

Fundamentals of
DIGITAL
COMMUNICATION

Upamanyu Madhow

CAMBRIDGE



Fundamentals of Digital Communication

This textbook presents the fundamental concepts underlying the design of modern digital communication systems, which include the wireline, wireless, and storage systems that pervade our everyday lives. Using a highly accessible, lecture style exposition, this rigorous textbook first establishes a firm grounding in classical concepts of modulation and demodulation, and then builds on these to introduce advanced concepts in synchronization, non-coherent communication, channel equalization, information theory, channel coding, and wireless communication. This up-to-date textbook covers turbo and LDPC codes in sufficient detail and clarity to enable hands-on implementation and performance evaluation, as well as “just enough” information theory to enable computation of performance benchmarks to compare them against. Other unique features include the use of complex baseband representation as a unifying framework for transceiver design and implementation; wireless link design for a number of modulation formats, including space-time communication; geometric insights into noncoherent communication; and equalization. The presentation is self-contained, and the topics are selected so as to bring the reader to the cutting edge of digital communications research and development.

Numerous examples are used to illustrate the key principles, with a view to allowing the reader to perform detailed computations and simulations based on the ideas presented in the text.

With homework problems and numerous examples for each chapter, this textbook is suitable for advanced undergraduate and graduate students of electrical and computer engineering, and can be used as the basis for a one or two semester course in digital communication. It will also be a valuable resource for practitioners in the communications industry.

Additional resources for this title, including instructor-only solutions, are available online at www.cambridge.org/9780521874144.

Upamanyu Madhow is Professor of Electrical and Computer Engineering at the University of California, Santa Barbara. He received his Ph.D. in Electrical Engineering from the University of Illinois, Urbana-Champaign, in 1990, where he later served on the faculty. A Fellow of the IEEE, he worked for several years at Telcordia before moving to academia.

Fundamentals of Digital Communication

Upamanyu Madhow
University of California, Santa Barbara



CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

Published in the United States of America by Cambridge University Press, New York

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning and research at the highest international levels of excellence.

www.cambridge.org

Information on this title: www.cambridge.org/9780521874144

© Cambridge University Press 2008

This publication is in copyright. Subject to statutory exception and to the provisions of relevant collective licensing agreements, no reproduction of any part may take place without the written permission of Cambridge University Press.

First published 2008

A catalogue record for this publication is available from the British Library

ISBN 978-0-521-87414-4 Hardback

Cambridge University Press has no responsibility for the persistence or accuracy of URLs for external or third-party Internet websites referred to in this publication, and does not guarantee that any content on such websites is, or will remain, accurate or appropriate.

Introduction

We define communication as information transfer between different points in space or time, where the term *information* is loosely employed to cover standard formats that we are all familiar with, such as voice, audio, video, data files, web pages, etc. Examples of communication between two points in space include a telephone conversation, accessing an Internet website from our home or office computer, or tuning in to a TV or radio station. Examples of communication between two points in time include accessing a storage device, such as a record, CD, DVD, or hard drive. In the preceding examples, the information transferred is directly available for human consumption. However, there are many other communication systems, which we do not directly experience, but which form a crucial part of the infrastructure that we rely upon in our daily lives. Examples include high-speed packet transfer between routers on the Internet, inter- and intra-chip communication in integrated circuits, the connections between computers and computer peripherals (such as keyboards and printers), and control signals in communication networks.

In *digital* communication, the information being transferred is represented in digital form, most commonly as binary digits, or *bits*. This is in contrast to *analog* information, which takes on a continuum of values. Most communication systems used for transferring information today are either digital, or are being converted from analog to digital. Examples of some recent conversions that directly impact consumers include cellular telephony (from analog FM to several competing digital standards), music storage (from vinyl records to CDs), and video storage (from VHS or beta tapes to DVDs). However, we typically consume information in analog form; for example, reading a book or a computer screen, listening to a conversation or to music. Why, then, is the world going digital? We consider this issue after first discussing the components of a typical digital communication system.

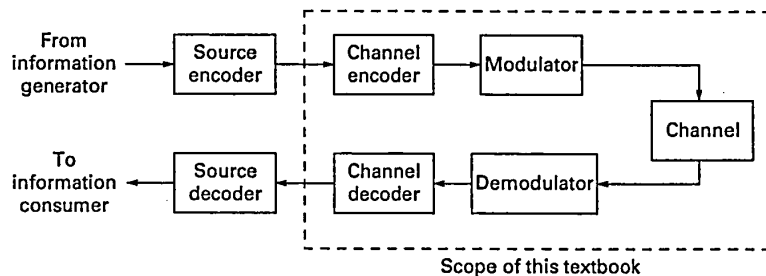
1.1 Components of a digital communication system

Consider the block diagram of a digital communication link depicted in Figure 1.1. Let us now briefly discuss the roles of the blocks shown in the figure.

Source encoder Information theory tells us that any information can be efficiently represented in digital form up to arbitrary precision, with the number of bits required for the representation depending on the required fidelity. The task of the source encoder is to accomplish this in a practical setting, reducing the redundancy in the original information in a manner that takes into account the end user's requirements. For example, voice can be intelligibly encoded into a 4 kbit/s bitstream for severely bandwidth constrained settings, or sent at 64 kbit/s for conventional wireline telephony. Similarly, audio encoding rates have a wide range – MP3 players for consumer applications may employ typical bit rates of 128 kbit/s, while high-end digital audio studio equipment may require around ten times higher bit rates. While the preceding examples refer to lossy source coding (in which a controlled amount of information is discarded), lossless compression of data files can also lead to substantial reductions in the amount of data to be transmitted.

Channel encoder and modulator While the source encoder eliminates unwanted redundancy in the information to be sent, the channel encoder introduces redundancy in a controlled fashion in order to combat errors that may arise from channel imperfections and noise. The output of the channel encoder is a codeword from a channel code, which is designed specifically for the anticipated channel characteristics and the requirements dictated by higher network layers. For example, for applications that are delay insensitive, the channel code may be optimized for error detection, followed by a request for retransmission. On the other hand, for real-time applications for which retransmissions are not possible, the channel code may be optimized for error correction. Often, a combination of error correction and detection may be employed. The modulator translates the discrete symbols output by the channel code into an analog waveform that can be transmitted over the

Figure 1.1 Block diagram of a digital communication link.



1.1 Components of a digital communication system

physical channel. The physical channel for an 802.11b based wireless local area network link is, for example, a band of 20 MHz width at a frequency of approximately 2.4 GHz. For this example, the modulator translates a bitstream of rate 1, 2, 5.5, or 11 Mbit/s (the rate varies, depending on the channel conditions) into a waveform that fits within the specified 20 MHz frequency band.

Channel The physical characteristics of communication channels can vary widely, and good channel models are critical to the design of efficient communication systems. While receiver thermal noise is an impairment common to most communication systems, the channel distorts the transmitted waveform in a manner that may differ significantly in different settings. For wireline communication, the channel is well modeled as a linear time-invariant system, and the transfer function in the band used by the modulator can often be assumed to be known at the transmitter, based on feedback obtained from the receiver at the link set-up phase. For example, in high-speed digital subscriber line (DSL) systems over twisted pairs, such channel feedback is exploited to send more information at frequencies at which the channel gain is larger. On the other hand, for wireless mobile communication, the channel may vary because of relative mobility between the transmitter and receiver, which affects both transmitter design (accurate channel feedback is typically not available) and receiver design (the channel must either be estimated, or methods that do not require accurate channel estimates must be used). Further, since wireless is a broadcast medium, multiple-access interference due to simultaneous transmissions must be avoided either by appropriate resource sharing mechanisms, or by designing signaling waveforms and receivers to provide robust performance in the presence of interference.

Demodulator and channel decoder The demodulator processes the analog received waveform, which is a distorted and noisy version of the transmitted waveform. One of its key tasks is synchronization: the demodulator must account for the fact that the channel can produce phase, frequency, and time shifts, and that the clocks and oscillators at the transmitter and receiver are not synchronized a priori. Another task may be channel equalization, or compensation of the intersymbol interference induced by a dispersive channel. The ultimate goal of the demodulator is to produce tentative decisions on the transmitted symbols to be fed to the channel decoder. These decisions may be “hard” (e.g., the demodulator guesses that a particular bit is 0 or 1), or “soft” (e.g., the demodulator estimates the likelihood of a particular bit being 0 or 1). The channel decoder then exploits the redundancy in the channel to code to improve upon the estimates from the demodulator, with its final goal being to produce an estimate of the sequence of information symbols that were the input to the channel encoder. While the demodulator and decoder operate independently in traditional receiver designs, recent advances in coding and

communication theory show that iterative information exchange between the demodulator and the decoder can dramatically improve performance.

Source decoder The source decoder converts the estimated information bits produced by the channel decoder into a format that can be used by the end user. This may or may not be the same as the original format that was the input to the source encoder. For example, the original source encoder could have translated speech into text, and then encoded it into bits, and the source decoder may then display the text to the end user, rather than trying to reproduce the original speech.

We are now ready to consider why the world is going digital. The two key advantages of the digital communication approach to the design of transmission and storage media are as follows:

Source-independent design Once information is transformed into bits by the source encoder, it can be stored or transmitted without interpretation: as long as the bits are recovered, the information they represent can be reconstructed with the same degree of precision as originally encoded. This means that the storage or communication medium can be independent of the source characteristics, so that a variety of information sources can share the same communication medium. This leads to significant economies of scale in the design of individual communication links as well as communication networks comprising many links, such as the Internet. Indeed, when information has to traverse multiple communication links in a network, the source encoding and decoding in Figure 1.1 would typically be done at the end points alone, with the network transporting the information bits put out by the source encoder without interpretation.

Channel-optimized design For each communication link, the channel encoder or decoder and modulator or demodulator can be optimized for the specific channel characteristics. Since the bits being transported are regenerated at each link, there is no “noise accumulation.”

The preceding framework is based on a separation of source coding and channel coding. Not only does this *separation principle* yield practical advantages as mentioned above, but we are also reassured by the source-channel separation theorem of information theory that it is theoretically optimal for point-to-point links (under mild conditions). While the separation approach is critical to obtaining the economies of scale driving the growth of digital communication systems, we note in passing that joint source and channel coding can yield superior performance, both in theory and practice, in certain settings (e.g., multiple-access and broadcast channels, or applications with delay or complexity constraints).

The scope of this textbook is indicated in Figure 1.1: we consider modulation and demodulation, channel encoding and decoding, and channel modeling.

Source encoding and decoding are not covered. Thus, we implicitly restrict attention to communication systems based on the separation principle.

1.2 Text outline

The objective of this text is to convey an understanding of the principles underlying the design of a modern digital communication link. An introduction to modulation techniques (i.e., how to convert bits into a form that can be sent over a channel) is provided in Chapter 2. We emphasize the important role played by the complex baseband representation for passband signals in both transmitter and receiver design, describe some common modulation formats, and discuss how to determine how much bandwidth is required to support a given modulation format. An introduction to demodulation (i.e., how to estimate the transmitted bits from a noisy received signal) for the classical additive white Gaussian noise (AWGN) channel is provided in Chapter 3. Our starting point is the theory of hypothesis testing. We emphasize the geometric view of demodulation first popularized by the classic text of Wozencraft and Jacobs, introduce the concept of soft decisions, and provide a brief exposure to link budget analysis (which is used by system designers for determining parameters such as antenna gains and transmit powers). Mastery of Chapters 2 and 3 is a prerequisite for the remainder of this book. The remaining chapters essentially stand on their own. Chapter 4 contains a framework for estimation of parameters such as delay and phase, starting from the derivation of the likelihood ratio of a signal in AWGN. Optimal noncoherent receivers are derived based on this framework. Chapter 5 describes the key ideas used in channel equalization, including maximum likelihood sequence estimation (MLSE) using the Viterbi algorithm, linear equalization, and decision feedback equalization. Chapter 6 contains a brief treatment of information theory, focused on the *computation* of performance benchmarks. This is increasingly important for the communication system designer, now that turbo-like codes provide a framework for approaching information-theoretic limits for virtually any channel model. Chapter 7 introduces error-correction coding. It includes convolutional codes, serial and parallel concatenated turbo codes, and low density parity check (LDPC) codes. It also provides a very brief discussion of how algebraic codes (which are covered in depth in coding theory texts) fit within modern communication link design, with an emphasis on Reed–Solomon codes. Finally, Chapter 8 contains an introduction to wireless communication, including channel modeling, the effect of fading, and a discussion of some modulation formats commonly used over the wireless channel that are not covered in the introductory treatment in Chapter 2. The latter include orthogonal frequency division multiplexing (OFDM), spread spectrum communication, continuous phase modulation, and space–time (or multiple antenna) communication.

1.3 Further reading

Useful resources for getting a quick exposure to many topics on communication systems are *The Communications Handbook* [1] and *The Mobile Communications Handbook* [2], both edited by Gibson. Standards for communication systems are typically available online from organizations such as the Institute for Electrical and Electronics Engineers (IEEE). Recently published graduate-level textbooks on digital communication include Proakis [3], Benedetto and Biglieri [4], and Barry, Lee, and Messerschmitt [5]. Undergraduate texts on communications include Haykin [6], Proakis and Salehi [7], Pursley [8], and Ziemer and Tranter [9]. Classical texts of enduring value include Wozencraft and Jacobs [10], which was perhaps the first textbook to introduce signal space design techniques, Viterbi [11], which provides detailed performance analysis of demodulation and synchronization techniques, Viterbi and Omura [12], which provides a rigorous treatment of modulation and coding, and Blahut [13], which provides an excellent perspective on the concepts underlying digital communication systems.

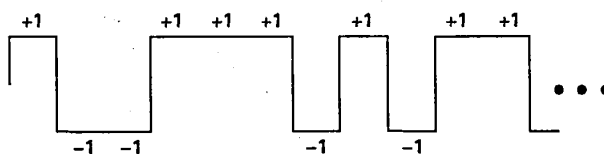
We do not cover source coding in this text. An information-theoretic treatment of source coding is provided in Cover and Thomas [14], while a more detailed description of compression algorithms is found in Sayood [15].

Finally, while this text deals with the design of individual communication links, the true value of these links comes from connecting them together to form communication networks, such as the Internet, the wireline phone network, and the wireless cellular communication network. Two useful texts on communication networks are Bertsekas and Gallager [16] and Walrand and Varaiya [17]. On a less technical note, Friedman [18] provides an interesting discussion on the immense impact of advances in communication networking on the global economy.

Modulation refers to the representation of digital information in terms of analog waveforms that can be transmitted over physical channels. A simple example is depicted in Figure 2.1, where a sequence of bits is translated into a waveform. The original information may be in the form of bits taking the values 0 and 1. These bits are translated into symbols using a bit-to-symbol map, which in this case could be as simple as mapping the bit 0 to the symbol $+1$, and the bit 1 to the symbol -1 . These symbols are then mapped to an analog waveform by multiplying with translates of a transmit waveform (a rectangular pulse in the example shown): this is an example of *linear modulation*, to be discussed in detail in Section 2.5. For the bit-to-symbol map just described, the bitstream encoded into the analog waveform shown in Figure 2.1 is 01100010100.

While a rectangular timelimited transmit waveform is shown in the example of Figure 2.1, in practice, the analog waveforms employed for modulation are often constrained in the frequency domain. Such constraints arise either from the physical characteristics of the communication medium, or from external factors such as government regulation of spectrum usage. Thus, we typically classify channels, and the signals transmitted over them, in terms of the frequency bands they occupy. In this chapter, we discuss some important modulation techniques, after first reviewing some basic concepts regarding frequency domain characterization of signals and systems. The material in this chapter is often covered in detail in introductory digital communication texts,

Figure 2.1 A simple example of binary modulation.



but we emphasize some specific points in somewhat more detail than usual. One of these is the complex baseband representation of passband signals, which is a crucial tool both for understanding and implementing modern communication systems. Thus, the reader who is familiar with this material is still encouraged to skim through this chapter.

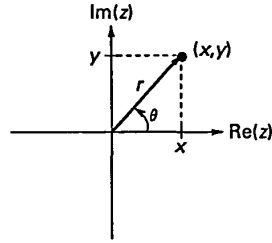
Map of this chapter In Section 2.1, we review basic notions such as the frequency domain representation of signals, inner products between signals, and the concept of baseband and passband signals. While currents and voltages in a circuit are always real-valued, both baseband and passband signals can be treated under a unified framework by allowing baseband signals to take on complex values. This complex baseband representation of passband signals is developed in Section 2.2, where we point out that manipulation of complex baseband signals is an essential component of modern transceivers. While the preceding development is for deterministic, finite energy signals, modeling of signals and noise in digital communication relies heavily on finite power, random processes. We therefore discuss frequency domain description of random processes in Section 2.3. This completes the background needed to discuss the main theme of this chapter: modulation. Section 2.4 briefly discusses the degrees of freedom available for modulation, and introduces the concept of bandwidth efficiency. Section 2.5 covers linear modulation using two-dimensional constellations, which, in principle, can utilize all available degrees of freedom in a bandlimited channel. The Nyquist criterion for avoidance of intersymbol interference (ISI) is discussed, in order to establish guidelines relating bandwidth to bit rate. Section 2.6 discusses orthogonal and biorthogonal modulation, which are nonlinear modulation formats optimized for power efficiency. Finally, Section 2.7 discusses differential modulation as a means of combating phase uncertainty. This concludes our introduction to modulation. Several other modulation formats are discussed in Chapter 8, where we describe some modulation techniques commonly employed in wireless communication.

2.1 Preliminaries

This section contains a description of just enough material on signals and systems for our purpose in this text, including the definitions of inner product, norm and energy for signals, convolution, Fourier transform, and baseband and passband signals.

Complex numbers A complex number z can be written as $z = x + jy$, where x and y are real numbers, and $j = \sqrt{-1}$. We say that $x = \text{Re}(z)$ is the real part of z and $y = \text{Im}(z)$ is the imaginary part of z . As depicted in Figure 2.2, it is often advantageous to interpret the complex number z as

Figure 2.2 A complex number z represented in the two-dimensional real plane.



a two-dimensional real vector, which can be represented in rectangular form as $(x, y) = (\text{Re}(z), \text{Im}(z))$, or in polar form as

$$r = |z| = \sqrt{x^2 + y^2},$$

$$\theta = \arg(z) = \tan^{-1} \frac{y}{x}.$$

Euler's identity We routinely employ this to decompose a complex exponential into real-valued sinusoids as follows:

$$e^{j\theta} = \cos \theta + j \sin \theta. \quad (2.1)$$

A key building block of communication theory is the relative geometry of the signals used, which is governed by the inner products between signals. Inner products for continuous-time signals can be defined in a manner exactly analogous to the corresponding definitions in finite-dimensional vector space.

Inner product The inner product for two $m \times 1$ complex vectors $\mathbf{s} = (s[1], \dots, s[m])^T$ and $\mathbf{r} = (r[1], \dots, r[m])^T$ is given by

$$\langle \mathbf{s}, \mathbf{r} \rangle = \sum_{i=1}^m s[i] r^*[i] = \mathbf{r}^H \mathbf{s}. \quad (2.2)$$

Similarly, we define the inner product of two (possibly complex-valued) signals $s(t)$ and $r(t)$ as follows:

$$\langle s, r \rangle = \int_{-\infty}^{\infty} s(t) r^*(t) dt. \quad (2.3)$$

The inner product obeys the following linearity properties:

$$\langle a_1 s_1 + a_2 s_2, r \rangle = a_1 \langle s_1, r \rangle + a_2 \langle s_2, r \rangle,$$

$$\langle s, a_1 r_1 + a_2 r_2 \rangle = a_1^* \langle s, r_1 \rangle + a_2^* \langle s, r_2 \rangle,$$

where a_1, a_2 are complex-valued constants, and s, s_1, s_2, r, r_1, r_2 are signals (or vectors). The complex conjugation when we pull out constants from the second argument of the inner product is something that we need to remain aware of when computing inner products for complex signals.

Energy and norm The energy E_s of a signal s is defined as its inner product with itself:

$$E_s = \|s\|^2 = \langle s, s \rangle = \int_{-\infty}^{\infty} |s(t)|^2 dt, \quad (2.4)$$

where $\|s\|$ denotes the *norm* of s . If the energy of s is zero, then s must be zero “almost everywhere” (e.g., $s(t)$ cannot be nonzero over any interval, no matter how small its length). For continuous-time signals, we take this to be equivalent to being zero everywhere. With this understanding, $\|s\| = 0$ implies that s is zero, which is a property that is true for norms in finite-dimensional vector spaces.

Cauchy–Schwartz inequality The inner product obeys the *Cauchy–Schwartz inequality*, stated as follows:

$$|\langle s, r \rangle| \leq \|s\| \|r\|, \quad (2.5)$$

with equality if and only if, for some complex constant a , $s(t) = ar(t)$ or $r(t) = as(t)$ almost everywhere. That is, equality occurs if and only if one signal is a scalar multiple of the other. The proof of this inequality is given in Problem 2.4.

Convolution The convolution of two signals s and r gives the signal

$$q(t) = (s * r)(t) = \int_{-\infty}^{\infty} s(u)r(t-u)du.$$

Here, the convolution is evaluated at time t , while u is a “dummy” variable that is integrated out. However, it is sometimes convenient to abuse notation and use $q(t) = s(t) * r(t)$ to denote the convolution between s and r . For example, this enables us to state compactly the following linear time invariance (LTI) property:

$$(a_1 s_1(t-t_1) + a_2 s_2(t-t_2)) * r(t) = a_1 (s_1 * r)(t-t_1) + a_2 (s_2 * r)(t-t_2),$$

for any complex gains a_1 and a_2 , and any time offsets t_1 and t_2 .

Delta function The delta function $\delta(t)$ is defined via the following “sifting” property: for any finite energy signal $s(t)$, we have

$$\int_{-\infty}^{\infty} \delta(t-t_0)s(t)dt = s(t_0). \quad (2.6)$$

In particular, this implies that convolution of a signal with a shifted version of the delta function gives a shifted version of the signal:

$$\delta(t-t_0) * s(t) = s(t-t_0). \quad (2.7)$$

Equation (2.6) can be shown to imply that $\delta(0) = \infty$ and $\delta(t) = 0$ for $t \neq 0$. Thus, thinking of the delta function as a signal is a convenient abstraction, since it is not physically realizable.

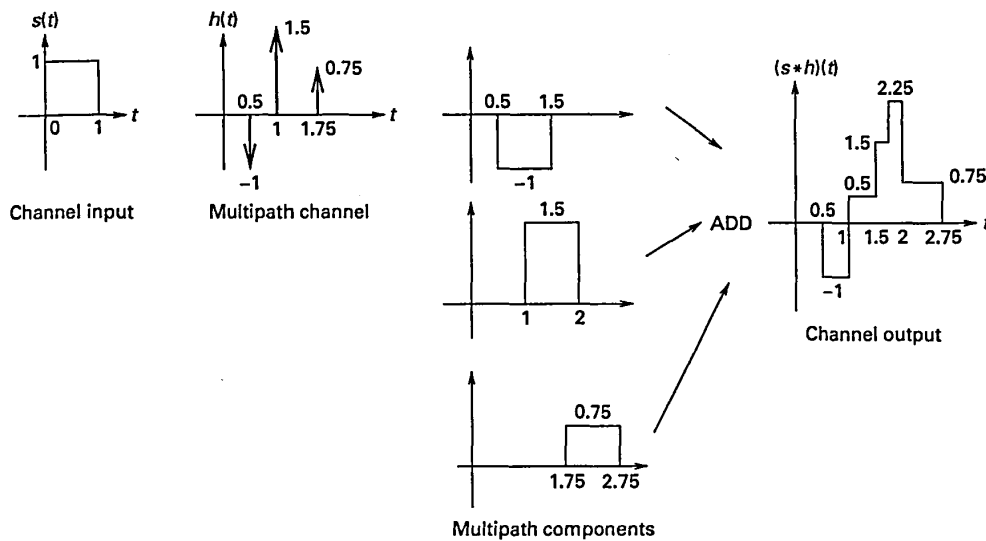


Figure 2.3 A signal going through a multipath channel.

Convolution plays a fundamental role in both modeling and transceiver implementation in communication systems, as illustrated by the following examples.

Example 2.1.1 (Modeling a multipath channel) The *channel* between the transmitter and the receiver is often modeled as an LTI system, with the received signal y given by

$$y(t) = (s * h)(t) + n(t),$$

where s is the transmitted waveform, h is the channel impulse response, and $n(t)$ is receiver thermal noise and interference. Suppose that the channel impulse response is given by

$$h(t) = \sum_{i=1}^M a_i \delta(t - t_i).$$

Ignoring the noise, a signal $s(t)$ passing through such a channel produces an output

$$y(t) = (s * h)(t) = \sum_{i=1}^M a_i s(t - t_i).$$

This could correspond, for example, to a wireless *multipath* channel in which the transmitted signal is reflected by a number of scatterers, each of which gives rise to a copy of the signal with a different delay and scaling. Typically, the results of propagation studies would be used to

obtain statistical models for the number of multipath components M , the delays $\{t_i\}$, and the gains $\{a_i\}$.

