^2. This system uses binary antipodal signaling (with equally likely inputs) with distance d between the points. We define a function

$$\tilde{Q}(x) = \begin{cases} \int_x^\infty \frac{1}{\sqrt{2}} e^{-\sqrt{2}u} du = \frac{1}{2} \cdot e^{-\sqrt{2}x} \text{ for } x \geq 0 \\ 1 - \int_{|x|}^\infty \frac{1}{\sqrt{2}} e^{-\sqrt{2}u} du = 1 - \frac{1}{2} \cdot e^{-\sqrt{2}|x|} \text{ for } x \leq 0 \end{cases} \tag{1.564}$$

a. Find the values $\tilde{Q}(-\infty)$, $\tilde{Q}(0)$, $\tilde{Q}(\infty)$, $\tilde{Q}(\sqrt{10})$. (2 pts)

b. For what x is $\tilde{Q}(x) = 10^{-6}$? (1 pt)

c. Find an expression for the probability of symbol error P_e in terms of d, σ, and the function $\tilde{Q}$. (2 pts)

d. Defining the SNR as SNR $= \frac{\bar{\mathcal{E}}_{\boldsymbol{x}}}{\sigma^2}$, find a new expression for P_e in terms of $\tilde{Q}$ and this SNR. (2 pts)

e. Find a general expression relating P_e to SNR, M, and $\tilde{Q}$ for PAM transmission. (2 pts)

f. What SNR is required for transmission at $\bar{b} = 1, 2$, and 3 when $P_e = 10^{-6}$? (2 pts)

g. Would you prefer Gaussian or exponential noise if you had a choice? (1 pt)

1.29 *QAM Design - 8 pts*

QAM transmission is to be used on an AWGN channel with SNR=27.5 dB at a symbol rate of $1/T = 5$ MHz used throughout this problem. You've been hired to design the transmission system. The desired probability of symbol error is $\bar{P}_e \leq 10^{-6}$.

a. (2 pts) List two basis functions that you would use for modulation.

b. (2 pts) Estimate the highest bit rate, $\bar{b}$, and data rate, R, that can be achieved with QAM with your design.

c. (1 pt) What signal constellation are you using?

d. (3 pts) By about how much (in dB) would $\bar{\mathcal{E}}_{\boldsymbol{x}}$ need to be increased to have 5 Mbps more data rate at the same probability of error? Does your answer change for $\mathcal{E}_{\boldsymbol{x}}$ or for $P_{\boldsymbol{x}}$?

1.30 *7HEX Constellation - 10 pts*

QAM transmission is used on an AWGN channel with $\frac{\mathcal{N}_0}{2} = .01$. The transmitted signal constellation points for the QAM signal are given by $\begin{bmatrix} \pm\frac{\sqrt{3}}{2} \\ \pm\frac{1}{2} \end{bmatrix}$, $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$, and $\begin{bmatrix} 0 \\ \pm 1 \end{bmatrix}$, with each constellation point equally likely.

a. (1 pt) Find M (message-set size) and $\bar{\mathcal{E}}_{\boldsymbol{x}}$ (energy per dimension) for this constellation.

b. (2 pts) Draw the constellation with decision regions indicated for an ML detector.

c. (2 pts) Find N_e and $d_{\min}$ for this constellation.

d. (2 pts) Compute a NNUB value for $\bar{P}_e$ for the ML detector of part b.

e. (1 pt) Determine $\bar{b}$ for this constellation (value may be non-integer).

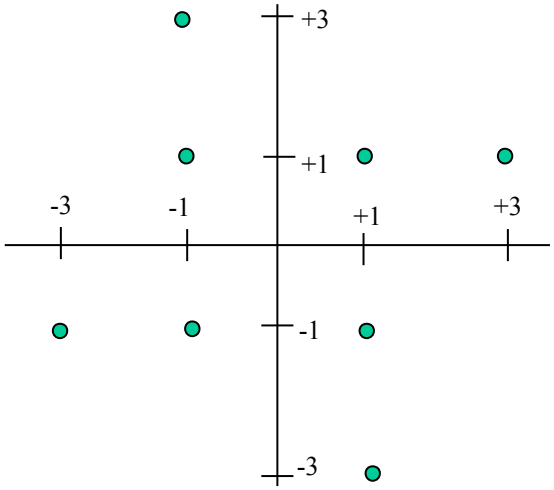


Figure 1.106: Constellation for 8 points.

f. (2 pts) For the same $\bar{b}$ as part e, how much better in decibels is the constellation of this problem than SQ QAM?

1.31 *Radial QAM Constellation - 7 pts*

The QAM "radial" constellation in Figure 1.106 is used for transmission on an AWGN with $\sigma^2 = .05$. All constellation points are equally likely.

a. (2pts) Find $\mathcal{E}_x$ and $\bar{\mathcal{E}}_{\boldsymbol{x}}$ for this constellation.

b. (3 pts) Find $\bar{b}$, $d_{\min}$, and N_e for this constellation.

c. (2 pts) Find P_e and $\bar{P}_e$ with the NNUB for an ML detector with this constellation.

1.32 *A concatenated QAM Constellation - 15 pts*

A set of 4 orthogonal basis functions $\{\varphi_1(t),\ \varphi_2(t),\ \varphi_3(t),\ \varphi_4(t)\}$ uses the following constellation in both the first 2 dimensions and again in the 2nd two dimensions: The constellation points are restricted

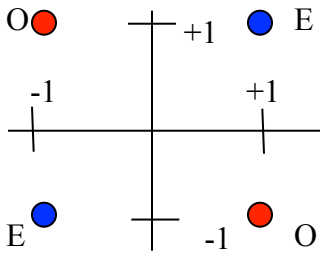


Figure 1.107: Constellation for Problem 1.32.

in that an E ("even") point may only follow an E point, and an O ("odd") point can only follow an O point. For instance, the 4-dimensional point $[+1\ +1\ -1\ -1]$ is permitted to occur, but the point $[+1\ +1\ -1\ +1]$ cannot occur.

a. (2 pts) Enumerate all M points as ordered-4-tuples.

b. (3 pts) Find b, $\bar{b}$, and the number of bits/Hz or bps/Hz.

c. (1 pt) Find $\mathcal{E}_{\boldsymbol{x}}$ and $\bar{\mathcal{E}}_{\boldsymbol{x}}$ (energy per dimension) for this constellation.

d. (2 pts) Find $d_{\min}$ for this constellation.

e. (2 pts) Find N_e and $\bar{N}_e$ for this constellation (you may elect to include only points at minimum distance in computing nearest neighbors).

f. (2 pts) Find P_e and $\bar{P}_e$ for this constellation using the NNUB if used on an AWGN with $\sigma^2 = 0.1$.

g. (3 pts) Compare this 4-dimensional constellation fairly (which requires increasing the number of points in the constellation to 6 to get the same data rate). 4QAM.

1.33 *Noise DAC - 15 pts*

A random variable $x_1$takes the 2 values ± 1 with equal probability independently of a second random variable x_2 that takes the values ± 2 also with equal probability. The two random variables are sumed to $x = x_1 + x_2$, and x can only be observed after zero-mean Gaussian noise of variance $\sigma^2 = .1$ is added, that is $y = x + n$ is observed where n is the noise.

a. (1 pt) What are the values that the discrete random variable x takes, and what are their probabilities? (1 pt)

b. (1 pt) What are the means and variances of x and y?

c. (2 pts) What is the lowest probability of error in detecting x given only an observation of y? Draw corresponding decision regions.

d. (1 pt) Relate the value of x with a table to the values of x_1 and x_2. Explain why this is called a "noisy DAC" channel.

e. (1 pt) What is the (approximate) lowest probability of error in detecting x_1 given only an observation of y?

f. (1 pt) What is the (approximate) lowest probability of error in detecting x_2 given only an observation of y?

g. Suppose additional binary independent random variables are added so that the two bipolar values for x_u are $\pm 2^{u-1}$, $u = 1, ..., U$. Which x_u has lowest probability of error for any AWG noise, and what is that P_e? (1 pt)

h. For $U = 2$, what is the lowest probability of error in detecting x_1 given an observation of y and a correct observation of x_2? (1 pt)

i. For $U = 2$, what is the lowest probability of error in detecting x_2 given an observation of y and a correct observation of x_1? (1 pt)

j. What is the lowest probability of error in any of parts e through i if $\sigma^2 = 0$? What does this mean in terms of the DAC? (1 pt)

k. Derive a general expression for the probability of error for all bits $u = 1, ..., U$ where $x = x_1 + x_2 + ... + x_U$ in AWGN with variance σ^2 for part g? (2 pts)

1.34 *Honeycomb QAM - 15 pts*

The QAM constellation in Figure 1.108 is used for transmission on an AWGN with symbol rate 10MHz and a carrier frequency of 100 MHz.