Example 2.1.2 (Matched filter) For a complex-valued signal $s(t)$, the matched filter is defined as a filter with impulse response $s_{MF}(t) = s^*(-t)$; see Figure 2.4 for an example. Note that $S_{MF}(f) = S^*(f)$. If the input to the matched filter is $x(t)$, then the output is given by

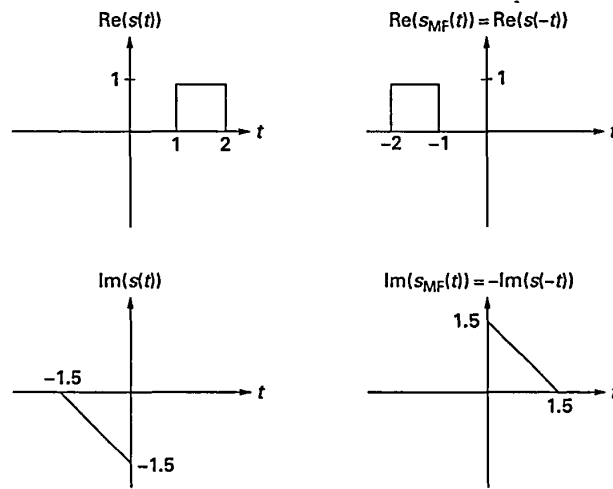


Figure 2.4 Matched filter for a complex-valued signal.

$$y(t) = (x * s_{MF})(t) = \int_{-\infty}^{\infty} x(u) s_{MF}(t-u) du = \int_{-\infty}^{\infty} x(u) s^*(u-t) du. \quad (2.8)$$

The matched filter, therefore, computes the inner product between the input x and all possible time translates of the waveform s , which can be interpreted as “template matching.” In particular, the inner product $\langle x, s \rangle$ equals the output of the matched filter at time 0. Some properties of the matched filter are explored in Problem 2.5. For example, if $x(t) = s(t - t_0)$ (i.e., the input is a time translate of s), then, as shown in Problem 2.5, the magnitude of the matched filter output is maximum at $t = t_0$. We can, then, intuitively see how the matched filter would be useful, for example, in delay estimation using “peak picking.” In later chapters, a more systematic development is used to reveal the key role played by the matched filter in digital communication receivers.

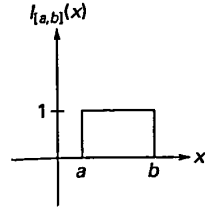


Figure 2.5 The indicator function of an interval has a boxcar shape.

Indicator function We use I_A to denote the indicator function of a set A , defined as

$$I_A(x) = \begin{cases} 1, & x \in A, \\ 0, & \text{otherwise.} \end{cases}$$

For example, the indicator function of an interval has a boxcar shape, as shown in Figure 2.5.

Sinc function The sinc function is defined as

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x},$$

where the value at $x = 0$, defined as the limit as $x \rightarrow 0$, is set as $\text{sinc}(0) = 1$. The sinc function is shown in Figure 2.19. Since $|\sin(\pi x)| \leq 1$, we have that $|\text{sinc}(x)| \leq (1/\pi x)$. That is, the sinc function exhibits a sinusoidal variation, with an envelope that decays as $1/x$. We plot the sinc function later in this chapter, in Figure 2.19, when we discuss linear modulation.

Fourier transform Let $s(t)$ denote a signal, and $S(f) = \mathcal{F}(s(t))$ denote its Fourier transform, defined as

$$S(f) = \int_{-\infty}^{\infty} s(t) e^{-j2\pi ft} dt. \quad (2.9)$$

The inverse Fourier transform is given by

$$s(t) = \int_{-\infty}^{\infty} S(f) e^{j2\pi ft} df. \quad (2.10)$$

Both $s(t)$ and $S(f)$ are allowed to take on complex values. We denote the relationship that $s(t)$ and $S(f)$ are a *Fourier transform pair* by $s(t) \leftrightarrow S(f)$.

Time-frequency duality in Fourier transform From an examination of the expressions (2.9) and (2.10), we obtain the following duality relation: if $s(t)$ has Fourier transform $S(f)$, then the signal $r(t) = S(t)$ has Fourier transform $R(f) = s(-f)$.

Important Fourier transform pairs

(i) The boxcar and the sinc functions form a pair:

$$s(t) = I_{[-\frac{T}{2}, \frac{T}{2}]}(t) \leftrightarrow S(f) = T \text{sinc}(fT). \quad (2.11)$$

(ii) The delta function and the constant function form a pair:

$$s(t) = \delta(t) \leftrightarrow S(f) \equiv 1. \quad (2.12)$$

We list only two pairs here, because most of the examples that we use in our theoretical studies can be derived in terms of these, using time-frequency duality and the properties of the Fourier transform below. On the other hand, closed form analytical expressions are not available for

many waveforms encountered in practice, and the Fourier or inverse Fourier transform is computed numerically using the discrete Fourier transform (DFT) in the sampled domain.

Basic properties of the Fourier transform Some properties of the Fourier transform that we use extensively are as follows (it is instructive to derive these starting from the definition (2.9)):

- (i) Complex conjugation in the time domain corresponds to conjugation and reflection around the origin in the frequency domain, and vice versa;

$$\begin{aligned} s^*(t) &\leftrightarrow S^*(-f), \\ s^*(-t) &\leftrightarrow S^*(f). \end{aligned} \quad (2.13)$$

- (ii) A signal $s(t)$ is real-valued (i.e., $s(t) = s^*(t)$) if and only if its Fourier transform is conjugate symmetric (i.e., $S(f) = S^*(-f)$). Note that conjugate symmetry of $S(f)$ implies that $\text{Re}(S(f)) = \text{Re}(S(-f))$ (real part is symmetric) and $\text{Im}(S(f)) = -\text{Im}(S(-f))$ (imaginary part is antisymmetric).

- (iii) Convolution in the time domain corresponds to multiplication in the frequency domain, and vice versa;

$$\begin{aligned} s(t) &= (s_1 * s_2)(t) \leftrightarrow S(f) = S_1(f)S_2(f), \\ s(t) &= s_1(t)s_2(t) \leftrightarrow S(f) = (S_1 * S_2)(f). \end{aligned} \quad (2.14)$$

- (iv) Translation in the time domain corresponds to multiplication by a complex exponential in the frequency domain, and vice versa;

$$\begin{aligned} s(t - t_0) &\leftrightarrow S(f)e^{-j2\pi f t_0}, \\ s(t)e^{j2\pi f_0 t} &\leftrightarrow S(f - f_0). \end{aligned} \quad (2.15)$$

- (v) Time scaling leads to reciprocal frequency scaling;

$$s(at) \leftrightarrow \frac{1}{|a|} S\left(\frac{f}{a}\right). \quad (2.16)$$

- (vi) *Parseval's identity* The inner product of two signals can be computed in either the time or frequency domain, as follows:

$$\langle s_1, s_2 \rangle = \int_{-\infty}^{\infty} s_1(t)s_2^*(t)dt = \int_{-\infty}^{\infty} S_1(f)S_2^*(f)df = \langle S_1, S_2 \rangle. \quad (2.17)$$

Setting $s_1 = s_2 = s$, we obtain the following expression for the energy E_s of a signal $s(t)$:

$$E_s = \|s\|^2 = \int_{-\infty}^{\infty} |s(t)|^2 dt = \int_{-\infty}^{\infty} |S(f)|^2 df. \quad (2.18)$$

2.1 Preliminaries

Energy spectral density The energy spectral density $E_s(f)$ of a signal $s(t)$ can be defined operationally as follows. Pass the signal $s(t)$ through an ideal narrowband filter with transfer function;

$$H_{f_0}(f) = \begin{cases} 1, & f_0 - \frac{\Delta f}{2} < f < f_0 + \frac{\Delta f}{2}, \\ 0, & \text{else.} \end{cases}$$

The energy spectral density $E_s(f_0)$ is defined to be the energy at the output of the filter, divided by the width Δf (in the limit as $\Delta f \rightarrow 0$). That is, the energy at the output of the filter is approximately $E_s(f_0)\Delta f$. But the Fourier transform of the filter output is

$$Y(f) = S(f)H(f) = \begin{cases} S(f), & f_0 - \frac{\Delta f}{2} < f < f_0 + \frac{\Delta f}{2}, \\ 0, & \text{else.} \end{cases}$$

By Parseval's identity, the energy at the output of the filter is

$$\int_{-\infty}^{\infty} |Y(f)|^2 df = \int_{f_0 - \frac{\Delta f}{2}}^{f_0 + \frac{\Delta f}{2}} |S(f)|^2 df \approx |S(f_0)|^2 \Delta f,$$

assuming that $S(f)$ varies smoothly and Δf is small enough. We can now infer that the energy spectral density is simply the magnitude squared of the Fourier transform:

$$E_s(f) = |S(f)|^2. \quad (2.19)$$

The integral of the energy spectral density equals the signal energy, which is simply a restatement of Parseval's identity.

Autocorrelation function The inverse Fourier transform of the energy spectral density $E_s(f)$ is termed the autocorrelation function $R_s(\tau)$, since it measures how closely the signal s matches delayed versions of itself. Since $|S(f)|^2 = S(f)S^*(f) = S(f)S_{MF}^*(f)$, where $s_{MF}(t) = s^*(-t)$ is the matched filter for s introduced earlier. We therefore have that

$$E_s(f) = |S(f)|^2 \leftrightarrow R_s(\tau) = (s * s_{MF})(\tau) = \int_{-\infty}^{\infty} s(u)s^*(u - \tau) du. \quad (2.20)$$

Thus, $R_s(\tau)$ is the outcome of passing the signal s through its matched filter, and sampling the output at time τ , or equivalently, correlating the signal s with a complex conjugated version of itself, delayed by τ .

While the preceding definitions are for finite energy deterministic signals, we revisit these concepts in the context of finite power random processes later in this chapter.

Baseband and passband signals A signal $s(t)$ is said to be *baseband* if

$$S(f) \approx 0, \quad |f| > W \quad (2.21)$$

for some $W > 0$. That is, the signal energy is concentrated in a band around DC. Similarly, a channel modeled as a linear time-invariant system is said to be baseband if its transfer function $H(f)$ satisfies (2.21).

A signal $s(t)$ is said to be *passband* if

$$S(f) \approx 0, \quad |f \pm f_c| > W \quad (2.22)$$

where $f_c > W > 0$. A channel modeled as a linear time-invariant system is said to be passband if its transfer function $H(f)$ satisfies (2.22).

Examples of baseband and passband signals are shown in Figures 2.6 and 2.7, respectively. We consider real-valued signals, since any signal that has a physical realization in terms of a current or voltage must be real-valued. As shown, the Fourier transforms can be complex-valued, but they must satisfy the conjugate symmetry condition $S(f) = S^*(-f)$. The bandwidth B is defined to be the size of the frequency interval occupied by $S(f)$, where we consider only the spectral occupancy for the positive frequencies

Figure 2.6 Example of the spectrum $S(f)$ for a real-valued baseband signal. The bandwidth of the signal is B .

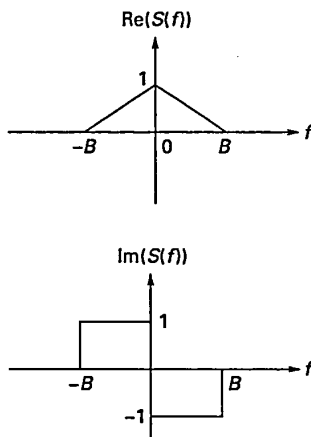
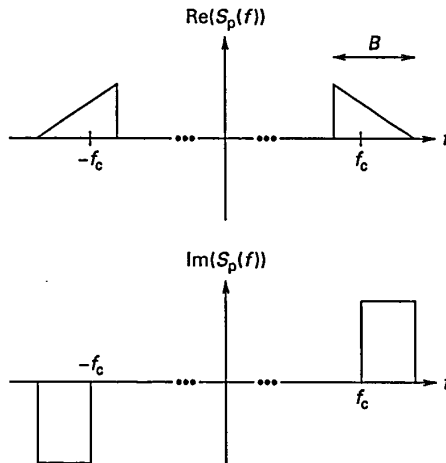


Figure 2.7 Example of the spectrum $S(f)$ for a real-valued passband signal. The bandwidth of the signal is B . The figure shows an arbitrarily chosen frequency f_c within the band in which $S(f)$ is nonzero. Typically, f_c is much larger than the signal bandwidth B .



for a real-valued signal $s(t)$. This makes sense from a physical viewpoint: after all, when the FCC allocates a frequency band to an application, say, around 2.4 GHz for unlicensed usage, it specifies the positive frequencies that can be occupied. However, in order to be clear about the definition being used, we occasionally employ the more specific term *one-sided* bandwidth, and also define the *two-sided* bandwidth based on the spectral occupancy for both positive and negative frequencies. For real-valued signals, the two-sided bandwidth is simply twice the one-sided bandwidth, because of the conjugate symmetry condition $S(f) = S^*(-f)$. However, when we consider the complex baseband representation of real-valued passband signals in the next section, the complex-valued signals which we consider do not, in general, satisfy the conjugate symmetry condition, and there is no longer a deterministic relationship between the two-sided and one-sided bandwidths. As we show in the next section, a real-valued passband signal has an equivalent representation as a complex-valued baseband signal, and the (one-sided) bandwidth of the passband signal equals the two-sided bandwidth of its complex baseband representation.

In Figures 2.6 and 2.7, the spectrum is shown to be exactly nonzero outside a well defined interval, and the bandwidth B is the size of this interval. In practice, there may not be such a well defined interval, and the bandwidth depends on the specific definition employed. For example, the bandwidth might be defined as the size of an appropriately chosen interval in which a specified fraction (say 99%) of the signal energy lies.

Example 2.1.3 (Fractional energy containment bandwidth) Consider a rectangular time domain pulse $s(t) = I_{[0,T]}$. Using (2.11) and (2.15), the Fourier transform of this signal is given by $S(f) = T \text{sinc}(fT)e^{-j\pi fT}$, so that

$$|S(f)|^2 = T^2 \text{sinc}^2(fT).$$

Clearly, there is no finite frequency interval that contains all of the signal energy. Indeed, it follows from a general *uncertainty* principle that strictly timelimited signals cannot be strictly bandlimited, and vice versa. However, most of the energy of the signal is concentrated around the origin, so that $s(t)$ is a baseband signal. We can now define the (one-sided) fractional energy containment bandwidth B as follows:

$$\int_{-B}^B |S(f)|^2 df = a \int_{-\infty}^{\infty} |S(f)|^2 df, \quad (2.23)$$

where $0 < a \leq 1$ is the fraction of energy contained in the band $[-B, B]$. The value of B must be computed numerically, but there are certain simplifications that are worth pointing out. First, note that T can be set to any convenient value, say $T = 1$ (equivalently, one unit of time is redefined to be T). By virtue of the scaling property (2.16), time scaling leads to

reciprocal frequency scaling. Thus, if the bandwidth for $T = 1$ is B_1 , then the bandwidth for arbitrary T must be $B_T = B_1 / T$. This holds regardless of the specific notion of bandwidth used, since the scaling property can be viewed simply as redefining the unit of frequency in a consistent manner with the change in our unit for time. The second observation is that the right-hand side of (2.23) can be evaluated in closed form using Parseval's identity (2.18). Putting these observations together, it is left as an exercise for the reader to show that (2.23) can be rewritten as

$$\int_{-B_1}^{B_1} \text{sinc}^2 f \, df = a, \quad (2.24)$$

which can be further simplified to

$$\int_0^{B_1} \text{sinc}^2 f \, df = \frac{a}{2}, \quad (2.25)$$

using the symmetry of the integrand around the origin. We can now evaluate B_1 numerically for a given value of a . We obtain $B_1 = 10.2$ for $a = 0.99$, and $B_1 = 0.85$ for $a = 0.9$. Thus, while the 90% energy containment bandwidth is moderate, the 99% energy containment bandwidth is large, because of the slow decay of the sinc function. For an arbitrary value of T , the 99% energy containment bandwidth is $B = 10.2/T$.

A technical note: (2.24) could also be inferred from (2.23) by applying a change of variables, replacing fT in (2.23) by f . This change of variables is equivalent to the scaling argument that we invoked.

2.2 Complex baseband representation

We often employ passband channels, which means that we must be able to transmit and receive passband signals. We now show that all the information carried in a real-valued passband signal is contained in a corresponding complex-valued baseband signal. This baseband signal is called the *complex baseband representation*, or *complex envelope*, of the passband signal. This equivalence between passband and complex baseband has profound practical significance. Since the complex envelope can be represented accurately in discrete time using a much smaller sampling rate than the corresponding passband signal $s_p(t)$, modern communication transceivers can implement complicated signal processing algorithms digitally on complex baseband signals, keeping the analog processing of passband signals to a minimum. Thus, the transmitter encodes information into the complex baseband waveform using encoding, modulation and filtering performed using digital signal processing (DSP). The complex baseband waveform is then *upconverted* to the corresponding passband signal to be sent on the channel. Similarly, the passband received waveform is *downconverted* to complex baseband by the receiver, followed

2.2 Complex baseband representation

by DSP operations for synchronization, demodulation, and decoding. This leads to a modular framework for transceiver design, in which sophisticated algorithms can be developed in complex baseband, independent of the physical frequency band that is ultimately employed for communication.

We now describe in detail the relation between passband and complex baseband, and the relevant transceiver operations. Given the importance of being comfortable with complex baseband, the pace of the development here is somewhat leisurely. For a reader who knows this material, quickly browsing this section to become familiar with the notation should suffice.

Time domain representation of a passband signal Any passband signal $s_p(t)$ can be written as

$$s_p(t) = \sqrt{2}s_c(t) \cos 2\pi f_c t - \sqrt{2}s_s(t) \sin 2\pi f_c t, \quad (2.26)$$

where $s_c(t)$ ("c" for "cosine") and $s_s(t)$ ("s" for "sine") are real-valued signals, and f_c is a frequency reference typically chosen in or around the band occupied by $S_p(f)$. The factor of $\sqrt{2}$ is included only for convenience in normalization (more on this later), and is often omitted in the literature.

In-phase and quadrature components The waveforms $s_c(t)$ and $s_s(t)$ are also referred to as the in-phase (or I) component and the quadrature (or Q) component of the passband signal $s_p(t)$, respectively.

Example 2.2.1 (Passband signal) The signal

$$s_p(t) = \sqrt{2}I_{[0,1]}(t) \cos 300\pi t - \sqrt{2}(1-|t|)I_{[-1,1]}(t) \sin 300\pi t$$

is a passband signal with I component $s_c(t) = I_{[0,1]}(t)$ and Q component $s_s(t) = (1-|t|)I_{[-1,1]}(t)$. Like Example 2.1.3, this example also illustrates that we do not require strict bandwidth limitations in our definitions of passband and baseband: the I and Q components are timelimited, and hence cannot be bandlimited. However, they are termed baseband signals because most of their energy lies in the baseband. Similarly, $s_p(t)$ is termed a passband signal, since most of its frequency content lies in a small band around 150 Hz.

Complex envelope The complex envelope, or complex baseband representation, of $s_p(t)$ is now defined as

$$s(t) = s_c(t) + js_s(t). \quad (2.27)$$

In the preceding example, the complex envelope is given by $s(t) = I_{[0,1]}(t) + j(1-|t|)I_{[-1,1]}(t)$.

Time domain relationship between passband and complex baseband We can rewrite (2.26) as

$$s_p(t) = \text{Re}(\sqrt{2}s(t)e^{j2\pi f_c t}). \quad (2.28)$$

To check this, plug in (2.27) and Euler's identity (2.1) on the right-hand side to obtain the expression (2.26).

Envelope and phase of a passband signal The complex envelope $s(t)$ can also be represented in polar form, defining the *envelope* $e(t)$ and *phase* $\theta(t)$ as

$$e(t) = |s(t)| = \sqrt{s_c^2(t) + s_s^2(t)}, \quad \theta(t) = \tan^{-1} \frac{s_s(t)}{s_c(t)}. \quad (2.29)$$

Plugging $s(t) = e(t)e^{j\theta(t)}$ into (2.28), we obtain yet another formula for the passband signal s :

$$s_p(t) = e(t) \cos(2\pi f_c t + \theta(t)). \quad (2.30)$$

The equations (2.26), (2.28) and (2.30) are three different ways of expressing the same relationship between passband and complex baseband in the time domain.

Example 2.2.2 (Modeling frequency or phase offsets in complex baseband) Consider the passband signal s_p (2.26), with complex baseband representation $s = s_c + js_s$. Now, consider a phase-shifted version of the passband signal

$$\tilde{s}_p(t) = \sqrt{2}s_c(t) \cos(2\pi f_c t + \theta(t)) - \sqrt{2}s_s(t) \sin(2\pi f_c t + \theta(t)),$$

where $\theta(t)$ may vary slowly with time. For example, a carrier frequency offset a and a phase offset b corresponds to $\theta(t) = 2\pi at + b$. We wish to find the complex envelope of $\tilde{s}_p$ with respect to f_c . To do this, we write $\tilde{s}_p$ in the standard form (2.28) as follows:

$$\tilde{s}_p(t) = \text{Re}(\sqrt{2}\tilde{s}(t)e^{j(2\pi f_c t + \theta(t))}).$$

Comparing with the desired form

$$\tilde{s}_p(t) = \text{Re}(\sqrt{2}\tilde{s}(t)e^{j2\pi f_c t}),$$

we can read off

$$\tilde{s}(t) = s(t)e^{j\theta(t)}. \quad (2.31)$$

Equation (2.31) relates the complex envelopes before and after a phase offset. We can expand out this "polar form" representation to obtain the corresponding relationship between the I and Q components. Suppressing time dependence from the notation, we can rewrite (2.31) as

$$\tilde{s}_c + j\tilde{s}_s = (s_c + js_s)(\cos \theta + j \sin \theta)$$

2.2 Complex baseband representation

using Euler's formula. Equating real and imaginary parts on both sides, we obtain

$$\begin{aligned}\tilde{s}_c &= s_c \cos \theta - s_s \sin \theta, \\ \tilde{s}_s &= s_c \sin \theta + s_s \cos \theta.\end{aligned}\tag{2.32}$$

This is a typical example of the advantage of working in complex baseband. Relationships between passband signals can be compactly represented in complex baseband, as in (2.31). For signal processing using real-valued arithmetic, these complex baseband relationships can be expanded out to obtain relationships involving real-valued quantities, as in (2.32).

Orthogonality of I and Q channels The passband waveform $x_c(t) = \sqrt{2}s_c(t) \cos 2\pi f_c t$ corresponding to the I component, and the passband waveform $x_s(t) = \sqrt{2}s_s(t) \sin 2\pi f_c t$ corresponding to the Q component, are orthogonal. That is,

$$\langle x_c, x_s \rangle = 0.\tag{2.33}$$

Since what we know about s_c and s_s (i.e., they are baseband) is specified in the frequency domain, we prove this result by computing the inner product in the frequency domain, using Parseval's identity (2.17):

$$\langle x_c, x_s \rangle = \langle X_c, X_s \rangle = \int_{-\infty}^{\infty} X_c(f) X_s^*(f) df.$$

We now need expressions for X_c and X_s . Since $\cos \theta = \frac{1}{2}(e^{j\theta} + e^{-j\theta})$ and $\sin \theta = \frac{1}{2j}(e^{j\theta} - e^{-j\theta})$ we have

$$x_c(t) = \frac{1}{\sqrt{2}}(s_c(t)e^{j2\pi f_c t} + s_c(t)e^{-j2\pi f_c t}) \leftrightarrow X_c(f) = \frac{1}{\sqrt{2}}(S_c(f - f_c) + S_c(f + f_c)),$$

$$x_s(t) = \frac{1}{\sqrt{2}j}(s_s(t)e^{j2\pi f_c t} - s_s(t)e^{-j2\pi f_c t}) \leftrightarrow X_s(f) = \frac{1}{\sqrt{2}j}(S_s(f - f_c) - S_s(f + f_c)).$$

The inner product can now be computed as follows:

$$\langle X_c, X_s \rangle = \frac{1}{2j} \int_{-\infty}^{\infty} [S_c(f - f_c) + S_c(f + f_c)] [S_s^*(f - f_c) - S_s^*(f + f_c)] df.\tag{2.34}$$

We now look more closely at the integrand above. Since f_c is assumed to be larger than the bandwidth of the baseband signals S_c and S_s , the translation of $S_c(f)$ to the right by f_c has zero overlap with a translation of $S_s^*(f)$ to the left by f_c . That is, $S_c(f - f_c)S_s^*(f + f_c) \equiv 0$. Similarly, $S_c(f + f_c)S_s^*(f - f_c) \equiv 0$. We can therefore rewrite the inner product in (2.34) as

$$\begin{aligned}\langle X_c, X_s \rangle &= \frac{1}{2j} \left[\int_{-\infty}^{\infty} S_c(f - f_c)S_s^*(f - f_c) df - \int_{-\infty}^{\infty} S_c(f + f_c)S_s^*(f + f_c) df \right] \\ &= \frac{1}{2j} \left[\int_{-\infty}^{\infty} S_c(f)S_s^*(f) df - \int_{-\infty}^{\infty} S_c(f)S_s^*(f) df \right] = 0,\end{aligned}\tag{2.35}$$

where we have used a change of variables to show that the integrals involved cancel out.

Exercise 2.2.1 Work through the details of an alternative, shorter, proof of (2.33) as follows. Show that

$$u(t) = x_c(t)x_s(t) = 2s_c(t)s_s(t) \cos 2\pi f_c t \sin 2\pi f_c t$$

is a passband signal (around what frequency?), and thus infer that

$$\int_{-\infty}^{\infty} u(t) dt = U(0) = 0.$$

Passband and complex baseband inner products For real passband signals a_p and b_p with complex envelopes a and b , respectively, the inner product satisfies

$$\langle u_p, v_p \rangle = \langle u_c, v_c \rangle + \langle u_s, v_s \rangle = \text{Re}(\langle u, v \rangle). \quad (2.36)$$

To show the first equality, we substitute the standard form (2.26) for u_p and v_p and use the orthogonality of the I and Q components. For the second equality, we write out the complex inner product $\langle u, v \rangle$,

$$\begin{aligned} \langle u, v \rangle &= \int_{-\infty}^{\infty} (u_c(t) + ju_s(t))(v_c(t) - jv_s(t)) dt \\ &= (\langle u_c, v_c \rangle + \langle u_s, v_s \rangle) + j(-\langle u_c, v_s \rangle + \langle u_s, v_c \rangle), \end{aligned} \quad (2.37)$$

and note that the real part gives the desired term.

Energy of complex envelope Specializing (2.36) to the inner product of a signal with itself, we infer that the energy of the complex envelope is equal to that of the corresponding passband signal (this is a convenient consequence of the specific scaling we have chosen in our definition of the complex envelope). That is,

$$\|s\|^2 = \|s_p\|^2. \quad (2.38)$$

To show this, set $u = v = s$ and $u_p = v_p = s_p$ in (2.36), noting that $\text{Re}(\langle s, s \rangle) = \text{Re}(\|s\|^2) = \|s\|^2$.

Frequency domain relationship between passband and complex baseband We first summarize the results relating $S_p(f)$ and $S(f)$. Let $S_p^+(f) = S_p(f)I_{f>0}$ denote the segment of $S_p(f)$ occupying positive frequencies. Then the complex envelope is specified as

$$S(f) = \sqrt{2}S_p^+(f + f_c). \quad (2.39)$$

Conversely, given the complex envelope $S(f)$ in the frequency domain, the passband signal is specified as

$$S_p(f) = \frac{S(f - f_c) + S^*(-f - f_c)}{\sqrt{2}}. \quad (2.40)$$

2.2 Complex baseband representation

We now derive and discuss these relationships. Define

$$v(t) = \sqrt{2}s(t)e^{j2\pi f_c t} \leftrightarrow V(f) = \sqrt{2}S(f - f_c). \quad (2.41)$$

By the time domain relationship between s_p and s , we have

$$\begin{aligned} s_p(t) &= \text{Re}(v(t)) = \frac{v(t) + v^*(t)}{2} \leftrightarrow S_p(f) = \frac{V(f) + V^*(-f)}{2} \\ &= \frac{S(f - f_c) + S^*(-f - f_c)}{\sqrt{2}}. \end{aligned} \quad (2.42)$$

If $S(f)$ has energy concentrated in the baseband, then the energy of $V(f)$ is concentrated around f_c , and the energy of $V^*(-f)$ is concentrated around $-f_c$. Thus, $S_p(f)$ is indeed passband. We also see from (2.42) that the symmetry condition $S_p(f) = S_p^*(-f)$ holds, which implies that $s_p(t)$ is real-valued. This is, of course, not surprising, since our starting point was the time domain expression (2.28) for a real-valued signal $s_p(t)$.

Figure 2.8 shows the relation between the passband signal $S_p(f)$, its scaled version $V(f)$ restricted to positive frequencies, and the complex baseband signal $S(f)$. As this example emphasizes, all of these spectra can, in general, be complex-valued. Equation (2.41) corresponds to starting with an arbitrary

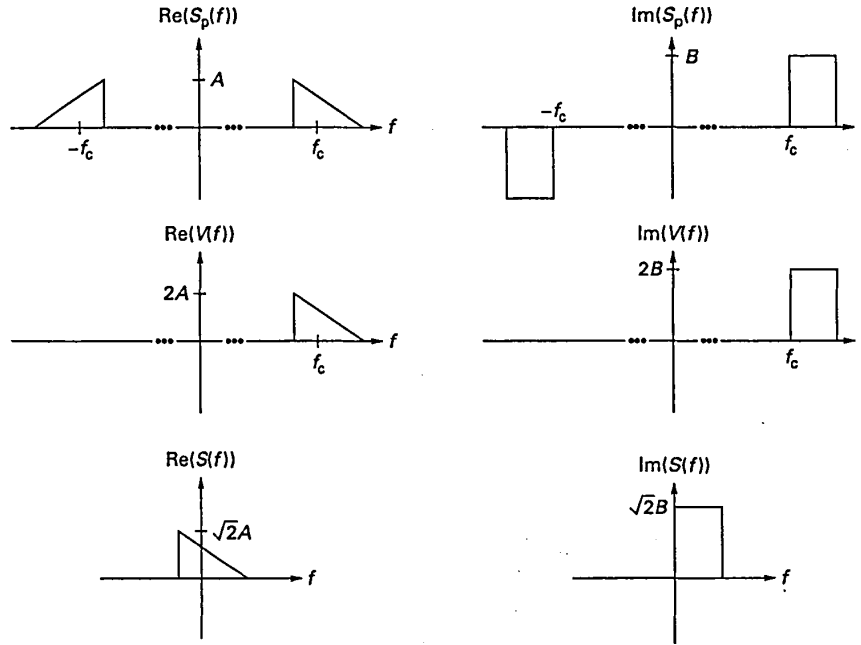


Figure 2.8 Frequency domain relationship between a real-valued passband signal and its complex envelope. The figure shows the spectrum $S_p(f)$ of the passband signal, its scaled restriction to positive frequencies $V(f)$, and the spectrum $S(f)$ of the complex envelope.

baseband signal $S(f)$ as in the bottom of the figure, and constructing $V(f)$ as depicted in the middle of the figure. We then use $V(f)$ to construct a conjugate symmetric passband signal $S_p(f)$, proceeding from the middle of the figure to the top. This example also shows that $S(f)$ does not, in general, obey conjugate symmetry, so that the baseband signal $s(t)$ is complex-valued. However, by construction, $S_p(f)$ is conjugate symmetric, and hence the passband signal $s_p(t)$ is real-valued.

General applicability of complex baseband representation We have so far seen that, given a complex baseband signal (or equivalently, a pair of real baseband signals), we can generate a real-valued passband signal using (2.26) or (2.28). But do these baseband representations apply to any real-valued passband signal? To show that they indeed do apply, we simply reverse the frequency domain operations in (2.41) and (2.42). Specifically, suppose that $s_p(t)$ is an arbitrary real-valued passband waveform. This means that the conjugate symmetry condition $S_p(f) = S_p^*(-f)$ holds, so that knowing the values of S_p for positive frequencies is enough to characterize the values for all frequencies. Let us therefore consider an appropriately scaled version of the segment of S_p for positive frequencies, defined as

$$V(f) = 2S_p^+(f) = \begin{cases} 2S_p(f), & f > 0, \\ 0, & \text{else.} \end{cases} \quad (2.43)$$

By the definition of V , and using the conjugate symmetry of S_p , we see that (2.42) holds. Note also that, since S_p is passband, the energy of V is concentrated around $+f_c$. Now, let us define the complex envelope of S_p by inverting the relation (2.41), as follows:

$$S(f) = \frac{1}{\sqrt{2}} V(f + f_c). \quad (2.44)$$

Since $V(f)$ is concentrated around $+f_c$, $S(f)$, which is obtained by translating it to the left by f_c , is baseband. Thus, starting from an arbitrary passband signal $S_p(f)$, we have obtained a baseband signal $S(f)$ that satisfies (2.41) and (2.42), which are equivalent to the time domain relationship (2.28). We refer again to Figure 2.8 to illustrate the relation between $S_p(f)$, $V(f)$ and $S(f)$. However, we now go from top to bottom: starting from an arbitrary conjugate symmetric $S_p(f)$, we construct $V(f)$, and then $S(f)$.

Upconversion and downconversion Equation (2.26) immediately tells us how to *upconvert* from baseband to passband. To *downconvert* from passband to baseband, consider

$$\begin{aligned} \sqrt{2}s_p(t) \cos(2\pi f_c t) &= 2s_c(t) \cos^2 2\pi f_c t - 2s_s(t) \sin 2\pi f_c t \cos 2\pi f_c t \\ &= s_c(t) + s_c(t) \cos 4\pi f_c t - s_s(t) \sin 4\pi f_c t. \end{aligned}$$

The first term on the extreme right-hand side is the I component, a baseband signal. The second and third terms are passband signals at $2f_c$, which we can

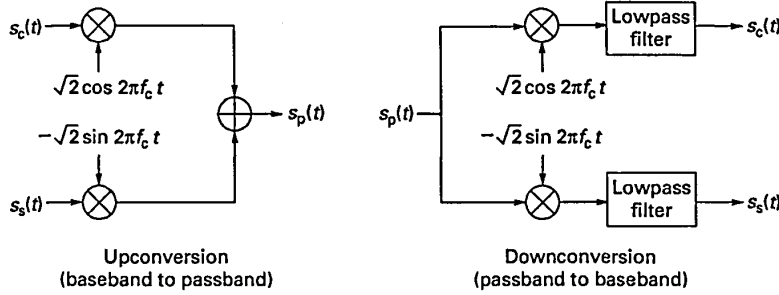


Figure 2.9 Upconversion from baseband to passband and downconversion from passband to baseband.

get rid of by lowpass filtering. Similarly, we can obtain the Q component by lowpass filtering $-\sqrt{2}s_p(t)\sin 2\pi f_c t$. The upconversion and downconversion operations are depicted in Figure 2.9.

Information resides in complex baseband The complex baseband representation corresponds to subtracting out the rapid, but predictable, phase variation due to the fixed reference frequency f_c , and then considering the much slower amplitude and phase variations induced by baseband modulation. Since the phase variation due to f_c is predictable, it cannot convey any information. Thus, all the information in a passband signal is contained in its complex envelope.

Example 2.2.3 (Linear modulation) Suppose that information is encoded into a complex number $b = b_c + jb_s = re^{j\theta}$, where b_c , b_s are real-valued numbers corresponding to its rectangular form, and $r \geq 0$, θ are real-valued and correspond to its polar form. Let $p(t)$ denote a baseband pulse (for simplicity, assume that p is real-valued). Then the *linearly modulated* complex baseband waveform $s(t) = bp(t)$ can be used to convey the information in b over a passband channel by upconverting to an arbitrary carrier frequency f_c . The corresponding passband signal is given by

$$\begin{aligned} s_p(t) &= \operatorname{Re} \left(\sqrt{2}s(t)e^{j2\pi f_c t} \right) = \sqrt{2}(b_c p(t) \cos 2\pi f_c t - b_s p(t) \sin 2\pi f_c t) \\ &= \sqrt{2}r \cos(2\pi f_c t + \theta). \end{aligned}$$

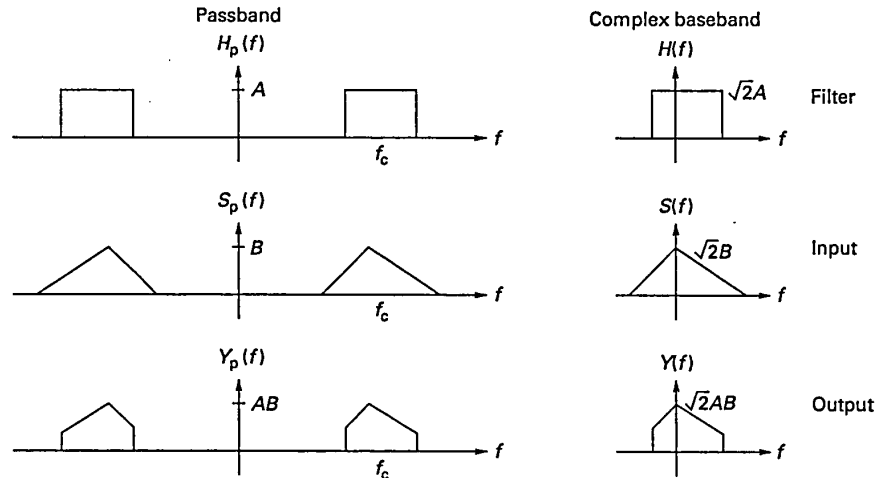
Thus, linear modulation in complex baseband by a complex symbol b can be viewed as separate amplitude modulation (by b_c , b_s) of the I component and the Q component, or as amplitude and phase modulation (by r , θ) of the overall passband waveform. In practice, we encode information in a stream of complex symbols $\{b[n]\}$ that linearly modulate time shifts of a basic waveform, and send the complex baseband waveform $\sum_n b[n]p(t - nT)$. Linear modulation is discussed in detail in Section 2.5.

Complex baseband equivalent of passband filtering We now state another result that is extremely relevant to transceiver operations; namely, any passband filter can be implemented in complex baseband. This result applies to filtering operations that we desire to perform at the transmitter (e.g., to conform to spectral masks), at the receiver (e.g., to filter out noise), and to a broad class of channels modeled as linear filters. Suppose that a passband signal $s_p(t)$ is passed through a passband filter with impulse response $h_p(t)$. Denote the filter output (which is clearly also passband) by $y_p(t) = (s_p * h_p)(t)$. Let y , s and h denote the complex envelopes for y_p , s_p and h_p , respectively, with respect to a common frequency reference f_c . Since real-valued passband signals are completely characterized by their behavior for positive frequencies, the passband filtering equation $Y_p(f) = S_p(f)H_p(f)$ can be separately (and redundantly) written out for positive and negative frequencies, because the waveforms are conjugate symmetric around the origin, and there is no energy around $f = 0$. Thus, focusing on the positive frequency segments $Y_+(f) = Y_p(f)I_{\{f>0\}}$, $S_+(f) = S_p(f)I_{\{f>0\}}$, $H_+(f) = H_p(f)I_{\{f>0\}}$, we have $Y_+(f) = S_+(f)H_+(f)$, from which we conclude that the complex envelope of y is given by

$$Y(f) = \sqrt{2}Y_+(f + f_c) = \sqrt{2}S_+(f + f_c)H_+(f + f_c) = \frac{1}{\sqrt{2}}S(f)H(f).$$

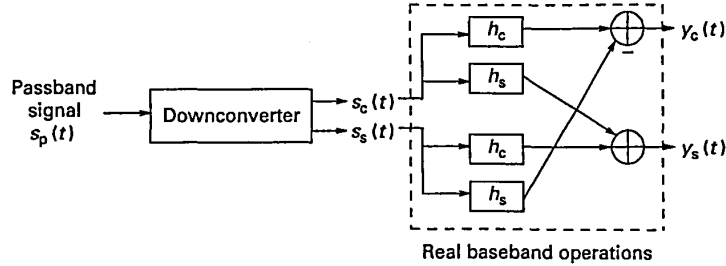
Figure 2.10 The relationship between passband filtering and its complex baseband analog.

Figure 2.10 depicts the relationship between the passband and complex baseband waveforms in the frequency domain, and supplies a pictorial proof of the preceding relationship. We now restate this important result in the time domain:



2.2 Complex baseband representation

Figure 2.11 Complex baseband realization of passband filter. The constant scale factors of $1/\sqrt{2}$ have been omitted.



$$y(t) = \frac{1}{\sqrt{2}}(s * h)(t). \quad (2.45)$$

That is, passband filtering can be implemented in complex baseband, using the complex baseband representation of the desired filter impulse response. As shown in Figure 2.11, this requires four real baseband filters: writing out the real and imaginary parts of (2.45), we obtain

$$y_c = \frac{1}{\sqrt{2}}(s_c * h_c - s_s * h_s), \quad y_s = \frac{1}{\sqrt{2}}(s_s * h_c + s_c * h_s). \quad (2.46)$$

Remark 2.2.1 (Complex baseband in transceiver implementations)

Given the equivalence of passband and complex baseband, and the fact that key operations such as linear filtering can be performed in complex baseband, it is understandable why, in typical modern passband transceivers, most of the intelligence is moved to baseband processing. For moderate bandwidths at which analog-to-digital and digital-to-analog conversion can be accomplished inexpensively, baseband operations can be efficiently performed in DSP. These digital algorithms are independent of the passband over which communication eventually occurs, and are amenable to a variety of low-cost implementations, including very large scale integrated circuits (VLSI), field programmable gate arrays (FPGA), and general purpose DSP engines. On the other hand, analog components such as local oscillators, power amplifiers, and low noise amplifiers must be optimized for the bands of interest, and are often bulky. Thus, the trend in modern transceivers is to accomplish as much as possible using baseband DSP algorithms. For example, complicated filters shaping the transmitted waveform to a spectral mask dictated by the FCC can be achieved with baseband DSP algorithms, allowing the use of relatively sloppy analog filters at passband. Another example is the elimination of analog phase locked loops for carrier synchronization in many modern receivers; the receiver instead employs a fixed analog local oscillator for downconversion, followed by a digital phase locked loop implemented in complex baseband.

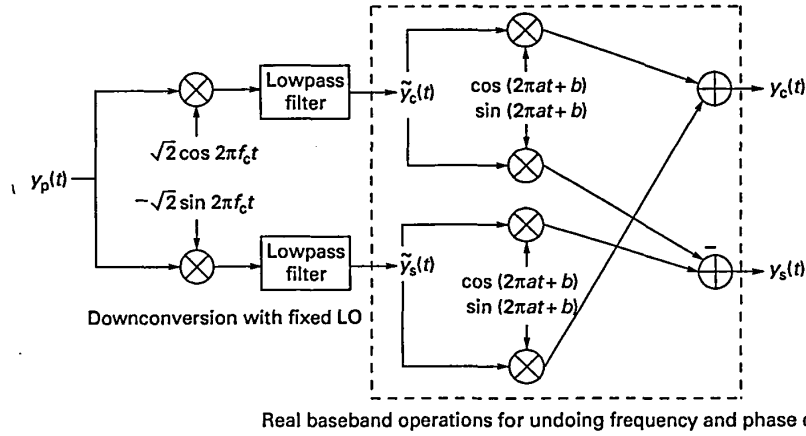


Figure 2.12 Undoing frequency and phase offsets in complex baseband after downconverting using a local oscillator at a fixed carrier frequency f_c . The complex baseband operations are expanded out into real arithmetic as shown.

Example 2.2.4 (Handling carrier frequency and phase offsets in complex baseband) As shown in Figure 2.12, a communication receiver uses a local oscillator with a fixed carrier frequency f_c to demodulate an incoming passband signal

$$y_p(t) = \sqrt{2}[y_c(t)\cos(2\pi(f_c + a)t + b) - y_s(t)\sin(2\pi(f_c + a)t + b)],$$

where a, b , are carrier frequency and phase offsets, respectively. Denote the I and Q components at the output of the downconverter as $\tilde{y}_c, \tilde{y}_s$, respectively, and the corresponding complex envelope as $\tilde{y} = \tilde{y}_c + j\tilde{y}_s$. We wish to recover y_c, y_s , the I and Q components relative to a reference that accounts for the offsets a and b . Typically, the receiver would estimate a and b using the downconverter output $\tilde{y}$; an example of an algorithm for such frequency and phase synchronization is discussed in the next chapter. Assuming that such estimates are available, we wish to specify baseband operations using real-valued arithmetic for obtaining y_c, y_s from the downconverter output. Equivalently, we wish to recover the complex envelope $y = y_c + jy_s$ from $\tilde{y}$. We can relate y and $\tilde{y}$ via (2.31) as in Example 2.2.2, and obtain

$$\tilde{y}(t) = y(t)e^{j(2\pi at + b)}.$$

This relation can now be inverted to get y from $\tilde{y}$:

$$y(t) = \tilde{y}(t)e^{-j(2\pi at + b)}.$$

Plugging in I and Q components explicitly and using Euler's formula, we obtain

$$y_c(t) + jy_s(t) = (\tilde{y}_c(t) + j\tilde{y}_s(t)) (\cos(2\pi at + b) - j\sin(2\pi at + b)).$$

Equating real and imaginary parts, we obtain equations involving real-valued quantities alone:

$$\begin{aligned} y_c &= \tilde{y}_c \cos(2\pi at + b) + \tilde{y}_s \sin(2\pi at + b) \\ y_s &= -\tilde{y}_c \sin(2\pi at + b) + \tilde{y}_s \cos(2\pi at + b). \end{aligned} \quad (2.47)$$

These computations are depicted in Figure 2.12.

Example 2.2.5 (Coherent and noncoherent reception) We see in the next two chapters that a fundamental receiver operation is to compare a noisy received signal against noiseless copies of the received signals corresponding to the different possible transmitted signals. This comparison is implemented by a correlation, or inner product. Let $y_p(t) = \sqrt{2}\text{Re}(y(t)e^{j2\pi f_c t})$ denote the noisy received passband signal, and $s_p(t) = \sqrt{2}\text{Re}(s(t)e^{j2\pi f_c t})$ denote a noiseless copy that we wish to compare it with, where $y = y_c + jy_s$ and $s = s_c + js_s$ are the complex envelopes of y_p and s_p , respectively. A coherent receiver (which is a building block for the optimal receivers in Chapter 3) for s implements the inner product $\langle y_p, s_p \rangle$. In terms of complex envelopes, we know from (2.36) that this can be written as

$$\langle y_p, s_p \rangle = \text{Re}(\langle y, s \rangle) = \langle y_c, s_c \rangle + \langle y_s, s_s \rangle. \quad (2.48)$$

Clearly, when $y = As$ (plus noise), where $A > 0$ is an arbitrary amplitude scaling, the coherent receiver gives a large output. However, coherent reception assumes carrier phase synchronization (in order to separate out and compute inner products with the I and Q components of the received passband signal). If, on the other hand, the receiver is not synchronized in phase, then (see Example 2.2.2) the complex envelope of the received signal is given by $y = Ae^{j\theta}s$ (plus noise), where $A > 0$ is the amplitude scale factor, and θ is an unknown carrier phase. Now, the coherent receiver gives the output

$$\begin{aligned} \langle y_p, s_p \rangle &= \text{Re}(\langle Ae^{j\theta}s, s \rangle) \text{ (plus noise)} \\ &= A \cos \theta \|s\|^2 \text{ (plus noise)}. \end{aligned}$$

In this case, the output can be large or small, depending on the value of θ . Indeed, for $\theta = \pi/2$, the signal contribution to the inner product becomes zero. The noncoherent receiver deals with this problem by using the magnitude, rather than the real part, of the complex inner

product $\langle y, s \rangle$. The signal contribution to this noncoherent correlation is given by

$$|\langle y, s \rangle| = |\langle Ae^{j\theta} s, s \rangle \text{ (plus noise)}| \approx A \|s\|^2 \text{ (ignoring noise),}$$

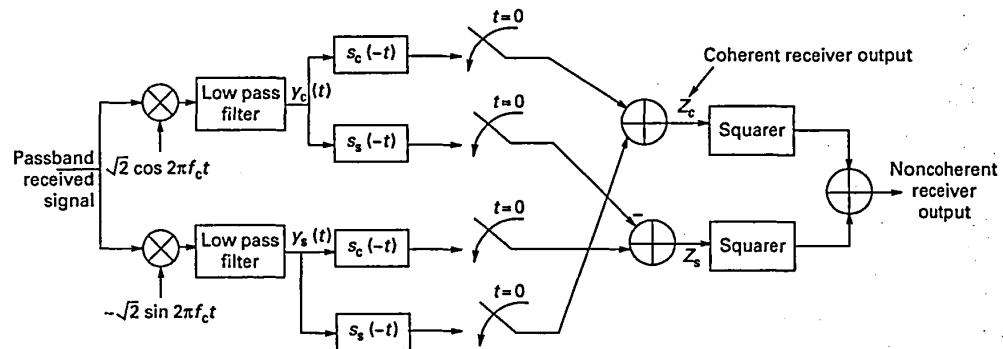
where we have omitted the terms arising from the nonlinear interaction between noise and signal in the noncoherent correlator output. The noiseless output from the preceding computation shows that we do get a large signal contribution regardless of the value of the carrier phase θ . It is convenient to square the magnitude of the complex inner product for computation. Substituting the expression (2.37) for the complex inner product, we obtain that the squared magnitude inner product computed by a noncoherent receiver requires the following real baseband computations:

$$|\langle y, s \rangle|^2 = (\langle u_c, v_c \rangle + \langle u_s, v_s \rangle)^2 + (-\langle u_c, v_s \rangle + \langle u_s, v_c \rangle)^2. \quad (2.49)$$

We see in Chapter 4 that the preceding computations are a building block for optimal noncoherent demodulation under carrier phase uncertainty. The implementations of the coherent and noncoherent receivers in complex baseband are shown in Figure 2.13.