Each of the solid constellation symbol possibilities is at the center of a perfect hexagon (all sides are equal) and the distance to any of the closest sides of the hexagon is $\frac{d}{2}$. The 6 empty points represent a possible message also, but each is used only every 6 symbol instants, so that for instance, the point labelled 0 is a potential message only on symbol instants that are integer multiples of 6. The 1 point can only be transmitted on symbol instants that are integer multiples of 6 plus one, the 2 point only on symbol instants that are integer multiples of 6 plus two, and so on. At any symbol instant, any of the points possible on that symbol are equally likely.

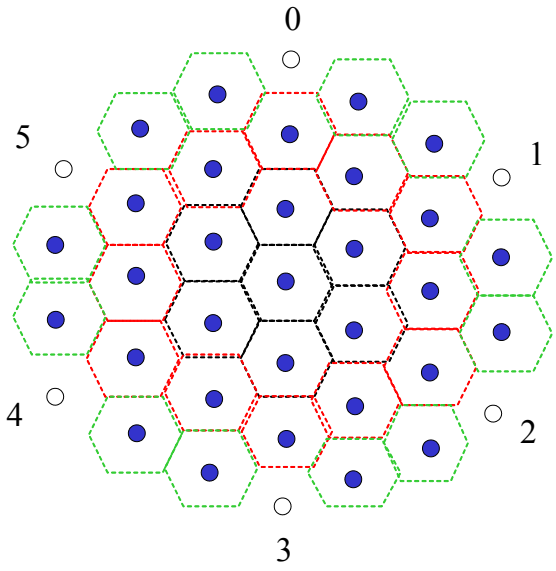


Figure 1.108: Constellation for Problem 1.34.

a. What is the number of messages that can be possibly transmitted on any single symbol? What are b and $\bar{b}$? (3 pts)

b. What is the data rate? (1 pt)

c. Draw the decision boundaries for time 0 of a ML receiver. (2 pts)

d. What is $d_{\min}$ in terms of d? (1 pt)

e. What are $\mathcal{E}_{\boldsymbol{x}}$ and $\bar{\mathcal{E}}_{\boldsymbol{x}}$ for this constellation in terms of d? (3 pts)

f. What is the number of average number nearest neighbors? (1 pt)

g. Determine the NNUB expression that tightly upper bounds $\bar{P}_e$ for this constellation in terms of SNR. (2 pts)

h. Compare this constellation fairly to Cross QAM transmission. (1 pt)

i. Describe an equivalent ML receiver that uses time-invariant decision boundaries and a constant decision device with a simple preprocessor to the decision device. (1 pt).

1.35 *Baseband Equivalents - 18 pts*

A channel with additive white Gaussian noise has the channel shown in Figure 1.109 with unit gain and and no phase distortion up to 50 MHz. The two-sided power spectral density of the noise is -103

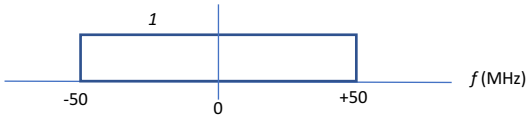


Figure 1.109: Channel Response.

dBm/Hz. The transmit power for a QAM modulator is 0 dBm = $\frac{\mathcal{E}_x}{T}$. The initial symbol rate is 1 MHz.

a. Suggest two ideal basis functions that use the lowest possible frequencies for this channel. (2 pts)

b. What is the SNR? (2 pts)

c. What is the data rate R if $\bar{P}_e \leq 10^{-7}$? (2 pts)

d. What is the constellation used for your answer in part c? (1 pt)

e. Draw the modulator, and specify input bits, the message m and the mapping into the in-phase component $x_I(t)$ and the quadrature component $x_Q(t)$. (3 pts)

f. Draw a simple demodulator. (3 pts)

g. What is the highest data rate for $\bar{P}_e \leq 10^{-7}$ using QAM or PAM and any symbol rate and/or carrier frequency? (3 pts)

h. What is the highest data rate for QAM/PAM if this channel had no filter and was purely an AWGN? (2 pts)

1.36 *Comparison - 18 pts*

Two passband transmission systems each use a transmit power ($\frac{\mathcal{E}_x}{T}$) of 0 dBm to transmit 16 Mbps with a carrier frequency of $f_c = 10^7$ Hz over an AWGN characterized by $\frac{N_0}{2} = -90$dBm/Hz. System 1 uses SSB with $M = 4$ (looks like 4 PAM) constellation, while System 2 uses 16 QAM. The basis functions are:

$$\begin{aligned} \text{System 1:} \quad \varphi_1(t) &= \tfrac{1}{\sqrt{T_1}} \cdot \text{sinc}\left(\tfrac{t}{T_1}\right) \cdot \cos\left(2\pi f_c t\right) - \tfrac{1}{\sqrt{T_1}} \cdot \text{sinc}\left(\tfrac{t}{T_1}\right) \cdot \sin\left(2\pi f_c t\right) \\ \text{System 2:} \quad \phi_1(t) &= \tfrac{1}{\sqrt{T_2}} \cdot \text{sinc}\left(\tfrac{t}{T_2}\right) \cdot \cos\left(2\pi f_c t\right) \\ \phi_2(t) &= \tfrac{1}{\sqrt{T_2}} \cdot \text{sinc}\left(\tfrac{t}{T_2}\right) \cdot \sin\left(2\pi f_c t\right) \quad . \end{aligned} \tag{1.565}$$

a. Complete Table 1.5 below (13 pts).

Quantity desired	System 1	System 2	points
Symbol rate			2
N			1
b			1
$\bar{b}$			1
E_x			2
$\bar{E}_x$			1
SNR			2
P_e			2
$\bar{P}_e$			1

Table 1.5: Table for Problem 1.35.

b. Let represent the total postive bandwidth allowed for transmission and suppose $W \leq 4$ MHz. Which system should be used? (1 pt).

c. Suppose$T \leq 125$ ns. Which system should be used? (1 pt)

d. With no restrictions on W nor T (but maintaining constant transmit power of 0 dBm) and a gap of $\Gamma = 8.8$ dB, what is the highest data rate that can be achieved if only QAM designs are allowed? (3 pts)

1.37 *Passband Representations - 9 pts*

Consider the following passband waveform:

$$x(t) = sinc^2(t) \cdot (1 + A\sin(4\pi t)) \cdot \cos(\omega_c t + \frac{\pi}{4}),$$

where $\omega_c >> 4\pi$.

Hint: It may be convenient in working this problem to use the identity $cos(a+b) = \cos a \cos b - \sin a \sin b$ to rewrite $x(t)$ in inphase and quadrature, and to realize/define $sinc^2(t)$ equal to a more general pulse shaping function $p(t)$, recognizing that this particular choice of $p(t)$ has a fourier transform that is well known and easily sketched.

a. Sketch (roughly) Re$[X(\omega)]$ and Im$[X(\omega)]$. (2 pts)

b. Find $x_{bb}(t)$, the baseband equivalent of $x(t)$. Sketch (roughly) $X_{bb}(\omega)$. (3 pts)

c. Find the $x_A(t)$ analytic equivalent of $x(t)$. (2 pts)

d. Find the Hilbert Transform of $x(t)$. (2 pts)

1.38 *A Two-Tap Channel - 8 pts*

The two equally likely baseband signals,$\tilde{x}_{bb,1}(t)$ and $\tilde{x}_{bb,2}(t)$ illustrated in the following figure are used to transmit a binary sequence over a channel. The use of the scaling phase splitter in Figure 1.58 is assumed. Note that the two signals do not seem to be of the form $(x_1 + jx_2) \cdot \varphi(t)$ directly. However, this form can be applied if one views each of these two signals as a succession of four "one-shot" inputs to the channel, each of which can be construed as of the form $(x_1 + jx_2) \cdot \varphi(t - iT/4),\ i = 0, 1, 2, 3$ – this view is not necessary, however, to work this problem. Equivalently, an easy representation is a four-dimensional symbol vector.

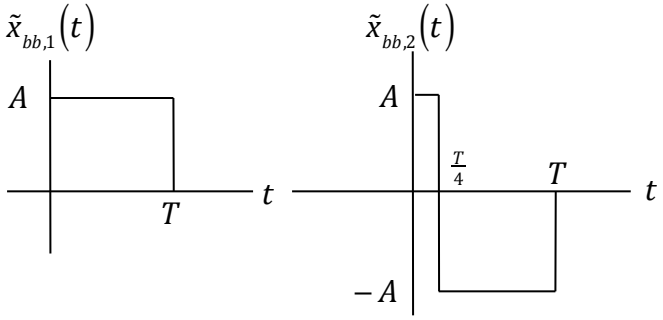


Figure 1.110: Two baseband signals.