Remark 2.2.2 (Bandwidth) Given the scarcity of spectrum and the potential for interference between signals operating in neighboring bands, determining the spectral occupancy of signals accurately is an important part of communication system design. As mentioned earlier, the spectral occupancy of a physical (and hence real-valued) signal is the smallest band of positive frequencies that contains most of the signal content. Negative frequencies are not included in the definition, since they contain no information beyond that already contained in the positive frequencies ($S(-f) = S^*(f)$ for real-valued $s(t)$). For complex baseband signals, however, information resides in both positive and negative frequencies, since the complex baseband representation is a translated version of the corresponding passband signal restricted to

Figure 2.13 Complex baseband implementations of coherent and noncoherent receivers. The real-valued correlations are performed using matched filters sampled at time zero.



positive frequencies. We therefore define the spectral occupancy of a complex baseband signal as the smallest band around the origin, including both positive and negative frequencies, that contains most of the signal content. These definitions of bandwidth are consistent: from Figure 2.8, it is evident that the bandwidth of a passband signal (defined based on positive frequencies alone) is equal to the bandwidth of its complex envelope (defined based on both positive and negative frequencies). Thus, it suffices to work in complex baseband when determining the spectral occupancy and bandwidth of passband signals.

2.3 Spectral description of random processes

So far, we have considered deterministic signals with finite energy. From the point of view of communication system design, however, it is useful to be able to handle random signals, and to allow the signal energy to be infinite. For example, consider the binary signaling example depicted in Figure 2.1. We would like to handle bitstreams of arbitrary length within our design framework, and would like our design to be robust to which particular bitstream was sent. We therefore model the bitstream as random (and demand good system performance averaged over these random realizations), which means that the modulated signal is modeled as a random process. Since the bitstream can be arbitrarily long, the energy of the modulated signal is unbounded. On the other hand, when averaged over a long interval, the power of the modulated signal in Figure 2.1 is finite, and tends to a constant, regardless of the transmitted bitstream. It is evident from this example, therefore, that we must extend our discussion of baseband and passband signals to random processes. Random processes serve as a useful model not only for modulated signals, but also for noise, interference, and for the input-output response of certain classes of communication channels (e.g., wireless mobile channels).

For a finite-power signal (with unbounded energy), a time-windowed realization is a deterministic signal with finite energy, so that we can employ our existing machinery for finite-energy signals. Our basic strategy is to define properties of a finite-power signal in terms of quantities that can be obtained as averages over a time window, in the limit as the time window gets large. These time averaged properties can be defined for any finite-power signal. However, we are interested mainly in scenarios where the signal is a realization of a random process, and we wish to ensure that properties we infer as a time average over one realization apply to most other realizations as well. In this case, a time average is meaningful as a broad descriptor of the random process only under an *ergodicity* assumption that the time average along a realization equals a corresponding statistical average across realizations. Moreover, while the time average provides a definition that has

operational significance (in terms of being implementable by measurement or computer simulation), when a suitable notion of ergodicity holds, it is often analytically tractable to compute the statistical average. In the forthcoming discussions, we discuss both time averages and statistical averages for several random processes of interest.

Power spectral density As with our definition of energy spectral density, let us define the power spectral density (PSD) for a finite-power signal $s(t)$ in operational terms. Pass the signal $s(t)$ through an ideal narrowband filter with transfer function

$$H_{f_0}(f) = \begin{cases} 1, & f_0 - \frac{\Delta f}{2} < f < f_0 + \frac{\Delta f}{2}, \\ 0, & \text{else.} \end{cases}$$

The PSD evaluated at f_0 , $S_s(f_0)$, can now be defined to be the measured power at the filter output, divided by the filter width Δf (in the limit as $\Delta f \rightarrow 0$).

The preceding definition directly leads to a procedure for computing the PSD based on empirical measurements or computer simulations. Given a physical signal or a computer model, we can compute the PSD by time-windowing the signal and computing the Fourier transform, as follows. Define the time-windowed version of s as

$$s_{T_0}(t) = s(t)I_{[-T_0/2, T_0/2]}(t), \quad (2.50)$$

where T_0 is the length of the observation interval. (The observation interval need not be symmetric about the origin, in general.) Since T_0 is finite, $s_{T_0}(t)$ has finite energy if $s(t)$ has finite power, and we can compute its Fourier transform

$$S_{T_0}(f) = \mathcal{F}(s_{T_0}).$$

The energy spectral density of s_{T_0} is given by $|S_{T_0}(f)|^2$, so that an estimate of the PSD of s is obtained by averaging this over the observation interval. We thus obtain an estimated PSD

$$\hat{S}_s(f) = \frac{|S_{T_0}(f)|^2}{T_0}. \quad (2.51)$$

The computations required to implement (2.51) are often referred to as a periodogram. In practice, the signal s is sampled, and the Fourier transform is computed using a DFT. The length of the observation interval determines the frequency resolution, while the sampling rate is chosen to be large enough to avoid significant aliasing. The multiplication by a rectangular time window in (2.50) corresponds to convolution in the frequency domain with the sinc function, which can lead to significant spectral distortion. It is common, therefore, to employ time windows that taper off at the edges of the observation interval, so as to induce a quicker decay of the frequency domain signal being convolved with. Finally, multiple periodograms can be averaged in order to get a less noisy estimate of the PSD.

Autocorrelation function As with finite-energy signals, the inverse Fourier transform of (2.51) has the interpretation of autocorrelation. Specifically, using (2.20), we have that the inverse Fourier transform of (2.51) is given by

$$\begin{aligned}\hat{S}_s(f) &= \frac{|S_{T_0}(f)|^2}{T_0} \leftrightarrow \hat{R}_s(\tau) = \frac{1}{T_0} \int_{-\infty}^{\infty} s_{T_0}(u) s_{T_0}^*(u - \tau) du \\ &= \frac{1}{T_0} \int_{-\frac{T_0}{2} + \max(0, \tau)}^{\frac{T_0}{2} + \min(0, \tau)} s_{T_0}(u) s_{T_0}^*(u - \tau) du \\ &\approx \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} s_{T_0}(u) s_{T_0}^*(u - \tau) du,\end{aligned}\quad (2.52)$$

where the last approximation neglects edge effects as T_0 gets large (for fixed τ). An alternative method for computing PSD, therefore, is first to compute the empirical autocorrelation function (again, this is typically done in discrete time), and then to compute the DFT. While these methods are equivalent in theory, in practice, the properties of the estimates depend on a number of computational choices, discussion of which is beyond the scope of this book. The interested reader may wish to explore the various methods for estimating PSD available in MATLAB or similar programs.

Formal definitions of PSD and autocorrelation function In addition to providing a procedure for computing the PSD, we can also use (2.51) to provide a formal definition of PSD by letting the observation interval get large:

$$\bar{S}_s(f) = \lim_{T_0 \rightarrow \infty} \frac{|S_{T_0}(f)|^2}{T_0}. \quad (2.53)$$

Similarly, we can take limits in (2.52) to obtain a formal definition of the autocorrelation function as follows:

$$\bar{R}_s(\tau) = \lim_{T_0 \rightarrow \infty} \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} s_{T_0}(u) s_{T_0}^*(u - \tau) du, \quad (2.54)$$

where the overbar notation denotes time averages along a realization. As we see shortly, we can also define the PSD and autocorrelation function as statistical averages across realizations; we drop the overbar notation when we consider these. More generally, we adopt the shorthand $\bar{f}(t)$ to denote the time average of $f(t)$. That is,

$$\bar{f}(t) = \lim_{T_0 \rightarrow \infty} \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} f(u) du.$$

Thus, the definition (2.54) can be rewritten as

$$\bar{R}_s(\tau) = \overline{s(u) s^*(u - \tau)}.$$

Baseband and passband random processes A random process is baseband if its PSD is baseband, and it is passband if its PSD is passband. Since the PSD is defined as a time average along a realization, this also means (by assumption) that the realizations of baseband and passband random processes are modeled as deterministic baseband and passband signals, respectively. In the next section, this assumption enables us to use the development of the complex baseband representation for deterministic passband signals in our discussion of the complex baseband representation of passband random processes.

Crosscorrelation function For finite-power signals s_1 and s_2 , we define the crosscorrelation function as the following time average:

$$\bar{R}_{s_1, s_2}(\tau) = \overline{s_1(u)s_2^*(u-\tau)}. \quad (2.55)$$

The cross-spectral density is defined as the Fourier transform of the cross-correlation function:

$$\bar{S}_{s_1, s_2}(f) = \mathcal{F}(\bar{R}_{s_1, s_2}(f)). \quad (2.56)$$

Example 2.3.1 (Autocorrelation function and power spectral density for a complex waveform) Let $s(t) = s_c(t) + js_s(t)$ be a complex-valued, finite-power, signal, where s_c and s_s are real-valued. Then the autocorrelation function of s can be computed as

$$\bar{R}_s(\tau) = \overline{s(t)s^*(t-\tau)} = \overline{(s_c(t) + js_s(t))(s_c(t-\tau) - js_s(t-\tau))}.$$

Simplifying, we obtain

$$\bar{R}_s(\tau) = [\bar{R}_{s_c}(\tau) + \bar{R}_{s_s}(\tau)] + j[\bar{R}_{s_c, s_s}(\tau) - \bar{R}_{s_s, s_c}(\tau)]. \quad (2.57)$$

Taking Fourier transforms, we obtain the PSD

$$\bar{S}_s(f) = [\bar{S}_{s_c}(f) + \bar{S}_{s_s}(f)] + j[\bar{S}_{s_c, s_s}(f) - \bar{S}_{s_s, s_c}(f)]. \quad (2.58)$$

We use this result in our discussion of the complex baseband representation of passband random processes in the next section.

Example 2.3.2 (Power spectral density of a linearly modulated signal)

The modulated waveform shown in Figure 2.1 can be written in the form

$$s(t) = \sum_{n=-\infty}^{\infty} b[n]p(t-nT),$$

where the bits $b[n]$ take the values ± 1 , and $p(t)$ is a rectangular pulse. Let us try to compute the PSD for this signal using the definition (2.53).

Anticipating the discussion of linear modulation in Section 2.5, let us consider a generalized version of Figure 2.1 in the derivation, allowing the symbols $b[n]$ to be complex-valued and $p(t)$ to be an arbitrary pulse in the derivation. Consider the signal $\hat{s}(t)$ restricted to the observation interval $[0, NT]$, given by

$$s_{T_0}(t) = \sum_{n=0}^{N-1} b[n]p(t - nT).$$

The Fourier transform of the time-windowed waveform is given by

$$S_{T_0}(f) = \sum_{n=0}^{N-1} b[n]P(f)e^{-j2\pi fnT} = P(f) \sum_{n=0}^{N-1} b[n]e^{-j2\pi fnT}.$$

The estimate of the power spectral density is therefore given by

$$\frac{|S_{T_0}(f)|^2}{T_0} = \frac{|P(f)|^2 \sum_{n=0}^{N-1} b[n]e^{-j2\pi fnT}}{NT}. \quad (2.59)$$

Let us now simplify the preceding expression and take the limit as $T_0 \rightarrow \infty$ (i.e., $N \rightarrow \infty$). Define the term

$$\begin{aligned} A &= \left| \sum_{n=0}^{N-1} b[n]e^{-j2\pi fnT} \right|^2 = \sum_{n=0}^{N-1} b[n]e^{-j2\pi fnT} \left(\sum_{m=0}^{N-1} b[m]e^{-j2\pi fmT} \right)^* \\ &= \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} b[n]b^*[m]e^{-j2\pi f(n-m)T}. \end{aligned}$$

Setting $k = n - m$, we can rewrite the preceding as

$$\begin{aligned} A &= \sum_{n=0}^{N-1} |b[n]|^2 + \sum_{k=1}^{N-1} e^{-j2\pi fkT} \sum_{n=k}^{N-1} b[n]b^*[n-k] \\ &\quad + \sum_{k=-(N-1)}^{-1} e^{-j2\pi fkT} \sum_{n=0}^{N-1+k} b[n]b^*[n-k]. \end{aligned}$$

Now, suppose that the symbols are uncorrelated, in that the time average of $b[n]b^*[n-k]$ is zero for $k \neq 0$. Also, denote the empirical average of $|b[n]|^2$ by σ_b^2 . Then the limit becomes

$$\lim_{N \rightarrow \infty} \frac{A}{N} = \sigma_b^2.$$

Substituting into (2.59), we can now infer that

$$\bar{S}_s(f) = \lim_{T_0 \rightarrow \infty} \frac{|S_{T_0}(f)|^2}{T_0} = \lim_{N \rightarrow \infty} \frac{|P(f)|^2 A}{NT} = \sigma_b^2 \frac{|P(f)|^2}{T}.$$

Thus, we have shown that the PSD of a linearly modulated signal scales as the magnitude squared of the spectrum of the modulating pulse.

The time averages discussed thus far interpret a random process $s(t)$ as a collection of deterministic signals, or *realizations*, evolving over time t , where the specific realization is chosen randomly. Next, we discuss methods for computing the corresponding statistical averages, which rely on an alternate view of $s(t)$, for fixed t , as a random variable which takes a range of values across realizations of the random process. If $\mathcal{T}$ is the set of allowable values for the index t , which is interpreted as time for our purpose here (e.g., $\mathcal{T} = (-\infty, \infty)$ when the time index can take any real value), then $\{s(t), t \in \mathcal{T}\}$ denotes a collection of random variables over a common probability space. The term *common probability space* means that we can talk about the joint distribution of these random variables.

In particular, the statistical averages of interest to us are the autocorrelation function and PSD for wide sense stationary and wide sense cyclostationary random processes (defined later). Since most of the signal and noise models that we encounter fall into one of these two categories, these techniques form an important part of the communication system designer's toolkit. The practical utility of a statistical average in predicting the behavior of a particular realization of a random process depends, of course, on the ergodicity assumption (discussed in more detail later) that time averages equal statistical averages for the random processes of interest.

Mean, autocorrelation, and autocovariance functions For a random process $s(t)$, the mean function is defined as

$$m_s(t) = E[s(t)] \quad (2.60)$$

and the autocorrelation function as

$$R_s(t_1, t_2) = E[s(t_1)s^*(t_2)]. \quad (2.61)$$

The autocovariance function of s is the autocorrelation function of the zero mean version of s , and is given by

$$C_s(t_1, t_2) = E[(s(t_1) - E[s(t_1)])(s(t_2) - E[s(t_2)])^*] = R_s(t_1, t_2) - m_s(t_1)m_s^*(t_2). \quad (2.62)$$

Crosscorrelation and crosscovariance functions For random processes s_1 and s_2 defined on a common probability space (i.e., we can talk about the joint distribution of samples from these random processes), the crosscorrelation function is defined as

$$R_{s_1, s_2}(t_1, t_2) = E[s_1(t_1)s_2^*(t_2)] \quad (2.63)$$

and the crosscovariance function is defined as

$$\begin{aligned} C_{s_1, s_2}(t_1, t_2) &= E[(s_1(t_1) - E[s_1(t_1)])(s_2(t_2) - E[s_2(t_2)])^*] \\ &= R_{s_1, s_2}(t_1, t_2) - m_{s_1}(t_1)m_{s_2}^*(t_2). \end{aligned} \quad (2.64)$$

2.3 Spectral description of random processes

Stationary random process A random process $s(t)$ is said to be stationary if it is statistically indistinguishable from a delayed version of itself. That is, $s(t)$ and $s(t-d)$ have the same statistics for any delay $d \in (-\infty, \infty)$.

For a stationary random process s , the mean function satisfies

$$m_s(t) = m_s(t-d)$$

for any t , regardless of the value of d . Choosing $d = t$, we infer that

$$m_s(t) = m_s(0).$$

That is, the mean function is a constant. Similarly, the autocorrelation function satisfies

$$R_s(t_1, t_2) = R_s(t_1-d, t_2-d)$$

for any t_1, t_2 , regardless of the value of d . Setting $\tau = t_2$, we have

$$R_s(t_1, t_2) = R_s(t_1 - t_2, 0).$$

That is, the autocorrelation function depends only on the difference of its arguments.

Stationarity is a stringent requirement that is not always easy to verify. However, the preceding properties of the mean and autocorrelation functions can be used as the defining characteristics for a weaker property termed wide sense stationarity.

Wide sense stationary (WSS) random process A random process s is said to be WSS if

$$m_s(t) \equiv m_s(0) \quad \text{for all } t$$

and

$$R_s(t_1, t_2) = R_s(t_1 - t_2, 0) \quad \text{for all } t_1, t_2.$$

In this case, we change notation and express the autocorrelation function as a function of $\tau = t_1 - t_2$ alone. Thus, for a WSS process, we can define the autocorrelation function as

$$R_s(\tau) = E[s(t)s^*(t-\tau)] \quad \text{for } s \text{ WSS} \quad (2.65)$$

with the understanding that the expectation is independent of t .

Power spectral density for a WSS process We define the PSD of a WSS process s as the Fourier transform of its autocorrelation function, as follows:

$$S_s(f) = \mathcal{F}(R_s(\tau)). \quad (2.66)$$

We sometimes also need the notion of joint wide sense stationarity of two random processes.

Jointly wide sense stationary random processes The random processes X and Y are said to be jointly WSS if (a) X is WSS, (b) Y is WSS, (c) the crosscorrelation function $R_{X,Y}(t_1, t_2) = E[X(t_1)Y^*(t_2)]$ depends on the time difference $\tau = t_1 - t_2$ alone. In this case, we can redefine the crosscorrelation function as $R_{X,Y}(\tau) = E[X(t)Y^*(t - \tau)]$.

Ergodicity A stationary random process s is ergodic if time averages along a realization equal statistical averages across realizations. For WSS processes, we are primarily interested in ergodicity for the mean and autocorrelation functions. For example, for a WSS process s that is ergodic in its autocorrelation function, the definitions (2.54) and (2.65) of autocorrelation functions give the same result, which gives us the choice of computing the autocorrelation function (and hence the PSD) as either a time average or a statistical average. Intuitively, ergodicity requires having “enough randomness” in a given realization so that a time average along a realization is rich enough to capture the statistics across realizations. Specific technical conditions for ergodicity are beyond our present scope, but it is worth mentioning the following intuition in the context of the simple binary modulated waveform depicted in Figure 2.1. If all bits take the same value over a realization, then the waveform is simply a constant taking value $+1$ or -1 : clearly, a time average across such a degenerate realization does not yield “typical” results. Thus, we need the bits in a realization to exhibit enough variation to obtain ergodicity. In practice, we often use line codes or scramblers specifically designed to avoid long runs of zeros or ones, in order to induce enough transitions for proper operation of synchronization circuits. It is fair to say, therefore, that there is typically enough randomness in the kinds of waveforms we encounter (e.g., modulated waveforms, noise and interference) that ergodicity assumptions hold.

Example 2.3.3 Armed with these definitions, let us revisit the binary modulated waveform depicted in Figure 2.1, or more generally, a linearly modulated waveform of the form

$$s(t) = \sum_{n=-\infty}^{\infty} b[n]p(t - nT). \quad (2.67)$$

When we delay this waveform by d , we obtain

$$s(t - d) = \sum_{n=-\infty}^{\infty} b[n]p(t - nT - d).$$

Let us consider the special case $d = kT$, where k is an integer. We obtain

$$s(t - kT) = \sum_{n=-\infty}^{\infty} b[n]p(t - (n+k)T) = \sum_{n=-\infty}^{\infty} b[n-k]p(t - nT), \quad (2.68)$$

where we have replaced $n+k$ by n in the last summation. Comparing (2.67) and (2.68), we note that the only difference is that the symbol sequence

$\{b[n]\}$ is replaced by a delayed version $\{b[n-k]\}$. If the symbol sequence is stationary, then it has the same statistics as its delayed version, which implies that $s(t)$ and $s(t-kT)$ are statistically indistinguishable. However, this is a property that only holds for delays that are integer multiples of the symbol time. For example, for the binary signaling waveform in Figure 2.1, it is immediately evident by inspection that $s(t)$ can be distinguished easily from $s(t-T/2)$ (e.g., from the location of the symbol edges). Slightly more sophisticated arguments can be used to show similar results for pulses that are more complicated than the rectangular pulse. We conclude, therefore, that a linearly modulated waveform of the form (2.67), with a stationary symbol sequence $\{b[n]\}$, is a *cyclostationary* random process, where the latter is defined formally below.

Cyclostationary random process The random process $s(t)$ is *cyclostationary* with respect to time interval T if it is statistically indistinguishable from $s(t-kT)$ for any integer k .

As with the concept of stationarity, we can relax the notion of cyclostationarity by considering only the first and second order statistics.

Wide sense cyclostationary random process The random process $s(t)$ is *wide sense cyclostationary* with respect to time interval T if the mean and autocorrelation functions satisfy the following:

$$m_s(t) = m_s(t-T) \quad \text{for all } t,$$

$$R_s(t_1, t_2) = R_s(t_1-T, t_2-T) \quad \text{for all } t_1, t_2.$$

We now state the following theorem regarding cyclostationary processes; this is proved in Problem 2.14.

Theorem 2.3.1 (Stationarizing a cyclostationary process) Let $s(t)$ be a cyclostationary random process with respect to the time interval T . Suppose that D is a random variable that is uniformly distributed over $[0, T]$, and independent of $s(t)$. Then $s(t-D)$ is a stationary random process. Similarly, if $s(t)$ is wide sense cyclostationary, then $s(t-D)$ is a WSS random process.

The random process $s(t-D)$ is a “stationarized” version of $s(t)$, with the random delay D transforming the periodicity in the statistics of $s(t)$ into time invariance in the statistics of $s(t-D)$. We can now define the PSD of s to be that of its stationarized version, as follows.

Computation of PSD for a (wide sense) cyclostationary process as a statistical average For $s(t)$ (wide sense) cyclostationary with respect to time interval T , we define the PSD as

$$S_s(f) = \mathcal{F}(R_s(\tau)),$$

where R_s is the “stationarized” autocorrelation function,

$$R_s(\tau) = E[s(t-D)s^*(t-D-\tau)],$$

with the random variable D chosen as in Theorem 2.3.1.

In Problem 2.14, we discuss why this definition of PSD for cyclostationary processes is appropriate when we wish to relate statistical averages to time averages. That is, when a cyclostationary process satisfies intuitive notions of ergodicity, then its time averaged PSD equals the statistically averaged PSD of the corresponding stationarized process. We then rederive the PSD for a linearly modulated signal, obtained as a time average in Example 2.3.2 and as a statistical average in Problem 2.22.

2.3.1 Complex envelope for passband random processes

For a passband random process $s_p(t)$ with PSD $S_{s_p}(f)$, we know that the time-windowed realizations are also approximately passband. We can therefore define the complex envelope for these time-windowed realizations, and then remove the windowing in the limit to obtain a complex baseband random process $s(t)$. Since we have defined this relationship on the basis of the deterministic time-windowed realizations, the random processes s_p and s obey the same upconversion and downconversion relationships (Figure 2.9) as deterministic signals. It remains to specify the relation between the PSDs of s_p and s , which we again infer from the relationships between the time-windowed realizations. For notational simplicity, denote by $\hat{s}_p(t)$ a realization of $s_p(t)$ windowed by an observation interval of length T_o ; that is, $\hat{s}_p(t) = s_p(t)I_{[-T_o/2, T_o/2]}$. Let $\hat{S}_p(f)$ denote the Fourier transform of $\hat{s}_p$, $\hat{s}(t)$ the complex envelope of $\hat{s}_p$, and $\hat{S}(f)$ the Fourier transform of $\hat{s}(t)$. We know that the PSD of s_p and s can be approximated as follows:

$$S_{s_p}(f) \approx \frac{|\hat{S}_p(f)|^2}{T_o}, \quad S_s(f) \approx \frac{|\hat{S}(f)|^2}{T_o}. \quad (2.69)$$

Furthermore, we know from the relationship (2.42) between deterministic passband signals and their complex envelopes that the following spectral relationships hold:

$$\hat{S}_p(f) = \frac{1}{\sqrt{2}} (\hat{S}(f - f_c) + \hat{S}^*(-f - f_c)).$$

Since the $\hat{S}(f)$ is (approximately) baseband, the right translate $\hat{S}(f - f_c)$ and the left translate $\hat{S}^*(-f - f_c)$ do not overlap, so that

$$|\hat{S}_p(f)|^2 = \frac{1}{2} (|\hat{S}(f - f_c)|^2 + |\hat{S}^*(-f - f_c)|^2).$$

Combining with (2.69), and letting the observation interval T_o get large, we obtain

$$S_{s_p}(f) = \frac{1}{2} (S_s(f - f_c) + S_s(-f - f_c)). \quad (2.70)$$

2.4 Modulation degrees of freedom

In a similar fashion, starting from the passband PSD and working backward, we can infer that

$$S_s(f) = 2S_{s_p}^+(f + f_c),$$

where $S_{s_p}^+(f) = S_{s_p}(f)I_{[f>0]}$ is the “right half” of the passband PSD.

As with deterministic signals, the definition of bandwidth for real-valued passband random processes is based on occupancy of positive frequencies alone, while that for complex baseband random processes is based on occupancy of both positive and negative frequencies. For a given passband random process, both definitions lead to the same value of bandwidth.

2.4 Modulation degrees of freedom

While analog waveforms and channels live in a continuous time space with uncountably infinite dimensions, digital communication systems employing such waveforms and channels can be understood in terms of vector spaces with finite, or countably infinite, dimensions. This is because the dimension, or degrees of freedom, available for modulation is limited when we restrict the time and bandwidth of the signaling waveforms to be used. Let us consider signaling over an ideally bandlimited passband channel spanning $f_c - W/2 \leq f \leq f_c + W/2$. By choosing f_c as a reference, this is equivalent to an ideally bandlimited complex baseband channel spanning $[-W/2, W/2]$. That is, modulator design corresponds to design of a set of complex baseband transmitted waveforms that are bandlimited to $[-W/2, W/2]$. We can now invoke Nyquist’s sampling theorem, stated below.

Theorem 2.4.1 (Nyquist’s sampling theorem) *Any signal $s(t)$ bandlimited to $[-W/2, W/2]$ can be described completely by its samples $\{s(n/W)\}$ at rate W . Furthermore, $s(t)$ can be recovered from its samples using the following interpolation formula:*

$$s(t) = \sum_{n=-\infty}^{\infty} s\left(\frac{n}{W}\right) p\left(t - \frac{n}{W}\right), \quad (2.71)$$

where $p(t) = \text{sinc}(Wt)$.

By the sampling theorem, the modulator need only specify the samples $\{s(n/W)\}$ to specify a signal $s(t)$ bandlimited to $[-W/2, W/2]$. If the signals are allowed to span a large time interval T_0 (large enough that they are still approximately bandlimited), the number of complex-valued samples that the modulator must specify is approximately WT_0 . That is, the set of possible transmitted signals lies in a finite-dimensional complex subspace of dimension WT_0 , or equivalently, in a real subspace of dimension $2WT_0$. To summarize, the dimension of the complex-valued signal space (i.e., the number of degrees of freedom available to the modulator) equals the time–bandwidth product.

The interpolation formula (2.71) can be interpreted as linear modulation (which has been introduced informally via several examples, and is considered in detail in the next section) at rate W using the samples $\{s(n/W)\}$ as the symbols, and the sinc pulse as the modulating pulse $g_{TX}(t)$. Linear modulation with the sinc pulse has the desirable characteristic, therefore, of being able to utilize all of the degrees of freedom available in a bandlimited channel. As we show in the next section, however, the sinc pulse has its problems, and in practice, it is necessary to back off from utilizing all available degrees of freedom, using modulating pulses that have less abrupt transitions in the frequency domain than the brickwall Fourier transform of the sinc pulse.

Bandwidth efficiency Bandwidth efficiency for a modulation scheme is defined to be the number of bits conveyed per degree of freedom. Thus, M -ary signaling in a D -dimensional signal space has bandwidth efficiency

$$\eta_B = \frac{\log_2 M}{D}. \quad (2.72)$$

The number of degrees of freedom in the preceding definition is taken to be the maximum available. Thus, for a bandlimited channel with bandwidth W , we would set $D = WT_0$ to obtain the number of complex degrees of freedom available to a modulator over a large time interval T_0 . In practice, the number of effective degrees of freedom is smaller owing to a variety of implementation considerations, as mentioned for the example of linear modulation in the previous paragraph. We do not include such considerations in our definition of bandwidth efficiency, in order to get a number that fundamentally characterizes a modulation scheme, independent of implementation variations.

To summarize, the set of possible transmitted waveforms in a timelimited and bandwidth-limited system lies in a finite-dimensional *signal space*. The broad implication of this observation is that we can restrict attention to discrete-time signals, or vectors, for most aspects of digital communication system design, even though the physical communication mechanism is based on sending continuous-time waveforms over continuous-time channels. In particular, we shall see in Chapter 3 that signal space concepts play an important role in developing a geometric understanding of receiver design. Signal space concepts are also useful for describing modulation techniques, as we briefly describe below (postponing a more detailed development to Chapter 3).

Signal space description of modulation formats Consider a modulation format in which one of M signals, $s_1(t), \dots, s_M(t)$, is transmitted. The *signal space* spanned by these signals is of dimension $n \leq M$, so we can represent each signal $s_i(t)$ by an n -dimensional vector $\mathbf{s}_i = (s_i[1], \dots, s_i[n])^T$, with

respect to some orthonormal basis $\psi_1(t), \dots, \psi_n(t)$ satisfying $\langle \psi_k, \psi_l \rangle = \delta_{kl}$, $1 \leq k, l \leq n$. That is, we have

$$s_i(t) = \sum_{l=1}^n s_{il} \psi_l(t), \quad s_{il} = \langle s_i, \psi_l \rangle = \int s_i(t) \psi_l^*(t) dt. \quad (2.73)$$

By virtue of (2.73), we can describe a modulation format by specifying either the signals $s_i(t)$, $1 \leq i \leq M$, or the vectors s_i , $1 \leq i \leq M$. More importantly, the *geometry* of the signal set is preserved when we go from continuous-time to vectors, in the sense that inner products, and Euclidean distances, are preserved: $\langle s_i, s_j \rangle = \langle s_i, s_j \rangle$ for $1 \leq i, j \leq M$. As we shall see in Chapter 3, it is this geometry that determines performance over the AWGN channel, which is the basic model upon which we build when designing most communication systems. Thus, we can design vectors with a given geometry, depending on the performance characteristics we desire, and then map them into continuous-time signals using a suitable orthonormal basis $\{\psi_k\}$. This implies that the same vector space design can be reused over very different physical channels, simply by choosing an appropriate basis matched to the channel's time-bandwidth constraints. An example of signal space construction based on linear modulation is provided in Section 2.5.4.

2.5 Linear modulation

We now know that we can encode information to be transmitted over a passband channel into a complex-valued baseband waveform. For a physical baseband channel, information must be encoded into a real-valued baseband waveform. We focus on more general complex baseband (i.e., passband) systems, with physical real baseband systems automatically included as a special case.

As the discussion in the previous section indicates, *linear modulation* is a technique of fundamental importance for communication over bandlimited channels. We have already had sneak previews of this modulation technique in Figure 2.1 and Examples 2.3.2, 2.2.3, and we now build on these for a more systematic exposition. The complex baseband transmitted waveform for linear modulation can be written as

$$u(t) = \sum_n b[n] g_{TX}(t - nT). \quad (2.74)$$

Here $\{b[n]\}$ are the transmitted *symbols*, typically taking values in a fixed symbol *alphabet*, or *constellation*. The modulating pulse $g_{TX}(t)$ is a fixed baseband waveform. The *symbol rate*, or *baud rate* is $1/T$, and T is termed the *symbol interval*.

2.5.1 Examples of linear modulation

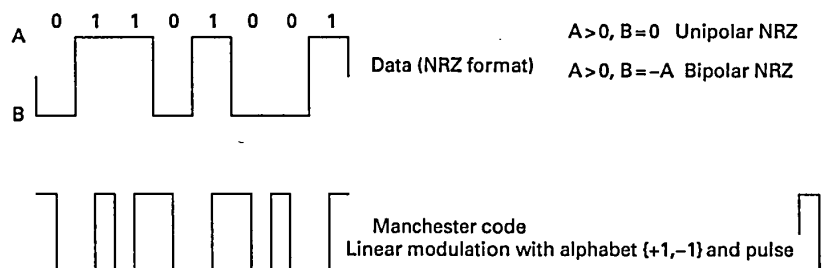
We now discuss some commonly used linear modulation formats for baseband and passband channels.

Baseband line codes Linear modulation over physical baseband channels is a special case of (2.74), with all quantities constrained to be real-valued, since $u(t)$ is actually the physical waveform transmitted over the channel. For such real baseband channels, methods of mapping bits to (real-valued) analog waveforms are often referred to as *line codes*. Examples of some binary line codes are shown in Figure 2.14 and can be interpreted as linear modulation with either a $\{-1, +1\}$ or a $\{0, 1\}$ alphabet.

If a clock is not sent in parallel with the modulated data, then bit timing must be extracted from the modulated signal. For the non return to zero (NRZ) formats shown in Figure 2.14, a long run of zeros or ones can lead to loss of synchronization, since there are no transitions in voltage to demarcate bit boundaries. This can be alleviated by precoding the transmitted data so that it has a high enough rate of transitions from 0 to 1, and vice versa. Alternatively, transitions can be guaranteed through choice of modulating pulse: the Manchester code shown in Figure 2.14 has transitions that are twice as fast as the bit rate. The spectral characteristics of baseband line codes are discussed further in Problem 2.23.

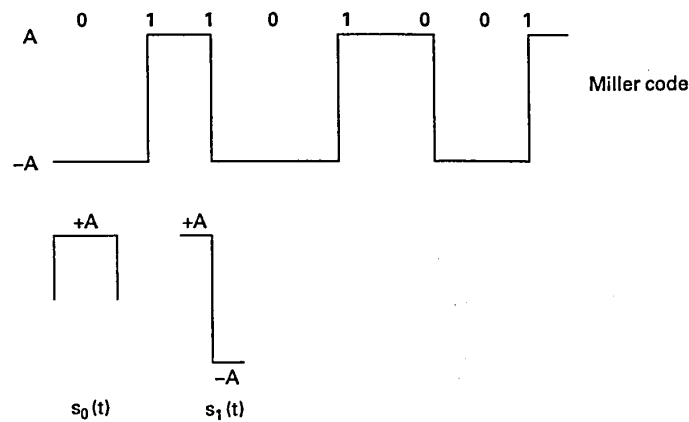
Linear memoryless modulation is not the only option The line codes in Figure 2.14 can be interpreted as memoryless linear modulation: the waveform corresponding to a bit depends only on the value of the bit, and is a translate of a single basic pulse shape. We note at this point that this is certainly not the only way to construct a line code. Specifically, the Miller code, depicted in Figure 2.15, is an example of a line code employing memory and nonlinear modulation. The code uses two different basic pulse shapes, $\pm s_1(t)$ to send 1, and $\pm s_0(t)$ to send 0. A sign change is enforced when 0 is followed by 0, in order to enforce a transition. For the sequences 01, 10 and 11, a transition is ensured because of the transition within $s_1(t)$. In this case, the sign of the waveform is chosen to delay the transition as much as possible; it is intuitively

Figure 2.14 Some baseband line codes using memoryless linear modulation.



2.5 Linear modulation

Figure 2.15 The Miller code is a nonlinear modulation format with memory.



plausible that this makes the modulated waveform smoother, and reduces its spectral occupancy.

Passband linear modulation For passband linear modulation, the symbols $\{b[n]\}$ in (2.74) are allowed to be complex-valued, so that they can be represented in the two-dimensional real plane. Thus, we often use the term *two-dimensional* modulation for this form of modulation. The complex baseband signal $u(t) = u_c(t) + ju_s(t)$ is upconverted to passband as shown in Figure 2.9.

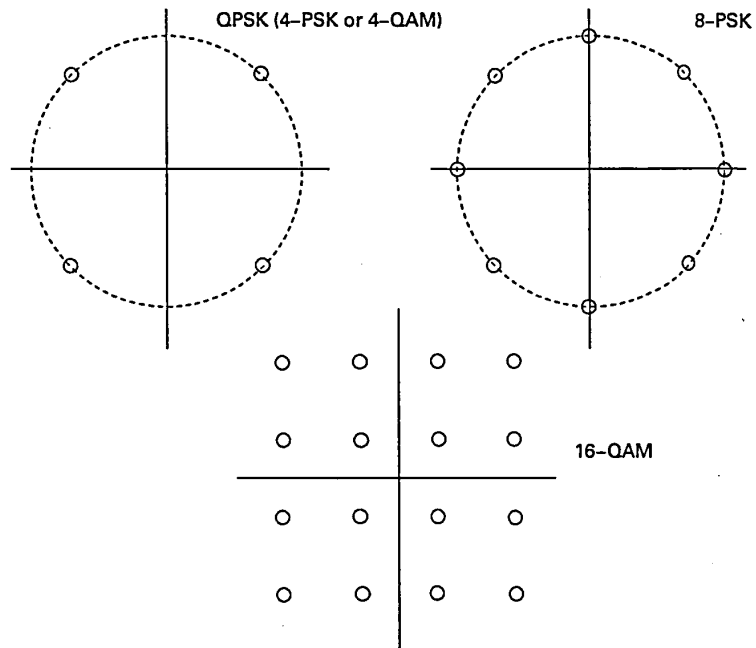
Two popular forms of modulation are phase shift keying (PSK) and quadrature amplitude modulation (QAM). Phase shift keying corresponds to choosing $\arg(b[n])$ from a constellation where the modulus $|b[n]|$ is constant. Quadrature amplitude modulation allows both $|b[n]|$ and $\arg(b[n])$ to vary, and often consists of varying $\text{Re}(b[n])$ and $\text{Im}(b[n])$ independently. Assuming, for simplicity, that $g_{TX}(t)$ is real-valued, we have

$$u_c(t) = \sum_n \text{Re}(b[n])g_{TX}(t - nT), \quad u_s(t) = \sum_n \text{Im}(b[n])g_{TX}(t - nT).$$

The term QAM refers to the variations in the amplitudes of I and Q components caused by the modulating symbol sequence $\{b[n]\}$. If the sequence $\{b[n]\}$ is real-valued, then QAM specializes to pulse amplitude modulation (PAM). Figure 2.16 depicts some well-known constellations, where we plot $\text{Re}(b)$ on the x -axis, and $\text{Im}(b)$ on the y -axis, as b ranges over all possible values for the signaling alphabet. Note that rectangular QAM constellations can be interpreted as modulation of the in-phase and quadrature components using PAM (e.g., 16-QAM is equivalent to I and Q modulation using 4-PAM).

Each symbol in a constellation of size M can be uniquely mapped to $\log_2 M$ bits. For a symbol rate of $1/T$ symbols per unit time, the *bit rate*

Figure 2.16 Some constellations for two-dimensional linear modulation.



is therefore $(\log_2 M)/T$ bits per unit time. Since the transmitted bits often contain redundancy because of a channel code employed for error correction or detection, the *information rate* is typically smaller than the bit rate.

Design choices Some basic choices that a designer of a linearly modulated system must make are: the transmitted pulse shape g_{TX} , the symbol rate $1/T$, the signaling constellation, the mapping from bits to symbols, and the channel code employed, if any. We now show that the symbol rate and pulse shape are determined largely by the available bandwidth, and by implementation considerations. The background needed to make the remaining choices is built up as we progress through this book. In particular, it will be seen later that the constellation size M and the channel code, if any, should be chosen based on channel quality measures such as the signal-to-noise ratio.

2.5.2 Spectral occupancy of linearly modulated signals

From Example 2.3.3, we know that the linearly modulated signal u in (2.74) is a cyclostationary random process if the modulating symbol sequence $\{b[n]\}$ is a stationary random process. Problem 2.22 discusses computation of the PSD for u as a statistical average across realizations, while Example 2.3.2

discusses computation of the PSD as a time average. We now summarize these results in the following theorem.

Theorem 2.5.1 (Power spectral density of a complex baseband linearly modulated signal) Consider a linearly modulated signal

$$u(t) = \sum_{n=-\infty}^{\infty} b[n]g_{TX}(t-nT).$$

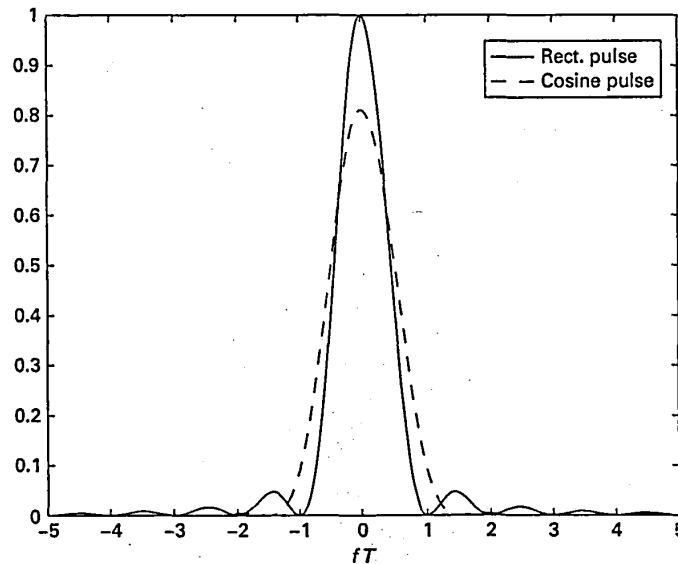
Assume that the symbol stream $b[n]$ is uncorrelated and has zero mean. That is, $E[b[n]b^*[m]] = E[|b[n]|^2]\delta_{nm}$ and $E[b[n]] = 0$ (the expectation is replaced by a time average when the PSD is defined as a time average). Then the PSD of u is given by

$$S_u(f) = \frac{E[|b[n]|^2]}{T} |G_{TX}(f)|^2. \quad (2.75)$$

Figure 2.17 shows the PSD (as a function of normalized frequency fT) for linear modulation using a rectangular timelimited pulse, as well as the cosine-shaped timelimited pulse used for minimum shift keying, which is discussed in Problem 2.24. The smoother shape of the cosine pulse leads to a faster decay of the PSD beyond the main lobe.

Theorem 2.5.1 implies that, for uncorrelated symbols, the shape of the PSD of a linearly modulated signal is determined completely by the spectrum of the modulating pulse $g_{TX}(t)$. A generalization of this theorem for correlated symbol sequences is considered in Problem 2.22, which also discusses the use of such correlations in spectrally shaping the transmitted signal. Another

Figure 2.17 PSD for linear modulation using rectangular and cosine timelimited pulses. The normalization is such that the power (i.e., the area under the PSD) is the same in both cases.



generalization of this theorem, discussed in Problem 2.23, is when the symbols $\{b[n]\}$ have nonzero mean, as is the case for some baseband line codes. In this case, the PSD has spectral lines at multiples of the symbol rate $1/T$, which can be exploited for symbol synchronization.

The preceding result, and the generalizations in Problems 2.22 and 2.23, do not apply to nonlinear modulation formats such as the Miller code shown in Figure 2.15. However, the basic concepts of analytical characterization of PSD developed in these problems can be extended to more general modulation formats with a Markovian structure, such as the Miller code. The details are straightforward but tedious, hence we do not discuss them further.

Once the PSD is known, the bandwidth of u can be characterized using any of a number of definitions. One popular concept (analogous to the energy containment bandwidth for a finite-energy signal) is the $1 - \epsilon$ power containment bandwidth, where ϵ is a small number: this is the size of the smallest contiguous band that contains a fraction $1 - \epsilon$ of the signal power. The fraction of the power contained is often expressed in terms of a percentage: for example, the 99% power containment bandwidth corresponds to $\epsilon = 0.01$. Since the PSD of the modulated signal u is proportional to $|G_{TX}(f)|^2$, the fractional power containment bandwidth is equal to the fractional energy containment bandwidth for $G_{TX}(f)$. Thus, the $1 - \epsilon$ power containment bandwidth B satisfies

$$\int_{-\frac{B}{2}}^{\frac{B}{2}} |G_{TX}(f)|^2 df = (1 - \epsilon) \int_{-\infty}^{\infty} |G_{TX}(f)|^2 df. \quad (2.76)$$

We use the two-sided bandwidth B for the complex baseband signal to quantify the signaling bandwidth needed, since this corresponds to the physical (one-sided) bandwidth of the corresponding passband signal. For real-valued signaling over a *physical* baseband channel, the one-sided bandwidth of u would be used to quantify the physical signaling bandwidth.

Normalized bandwidth Time scaling the modulated waveform $u(t)$ preserves its shape, but corresponds to a change of symbol rate. For example, we can double the symbol rate by using a time compressed version $u(2t)$ of the modulated waveform in (2.74):

$$u_2(t) = u(2t) = \sum_n b[n] g_{TX}(2t - nT) = \sum_n b[n] g_{TX} \left(2 \left(t - n \frac{T}{2} \right) \right).$$

Time compression leads to frequency dilation by a factor of two, while keeping the signal power the same. It is intuitively clear that the PSD $S_{u_2}(f) = 1/2 S_u(f/2)$, regardless of what definition we use to compute it. Thus, whatever our notion of bandwidth, changing the symbol rate in this fashion leads to a proportional scaling of the required bandwidth. This has the following important consequence. Once we have arrived at a design for a given symbol rate $1/T$, we can reuse it without any change for a different symbol rate a/T , simply by replacing $g_{TX}(t)$ with $g_{TX}(at)$ (i.e., $G_{TX}(f)$ with a scaled version of $G_{TX}(f/a)$). If the bandwidth required was B , then the new bandwidth

required is aB . Thus, it makes sense to consider the *normalized bandwidth* BT , which is invariant to the specific symbol rate employed, and depends only on the *shape* of the modulating pulse $g_{TX}(t)$. In doing this, it is also convenient to consider the *normalized time* t/T and normalized frequency fT . Equivalently, we can, without loss of generality, set $T = 1$ to compute the normalized bandwidth, and then simply scale the result by the desired symbol rate.

Example 2.5.1 (Fractional power containment bandwidth with time-limited pulse) We wish to determine the 99% power containment bandwidth when signaling at 100 Mbps using 16-QAM, using a rectangular transmit pulse shape timelimited over the symbol interval. Since there are $\log_2 16 = 4$ bit/symbol, the symbol rate is given by

$$1/T = \frac{100 \text{ Mbps}}{4 \text{ bit/symbol}} = 25 \text{ Msymbol/s.}$$

Let us first compute the normalized bandwidth B_1 for $T = 1$. The transmit pulse is $g_{TX}(t) = I_{[0,1]}(t)$, so that

$$|G_{TX}(f)|^2 = |\text{sinc}(f)|^2.$$

We can now substitute into (2.76) to compute the power containment bandwidth B . We have actually already solved this problem in Example 2.1.3, where we computed $B_1 = 10.2$ for 99% energy containment. We therefore find that the bandwidth required is

$$B = \frac{B_1}{T} = 10.2 \times 25 \text{ MHz} = 260 \text{ MHz.}$$

This is clearly very wasteful of bandwidth. Thus, if we are concerned about strict power containment within the allocated band, we should not be using rectangular timelimited pulses. On the other hand, if we are allowed to be sloppier, and can allow 10% of the power to spill outside the allocated band, then the required bandwidth is less than 25 MHz ($B_1 = 0.85$ for $\alpha = 0.9$, from Example 2.1.3).

2.5.3 The Nyquist criterion: relating bandwidth to symbol rate

Typically, a linearly modulated system is designed so as to avoid intersymbol interference at the receiver, assuming an ideal channel, as illustrated in Figure 2.18, which shows symbols going through a transmit filter, a channel (also modeled as a filter), and a receive filter (noise is ignored for now). Since symbols are being fed into the transmit filter at rate $1/T$, it is natural to expect that we can process the received signal such that, in the absence of channel distortions and noise, samples at rate $1/T$ equal the transmitted symbols. This expectation is fulfilled when the cascade of the transmit filter, the channel

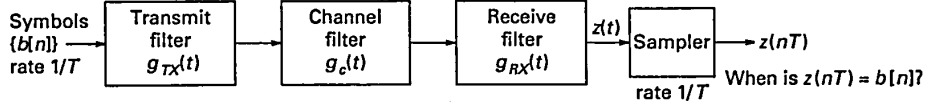


Figure 2.18 Set-up for applying Nyquist criterion.

filter, and the receive filter satisfy the Nyquist criterion for ISI avoidance, which we now state.

From Figure 2.18, the noiseless signal at the output of the receive filter is given by

$$z(t) = \sum_n b[n]x(t - nT), \quad (2.77)$$

where

$$x(t) = (g_{TX} * g_c * g_{RX})(t)$$

is the overall system response to a single symbol. The Nyquist criterion answers the following question: when is $z(nT) = b[n]$? That is, when is there no ISI in the symbol-spaced samples? The answer is stated in the following theorem.

Theorem 2.5.2 (Nyquist criterion for ISI avoidance) *Intersymbol interference can be avoided in the symbol-spaced samples, i.e.,*

$$z(nT) = b[n] \quad \text{for all } n \quad (2.78)$$

if

$$x(mT) = \delta_{m0} = \begin{cases} 1, & m = 0, \\ 0, & m \neq 0. \end{cases} \quad (2.79)$$

Letting $X(f)$ denote the Fourier transform of $x(t)$, the preceding condition can be equivalently written as

$$1/T \sum_{k=-\infty}^{\infty} X\left(f + \frac{k}{T}\right) = 1 \quad \text{for all } f. \quad (2.80)$$

Proof of Theorem 2.5.2 It is immediately obvious that the time domain condition (2.79) gives the desired ISI avoidance (2.78). It can be shown that this is equivalent to the frequency domain condition (2.80) by demonstrating that the sequence $\{x(-mT)\}$ is the Fourier series for the periodic waveform

$$B(f) = 1/T \sum_{k=-\infty}^{\infty} X\left(f + \frac{k}{T}\right)$$

obtained by summing all the aliased copies $X(f + k/T)$ of the Fourier transform of x . Thus, for the sequence $\{x(mT)\}$ to be a discrete delta, the periodic function $B(f)$ must be a constant. The details are developed in Problem 2.15. $\square$

2.5 Linear modulation

A pulse $x(t)$ or $X(f)$ is said to be Nyquist at rate $1/T$ if it satisfies (2.79) or (2.80), where we permit the right-hand sides to be scaled by arbitrary constants.