The baseband equivalent channel impulse response is

$$h_{bb}(t) = 4 \cdot \delta(t) - 2 \cdot \delta(t - T)$$

The transmission rate is $R = \frac{1}{2T}$ bits per second to avoid overlay of successive transmissions.

a. Sketch the two possible baseband-equivalent noise–free received waveforms. (2 pts)

b. Compute the squared distance between the two baseband possible signals as the integral of the squared difference between the two signals at the channel input. Compute the same distance after filtering by the channel impulse response.(2 pts)

c. Determine P_e for transmission of the two corresponding messages where $n_{bb}(t)$ has autocorrelation $r_{bb}(t) = \mathcal{N}_0 \cdot \delta(t)$ with $\mathcal{N}_0 = \frac{\mathcal{E}_x}{20}$ where $\mathcal{E}_x$ is the average energy before filtering by the channel response. Compare this to the situation where the channel simply passes $x_{bb}(t)$ with no distortion and gain A, where the channel would then have $h_{bb}(t) = 2 \cdot \delta(t)$. (4 pts)

1.39 *A Bandpass Channel.* (from **Proakis** - 10 pts)
The input $x(t)$ to a bandpass filter is

$$x(t) = u(t) \cdot \cos(w_c t)$$

where

$$u(t) \quad = \quad \begin{cases} A & \text{if } t \in [0, T] \\ 0 & \text{otherwise} \end{cases}$$

Please assume that w_c is sufficiently high that $x(t)$ has only a neglible amount of energy near DC.

a. (3 pts) Determine the output $y(t) = g(t) * x(t)$ of a bandpass filter for all $t \geq 0$ if the impulse response of the filter is,

$$g(t) \quad = \quad \begin{cases} \frac{2}{T} \cdot e^{-t/T} \cdot \cos(\omega_c t) & \text{if } t \in [0, T] \\ 0 & \text{otherwise} \end{cases}$$

b. Sketch the equivalent lowpass output of the filter if it is passed through the scaling phase-splitter, $\tilde{y}_{bb}(t) =?$ (1 pt)

c. Assume the baseband equivalent noise at the output of the scaling phase splitter has variance $\mathcal{N}_0$ and that there are two dimensions. What is the SNR of the output? (3 pts)

d. For what value of A is the $\text{SNR}_{channel\ output}$=13 dB if the power spectral density of the channel's AWGN ($\frac{\mathcal{N}_0}{2}$) is - 30 dBm/Hz and $1/T = 1000$ Hz? Repeat for -100 dBm/hz and 1 MHz respectively. (3 pts)

1.40 *Passband Equivalent System - 5 pts*
A baseband-equivalent waveform ($w_c > 2\pi$)

$$\tilde{x}_{bb}(t) = (x_1 + jx_2) \cdot \text{sinc(t)}$$

is convolved with the complex filter

$$w_1(t) = \delta(t) - \jmath\delta(t-1)$$

a. (1 pt) Find

$$y(t) = w_1(t) * \tilde{x}_{bb}(t) \ .$$

b. (1 pt) Suppose $y(t)$ is convolved with a second complex filter

$$w_2(t) = 2\jmath \cdot \text{sinc(t)}$$

to get the complex filtered signal

$$\begin{aligned} z(t) &= w_2(t) * y(t) \\ &= w_2(t) * w_1(t) * \tilde{x}_{bb}(t) \\ &= w(t) * \tilde{x}_{bb}(t) \ , \end{aligned}$$

so that $z(t)$ is complex, yet corresponds to some passband signal. First find the complex signal $z(t)$. Note that $sinc(t) * sinc(t-k) = sinc(t-k)$, when k is an integer.

c. (3 pts) Let us define an analytic complex signal $z_A(t) = z(t) \cdot e^{\jmath\omega_c t}$ with real part:

$$\tilde{z}(t) = \Re\left\{z(t) \cdot e^{\jmath\omega_c t}\right\} = \tilde{w}(t) * \tilde{x}(t) \quad ,$$

that is the result of some **real** filter $\tilde{w}(t)$ acting on the transmitted passband signal $\tilde{x}(t) = \Re\{\tilde{x}_{bb}(t)e^{\jmath\omega_c t}\}$ (when convolved with the passband $\tilde{x}(t)$ will produce $\tilde{z}(t)$). Show that

$$\tilde{w}(t) = 4 \cdot \text{sinc}(t-1) \cdot \cos(\omega_c t) - 4 \cdot \text{sinc}(t) \cdot \sin(\omega_c t)$$

and that correspondingly.

$$\tilde{w}_{bb}(t) = 4 \cdot [\text{sinc}(t-1) - \jmath\text{sinc}(t)] \quad .$$

Hint: Use baseband calculations.

1.41 *Matlab demodulator - 10 pts*

This problem uses three matlab files: **x.mat, plt_fft.m,** and **plt_cplx.m**. These files are available at the web-site

`http://web.stanford.edu/group/cioffi/ee379a/ \;\;\; .`

This problem's objective is to demodulate three symbols of a passband 4 QAM signal. The baseband basis function is a windowed sinc function. The sampling rate that provided the digital received signal was 1000 Hz. The received signal that this problem uses is the real part of the analytic signal. Usually the signal will have been convolved with a channel response and had noise added, but this problem ignores noise (so it is zero).

a. First, download the three needed files. Execute **load x.mat** which will create a 747 point vector $\boldsymbol{x}$ which contains 3 symbol periods of received signal. (Each symbol period is 249 samples). The function **plt_cplx** plots the real and imaginary parts of a complex vector. It takes two arguments, the vector and the plot title. Execute **plt_cplx(x, 'received signal')** to produce a plot (and provide your plot). The received signal should be real. While the constellation points are not readily evident, the three symbols should be fairly evident. (1 pt)

b. Now, execute **plt_fft(x, 'received signal')** to plot an FFT of the received signal. Provide the plot. Neglect powers that are 50 dB smaller than other signals and report the frequency range to which this signal is bandlimited (2 pts)

c. Now use the function **hilbert()** provided by MATLAB to recover the analytic signal **x_A**. You might want to execute **help hilbert** to get started. Use **plt_fft** to plot the FFT of **x_A** (and provide the plot). How is this signal different from **x**? The discussion may again neglect signals 50 dB below peak signals. (2 pts)

d. The carrier frequency is 250 Hz. As mentioned before, the sampling frequency is 1000 Hz. Show that the discrete time radian carrier frequency is $\frac{\pi}{2}$ radians/sec. (1 pt)

e. (4 pts) Demodulate **x_A** to create the baseband signal **x_bb** by executing the following command:

x_bb = x_A .* exp (-j * 0.5 * pi * [0:746]);

Plot both the FFT and complex time sequence as before (and provide the plots). In what range of frequencies is **x_bb** non-negligible? Why? By examining the complex time sequence plot, decode the received signal. The complex constellation points have been labeled as shown below. Correct selection of the correct constellation point for each symbol need only consider the sign of the real and imaginary scale factors multiplying the windowed sinc pulse.