Minimum bandwidth Nyquist pulse The minimum bandwidth Nyquist pulse is

$$X(f) = \begin{cases} T, & |f| \leq \frac{1}{2T}, \\ 0, & \text{else,} \end{cases}$$

corresponding to the time domain pulse

$$x(t) = \text{sinc}\left(\frac{t}{T}\right).$$

The need for excess bandwidth The sinc pulse is not used in practice because it decays too slowly: the $1/t$ decay implies that the signal $z(t)$ in (2.77) can exhibit arbitrarily large fluctuations, depending on the choice of the sequence $\{b[n]\}$. It also implies that the ISI caused by sampling errors can be unbounded (see Problem 2.21). Both of these phenomena are related to the divergence of the series $\sum_{n=1}^{\infty} 1/n$, which determines the worst-case contribution from “distant” symbols at a given instant of time. Since the series $\sum_{n=1}^{\infty} 1/n^a$ converges for $a > 1$, these problems can be fixed by employing a pulse $x(t)$ that decays as $1/t^a$ for $a > 1$. A faster time decay implies a slower decay in frequency. Thus, we need *excess bandwidth*, beyond the minimum bandwidth dictated by the Nyquist criterion, to fix the problems associated with the sinc pulse. The (fractional) excess bandwidth for a linear modulation scheme is defined to be the fraction of bandwidth over the minimum required for ISI avoidance at a given symbol rate.

Raised cosine pulse An example of a pulse with a fast enough time decay is the frequency domain raised cosine pulse shown in Figure 2.20, and specified as

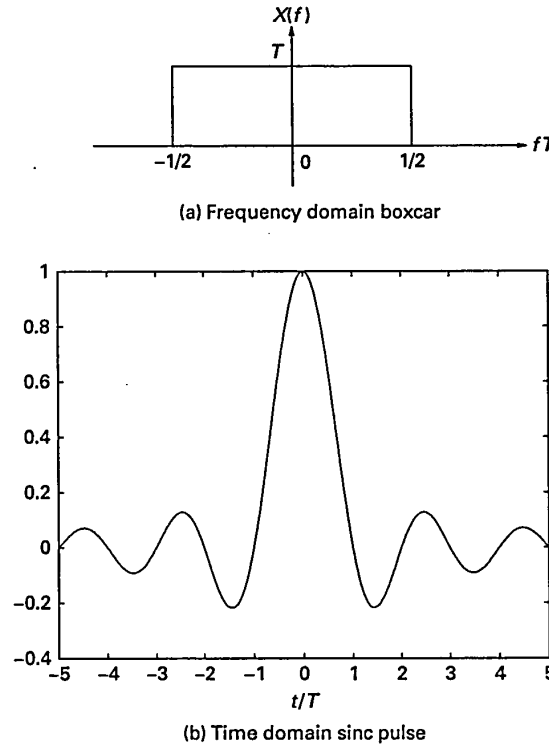
$$S(f) = \begin{cases} T, & |f| \leq \frac{1-a}{2T}, \\ \frac{T}{2} \left[1 - \sin\left(\left(|f| - \frac{1}{2T}\right) \frac{\pi T}{a}\right) \right], & \frac{1-a}{2T} \leq |f| \leq \frac{1+a}{2T}, \\ 0, & |f| > \frac{1+a}{2T}, \end{cases}$$

where a is the fractional excess bandwidth, typically chosen in the range where $0 \leq a < 1$. As shown in Problem 2.16, the time domain pulse $s(t)$ is given by

$$s(t) = \text{sinc}\left(\frac{t}{T}\right) \frac{\cos \pi a \frac{t}{T}}{1 - \left(\frac{2at}{T}\right)^2}.$$

This pulse inherits the Nyquist property of the sinc pulse, while having an additional multiplicative factor that gives an overall $(1/t^3)$ decay with

Figure 2.19 Sinc pulse for minimum bandwidth ISI-free signaling at rate $1/T$. Both time and frequency axes are normalized to be dimensionless.

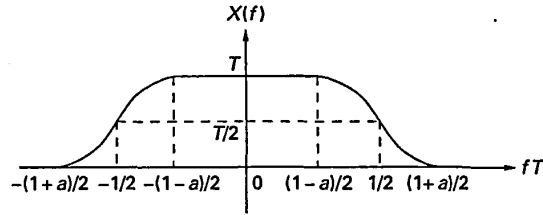


time. The faster time decay compared with the sinc pulse is evident from a comparison of Figures 2.20(b) and 2.19(b).

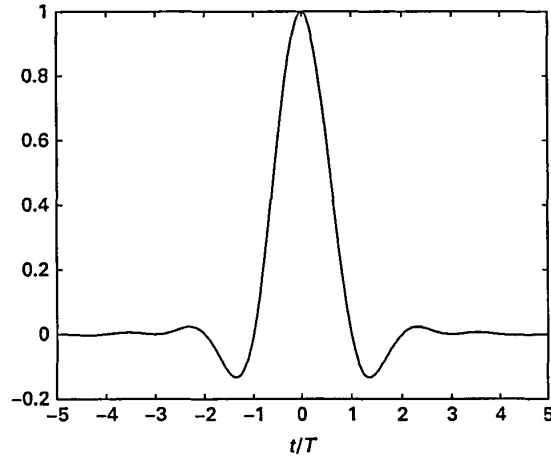
The Nyquist criterion applies to the cascade of the transmit, channel, and receive filters. How is Nyquist signaling done in practice, since the channel is typically not within our control? Typically, the transmit and receive filters are designed so that the cascade $G_{TX}(f)G_{RX}(f)$ is Nyquist, and the ISI introduced by the channel, if any, is handled separately. A typical choice is to set G_{TX} and G_{RX} to be square roots (in the frequency domain) of a Nyquist pulse. Such a pulse is called a *square root Nyquist* pulse. For example, the square root raised cosine (SRRC) pulse is often used in practice. Another common choice is to set G_{TX} to be a Nyquist pulse, and G_{RX} to be a wideband filter whose response is flat over the band of interest.

We had argued in Section 2.4, using Nyquist's sampling theorem, that linear modulation using the sinc pulse takes up all of the degrees of freedom in a bandlimited channel. The Nyquist criterion for ISI avoidance may be viewed loosely as a converse to the preceding result, saying that if there are not enough degrees of freedom, then linear modulation incurs ISI. The relation between these two observations is not accidental: both Nyquist's sampling

Figure 2.20 Raised cosine pulse for minimum bandwidth ISI-free signaling at rate $1/T$, with excess bandwidth α . Both time and frequency axes are normalized to be dimensionless.



(a) Frequency domain raised cosine



(b) Time domain pulse (excess bandwidth $\alpha = 0.5$)

theorem and the Nyquist criterion are based on the Fourier series relationship between the samples of a waveform and its aliased Fourier transform.

Bandwidth efficiency We define the bandwidth efficiency of linear modulation with an M -ary alphabet as

$$\eta_B = \log_2 M \text{ bit/symbol.}$$

This is consistent with the definition (2.72) in Section 2.4, since one symbol in linear modulation takes up one degree of freedom. Since the Nyquist criterion states that the minimum bandwidth required equals the symbol rate, knowing the bit rate R_b and the bandwidth efficiency η_B of the modulation scheme, we can determine the symbol rate, and hence the minimum required bandwidth $B_{\min}$.

$$B_{\min} = \frac{R_b}{\eta_B}.$$

This bandwidth would then be expanded by the excess bandwidth used in the modulating pulse, which (as discussed already in Section 2.4) is not included

in our definition of bandwidth efficiency, because it is a highly variable quantity dictated by a variety of implementation considerations. Once we decide on the fractional excess bandwidth a , the actual bandwidth required is

$$B = (1 + a)B_{\min} = (1 + a)\frac{R_b}{\eta_B}.$$

2.5.4 Linear modulation as a building block

Linear modulation can be used as a building block for constructing more sophisticated waveforms, using a square root Nyquist pulse as the modulating waveform. To see this, let us first describe the square root Nyquist property in the time domain. Suppose that $\psi(t)$ is square root Nyquist at rate $1/T_c$. This means that $Q(f) = |\Psi(f)|^2 = \Psi(f)\Psi^*(f)$ is Nyquist at rate $1/T_c$. Note that $\Psi^*(f)$ is simply the frequency domain representation of $\psi_{MF}(t) = \psi^*(-t)$, the matched filter for $\psi(t)$. This means that

$$Q(f) = \Psi(f)\Psi^*(f) \leftrightarrow q(t) = (\psi * \psi_{MF})(t) = \int \psi(s)\psi^*(s-t) ds. \quad (2.81)$$

That is, $q(t)$ is the autocorrelation function of $\psi(t)$, obtained by passing ψ through its matched filter. Thus, ψ is square root Nyquist if its autocorrelation function q is Nyquist. That is, the autocorrelation function satisfies $q(kT_c) = \delta_{k0}$ for integer k .

We have just shown that the translates $\{\psi(t - kT_c)\}$ are orthonormal. We can now use these as a basis for signal space constructions. Representing a signal $s_i(t)$ in terms of these basis functions is equivalent to linear modulation at rate $1/T_c$ as follows:

$$s_i(t) = \sum_{k=0}^{N-1} s_i[k]\psi(t - kT_c), \quad i = 1, \dots, M,$$

where $\mathbf{s}_i = (s_i[0], \dots, s_i[N-1])$ is a code vector that is mapped to continuous time by linear modulation using the waveform ψ . We often refer to $\psi(t)$ as the *chip waveform*, and $1/T_c$ as the *chip rate*, where N chips constitute a single symbol. Note that the continuous-time inner product between the signals thus constructed is determined by the discrete-time inner product between the corresponding code vectors:

$$\langle s_i, s_j \rangle = \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} s_i[k]s_j^*[l] \int \psi(t - kT_c)\psi^*(t - lT_c)dt = \sum_{k=0}^{N-1} s_i[k]s_j^*[k] = \langle \mathbf{s}_i, \mathbf{s}_j \rangle,$$

where we have used the orthonormality of the translates $\{\psi(t - kT_c)\}$.

Examples of square root Nyquist chip waveforms include a rectangular pulse timelimited to an interval of length T_c , as well as bandlimited pulses such as the square root raised cosine. From Theorem 2.5.1, we see that the PSD of the modulated waveform is proportional to $|\Psi(f)|^2$ (it is typically a good approximation to assume that the chips $\{s_i[k]\}$ are uncorrelated). That is, the bandwidth occupancy is determined by that of the chip waveform ψ .

In the next section, we apply the preceding construction to obtain waveforms for orthogonal modulation. In Chapter 8, I discuss direct sequence spread spectrum systems based on this construction.

2.6 Orthogonal and biorthogonal modulation

The number of possible transmitted signals for orthogonal modulation equals the number of degrees of freedom M available to the modulator, since we can fit only M orthogonal vectors in an M -dimensional signal space. However, as discussed below, whether we need M real degrees of freedom or M complex degrees of freedom depends on the notion of orthogonality required by the receiver implementation.

Frequency shift keying A classical example of orthogonal modulation is frequency shift keying (FSK). The complex baseband signaling waveforms for M -ary FSK over a signaling interval of length T are given by

$$s_i(t) = e^{j2\pi f_i t} I_{[0,T]}, \quad i = 1, \dots, M,$$

where the frequency shifts $|f_i - f_j|$ are chosen to make the M waveforms orthogonal. The bit rate for such a system is therefore given by $(\log_2 M)/T$, since $\log_2 M$ bits are conveyed over each interval of length T . To determine the bandwidth needed to implement such an FSK scheme, we must determine the minimal frequency spacing such that the $\{s_i\}$ are orthogonal. Let us first discuss what orthogonality means.

We have introduced the concepts of coherent and noncoherent reception in Example 2.2.5, where we correlated the received waveform against copies of the possible noiseless received waveforms corresponding to different transmitted signals. In practical terms, therefore, orthogonality means that, if s_i is sent, and we are correlating the received signal against s_j , $j \neq i$, then the output of the correlator should be zero (ignoring noise). This criterion leads to two different notions of orthogonality, depending on the assumptions we make on the receiver's capabilities.

Orthogonality for coherent and noncoherent systems Consider two complex baseband waveforms $u = u_c + ju_s$ and $v = v_c + jv_s$, and their passband equivalents $u_p(t) = \text{Re}(\sqrt{2}u(t)e^{j2\pi f_c t})$ and $v_p(t) = \text{Re}(\sqrt{2}v(t)e^{j2\pi f_c t})$, respectively. From (2.36), we know that

$$\langle u_p, v_p \rangle = \text{Re}(\langle u, v \rangle) = \langle u_c, v_c \rangle + \langle u_s, v_s \rangle. \quad (2.82)$$

Thus, one concept of orthogonality between complex baseband waveforms is that their passband equivalents (with respect to a common frequency and phase reference) are orthogonal. This requires that $\text{Re}(\langle u, v \rangle) = 0$. In the inner product $\text{Re}(\langle u, v \rangle)$, the I and Q components are correlated separately

and then summed up. At a practical level, extracting the I and Q components from a passband waveform requires a *coherent* system, in which an accurate frequency and phase reference is available for downconversion.

Now, suppose that we want the passband equivalents of u and v to remain orthogonal in *noncoherent* systems in which an accurate phase reference may not be available. Mathematically, we want $u_p(t) = \text{Re}(\sqrt{2}u(t)e^{j2\pi f_c t})$ and $\hat{v}_p(t) = \text{Re}(\sqrt{2}v(t)e^{j(2\pi f_c t + \theta)})$ to remain orthogonal, regardless of the value of θ . The complex envelope of $\hat{v}_p$ with respect to f_c is $\hat{v}(t) = v(t)e^{j\theta}$, so that, applying (2.82), we have

$$\langle u_p, \hat{v}_p \rangle = \text{Re}(\langle u, \hat{v} \rangle) = \text{Re}(\langle u, v \rangle e^{-j\theta}). \quad (2.83)$$

It is easy to see that the preceding inner product is zero for all possible θ if and only if $\langle u, v \rangle = 0$; set $\theta = 0$ and $\theta = \pi/2$ in (2.83) to see this.

We therefore have two different notions of orthogonality, depending on which of the inner products (2.82) and (2.83) is employed:

$$\begin{aligned} \text{Re}(\langle s_i, s_j \rangle) &= 0 && \text{Coherent orthogonality criterion} \\ \langle s_i, s_j \rangle &= 0 && \text{Noncoherent orthogonality criterion.} \end{aligned} \quad (2.84)$$

It is left as an exercise (Problem 2.25) to show that a tone spacing of $1/2T$ provides orthogonality in coherent FSK, while a tone spacing of $1/T$ is required for noncoherent FSK. The bandwidth for coherent M -ary FSK is therefore approximately $M/2T$, which corresponds to a time-bandwidth product of approximately $M/2$. This corresponds to a complex vector space of dimension $M/2$, or a real vector space of dimension M , in which we can fit M orthogonal signals. On the other hand, M -ary noncoherent signaling requires M complex dimensions, since the complex baseband signals must remain orthogonal even under multiplication by complex-valued scalars. This requirement doubles the bandwidth requirement for noncoherent orthogonal signaling.

Bandwidth efficiency We can conclude from the example of orthogonal FSK that the bandwidth efficiency of orthogonal signaling is $\eta_B = (\log_2 M + 1)/M$ bit/complex dimension for coherent systems, and $\eta_B = (\log_2 M)/M$ bit/complex dimension for noncoherent systems. This is a general observation that holds for any realization of orthogonal signaling. In a signal space of complex dimension D (and hence real dimension $2D$), we can fit $2D$ signals satisfying the coherent orthogonality criterion, but only D signals satisfying the noncoherent orthogonality criterion. As M gets large, the bandwidth efficiency tends to zero. In compensation, as we see in Chapter 3, the power efficiency of orthogonal signaling for large M is the “best possible.”

Orthogonal Walsh–Hadamard codes Section 2.5.4 shows how to map vectors to waveforms while preserving inner products, by using linear modulation with a square root Nyquist chip waveform. Applying this construction,

2.7 Differential modulation

the problem of designing orthogonal waveforms $\{s_i\}$ now reduces to designing orthogonal code vectors $\{s_i\}$. Walsh-Hadamard codes are a standard construction employed for this purpose, and can be constructed recursively as follows: at the n th stage, we generate 2^n orthogonal vectors, using the 2^{n-1} vectors constructed in the $n-1$ stage. Let H_n denote a matrix whose rows are 2^n orthogonal codes obtained after the n th stage, with $H_0 = (1)$. Then

$$H_n = \begin{pmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{pmatrix}.$$

We therefore get

$$H_1 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad H_2 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}, \quad \text{etc.}$$

The signals $\{s_i\}$ obtained above can be used for noncoherent orthogonal signaling, since they satisfy the orthogonality criterion $\langle s_i, s_j \rangle = 0$ for $i \neq j$. However, just as for FSK, we can fit twice as many signals into the same number of degrees of freedom if we used the weaker notion of orthogonality required for coherent signaling, namely $\text{Re}(\langle s_i, s_j \rangle) = 0$ for $i \neq j$. It is easy to check that for M -ary Walsh-Hadamard signals $\{s_i, i = 1, \dots, M\}$, we can get $2M$ orthogonal signals for coherent signaling: $\{s_i, js_i, i = 1, \dots, M\}$. This construction corresponds to independently modulating the I and Q components with a Walsh-Hadamard code.

Biorthogonal modulation Given an orthogonal signal set, a biorthogonal signal set of twice the size can be obtained by including a negated copy of each signal. Since signals s and $-s$ cannot be distinguished in a noncoherent system, biorthogonal signaling is applicable to coherent systems. Thus, for an M -ary Walsh-Hadamard signal set $\{s_i\}$ with M signals obeying the noncoherent orthogonality criterion, we can construct a coherent orthogonal signal set $\{s_i, js_i\}$ of size $2M$, and hence a biorthogonal signal set of size $4M$, e.g., $\{s_i, js_i, -s_i, -js_i\}$.

2.7 Differential modulation

Differential modulation uses standard PSK constellations, but encodes the information in the phase transitions between successive symbols rather than in the absolute phase of one symbol. This allows recovery of the information even when there is no absolute phase reference.

Differential modulation is useful for channels in which the amplitude and phase may vary over time (e.g., for a wireless mobile channel), or if there is a residual carrier frequency offset after carrier synchronization. To see why,

consider linear modulation of a PSK symbol sequence $\{b[n]\}$. Under ideal Nyquist signaling, the samples at the output of the receive filter obey the model

$$r[n] = h[n]b[n] + \text{noise}$$

where $h[n]$ is the channel gain. If the phase of $h[n]$ can vary arbitrarily fast with n , then there is no hope of conveying any information in the carrier phase. However, if $h[n]$ varies slowly enough that we can approximate it as piecewise constant over at least two symbol intervals, then we can use phase transitions to convey information. Figure 2.21 illustrates this for two successive noiseless received samples for a QPSK alphabet, comparing $b[n]b^*[n-1]$ with $r[n]r^*[n-1]$. We see that, ignoring noise, these two quantities have the same phase. Thus, even when the channel imposes an arbitrary phase shift, as long as the phase shift is roughly constant over two consecutive symbols, the phase *difference* is unaffected by the channel, and hence can be used to convey information. On the other hand, we see from Figure 2.21 that the *amplitude* of $b[n]b^*[n-1]$ differs from that of $r[n]r^*[n-1]$. Thus, some form of explicit amplitude estimation or tracking is required in order to generalize differential modulation to QAM constellations. How best to design differential modulation for QAM alphabets is still a subject of ongoing research, and we do not discuss it further.

Figure 2.22 shows an example of how two information bits can be mapped to phase transitions for differential QPSK. For example, if the information bits at time n are $i[n] = 00$, then $b[n]$ has the same phase as $b[n-1]$. If

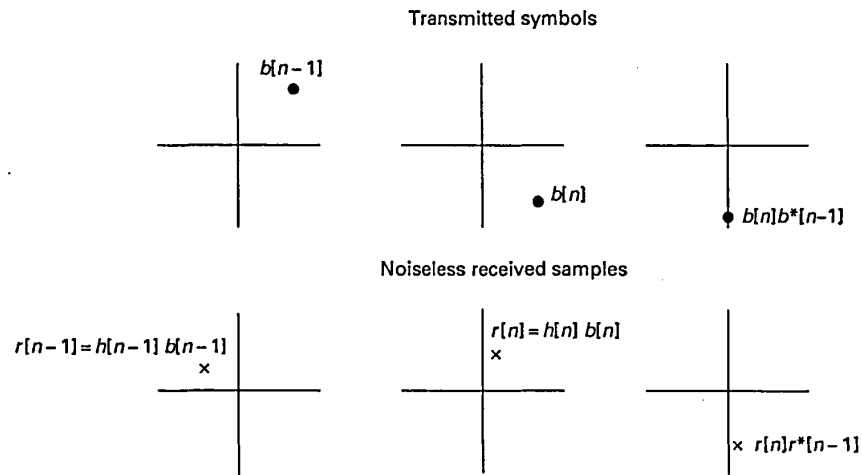
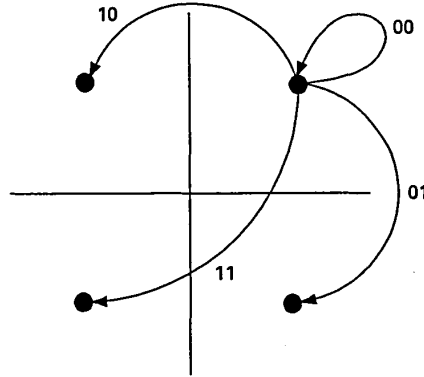


Figure 2.21 Ignoring noise, the phase transitions between successive symbols remain unchanged after an arbitrary phase offset induced by the channel. This motivates differential modulation as a means of dealing with unknown or slowly time-varying channels.

Figure 2.22 Mapping information bits to phase transitions in differential QPSK.



$i[n] = 10$, then $b[n] = e^{j\pi/2}b[n-1]$, and so on, where the symbols $b[n]$ take values in $\{e^{j\pi/4}, e^{j3\pi/4}, e^{j5\pi/4}, e^{j7\pi/4}\}$.

We now discuss the special case of binary differential PSK (DPSK), which has an interesting interpretation as orthogonal modulation. For a BPSK alphabet, suppose that the information bits $\{i[n]\}$ take values in $\{0, 1\}$, the transmitted symbols $\{b[n]\}$ take values in $\{-1, +1\}$, and the encoding rule is as follows:

$$b[n] = b[n-1] \text{ if } i[n]=0,$$

$$b[n] = -b[n-1] \text{ if } i[n]=1.$$

If we think of the signal corresponding to $i[n]$ as $s[n] = (b[n-1], b[n])$, then $s[n]$ can take the following values:

$$\text{for } i[n] = 0, s[n] = \pm s_0, \text{ where } s_0 = (+1, +1),$$

$$\text{for } i[n] = 1, s[n] = \pm s_1, \text{ where } s_1 = (+1, -1).$$

The signals s_0 and s_1 are orthogonal. Note that $s[n] = (b[n-1], b[n]) = b[n-1](1, b[n]/b[n-1])$. Since $b[n]/b[n-1]$ depends only on the information bit $i[n]$, the direction of $s[n]$ depends only on $i[n]$, while there is a sign ambiguity due to $b[n-1]$. Not knowing the channel $h[n]$ would impose a further phase ambiguity. Thus, binary DPSK can be interpreted as binary noncoherent orthogonal signaling, with the signal duration spanning two symbol intervals. However, there is an important distinction from standard binary noncoherent orthogonal signaling, which conveys one bit using two complex degrees of freedom. Binary DPSK uses the available degrees of freedom more efficiently by employing overlapping signaling intervals for sending successive information bits: the signal $(b[n], b[n-1])$ used to send $i[n]$ has one degree of freedom in common with the signal $(b[n+1], b[n])$ used to send $i[n+1]$. In particular, we need $n+1$ complex degrees of freedom to send n bits. Thus, for large enough n , binary DPSK needs one complex degree of

freedom per information bit, so that its bandwidth efficiency is twice that of standard binary noncoherent orthogonal signaling.

A more detailed investigation of noncoherent communication and differential modulation is postponed to Chapter 4, after we have developed the tools for handling noise and noncoherent processing.

2.8 Further reading

Additional modulation schemes (and corresponding references for further reading) are described in Chapter 8, when we discuss wireless communication. Analytic computations of PSD for a variety of modulation schemes, including line codes with memory, can be found in the text by Proakis [3]. Averaging techniques for simulation-based computation of PSD are discussed in Chapter 8, Problem 8.29.

2.9 Problems

2.9.1 Signals and systems

Problem 2.1 A signal $s(t)$ and its matched filter are shown in Figure 2.4.

- Sketch the real and imaginary parts of the output waveform $y(t) = (s * s_{\text{MF}})(t)$ when $s(t)$ is passed through its matched filter.
- Draw a rough sketch of the magnitude $|y(t)|$. When is the output magnitude the largest?