1.42 *Baseband Analysis - 4 pts*

Consider the two baseband equivalent signals, $\tilde{x}_{bb,1}(t)$ and $\tilde{x}_{bb,2}(t)$.

$$\tilde{x}_{bb,1}(t) = \begin{cases} A(1+j) & \text{if } t \in [0,T] \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{x}_{bb,2}(t) = \begin{cases} A(1-j) & \text{if } t \in \left[0, \frac{3}{4}T\right] \\ -A(1-j) & \text{if } t \in \left(\frac{3}{4}T, T\right] \\ 0 & \text{otherwise} \end{cases}$$

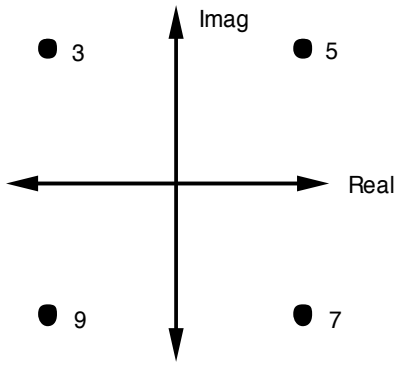


Figure 1.111: 4-QAM constellation with message labels 3, 5, 7 & 9

These signals can be used to transmit a binary signal set.

$$\mathrm{p}_{\boldsymbol{x}_{bb}}(1) = \mathrm{p}_{\boldsymbol{x}_{bb}}(2) = \frac{1}{2}$$

The transmitted signals are corrupted by AWGN having a baseband equivalent representation corresponding to the scaled phase splitter of Figure 1.58, $\tilde{n}_{bb}(t)$, with an autocorrelation function

$$r_{\tilde{n}_{bb}}(\tau) = E\left[\tilde{n}^*_{bb}(t)\tilde{n}_{bb}(t+\tau)\right] = \mathcal{N}_0\delta(t)$$

a. Find $\mathcal{E}_{\boldsymbol{x}}$. (2 pts)

b. Find P_e as a function of A, T and $\mathcal{N}_0$. (1 pt)

c. Find $\frac{A^2}{N_0}$ in terms of T if SNR=12.5 dB. Compute the probability of error P_e (a numerical value is required). (1 pt)

1.43 *Twisted pairs - 3 pts*

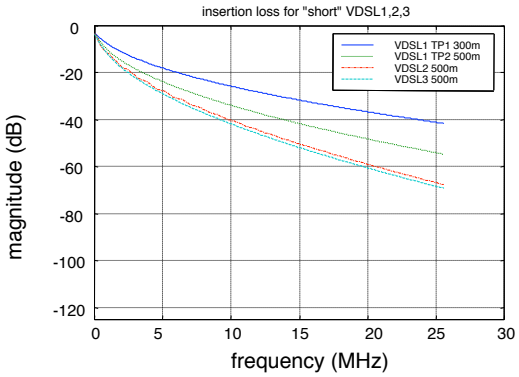


Figure 1.112: Insertion loss for 26-gauge (TP1) and 24-guage (TP2) twisted-pair phone line.

Figure 1.112 shows the magnitude(in dB) of the insertion losses (which is 6 dB more than the transfer function $H(f)$) for several lengths of two-types of twisted pair. Suppose the signals are passband, with frequencies ranging from 6MHz to 12 MHz. You want to analyse this transmission system in baseband. Assume a carrier frequency of $f = 9$ MHz. Assume the receiver uses a scaling phase splitter.

a. Draw the frequency responses of the complex channels to which you would apply the complex modulator input $\tilde{x}_{bb}(t)$, corresponding to the scaling in Figure 1.58. (1 pt)

b. Compute the noise power spectral density (two-sided) of the WGN that you would add to each of your complex channel outputs to model transmission if the one-sided power spectral density of the AWGN noise on the channel is given as -140 dBm/Hz. (2 pts)

1.44 *Signal Transformation Practice - 4 pts*

Find the Hilbert transform of:

$$x(t) = sinc(\frac{t}{T})\cos(\omega_c t + \frac{\pi}{4})$$

where

$$\omega_c \geq \frac{\pi}{T} \quad .$$

1.45 *Baseband Analysis with Parseval's Help - 5 pts*

The two baseband equivalent signals at the modulator output using the scaling in Figure 1.58 for binary transmission:

$$\begin{aligned} \tilde{x}_0(t) &= \frac{1}{\sqrt{T}}\text{sinc}\left(\frac{t}{T}\right) + \jmath\sqrt{\frac{2}{T}}\cos\left(\frac{\pi t}{T}\right)\cdot\text{sinc}(t/T) \\ \tilde{x}_1(t) &= \jmath\sqrt{\frac{2}{T}}\sin\left(\frac{\pi t}{T}\right)\cdot\text{sinc}(t/T) \end{aligned}$$

are transmitted over an AWGN with $\frac{\mathcal{N}_0}{2} = .02$.

a. Find $\tilde{X}_0(f)$, $\tilde{X}_1(f)$, and $\tilde{X}_0(f) - \tilde{X}_1(f)$. (3 pts)

b. Determine P_e. (2 pts)

1.46 *Phase Distortion Only - 8 pts*

A passband channel has complex baseband equivalent impulse response

$$h_{bb}(t) = (1+\jmath)\delta(t) \quad .$$

A 4 QAM (QPSK) input with the constellation labeling below in Figure 1.113 is input to this channel. WGN is added at the output of this channel with power spectral density $\frac{\mathcal{N}_0}{2} = .04$.

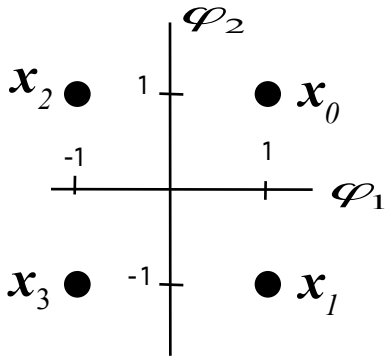


Figure 1.113: Channel input for Problem 1.46

a. Calculate and draw the constellation for the corresponding four signal constellation points at the output of this channel and a scaling demodulator. Call these points $\tilde{y}_0$, ..., $\tilde{y}_3$ where the subscripts correspond to the subscripts on the channel input. (4 pts)

b. Write a quadrature decomposition for $\tilde{y}_0$'s corresponding passband modulated signal. (2 pts)

c. Sketch the baseband AWGN power spectral density. (2 pts)

1.47 *64 Single Sideband (SSB) - (18 pts)*

Let $m(t)$ be an 8-PAM signal with

$$m(t) = \sum_k x_k \cdot p(t - kT)$$

where $x_k = \pm 1, \pm 3, \pm 5, \pm 7$ and $p(t) = \text{sinc}(2t/T)$. Also let $\omega_c >> 2\pi/T$ while forming the single-sideband (SSB) modulated signal

$$x(t) = m(t)\cos(\omega_c t) - \check{m}(t)\sin(\omega_c t)$$

for transmission over an AWGN with $\frac{\mathcal{N}_0}{2}$ as the power spectral density..

a. Find an analytic signal $m_A(t)$ that is equivalent to $x(t)$. Also find $x_A(t)$ in terms of $m_A(t)$. (3 pts)

b. Find the Hilbert transform of $p(t)$, and rewrite your answer to part a without using $m(t)$. (3 pts)

c. Write an expression for $\check{m}(t)$ in terms of x_k that uses your result from part b. Evaluate this $m(t)$ and $\check{m}(t)$ in terms of integer multiple of $T/2$ sampling instants, $n(T/2)$. What does this tell you about the pure phase changing of the Hilbert Transform and any interference at the channel output between successive transmitted symbols x_k? (2 pts)

d. What is the noise sample variance for the corresponding channel output samples of $m(t)$? Find the SNR for this sequence $m(nT/2)$. Compare this SNR with that of 64SQ QAM on the same channge with symbol rate $1/T$ and the same energy per dimension. (2 pts)

e. Find the single basis function for this SSB transmission system. Was straightforward estimation in part d of the symbol values from $m(nT/2)$ an ML detector? Why or why not? Now compare again to 64 QAM. (4 pts)

f. Draw a block diagram of a ML receiver for this SSB transmission system using $\check{m}(t)$ also. (4 pts)

g. Why is this signal called **64**-VSB or equivalently **64**-SSB (and not 8 SSB)? (1 pt)

h. For what $\frac{\mathcal{N}_0}{2}$ is $P_e < 10^{-6}$? (answer in terms of T) (1 pt)

1.48 *Complex Channel - Duobinary* $1 + D$ *- 19 pts*

A complex channel, derived through the scaling of Figure 1.58, has binary inputs $\tilde{x}_{bb}(t)$ in Figure 1.114 below. Let the passband filter be $\frac{h_{bb}(t)}{2} = \delta(t)$ and the SNR $= \frac{\bar{\mathcal{E}}_{\boldsymbol{x}}}{\sigma^2}$=10 dB and the signal is considered two dimensional (one real and one imaginary dimension) for computation of $\bar{\mathcal{E}}_{\boldsymbol{x}}$.