Problem 2.2 For $s(t) = \text{sinc}(t)\text{sinc}(2t)$:

- Find and sketch the Fourier transform $S(f)$.
- Find and sketch the Fourier transform $U(f)$ of $u(t) = s(t)\cos(100\pi t)$ (sketch real and imaginary parts separately if $U(f)$ is complex-valued).

Problem 2.3 For $s(t) = (10 - |t|)I_{[-10,10]}(t)$:

- Find and sketch the Fourier transform $S(f)$.
- Find and sketch the Fourier transform $U(f)$ of $u(t) = s(t)\sin(1000\pi t)$ (sketch real and imaginary parts separately if $U(f)$ is complex-valued).

Problem 2.4 In this problem, we prove the Cauchy–Schwartz inequality (2.5), restated here for convenience,

$$|\langle s, r \rangle| = \left| \int s(t)r^*(t) dt \right| \leq \|s\| \|r\|,$$

2.9 Problems

for any complex-valued signals $s(t)$ and $r(t)$, with equality if and only if one signal is a scalar multiple of the other. For simplicity, we assume in the proof that $s(t)$ and $r(t)$ are real-valued.

- (a) Suppose we try to approximate $s(t)$ by $ar(t)$, a scalar multiple of r , where a is a real number. That is, we are trying to approximate s by an element in the subspace spanned by r . Then the error in the approximation is the signal $e(t) = s(t) - ar(t)$. Show that the energy of the error signal, as a function of the scalar a , is given by

$$J(a) = \|e\|^2 = \|s\|^2 + a^2\|r\|^2 - 2a\langle s, r \rangle.$$

- (b) Note that $J(a)$ is a quadratic function of a with a global minimum. Find the minimizing argument $a_{\min}$ by differentiation and evaluate $J(a_{\min})$. The Cauchy-Schwartz inequality now follows by noting that the minimum error energy is nonnegative. That is, it is a restatement of the fact that $J(a_{\min}) \geq 0$.
- (c) Infer the condition for equality in the Cauchy-Schwartz inequality.

Note For a rigorous argument, the case when $s(t) = 0$ or $r(t) = 0$ almost everywhere should be considered separately. In this case, it can be verified directly that the Cauchy-Schwartz condition is satisfied with equality.

- (d) Interpret the minimizing argument $a_{\min}$ as follows: the signal $a_{\min}r(t)$ corresponds to the projection of $s(t)$ along a unit “vector” in the direction of $r(t)$. The Cauchy-Schwartz inequality then amounts to saying that the error incurred in the projection has nonnegative energy, with equality if $s(t)$ lies in the subspace spanned by $r(t)$.

Problem 2.5 Let us now show why using a matched filter makes sense for delay estimation, as asserted in Example 2.1.2. Suppose that $x(t) = As(t - t_0)$ is a scaled and delayed version of s . We wish to design a filter h such that, when we pass x through h , we get a peak at time t_0 , and we wish to make this peak as large as possible. Without loss of generality, we scale the filter impulse response so as to normalize it as $\|h\| = \|s\|$.

- (a) Using the Cauchy-Schwartz inequality, show that the output y is bounded at all times as follows:

$$|y(t)| \leq \|A\| \|s\|^2.$$

- (b) Using the condition for equality in Cauchy-Schwartz, show that $y(t_0)$ attains the upper bound in (a) if and only if $h(t) = s^*(-t)$ (we are considering complex-valued signals in general, so be careful with the complex conjugates). This means two things: $y(t)$ must have a peak at $t = t_0$, and this peak is an upper bound for the output of any other choice of filter (subject to the normalization we have adopted) at any time.

Note We show in the next chapter that, under a suitable noise model, the matched filter is the optimal form of preprocessing in a broad class of digital communication systems.

2.9.2 Complex baseband representation

Problem 2.6 Consider a real-valued passband signal $x_p(t)$ whose Fourier transform for positive frequencies is given by

$$\begin{aligned} \operatorname{Re}(X_p(f)) &= \begin{cases} \sqrt{2}, & 20 \leq f \leq 22, \\ 0, & 0 \leq f < 20, \\ 0, & 22 < f < \infty, \end{cases} \\ \operatorname{Im}(X_p(f)) &= \begin{cases} \frac{1}{\sqrt{2}}(1 - |f - 22|), & 21 \leq f \leq 23, \\ 0, & 0 \leq f < 21, \\ 0, & 23 < f < \infty. \end{cases} \end{aligned}$$

- Sketch the real and imaginary parts of $X_p(f)$ for both positive and negative frequencies.
- Specify the time domain waveform that you get when you pass $\sqrt{2}x_p(t)\cos(40\pi t)$ through a low pass filter.

Problem 2.7 Let $v_p(t)$ denote a real passband signal, with Fourier transform $V_p(f)$ specified as follows for negative frequencies:

$$V_p(f) = \begin{cases} f + 101, & -101 \leq f \leq -99, \\ 0, & f < -101 \text{ or } -99 < f \leq 0. \end{cases}$$

- Sketch $V_p(f)$ for both positive and negative frequencies.
- Without explicitly taking the inverse Fourier transform, can you say whether $v_p(t) = v_p(-t)$ or not?
- Choosing $f_0 = 100$, find real baseband waveforms $v_c(t)$ and $v_s(t)$ such that

$$v_p(t) = \sqrt{2}(v_c(t)\cos 2\pi f_0 t - v_s(t)\sin 2\pi f_0 t).$$

- Repeat (c) for $f_0 = 101$.

Problem 2.8 Consider the following two passband signals:

$$u_p(t) = \sqrt{2} \operatorname{sinc}(2t) \cos 100\pi t$$

and

$$v_p(t) = \sqrt{2} \operatorname{sinc}(t) \sin\left(101\pi t + \frac{\pi}{4}\right).$$

2.9 Problems

- (a) Find the complex envelopes $u(t)$ and $v(t)$ for u_p and v_p , respectively, with respect to the frequency reference $f_c = 50$.
- (b) What is the bandwidth of $u_p(t)$? What is the bandwidth of $v_p(t)$?
- (c) Find the inner product $\langle u_p, v_p \rangle$, using the result in (a).
- (d) Find the convolution $y_p(t) = (u_p * v_p)(t)$, using the result in (a).

Problem 2.9 Let $u(t)$ denote a real baseband waveform with Fourier transform for $f > 0$ specified by

$$U(f) = \begin{cases} e^{j\pi f} & 0 < f < 1, \\ 0 & f > 1. \end{cases}$$

- (a) Sketch $\text{Re}(U(f))$ and $\text{Im}(U(f))$ for both positive and negative frequencies.
- (b) Find $u(t)$.

Now, consider the bandpass waveform $v(t)$ generated from $u(t)$ as follows:

$$v(t) = \sqrt{2}u(t) \cos 200\pi t.$$

- (c) Sketch $\text{Re}(V(f))$ and $\text{Im}(V(f))$ for both positive and negative frequencies.
- (d) Let $y(t) = (v * h_{\text{hp}})(t)$ denote the result of filtering $v(t)$ using a high pass filter with transfer function

$$H_{\text{hp}}(f) = \begin{cases} 1 & |f| \geq 100 \\ 0 & \text{else.} \end{cases}$$

Find real baseband waveforms y_c, y_s such that

$$y(t) = \sqrt{2}(y_c(t) \cos 200\pi t - y_s(t) \sin 200\pi t).$$

- (e) Finally, pass $y(t) \cos 200\pi t$ through an ideal low pass filter with transfer function

$$H_{\text{lp}}(f) = \begin{cases} 1 & |f| \leq 1 \\ 0 & \text{else.} \end{cases}$$

How is the result related to $u(t)$?

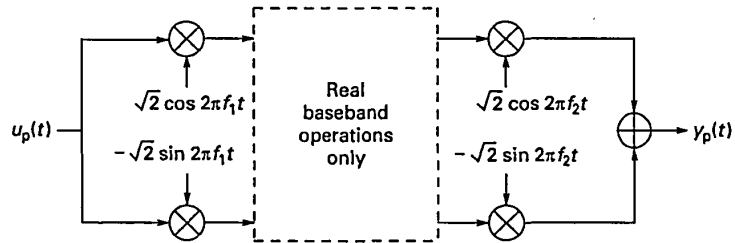
Remark It is a good idea to draw pictures of what is going on in the frequency domain to get a good handle on this problem.

Problem 2.10 Consider a passband filter whose transfer function for $f > 0$ is specified by

$$H_p(f) = \begin{cases} 1 & f_c - 2 \leq f \leq f_c \\ 1 - f + f_c & f_c \leq f \leq f_c + 1 \\ 0 & \text{else.} \end{cases} \quad (f_c \gg 1) \quad (2.85)$$

Let $y_p(t)$ denote the output of the filter when fed by a passband signal $u_p(t)$. We would like to generate $y_p(t)$ from $u_p(t)$ using baseband processing in the system shown in Figure 2.23.

Figure 2.23 Implementation of a passband filter using downconversion, baseband operations and upconversion (Problem 2.10).



- (a) For $f_1 = f_2 = f_c$, sketch the baseband processing required, specifying completely the transfer function of all baseband filters used. Be careful with signs.
- (b) Repeat (a) for $f_1 = f_c + 1/2$ and $f_2 = f_c - 1/2$.

Hint The inputs to the black box are the real and imaginary parts of the complex baseband representation for $u(t)$ centered at f_1 . Hence, we can use baseband filtering to produce the real and imaginary parts for the complex baseband representation for the output $y(t)$ using f_1 as center frequency. Then use baseband processing to construct the real and imaginary parts of the complex baseband representation for $y(t)$ centered at f_2 . These will be the output of the black box.

Problem 2.11 Consider a pure sinusoid $s_p(t) = \cos 2\pi f_c t$, which is the simplest possible example of a passband signal with finite power.

- (a) Find the time-averaged PSD $\bar{S}_s(f)$ and autocorrelation function $\bar{R}_s(\tau)$, proceeding from the definitions. Check that the results conform to your intuition.
- (b) Find the complex envelope $s(t)$, and its time-averaged PSD and autocorrelation function. Check that the relation (2.70) holds for the passband and baseband PSDs.

2.9.3 Random processes

Problem 2.12 Consider a passband random process $n_p(t) = \text{Re}(\sqrt{2}n(t)e^{j2\pi f_c t})$ with complex envelope $n(t) = n_c(t) + jn_s(t)$.

- (a) Given the time-averaged PSD for n_p , can you find the time-averaged PSD for n ? Specify any additional information you might need.
- (b) Given the time-averaged PSD for n_p , can you find the time-averaged PSDs for n_c and n_s ? Specify any additional information you might need.
- (c) Now, consider a statistical description of n_p . What are the conditions on n_c and n_s for n_p to be WSS? Under these conditions, what are the relations between the statistically averaged PSDs of n_p , n , n_c and n_s ?

Problem 2.13 We discuss passband white noise, an important noise model used extensively in Chapter 3, in this problem. A passband random process

2.9 Problems

$n_p(t) = \text{Re}(\sqrt{2}n(t)e^{j2\pi f_c t})$ with complex envelope $n(t) = n_c(t) + jn_s(t)$ has PSD

$$S_n(f) = \begin{cases} \frac{N_0}{2}, & |f - f_c| \leq \frac{W}{2} \text{ or } |f + f_c| \leq \frac{W}{2}, \\ 0, & \text{else,} \end{cases}$$

where n_c, n_s are independent and identically distributed zero mean random processes.

- Find the PSD $S_n(f)$ for the complex envelope n .
- Find the PSDs $S_{n_c}(f)$ and $S_{n_s}(f)$ if possible. If this is not possible from the given information, say what further information is needed.

Problem 2.14 In this problem, we prove Theorem 2.3.1 regarding (wide sense) stationarization of a (wide sense) cyclostationary process. Let $s(t)$ be (wide sense) cyclostationary with respect to the time interval T . Define $v(t) = s(t - D)$, where D is uniformly distributed over $[0, T]$ and independent of s .

- Suppose that s is cyclostationary. Use the following steps to show that v is stationary; that is, for any delay a , the statistics of $v(t)$ and $v(t - a)$ are indistinguishable.
 - Show that $a + D = kT + \tilde{D}$, where k is an integer, and $\tilde{D}$ is a random variable which is independent of s , and uniformly distributed over $[0, T]$.
 - Show that the random process $\tilde{s}$ defined by $\tilde{s}(t) = s(t - kT)$ is statistically indistinguishable from s .
 - Show that the random process $\tilde{v}$ defined by $\tilde{v}(t) = v(t - a) = \tilde{s}(t - \tilde{D})$ is statistically indistinguishable from v .
- Now, suppose that s is wide sense cyclostationary. Use the following steps to show that u is WSS.
 - Show that $m_v(t) = 1/T \int_0^T m_s(\nu) d\nu$ for all t . That is, the mean function of v is constant.
 - Show that

$$R_v(t_1, t_2) = 1/T \int_0^T R_s(t + t_1 - t_2, t) dt.$$

This implies that the autocorrelation function of v depends only on the time difference $t_1 - t_2$.

- Now, let us show that, under an intuitive notion of ergodicity, the autocorrelation function for s , computed as a time average along a realization, equals the autocorrelation function computed as a statistical average for its stationarized version u . This means, for example, that it is the stationarized version of a cyclostationary process which is relevant for computation of PSD as a statistical average.

- (i) Show that $s(t)s^*(t-\tau)$ has the same statistics as $s(t+T)s^*(t+T-\tau)$ for any t .
- (ii) Show that the time-averaged autocorrelation estimate

$$\hat{R}_s(\tau) = \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} s(t)s^*(t-\tau) dt$$

can be rewritten, for $T_0 = KT$ (K a large integer), as

$$\hat{R}_s(\tau) \approx 1/T \int_0^T \frac{1}{K} \sum_{k=-K/2}^{K/2} s(t+kT)s^*(t+kT-\tau) dt.$$

- (iii) Invoke the following intuitive notion of ergodicity: the time average of the identically distributed random variables $\{s(t+kT)s^*(t+kT-\tau)\}$ equals its statistical average $E[s(t)s^*(t-\tau)]$. Infer that

$$\hat{R}_s(\tau) \rightarrow 1/T \int_0^T R_s(t, t-\tau) dt = R_v(\tau)$$

as K (and T_0) becomes large.

2.9.4 Modulation

Problem 2.15 In this problem, we derive the Nyquist criterion for ISI avoidance. Let $x(t)$ denote a pulse satisfying the time domain Nyquist condition for signaling at rate $1/T$: $x(mT) = \delta_{m0}$ for all integer m . Using the inverse Fourier transform formula, we have

$$x(mT) = \int_{-\infty}^{\infty} X(f) e^{j2\pi f m T} df.$$

- (a) Observe that the integral can be written as an infinite sum of integrals over segments of length $1/T$:

$$x(mT) = \sum_{k=-\infty}^{\infty} \int_{\frac{k-1}{T}}^{\frac{k+1}{T}} X(f) e^{j2\pi f m T} df.$$

- (b) In the integral over the k th segment, make the substitution $\nu = f - k/T$. Simplify to obtain

$$x(mT) = T \int_{-\frac{1}{2T}}^{\frac{1}{2T}} B(\nu) e^{-j2\pi \nu m T} d\nu,$$

where $B(f) = 1/T \sum_{k=-\infty}^{\infty} X(f + k/T)$.

- (c) Show that $B(f)$ is periodic in f with period $P = 1/T$, so that it can be written as a Fourier series involving complex exponentials:

$$B(f) = \sum_{m=-\infty}^{\infty} a[m] e^{j2\pi \frac{m}{P} f},$$

where the Fourier series coefficients $\{a[m]\}$ are given by

$$a[m] = \frac{1}{P} \int_{-\frac{P}{2}}^{\frac{P}{2}} B(f) e^{-j2\pi \frac{m}{P} f} df.$$

2.9 Problems

- (d) Conclude that $x(mT) = a[-m]$, so that the Nyquist criterion is equivalent to $a[m] = \delta_{m0}$. This implies that $B(f) \equiv 1$, which is the desired frequency domain Nyquist criterion.

Problem 2.16 In this problem, we derive the time domain response of the frequency domain raised cosine pulse. Let $R(f) = I_{[-\frac{1}{2}, \frac{1}{2}]}(f)$ denote an ideal boxcar transfer function, and let $C(f) = \pi/2a \cos(\pi/af) I_{[-\frac{a}{2}, \frac{a}{2}]}$ denote a cosine transfer function.

- (a) Sketch $R(f)$ and $C(f)$, assuming that $0 < a < 1$.
 (b) Show that the frequency domain raised cosine pulse can be written as

$$S(f) = (R * C)(f).$$

- (c) Find the time domain pulse $s(t) = r(t)c(t)$. Where are the zeros of $s(t)$? Conclude that $s(t/T)$ is Nyquist at rate $1/T$.
 (d) Sketch an argument that shows that, if the pulse $s(t/T)$ is used for BPSK signaling at rate $1/T$, then the magnitude of the transmitted waveform is always finite.

Problem 2.17 Consider a pulse $s(t) = \text{sinc}(at)\text{sinc}(bt)$, where $a \geq b$.

- (a) Sketch the frequency domain response $S(f)$ of the pulse.
 (b) Suppose that the pulse is to be used over an ideal real baseband channel with one-sided bandwidth 400 Hz. Choose a and b so that the pulse is Nyquist for 4-PAM signaling at 1200 bit/s and exactly fills the channel bandwidth.
 (c) Now, suppose that the pulse is to be used over a passband channel spanning the frequencies 2.4–2.42 GHz. Assuming that we use 64-QAM signaling at 60 Mbit/s, choose a and b so that the pulse is Nyquist and exactly fills the channel bandwidth.
 (d) Sketch an argument showing that the magnitude of the transmitted waveform in the preceding settings is always finite.

Problem 2.18 Consider the pulse

$$p(t) = \begin{cases} 1 - \frac{|t|}{T}, & 0 \leq |t| \leq T, \\ 0, & \text{else.} \end{cases}$$

Let $P(f)$ denote the Fourier transform of $p(t)$.

- (a) **True or False** The pulse $p(t)$ is Nyquist at rate $1/T$.
 (b) **True or False** The pulse $p(t)$ is square root Nyquist at rate $1/T$ (i.e., $|P(f)|^2$ is Nyquist at rate $1/T$).

Problem 2.19 Consider the pulse $p(t)$, whose Fourier transform satisfies:

$$P(f) = \begin{cases} 1, & 0 \leq |f| \leq A, \\ \frac{B-|f|}{B-A}, & A \leq |f| \leq B, \\ 0, & \text{else,} \end{cases}$$

where $A = 250$ kHz and $B = 1.25$ MHz.

- (a) **True or False** The pulse $p(t)$ can be used for Nyquist signaling at rate 3 Mbps using an 8-PSK constellation.
- (b) **True or False** The pulse $p(t)$ can be used for Nyquist signaling at rate 4.5 Mbps using an 8-PSK constellation.

Problem 2.20 True or False Any pulse timelimited to duration T is square root Nyquist (up to scaling) at rate $1/T$.

Problem 2.21 (Effect of timing errors) Consider digital modulation at rate $1/T$ using the sinc pulse $s(t) = \text{sinc}(2Wt)$, with transmitted waveform

$$y(t) = \sum_{n=1}^{100} b_n s(t - (n-1)T),$$

where $1/T$ is the symbol rate and $\{b_n\}$ is the bitstream being sent (assume that each b_n takes one of the values ± 1 with equal probability). The receiver makes bit decisions based on the samples $r_n = y((n-1)T)$, $n = 1, \dots, 100$.

- (a) For what value of T (as a function of W) is $r_n = b_n$, $n = 1, \dots, 100$?

Remark In this case, we simply use the sign of the n th sample r_n as an estimate of b_n .

- (b) For the choice of T as in (a), suppose that the receiver sampling times are off by $0.25T$. That is, the n th sample is given by $r_n = y((n-1)T + 0.25T)$, $n = 1, \dots, 100$. In this case, we do have ISI of different degrees of severity, depending on the bit pattern. Consider the following bit pattern:

$$b_n = \begin{cases} (-1)^{n-1} & 1 \leq n \leq 49, \\ (-1)^n & 50 \leq n \leq 100. \end{cases}$$

Numerically evaluate the 50th sample r_{50} . Does it have the same sign as the 50th bit b_{50} ?

Remark The preceding bit pattern creates the worst possible ISI for the 50th bit. Since the sinc pulse dies off slowly with time, the ISI contribu-

2.9 Problems

tions from the 99 bits other than the 50th sample sum up to a number larger in magnitude, and opposite in sign, relative to the contribution due to b_{50} . A decision on b_{50} based on the sign of r_{50} would therefore be wrong. This sensitivity to timing error is why the sinc pulse is seldom used in practice.

- (c) Now, consider the digitally modulated signal in (a) with the pulse $s(t) = \text{sinc}(2Wt)\text{sinc}(Wt)$. For ideal sampling as in (a), what are the two values of T such that $r_n = b_n$?
- (d) For the smaller of the two values of T found in (c) (which corresponds to faster signaling, since the symbol rate is $1/T$), repeat the computation in (b). That is, find r_{50} and compare its sign with b_{50} for the bit pattern in (b).
- (e) Find and sketch the frequency response of the pulse in (c). What is the excess bandwidth relative to the pulse in (a), assuming Nyquist signaling at the same symbol rate?
- (f) Discuss the impact of the excess bandwidth on the severity of the ISI due to timing mismatch.

Problem 2.22 (PSD for linearly modulated signals) Consider the linearly modulated signal

$$s(t) = \sum_{n=-\infty}^{\infty} b[n]p(t-nT).$$

- (a) Show that s is cyclostationary with respect to the interval T if $\{b[n]\}$ is stationary.
- (b) Show that s is wide sense cyclostationary with respect to the interval T if $\{b[n]\}$ is WSS.
- (c) Assume that $\{b[n]\}$ is zero mean, WSS with autocorrelation function $R_b[k] = \mathbb{E}[b[n]b^*[n-k]]$. The z -transform of R_b is denoted by $S_b(z) = \sum_{k=-\infty}^{\infty} R_b[k]z^{-k}$. Let $v(t) = s(t-D)$ denote the stationarized version of s , where D is uniform over $[0, T]$ and independent of s . Show that the PSD of v is given by

$$S_v(f) = S_b(e^{j2\pi fT}) \frac{|P(f)|^2}{T}. \quad (2.86)$$

For uncorrelated symbols with equal average energy (i.e., $R_b[k] = \sigma_b^2 \delta_{k0}$), we have $S_b(z) \equiv \sigma_b^2$, and the result reduces to Theorem 2.5.1.

- (d) **Spectrum shaping via line coding** We can design the sequence $\{b[n]\}$ using a line code so as to shape the PSD of the modulated signal v . For example, for physical baseband channels, we might want to put a null at DC. For example, suppose that we wish to send i.i.d. symbols $\{a[n]\}$ which take values ± 1 with equal probability. Instead of sending $a[n]$

directly, we can send $b[n] = a[n] - a[n-1]$. The transformation from $a[n]$ to $b[n]$ is called a line code.

- (i) What is the range of values taken by $b[n]$?
- (ii) Show that there is a spectral null at DC.
- (iii) Find a line code of the form $b[n] = a[n] + ka[n-1]$ which puts a spectral null at $f = 1/2T$.

Remark The preceding line code can be viewed as introducing ISI in a controlled fashion, which must be taken into account in receiver design. The techniques for dealing with controlled ISI (introduced by a line code) and uncontrolled ISI (introduced by channel distortion) operate on the same principles. Methods for handling ISI are discussed in Chapter 5.

Problem 2.23 (Linear modulation using alphabets with nonzero mean)
Consider again the linearly modulated signal

$$s(t) = \sum_{n=-\infty}^{\infty} b[n]p(t-nT),$$

where $\{b[n]\}$ is WSS, but with nonzero mean $\bar{b} = \mathbb{E}[b[n]]$.

- (a) Show that we can write s as a sum of a deterministic signal $\bar{s}$ and a zero mean random signal $\tilde{s}$ as follows:

$$s(t) = \bar{s}(t) + \tilde{s}(t),$$

where

$$\bar{s}(t) = \bar{b} \sum_{n=-\infty}^{\infty} p(t-nT)$$

and

$$\tilde{s}(t) = \sum_{n=-\infty}^{\infty} \tilde{b}[n]p(t-nT),$$

where $\tilde{b}[n] = b[n] - \bar{b}$ is zero mean, WSS with autocorrelation function $R_{\tilde{b}}[k] = C_b[k]$, where $C_b[k]$ is the autocovariance function of the symbol sequence $\{b[n]\}$.

- (b) Show that the PSD of s is the sum of the PSDs of $\bar{s}$ and $\tilde{s}$, by showing that the two signals are uncorrelated.
- (c) Note that the PSD of $\tilde{s}$ can be found using the result of Problem 2.22(c). It remains to find the PSD of $\bar{s}$. Note that $\bar{s}$ is periodic with period T . It can therefore be written as a Fourier series

$$\bar{s}(t) = \sum_k a[k]e^{j2\pi kt/T},$$

where

$$a[n] = 1/T \int_0^T \bar{s}(t)e^{-j2\pi nt/T} dt.$$

Argue that the PSD of $\bar{s}$ is given by

$$S_{\bar{s}}(f) = \sum_k |a[k]|^2 \delta\left(f - \frac{k}{T}\right).$$

- (d) Find the PSD for the unipolar NRZ baseband line code in Figure 2.14 (set $A = 1$ and $B = 0$ in the NRZ code in the figure).