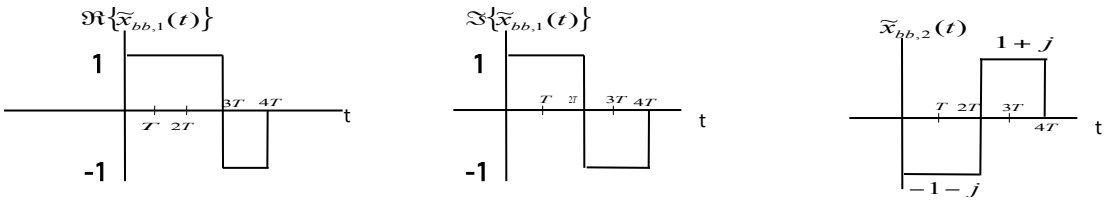


Figure 1.114: Baseband channel inputs.

a. write $\tilde{x}_{bb,1}(t)$ and $\tilde{x}_{bb,2}(t)$ in the form of 4 successive transmissions with symbol rate $1/T$. Each symbol must be of the form $(x_1 + jx_2) \cdot \varphi(t - kT)$ $k = 0, 1, 2, 3$. Thus, you must find x_1, x_2, and φ. (4pts)

b. Find $d_{\min}$ with $T = 1$. (2 pts)

c. Find P_e for an ML detector with $T = 1$. (2 pts)

d. For the remainder of this problem, let $\frac{1}{2}h_{bb}(t) = \delta(t) + \delta(t-1)$ and $T = 1$. Find the baseband-equivalents channel outputs prior to addition of baseband noise, $\tilde{n}_{bb}(t)$. (4 pts)

e. Find $d_{\min}$. (2 pts)

f. Find P_e with an ML detector, assuming all inputs are equally likely. (2 pts)

g. Has the distortion introduced by the new $h_{bb}(t)$ improved or degraded this system? Why? (3 pts)

1.49 *Linear Frequency Decrease in Baseband 11 pts*

Figure 1.115 shows the Fourier transform, $H(f)$, of a bandlimited channel's impulse response, $h(t)$. The input to the channel is

$$x(t) = \sqrt{2}\left\{\left[\sum_k a_k \cdot \varphi(t-kT)\right] \cdot \cos(\omega_c t) - \left[\sum_k b_k \cdot \varphi(t-kT)\right] \cdot \sin(\omega_c t)\right\} \quad . \tag{1.566}$$

This channel is used for passband transmission with QAM and has AWGN with (2-sided) power spectral density $\sigma^2 = .01$.

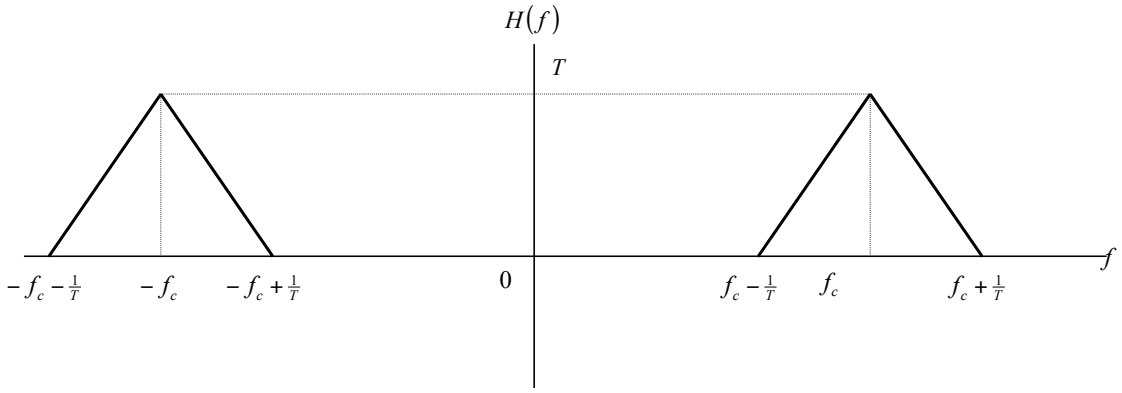


Figure 1.115: Figure for "baseband equivalents".

a. Draw $H_{bb}(f)$. (1 pt)

b. Find $h_{bb}(t)$. (2 pts)

c. Find the input $\tilde{x}_{bb}(t)$ as per Figure 1.58. (2 pts)

d. Find the complex channel model, including the channel complex impulse response and a numerical value for the noise power spectral density corresponding to the complex input you found in part c. The channel output should be the $\tilde{y}_{bb}(t)$ of Figure 1.58. (2 pts)

e. Find the analytic equivalent $h_A(t)$. (2 pts)

f. Write a simple expression for the Hilbert transform of $h(t)$. $\check{h}(t)$ =?(2 pts)

1.50 *Linear Frequency Decrease in Passband - 10 pts*

The Fourier transform of the impulse response of a channel is shown in Figure 1.116. The power spectral density of the additive Gaussian noise at the output of the channel is shown in tFigure 1.117. You are given the following integrals to avoid any need for doing integration in this problem (i.e., you can plug the formulae)

$$\int x^2 e^{bx} dx = \frac{e^{bx}}{b^3}\left(b^2x^2 - 2bx + 2\right) \quad ; \quad \int x e^{bx} dx = \frac{e^{bx}}{b^2}(bx-1) \quad ; \quad \int e^{bx} dx = \frac{e^{bx}}{b} \quad . \tag{1.567}$$

Throughout this problem, please use the scaling QAM demodulator of Figure 1.60.

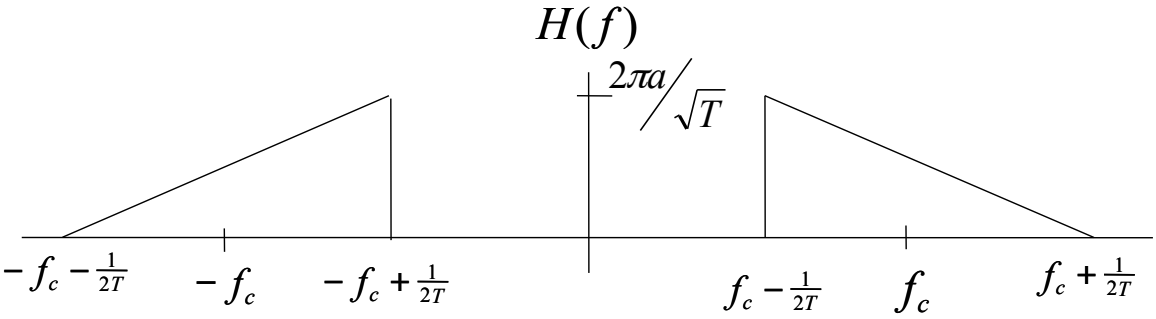


Figure 1.116: Channel Response.

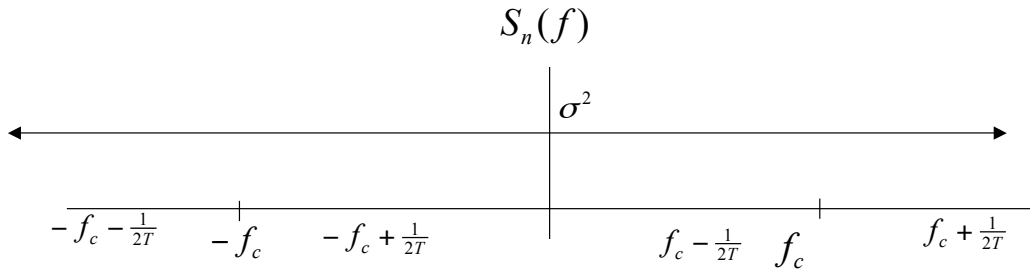


Figure 1.117: Noise Power Spectral Density.

a. Draw the Fourier transform of the complex channel, $\frac{1}{2}H_{bb}(f)$, that is used in this text to model the channel. (2 pts)

b. Find the power spectral density per real dimension of the noise in the complex baseband-equivalent channel that results from the scaled demodulator. (1 pt)

c. Find the complex-equivalent pulse response (time-domain) of the channel in part a if the transmitter uses the basis functions of QAM with $\varphi(t) = \frac{1}{\sqrt{T}} \cdot \text{sinc}\frac{t}{T}$. Intepret if $a \cdot \sqrt{T} = 1$ (4 pts)

d. If $\sigma^2 = 1$, and the transmitter sends 16 SQ QAM with constellation points $\begin{bmatrix} \pm 3 \\ \pm 1 \end{bmatrix}$, what is the lowest upper bound on the best possible SNR for a symbol-by-symbol detector on this channel? (3 pts)

1.51 *Mini-Design - 12 pts*

An AWGN with SNR=22 dB has baseband channel transfer function (there is no energy gain or loss in the channel):

$$H(f) = \begin{cases} 1 & |f| < 500 \text{ kHz} \\ 0 & |f| \geq 500 \text{ kHz} \end{cases} \tag{1.568}$$

a. Find an integer $\bar{b}$ for QAM transmission with $P_e < 10^{-6}$ and compare with value found by the "gap approximation." (2 pts)

b. Find the largest possible symbol rate and corresponding data rate. (2 pts)

c. Draw the corresponding signal constellation and label your points with bits. (2 pts) - hint, don't concern yourself with clever bit labelings, just do it.

d. Find $\bar{P}_b$ and N_b for your design in part b. (3 pts)

e. Repart part b for PAM transmission on this channel and explain the difference in data rates. (3 pts)

1.52 *Offset Carrier - 12 pts*

For the AWGN channel with transfer function shown in Figure 1.118, a transmitted signal cannot exceed 1 mW (0 dBm) and the power spectral density is also limited according to $S_x(f) \leq$-83 dBm/Hz(two-sided psd). The two-sided noise power spectral density is $\sigma^2 =$-98 dBm/Hz. The carrier frequency is $f_c =$100 MHz for QAM transmission. The probability of error is $P_e = 10^{-6}$.

Figure 1.118: Channel Response.

a. Find the baseband channel model, $\frac{1}{2}H_{bb}(f)$, for the scaled demodulator of Chapter 2. (2 pts)

b. Find the largest symbol rate that can be used with the 100 MHz carrier frequency? (1 pts)

c. What is the maximum signal power at the channel output with QAM? (2 pts)

d. What QAM data rate can be achieved with the symbol rate of part b? (2 pts)

e. Change the carrier frequency to a value that allows the best QAM data rate. (2 pts)

f. What is the new data rate for your answer in part e? (3 pts)

1.53 *Complex Channel and Design- Midterm 2003 - 15 pts*

Let x_k represent the successive independent transmitted symbols of a QAM constellation that can have only an integer number of bits for each symbol and for which each message is equally likely. Also let the pulse response of a filtered AWGN channel be $p(t) = \sqrt{\frac{1}{T}} \cdot \text{sinc}\left(\frac{t}{T}\right)$ where $\frac{1}{T} = 5$ Mhz is the symbol rate of the QAM transmission. The carrier frequency is 100 MHz. The transmitted signal has $\bar{\mathcal{E}}_{\boldsymbol{x}} = 1.2$ and the AWGN psd is $\sigma^2 = .01$. An expression for the modulated signal is

$$x(t) = \Re\left\{\sum_k x_k \cdot p(t - kT) \cdot e^{\jmath\omega_c t}\right\} \quad . \tag{1.569}$$

Define $g(t) = p(t) \cdot e^{\jmath\omega_c t}$.

a. Find $x_A(t)$ and $x_{bb}(t)$. (2 pts)

b. Use the gap approximation to determine the number of bits per dimension, data rate, and the number of bits/second/Hz that can be transmitted on this channel with $\bar{P}_e \leq 10^{-6}$. (3 pts)

c. Suppose $A_k = x_k \cdot e^{\jmath\omega_c kT}$ is the actual message symbol sequence of interest for the rest of this problem. How do your answers in part b change (if at all)? Why? (2 pts)

d. Find the analytic equivalent of the channel output, $y_A(t)$ in terms of only the message sequence A_k and $g(t)$ without any direct use of the carrier frequency. (2 pts)

e. Draw an optimum (MAP) detector for A_k that does not use the carrier frequency directly. (3 pts)

f. Augment your answer in part e with a simple rotation that provides the MAP detector for x_k. (1 pt)

g. This approach is used in some communications systems where the symbol rate and carrier frequency can be co-generated from the same oscillator, hence the knowledge of the symbol rate in the receiver tacitly implies then also knowing the carrier frequency. This is why the carrier was eliminated in the receiver of part e. Suggest a receiver implementation problem with this approach in general to replace QAM systems that would be used in transmission with symbol rates of up to 10 MHz and carriers above 1 GHz. (Hint - what does $g(t)$ look like?) (2 pts)

1.54 *Partial Response Class IV as a Passband Channel - 15 pts*

A 16 QAM constellation is used to transmit a message over a filtered AWGN with SNR $= \frac{\bar{\mathcal{E}}_{\boldsymbol{x}}}{\sigma^2} = 20$ dB. The QAM symbol is given by $\sqrt{\frac{2}{T}} \cdot \left[x_1 \cdot f(t) + x_2 \check{f}(t)\right]$ where $f(t) = \text{sinc}\left(\frac{2t}{T}\right)$. The real channel filter in has the shape shown in Figure 1.119 and is zero outside the frequency band of $(-1/T, 1/T)$.

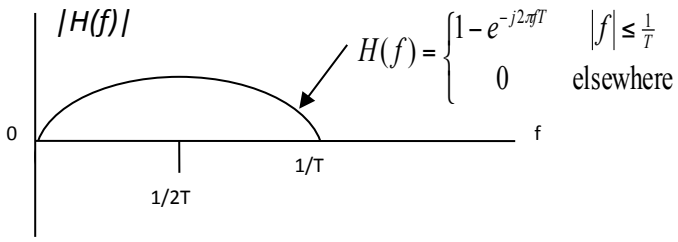


Figure 1.119: Channel Response.

a. Show $f(t) = \text{sinc}\left(\frac{t}{T}\right) \cdot \cos\left(\frac{\pi t}{T}\right)$ using frequency-domain arguments. Then find the Hilbert Transform $\check{f}(t)$. (2 pts)

b. Find a quadrature representation for $x(t)$ and the appropriate carrier frequency f_c. (2 pts)

c. Find $x_{bb}(t)$ and $x_A(t)$. (2 pts)

d. Find $h(t)$, $h_{bb}(t)$, and $h_A(t)$. **Hint:** multiplication of a transfer function by the ideal lowpass filter is the same as convolving with a sinc function, which may be useful in converting the obvious frequency-domain answers to the time domain. (2 pts)

e. For the scaled phase splitter of Chapter 2, find the output $y_{bb}(t)$. (2 pts)

f. Design (draw) an optimum receiver for this single transmitted message using only one sampling device, one delay element, one matched filter, and one adder. Draw the decision regions for a single complex value that ultimately emanates from your receiver. (3 pts)

g. Calculate P_e for this optimum receiver. (2 pts)

1.55 *Vestigially Symmetric Channel Output - 12 pts*

The Fourier Transform $Y(f)$ of the output after QAM modulation for symbol x_k and channel and receiver filtering is shown in Figure 1.34. Assume that the receiver filter is of unit norm (unit energy) but is before the scaling phase-splitting operation. The AWGN of this channel (before receiver filtering) has power spectral density of $\frac{N_0}{2} = 1$, and let $\bar{\mathcal{E}}_{\boldsymbol{x}} = 1$.

a. Find $\tilde{y}_{bb}(t)$ and its Fourier Transform $\tilde{Y}_{bb}(f)$ after the scaling phase splitter. (2 pts)

b. Find the corresponding power spectral density of the baseband-equivalent noise $\tilde{S}_{bb}(f)$ (2 pts)

c. Find the highest symbol rate for which there is no ISI. (2 pts)

d. Find the data rate if $\bar{P}_e \leq 10^{-6}$ for the symbol rate of part c. (4 pts)

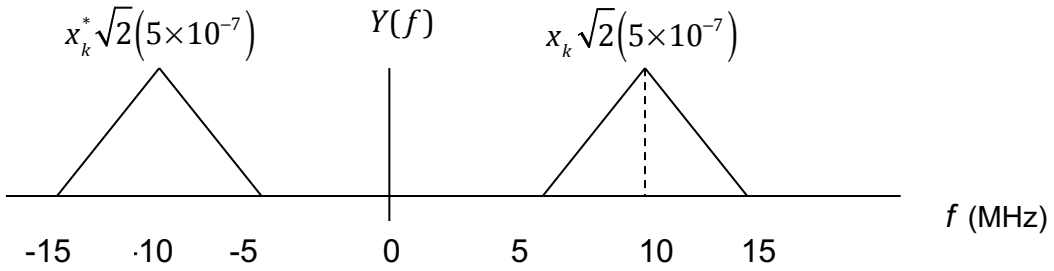


Figure 1.120: Channel Response.

1.56 *Baseband Equivalents - Midterm 2008 - 11 pts*

Figure 1.35 shows the Fourier Transform $H(f)$ of a bandlimited channel's impulse response $h(t)$. The input to the channel is

$$x(t) = \sqrt{2} \cdot \left\{ \left[\sum_k a_k \cdot \varphi(t - kT) \right] \cdot \cos(2\pi f_c t) - \left[\sum_k b_k \cdot \varphi(t - kT) \right] \cdot \sin(2\pi f_c t) \quad . \right\}$$

The channel is used for passband transmission with QAM and has AWGN with (2-sided) power spectral density $\frac{\mathcal{N}_0}{2} = 0.01$.

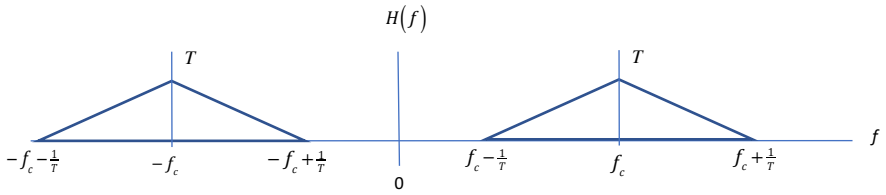


Figure 1.121: Channel Response.

a. Draw $H_{bb}(f)$. (1 pt)

b. Find $h_{bb}(t)$. (2 pts)

c. Find the input $\tilde{x}_{bb}(t)$ in terms of $\varphi(t)$. (1 pt)

d. For all that follows, let $\varphi(t) = \sqrt{\frac{2}{T}} \cdot \text{sinc}\left(\frac{2t}{T}\right)$. Show the complex channel model, including the channel complex impulse response and a numerical value for the noise power spectral density corresponding to the input you found in part c. Find the channel output $\tilde{y}_{bb}(t)$. (2 pts)

e. Find the analytic equivalent $h_A(t)$. (2 pts)

f. Write a simple expression for the Hilbert transform of $h(t)$, $\check{h}(t) =?$ (2 pts)

g. Find $\tilde{y}(t)$. (1 pt)

1.57 *Finite Field Linear Symmetric Channels (10 pts)*

This problem addresses a symmetric DMC that has the same input and output alphabet, namely a Galois Field with $M' = M = q$ as the size of the field $\mathscr{G}F(q)$. For this problem, this field's important properties are its closure under addition, and existence of a unique additive inverse for each element, and also a unique 0 element. Of course, the symmetric DMC has transition-probability matrix columns that are all permutations of one another.

This problem constructs an argument that the channel also has a description of the form

$$y = x \oplus e$$

where $\oplus$ is the field's addition operation (and $\ominus$ is its inverse). The elements are $0, ..., M-1$. This essentially amounts to adding integers that "'wrap around a circle" at $M = 0$. The signal e is an error or noise signal that is added to the channel input x to obtain the channel output y. The inputs are all equally likely.

a. First, to develop insight, set $M = 4$ and construct a 4×4 probability-transition matrix $P_{\boldsymbol{y/x}}$ using the column elements $0.8, 0.1, 0.05, 0.05$ (1 pt)

b. State the ML decision rule for your matrix in Part a. (1 pt)

c. Repeat Part a so that $e = 1$. And, again for $e = 0$, $e = 2$, and $e = 3$. (4 pts)

d. What can be said of the ML detector's error probability for each of your answers in Part c (1 pt)

e. Sketch a proof for why every symmetric DMC with inputs/outputs in $\mathscr{G}F(M)$ has a linear additive noise model. (2 pts)

f. Write $\max_x p_{y/x}$ in terms of p_e for any y value. (1 pt)

1.58 *Binary Symmetric Channel Creation (10 pts)*

A binary symmetric channel arises from the use of a gray-coded 16SQ QAM constellation (gray code means adjacent constellation points differ in only one bit position).

a. Draw such a 16SQ constellation and label the points with 4 bits each. (2 pts)

b. Find P_b and $\bar{P}_b$ in terms of SNR for the answer in Part a (4 pts)

c. If the AWGN has $SNR = 16$ dB, find the p for the BSC (3 pts)

d. Do you think another bit encoding might occur that leads to a BSC with better error probability? (1 pt)

1.59 *Binary Erasure Channel Creation (10 pts)*

This problem investigates conversion of AWGN Channels with ML PAM/QAM detectors to Binary Erasure Channels. Noise values that cause received channel output samples to be close to decision boundaries are offset by $\pm t/(2\sigma)$ where t is the "width" of the erasure region.

a. Draw the erasure regions for 4PAM and 16SQAM PAM. How many are there for each constellation? (3 pts)

b. Does the different number of erasure regions affect the value of p for the BEC for the two $\bar{b} = 2$ constellations in Part a? (1 pt)

c. Approximate accurately p for both these situations in terms of SNR. (3 pts)

d. If $p_{BEC} = .001$, find the ratio t/d and interpret the answer. (3 pts)

1.60 *Real Rectifier Nonlinear Channel (13 pts)*

The nonlinear channel function $|x(t)|$ is often used in wireless "non-coherent" demodulators known as discriminators as in Chapter 6. This channel also appears (effectively as $|x(t)|^2$ in non-coherent optical-fiber channels also where a photo diode can only measure intensity/power of light impinging upon it). This problem explores application of this chapter's nonlinear model to the this rectifier channel. This rectifier may be followed by additional complex nonlinearity to complete the discriminator receiver as explored further in Problem 1.61.

a. (3 pts) Find the minimum number of domains $|S|$ if the input $x(t)$ is real? What are the corresponding $\Theta_s(x)$? What are the corresponding $\mathcal{R}_s(y)$?

b. (3 pts) Suppose 8PAM with distance $d = 2$ were input to the real channel of Part a. Do any points overlap in the output ranges (without noise)? If so, which points? What is the effective number of messages in this case?

c. (3 pts) What is the percentage rate reduction for the real channel with the input 8 PAM of Part b? In terms of power increase in dB relative to transmission of the same number of messages on a linear channel of $f(x) = x$? Is there a more effective input design for the original nonlinear rectifier channel for symbol-by-symbol ML detection that loses less energy?

d. (2 pts) Describe a ML detector for the nonlinear channel with AWGN for the more effective design of Part c.

e. (2 pts) Create a simple neural-net model for this channel.

1.61 *Complex Amplitude Nonlinear Saturation Channel (10 pts)*

For a complex (inphase/quadrature baseband two-dimensional) input, the nonlinear saturation-channel function

$$f(x) = A \cdot e^{\jmath\theta_x}$$

is sometimes in wireless continuous-phase channels (see Chapter 6). θ_x is the phase of the complex input. A is a positive-real constant. For this staturation channel an amplifier is driven into saturation for maximum energy efficient and all output constellation points (with no noise) will have a constant magnitude A. This problem explores application of this chapter's nonlinear model to the complex-amplitude channel. This channel could cascade with the previous problems real channel to "strip the phase" after the absolute value and provide this phase as an output essentially, which is what the final portion of a discriminator does.

This problem will assume that the amplitude and phase of the complex signal are individually available as real quantities.

a. (2 pts) Find the minimum number of domains $|S|$ if the input $x(t)$ is 8PSK? Find the corresponding domains $\Theta_s(x)$?

b. (2 pts) Find the corresponding $\mathcal{R}_s(y)$.

c. (2 pts) Show a ML detector for this channel with AWGN.

d. (2 pts) Use the available amplitude and phase as inputs to construct a neural net model for this channel.

e. (2 pts) Comment on the difficulty of using the real and imaginary parts instead to do a neural-net model (don't attempt to design this yourself, just comment on what makes it difficult to do by hand).

1.62 *Rayleigh Distribution (4 pts)*

Show that the square root of the sum of the squares of two equal variance independent Gaussian random variables has a Rayleigh distribution. Hint: Use polar coordinates.

1.63 *Coherence Time and Bandwidth (20 pts)*

Employees in a business typically walk at a speed of $v = 1.5$ m/s. A wireless indoor transmission system uses 64 SQ QAM with a carrier frequency of $f_c = 6$ GHz at symbol rate $1/T = 300$ MHz. The average SNR is 27 dB.

a. What is the data rate of the system? (1 pt)

b. Compute and compare the average error probabilities with no fading and with Rayleigh fading. (1 pt)

c. What is the maximum Doppler Frequency f_d?(1 pt)

d. Estimate the coherence time T_Δ. (1 pt)

e. How often might a designer want to insert a known training (or "synchronization") symbol to ensure gain is accurately estimated, perhaps with 1dB accuracy permitted? Is this acceptable in your opinion? (no absolutely correct answer here.) What might you want to do with the constellation? (5 pts)

Now, for coherence bandwidth and refer to Example 1.6.3 to compute the coherence bandwidth for a smaller home. The symbol period is 312.5 μs.

f. Is the first term ($a_k = 10^{-L(d)/20}$) in Equation (1.539) necessary to compute the coherence bandwidth? Why or Why not? (2 pts)

g. Using a value of $K = 1$ for the indoor wireless channel model, find the values of $a_{1,k}$ and $a_{2,k}$ for $k = 0, ..., 8$ for each of Cluster 1 and Cluster 2, by using one standard deviation for the u_k value. Summarize in similar table format. (3 pts)

h. Compute the power-delay profile that includes effects of both clusters in absolute scale (not dB). (2 pts)

i. Compute the RMS delay spread τ_{rms} and coherence bandwidth $W_\Delta$for this channel (2 pts)

j. What does your answer in Part i imply relative to the symbol period. (1 pt)

1.64 *4 × 4 Markov (12 pts)*

A 4 ergodic Markov AWGN channel using QPSK has probability transition matrix for the gain parameter values (in dB) $\in \{11, 12, 14, 20\}$.

$$P_g = \begin{bmatrix} .1 & .8 & .05 & .05 \\ .05 & .1 & .8 & .05 \\ .05 & .05 & .1 & .8 \\ .8 & .05 & .05 & .1 \end{bmatrix}$$

a. Draw the state-machine for this channel including transition probability labels (3 pts)

b. The diagram for Part a's solution is called "fully connected." Explain this name. (1 pt)

c. Find the stationary distribution for g. (2 pts)

d. Find ergodic-average symbol-error probability. (2 pts)

e. The stationary distribution has a special form. Why? (This may combine knowledge from Section 1.4 and Section 1.6.)

1.65 *Exploring Fading Impact (10 pts)*

This problem explores the effect of flat Rayleigh fading on an uncoded 16 QAM transmission with through an AWGN with 21.5 dB SNR. The symbol rate is $1/T = 100$ MHZ with $f_c = 2$ GHz.

a. (2 pts) What is the data rate of this system and symbol error rate if there is no fading?

b. (2 pts) What does theory predict to be the average $< P_e >$=? for this channel?

c. (3 pts) Generate 1000 16SQ QAM symbols, add noise with no fading, and average 10^6 runs to determine $< P_e >$, and compare to Part a.

d. (1 pt) If a vehicle communicates on this channel (with Gaussian noise level remaining constant) and moving at 100 mph directly away from a stationary transmitter (cell site), what is the doppler frequency?

e. (4 pts) Repeat Part c with Part d's doppler shift, using the matlab RayleighChannel object. First, show the path gains progress over the 1000 symbols in a graph for an isolated run. Then run for the dopper channel, but with 10^4 runs. Show the new $< P_e >$ and compare to Part c's simulated answer and Part b's theoretical projection.

Bibliography

[1] J.M. Wozencraft and I.M. Jacobs. *"Principles of Communication Engineering"*. Wiley, New York, 1965.

[2] B.W. Lindgren. *"Statistical Theory, Third Edition"*. Macmillan, New York, 1968.

[3] G. David Forney, Jr. and Lee-Fang Wei. "Multidimensional Constellations-Part I: Introduction, Figures of Merit, and Generalized Cross Constellations" *IEEE Journal on Selected Areas in Communications* Vol 7, No. 6, August 1989, pp. 877-892

[4] K. Mueller and J.J. Werner "A Hardware-Efficient Passband-Equalizer Structure for Data Transmission" *IEEE Transactions on Communications* Vol 30, No. 3, March 1982, pp. 538-541

[5] Oskar Perron (1907) and George Frobenious (1912) "Wikipedia page: The Perron Frobenious Theorem" https://en.wikipedia.org/wiki/Perron\OT1\textendashFrobenius_theorem see the original references at the Wikipedia page

[6] Theodore S. Rappaport *"Wireless Communications: Principles and Practices"*. Prentice Hall, Upper Saddle River, NJ, 1995.

[7] Douglas Lind and Brian Marcus *"Symbolic Dynamics and Coding"*. Cambridge University Press, Cambridge, UK, 1995

[8] Miguel Griot and Wen-Yen Weng and Richard D. Wesel *"A Tighter Bhattacharyya Bound for Decoding Error Probability"*. *IEEE Communications Letters* Vol 11, No. 4, April 2007, pp. 346-347

[9] Thomas Kailath *"The divergence and Bhattacharyya distance measures in signal selection"*. *IEEE Transactions on Communications Technology.* Vol 15, No. 2, February 1967, pp. unknown

Index

χ-square distribution, 144
8AMPM, 56
8SQ, 56

ABC, 126
AWGN
    baseband-equivalent, 103
    channel, 41
    filtered, 93, 115
    Matrix, 122
    vector, 42
awgn, 5

BEC, 128
Bhattacharya Bound, 67
bi-phase level, 67
biorthogonal constellation, 74
bit-error probability
    average, 60
bit-error rate
    average, 59
bit-rate
    ergodic, 143
bits
    per dimension, 62
block orthogonal, 69
block-error rate, 61
bock-error probability, 61
BPSK, 30
BPSK binary phase-shift keying, 26
BSC, 127

cable modem, 82
CAP, 85, 86, 112
capacity, 69
carrier frequency, 94
channel
    2-ray, 106
    baseband equivalent, 99
    binary-erasure, 128
    binary-symmetric, 127
    discrete, 124
    discrete-memoryless, 124
    fading, 143
    nonlinear, 130
    one-shot, 115
    scaled model, 101
    statistically parameterized, 139
    symmetric, 126
    time-varying, 139
    transfer amplitude, 143
channel gain, 145
circular constellations, 75
coding
    gain, 89
    lattice, 88
coding gain
    fundamental, 90
    shaping, 90
coherence, 148
    bandwidth, 148, 150
    spatial, 148, 151
    time, 148
constellation, 7
    figure of merit, 64
continuous approximation, 89
correlation coefficient, 148

data rate, 7, 8, 62
decision
    region, 16
delay, 150
    group, 150
    phase, 150
delay-spread
    root-mean-square, 150
delta
    discrete, 28
demodulation, 36
demodulator, 26
    correlative, 36
    matched-filter., 37
detector
    optimum, 7
discrete data transmission, 4
distance
    minimum, 53
distribution

χ-square, 144
log-normal, 144
exponential, 145
Rayleigh, 144
Ricean, 144
DOCSIS, 82
Doppler frequency, 149

energy
average, 13
peak, 91
per bit, 64
per dimension, 63
erasure, 128
ergodic, 140
erred second, 61
error
event, 23, 121
probability, 14
bit, 22
calculation, 17
optimum, 17
error probabilty
ergodic-average, 141
error-free second, 61
Ethernet, 31
expectation
angle-bracket, 140

factorization
canonical, 118
fading
flat, 144
frequency-selective, 144
Rayleigh, 146
shadow, 144
five nines, 61, 141
Forney, G.D., 87
frequency modulation, 67
frequency modulation code, 31
FSK, 70

gap, 87

hexagonal constellations, 92
Hilbert Transform, 95

inner product, 34, 35
inphase component, 94
invariance
rotational, 49
translational, 51
irrelevancy, 19

Kronecker delta, 28

lattice, 88
line-of-sight, 144, 146
LLR, 23
log likelihood ratio, 23
Log-Normal
distribution, 146

Manchester Coding, 67
Manchester Encoding, 31
MAP Detector, 14
margin, 64
maxium likelihood detector, 15
MIMO, 38
minimax, 15
minimum energy translate, 52
minimum phase, 118
modulated waveform, 29
modulation, 5
baseband, 92
passband, 92
modulator, 26, 29
multipath, 107

nearest neighbors, 57
NNUB
correlated noise, 121
noise
colored, 114
radio-frequency, 114
NRZ, 67

one-shot, 5
OQAM, 86
orthonormal basis, 28
outage, 143
outage probability, 141

PageRank, 157
Paley-Wiener Criterion, 117
PAR, 91
parameter memory, 148
Parseval's Identity, 35
path loss, 144
peak-to-average, 91
phase
distortion
zero, 101
power, 13
power-delay profile, 144, 150
probability
correct, 14
of error, 14
probability transition matrix, 124
PSK, 75
pulse response, 118

pulse-duration modulation, 73
pulse-position modulation, 74

QAM
    OQAM, 85
QPSK, 45, 68
quadrature component, 94
quadrature decomposition, 94
quality of service, 61

Rayleigh
    distribution, 144
    error probability, 145
reversibility, 21, 120
Ricean
    distribution, 146
RZ, 70

sample average, 140
    ergodic, 140
scattering, 144
    macroscopic, 144
    microscopic, 144
    multi-path, 144
    rich, 147, 151
shaping
    gain, 90
    lattice, 88
signal
    baseband-equivalent, 95
    carrier-modulated, 94
    passband, 93
simplex constellation, 72
SNR
    channel, 64, 143
soft decision, 23
soft information, 23
spectral efficiency, 63
speed of light, 149
symbol, 4, 8
    periiod, 7
    perioid, 8
    rate, 7
    vector, 7

union bound, 52
    nearest neighbor, 57, 58

vector
    coding, 151
vector coding, 131
volume
    fundamental, 88
Vornoi
    boundary, 88
Voronoi
    region, 88
    volume, 88
VSB, 85