Problem 2.24 (OQPSK and MSK) Linear modulation with a bandlimited pulse can perform poorly over nonlinear passband channels. For example, the output of a passband hardlimiter (which is a good model for power amplifiers operating in a saturated regime) has constant envelope, but a PSK signal employing a bandlimited pulse has an envelope that passes through zero during a 180 degree phase transition, as shown in Figure 2.24. One way to alleviate this problem is to not allow 180 degree phase transitions. Offset QPSK (OQPSK) is one example of such a scheme, where the transmitted signal is given by

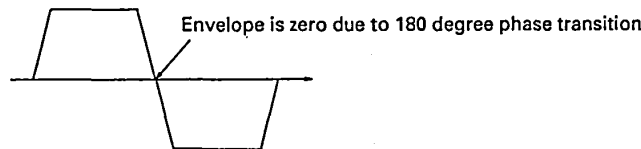
$$s(t) = \sum_{n=-\infty}^{\infty} b_c[n]g_{\text{TX}}(t-nT) + j b_s[n]g_{\text{TX}}\left(t-nT-\frac{T}{2}\right), \quad (2.87)$$

where $\{b_c[n]\}$, $b_s[n]$ are ± 1 BPSK symbols modulating the I and Q channels, with the I and Q signals being staggered by half a symbol interval. This leads to phase transitions of at most 90 degrees at integer multiples of the *bit time* $T_b = T/2$. Minimum shift keying (MSK) is a special case of OQPSK with timelimited modulating pulse

$$g_{\text{TX}}(t) = \sin\left(\frac{\pi t}{T}\right) I_{[0,T]}(t). \quad (2.88)$$

- Sketch the I and Q waveforms for a typical MSK signal, clearly showing the timing relationship between the waveforms.
- Show that the MSK waveform has constant envelope (an extremely desirable property for nonlinear channels).
- Find an analytical expression for the PSD of an MSK signal, assuming that all bits sent are i.i.d., taking values ± 1 with equal probability. Plot the PSD versus normalized frequency fT .
- Find the 99% power containment normalized bandwidth of MSK. Compare with the minimum Nyquist bandwidth, and the 99% power containment bandwidth of OQPSK using a rectangular pulse.

Figure 2.24 The envelope of a PSK signal passes through zero during a 180 degree phase transition, and gets distorted over a nonlinear channel.



- (e) Recognize that Figure 2.17 gives the PSD for OQPSK and MSK, and reproduce this figure, normalizing the area under the PSD curve to be the same for both modulation formats.

Problem 2.25 (FSK tone spacing) Consider two real-valued passband pulses of the form

$$s_0(t) = \cos(2\pi f_0 t + \phi_0), \quad 0 \leq t \leq T,$$

$$s_1(t) = \cos(2\pi f_1 t + \phi_1), \quad 0 \leq t \leq T,$$

where $f_1 > f_0 \gg 1/T$. The pulses are said to be *orthogonal* if $\langle s_0, s_1 \rangle = \int_0^T s_0(t)s_1(t)dt = 0$.

- (a) If $\phi_0 = \phi_1 = 0$, show that the minimum frequency separation such that the pulses are orthogonal is $f_1 - f_0 = 1/2T$.
 (b) If ϕ_0 and ϕ_1 are arbitrary phases, show that the minimum separation for the pulses to be orthogonal regardless of ϕ_0, ϕ_1 is $f_1 - f_0 = 1/T$.

Remark The results of this problem can be used to determine the bandwidth requirements for coherent and noncoherent FSK, respectively.

Problem 2.26 (Walsh–Hadamard codes)

- (a) Specify the Walsh–Hadamard codes for 8-ary orthogonal signaling with noncoherent reception.
 (b) Plot the baseband waveforms corresponding to sending these codes using a square root raised cosine pulse with excess bandwidth of 50%.
 (c) What is the fractional increase in bandwidth efficiency if we use these eight waveforms as building blocks for biorthogonal signaling with coherent reception?

Problem 2.27 (Bandwidth occupancy as a function of modulation format)

We wish to send at a rate of 10 Mbit/s over a passband channel. Assuming that an excess bandwidth of 50% is used, how much bandwidth is needed for each of the following schemes: QPSK, 64-QAM, and 64-ary noncoherent orthogonal modulation using a Walsh–Hadamard code?

Problem 2.28 (Binary DPSK) Consider binary DPSK with encoding as described in Section 2.7. Assume that we fix $b[0] = -1$, and that the stream of information bits $\{i[n], n = 1, \dots, 10\}$ to be sent is 0110001011.

- (a) Find the transmitted symbol sequence $\{b[n]\}$ corresponding to the preceding bit sequence.
 (b) Assuming that we use a rectangular timelimited pulse, draw the corresponding complex baseband transmitted waveform. Is the Q component being used?

2.9 Problems

- (c) Now, suppose that the channel imposes a phase shift of $-\pi/6$. Draw the I and Q components of the noiseless received complex baseband signal.
- (d) Suppose that the complex baseband signal is sent through a matched filter to the rectangular timelimited pulse, and is sampled at the peaks. What are the received samples $\{r[n]\}$ that are obtained corresponding to the transmitted symbol sequence $\{b[n]\}$?
- (e) Find $r[2]r^*[1]$. How do you figure out the information bit $i[2]$ based on this complex number?

Problem 2.29 (Differential QPSK) Consider differential QPSK as shown in Figure 2.22. Suppose that $b[0] = e^{-j\pi/4}$, and that $b[1], b[2], \dots, b[10]$ are determined by using the mapping shown in the figure, where the information bit sequence to be sent is given by 00, 11, 01, 10, 10, 01, 11, 00, 01, 10.

- (a) Specify the phases $\arg(b[n])$, $n = 1, \dots, 10$.
- (b) If you received noisy samples $r[1] = 2 - j$ and $r[2] = 1 + j$, what would be a sensible decision for the pair of bits corresponding to the phase transition from $n = 1$ to $n = 2$? Does this match the true value of these bits? (A systematic treatment of differential demodulation in the presence of noise is given in Chapter 4.)

Information-theoretic limits and their computation

Information theory (often termed Shannon theory in honor of its founder, Claude Shannon) provides fundamental benchmarks against which a communication system design can be compared. Given a channel model and transmission constraints (e.g., on power), information theory enables us to compute, at least in principle, the highest rate at which reliable communication over the channel is possible. This rate is called the *channel capacity*.

Once channel capacity is computed for a particular set of system parameters, it is the task of the communication link designer to devise coding and modulation strategies that approach this capacity. After 50 years of effort since Shannon's seminal work, it is now safe to say that this goal has been accomplished for some of the most common channel models. The proofs of the fundamental theorems of information theory indicate that Shannon limits can be achieved by random code constructions using very large block lengths. While this appeared to be computationally infeasible in terms of both encoding and decoding, the invention of turbo codes by Berrou *et al.* in 1993 provided implementable mechanisms for achieving just this. Turbo codes are random-looking codes obtained from easy-to-encode convolutional codes, which can be decoded efficiently using iterative decoding techniques instead of ML decoding (which is computationally infeasible for such constructions). Since then, a host of "turbo-like" coded modulation strategies have been proposed, including rediscovery of the low density parity check (LDPC) codes invented by Gallager in the 1960s. These developments encourage us to postulate that it should be possible (with the application of sufficient ingenuity) to devise capacity-achieving turbo-like coded modulation strategies for a very large class of channels. Thus, it is more important than ever to characterize information-theoretic limits when setting out to design a communication system, both in terms of setting design goals and in terms of gaining intuition on design parameters (e.g., size of constellation to use). The goal of this chapter, therefore, is to provide enough exposure to Shannon theory to enable computation of capacity benchmarks, with the focus on the AWGN channel and some variants. There is no attempt to give a complete,

or completely rigorous, exposition. For this purpose, the reader is referred to information theory textbooks mentioned in Section 6.5.

The techniques discussed in this chapter are employed in Chapter 8 in order to obtain information-theoretic insights into wireless systems. Constructive coding strategies, including turbo-like codes, are discussed in Chapter 7.

We note that the *law of large numbers (LLN)* is a key ingredient of information theory: if $X_1, \dots, X_n$ are i.i.d. random variables, then their empirical average $(X_1 + \dots + X_n)/n$ tends to the statistical mean $\mathbb{E}[X_1]$ (with probability one) as $n \rightarrow \infty$ under rather general conditions. Moreover, associated with the LLN are *large deviations* results that say that the probability of $O(1)$ deviation of the empirical average from the mean decays exponentially with n . These can be proved using the Chernoff bound (see Appendix B). In this chapter, when we invoke the LLN to replace an empirical average or sum by its statistical counterpart, we implicitly rely on such large deviations results as an underlying mathematical justification, although we do not provide the technical details behind such justification.

Map of this chapter In Section 6.1, we compute the capacity of the continuous and discrete-time AWGN channels using geometric arguments, and discuss the associated power–bandwidth tradeoffs. In Section 6.2, we take a more systematic view, discussing some basic quantities and results of Shannon theory, including the discrete memoryless channel model and the channel coding theorem. This provides a framework for the capacity computations in Section 6.3, where we discuss how to compute capacity under input constraints (specifically focusing on computing AWGN capacity with standard constellations such as PAM, QAM, and PSK). We also characterize the capacity for parallel Gaussian channels, and apply it for modeling dispersive channels. Finally, Section 6.4 provides a glimpse of optimization techniques for computing capacity in more general settings.

6.1 Capacity of AWGN channel: modeling and geometry

In this section, we discuss fundamental benchmarks for communication over a bandlimited AWGN channel.

Theorem 6.1.1 *For an AWGN channel of bandwidth W and received power P , the channel capacity is given by the formula*

$$C = W \log_2 \left(1 + \frac{P}{N_0 W} \right) \text{ bit/s.} \quad (6.1)$$

Let us first discuss some implications of this formula, and then provide some insight into why the formula holds, and how one would go about achieving the rate promised by (6.1).

Consider a communication system that provides an information rate of R bit/s. Denoting by E_b the energy per information bit, the transmitted power is $P = E_b R$. For reliable transmission, we must have $R < C$, so that we have from (6.1):

$$R < W \log_2 \left(1 + \frac{E_b R}{N_0 W} \right).$$

Defining $r = R/W$ as the *spectral efficiency*, or information rate per unit of bandwidth, of the system, we obtain the condition

$$r < \log_2 \left(1 + \frac{E_b r}{N_0} \right).$$

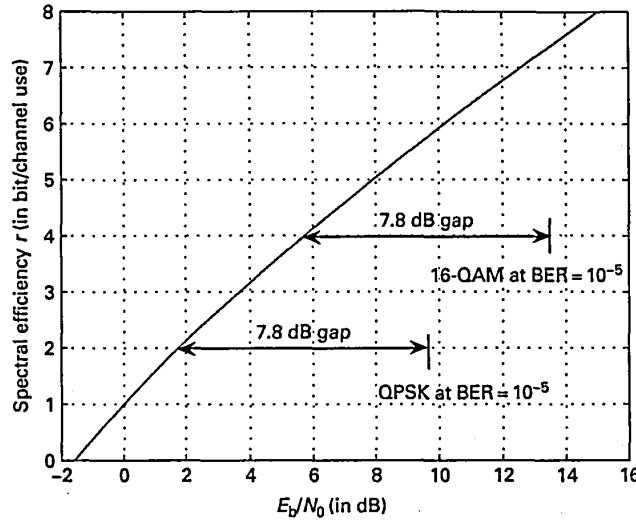
This implies that, for reliable communication, the signal-to-noise ratio must exceed a threshold that depends on the operating spectral efficiency:

$$\frac{E_b}{N_0} > \frac{2^r - 1}{r}. \quad (6.2)$$

“Reliable communication” in an information-theoretic context means that the error probability tends to zero as codeword lengths get large, while a practical system is deemed reliable if it operates at some desired, nonzero but small, error probability level. Thus, we might say that a communication system is operating 3 dB away from Shannon capacity at a bit error probability of 10^{-6} , meaning that the operating E_b/N_0 for a BER of 10^{-6} is 3 dB higher than the minimum required based on (6.2).

Equation (6.2) brings out a fundamental tradeoff between power and bandwidth. The required E_b/N_0 , and hence the required power (assuming that the information rate R and noise PSD N_0 are fixed) increase as we increase the spectral efficiency r , while the bandwidth required to support a given information rate decreases if we increase r . Taking the log of both sides of (6.2), we see that the spectral efficiency and the required E_b/N_0 in dB have an approximately linear relationship. This can be seen from Figure 6.1, which plots achievable spectral efficiency versus E_b/N_0 (dB). Reliable communication is not possible above the curve. In comparing a specific coded modulation scheme with the Shannon limit, we compare the E_b/N_0 required to attain a certain reference BER (e.g., 10^{-5}) with the minimum possible E_b/N_0 , given by (6.2) at that spectral efficiency (excess bandwidth used in the modulating pulse is not considered, since that is a heavily implementation-dependent parameter). With this terminology, uncoded QPSK achieves a BER of 10^{-5} at an E_b/N_0 of about 9.5 dB. For the corresponding spectral efficiency $r = 2$, the Shannon limit given by (6.2) is 1.76 dB, so that uncoded QPSK is about 7.8 dB away from the Shannon limit at a BER of 10^{-5} . A similar gap also exists for uncoded 16-QAM. As we shall see in the next chapter, the gap to Shannon capacity can be narrowed considerably by the use of channel coding. For example, suppose that we use a rate 1/2 binary code (1 information bit/2 coded bits), with the coded bits mapped to a QPSK constellation (2 coded bits/channel use). Then the spectral efficiency

Figure 6.1 Spectral efficiency as a function of E_b/N_0 (dB). The large gap to capacity for uncoded constellations (at a reference BER of 10^{-5}) shows the significant potential benefits of channel coding, which I discuss in Chapter 7.



is $r = 1/2 \times 2 = 1$, and the corresponding Shannon limit is 0 dB. We now know how to design turbo-like codes that get within a fraction of a dB of this limit.

The preceding discussion focuses on spectral efficiency, which is important when there are bandwidth constraints. What if we have access to unlimited bandwidth (for a fixed information rate)? As discussed below, even in this scenario, we cannot transmit at arbitrarily low powers: there is a fundamental limit on the smallest possible value of E_b/N_0 required for reliable communication.

Power-limited communication As we let the spectral efficiency $r \rightarrow 0$, we enter a power-limited regime. Evaluating the limit (6.2) tells us that, for reliable communication, we must have

$$\frac{E_b}{N_0} > \ln 2 \quad (-1.6 \text{ dB}) \quad \text{Minimum required for reliable communication.} \quad (6.3)$$

That is, even if we let bandwidth tend to infinity for a fixed information rate, we cannot reduce E_b/N_0 below its minimum value of -1.6 dB. As we have seen in Chapters 3 and 4, M -ary orthogonal signaling is asymptotically optimum in this power-limited regime, both for coherent and noncoherent communication.

Let us now sketch an intuitive proof of the capacity formula (6.1). While the formula refers to a continuous-time channel, both the proof of the capacity formula, and the kinds of constructions we typically employ to try to achieve capacity, are based on discrete-time constructions.

6.1.1 From continuous to discrete time

Consider an ideal complex WGN channel bandlimited to $[-W/2, W/2]$. If the transmitted signal is $s(t)$, then the received signal

$$y(t) = (s * h)(t) + n(t),$$

where h is the impulse response of an ideal bandlimited channel, and $n(t)$ is complex WGN. We wish to design the set of possible signals that we would send over the channel so as to maximize the rate of reliable communication, subject to a constraint that the signal $s(t)$ has average power at most P .

To start with, note that it does not make sense for $s(t)$ to have any component outside of the band $[-W/2, W/2]$, since any such component would be annihilated once we pass it through the ideal bandlimited filter h . Hence, without loss of generality, $s(t)$ must be bandlimited to $[-W/2, W/2]$ for an optimal signal set design. We now recall the discussion on modulation degrees of freedom from Chapter 2 in order to obtain a discrete-time model.

By the sampling theorem, a signal bandlimited to $[-W/2, W/2]$ is completely specified by its samples at rate W , $\{s(i/W)\}$. Thus, signal design consists of specifying these samples, and modulation for transmission over the ideal bandlimited channel consists of invoking the interpolation formula. Thus, once we have designed the samples, the complex baseband waveform that we send is given by

$$s(t) = \sum_{i=-\infty}^{\infty} s(i/W) p\left(t - \frac{i}{W}\right), \quad (6.4)$$

where $p(t) = \text{sinc}(Wt)$ is the impulse response of an ideal bandlimited pulse with transfer function $P(f) = \frac{1}{W} I_{[-\frac{W}{2}, \frac{W}{2}]}$. As noted in Chapter 2, this is linear modulation at symbol rate W with symbol sequence $\{s(i/W)\}$ and transmit pulse $p(t) = \text{sinc}(Wt)$, which is the minimum bandwidth Nyquist pulse at rate W . The translates $\{p(t - i/W)\}$ form an orthogonal basis for the space of ideally bandlimited functions, so that (6.4) specifies a basis expansion of $s(t)$.

For signaling under a power constraint P over a (large) interval T_0 , the transmitted signal energy should satisfy

$$\int_0^{T_0} |s(t)|^2 dt \approx PT_0.$$

Let $P_s = \mathbb{E}[|s(1/W)|^2]$ denote the average power per sample. Since energy is preserved under the basis expansion (6.4), and we have about $T_0 W$ samples in this interval, we also have

$$T_0 W P_s \|p\|^2 \approx PT_0.$$

For $p(t) = \text{sinc}(Wt)$, we have $\|p\|^2 = 1/W$, so that $P_s = P$. That is, for the scaling adopted in (6.4), the samples obey the same power constraint as the continuous-time signal.

6.1 Capacity of AWGN channel: modeling and geometry

When the bandlimited signal s passes through the ideally bandlimited complex AWGN channel, we get

$$y(t) = s(t) + n(t), \quad (6.5)$$

where n is complex WGN. Since s is linearly modulated at symbol rate W using modulating pulse p , we know that the optimal receiver front end is to pass the received signal through a filter matched to $p(t)$, and to sample at the symbol rate W . For notational convenience, we use a receive filter transfer function $G_R(f) = I_{[-\frac{W}{2}, \frac{W}{2}]}$ which is a scalar multiple of the matched filter $P^*(f) = P(f) = \frac{1}{W} I_{[-\frac{W}{2}, \frac{W}{2}]}$. This ideal bandlimited filter lets the signal $s(t)$ through unchanged, so that the signal contributions to the output of the receive filter, sampled at rate W , are $\{s(i/W)\}$. The noise at the output of the receive filter is bandlimited complex WGN with PSD $N_0 I_{[-\frac{W}{2}, \frac{W}{2}]}$, from which it follows that the noise samples at rate W are independent complex Gaussian random variables with covariance $N_0 W$. To summarize, the noisy samples at the receive filter output can be written as

$$y[i] = s(i/W) + N[i], \quad (6.6)$$

where the signal samples are subject to an average power constraint $\mathbb{E}[|s(i/W)|^2] \leq P$, and $\{N[i]\}$ are i.i.d., zero mean, proper complex Gaussian noise samples with $\mathbb{E}[|N[i]|^2] = N_0 W$.

Thus, we have reduced the continuous-time bandlimited passband AWGN channel model to the discrete-time complex WGN channel model (6.6) that we get to use W times per second if we employ bandwidth W . We can now characterize the capacity of the discrete-time channel, and then infer that of the continuous-time bandlimited channel.

6.1.2 Capacity of the discrete-time AWGN channel

Since the real and imaginary part of the discrete-time complex AWGN model (6.6) can be interpreted as two uses of a real-valued AWGN channel, we consider the latter first.

Consider a discrete-time real AWGN channel in which the output at any given time

$$Y = X + Z, \quad (6.7)$$

where X is a real-valued input satisfying $\mathbb{E}[X^2] \leq S$, and $Z \sim N(0, N)$ is real-valued AWGN. The noise samples over different channel uses are i.i.d. This is an example of a discrete memoryless channel, where $p(Y|X)$ is specified for a single channel use, and the channel outputs for multiple channel uses are conditionally independent given the inputs. A signal, or codeword, over such a channel is a vector $\mathbf{X} = (X_1, \dots, X_n)^T$, where X_i is the input for the i th channel use. A code of rate R bits per channel use can be constructed by designing a set of 2^{nR} such signals $\{\mathbf{X}^k, k = 1, \dots, 2^{nR}\}$, with each signal

having an equal probability of being chosen for transmission over the channel. Thus, nR bits are conveyed over n channel uses. Capacity is defined as the largest rate R for which the error probability tends to zero as $n \rightarrow \infty$.

Shannon has provided a general framework for computing capacity for a discrete memoryless channel, which we discuss in Section 6.3. However, we provide here a heuristic derivation of capacity for the AWGN channel (6.7), that specifically utilizes the geometry induced by AWGN.

Sphere packing based derivation of capacity formula For a transmitted signal $\mathbf{X}^J$, the n -dimensional output vector $\mathbf{Y} = (Y_1, \dots, Y_n)^T$ is given by

$$\mathbf{Y} = \mathbf{X}^J + \mathbf{Z}, \quad \mathbf{X}^J \text{ sent},$$

where $\mathbf{Z}$ is a vector of i.i.d. $\mathcal{N}(0, N)$ noise samples. For equal priors, the MPE and ML rules are equivalent. The ML rule for the AWGN channel is the minimum distance rule

$$\delta_{\text{ML}}(\mathbf{Y}) = \arg \min_{1 \leq k \leq 2^{nR}} \|\mathbf{Y} - \mathbf{X}^k\|^2.$$

Now, the noise vector $\mathbf{Z}$ that perturbs the transmitted signal has energy

$$\|\mathbf{Z}\|^2 = \sum_{i=1}^N Z_i^2 \approx n\mathbb{E}[Z_1^2] = nN,$$

where we have invoked the LLN. This implies that, if we draw a sphere of radius $\sqrt{nN}$ around a signal $\mathbf{X}^J$, then, with high probability, the received vector $\mathbf{Y}$ lies within the sphere when $\mathbf{X}^J$ is sent. Calling such a sphere the “decoding sphere” for $\mathbf{X}^J$, the minimum distance rule would lead to very small error probability if the decoding spheres for different signals were disjoint. We now wish to estimate how many such decoding spheres we can come up with; this gives the value of 2^{nR} for which reliable communication is possible.

Since X is independent of Z (the transmitter does not know the noise realization) in the model (6.7), the input power constraint implies an output power constraint

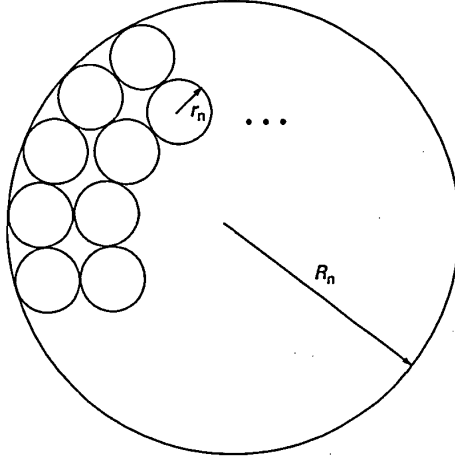
$$\mathbb{E}[Y^2] = \mathbb{E}[(X+Z)^2] = \mathbb{E}[X^2] + \mathbb{E}[Z^2] + 2\mathbb{E}[X]\mathbb{E}[Z] = \mathbb{E}[X^2] + \mathbb{E}[Z^2] \leq S + N. \quad (6.8)$$

Invoking the law of large numbers again, the received signal energy satisfies

$$\mathbb{E}[\|\mathbf{Y}\|^2] \approx n(S + N),$$

so that, with high probability, the received signal vector lies within an n -dimensional sphere with radius $R_n = \sqrt{n(S + N)}$. The problem of signal design for reliable communication now boils down to packing disjoint decoding spheres of radius $r_n = \sqrt{nN}$ within a sphere of radius R_n , as shown in Figure 6.2. The volume of an n -dimensional sphere of radius r equals $K_n r^n$,

Figure 6.2 Decoding spheres of radius $r_n = \sqrt{nN}$ are packed inside a sphere of radius $R_n = \sqrt{n(S+N)}$.



and the number of decoding spheres we can pack is roughly the following ratio of volumes:

$$\frac{K_n R_n^n}{K_n r_n^n} = \frac{K_n (\sqrt{n(S+N)})^n}{K_n (\sqrt{nN})^n} \approx 2^{nR}.$$

Solving, we obtain that the rate $R = 1/2 \log_2(1 + S/N)$. We show in Section 6.3 that this rate exactly equals the capacity of the discrete-time real AWGN channel. (It is also possible to make the sphere packing argument rigorous, but we do not attempt that here.) We now state the capacity formula formally.

Theorem 6.1.2 (Capacity of discrete-time real AWGN channel) *The capacity of the discrete-time real AWGN channel (6.7) is*

$$C_{\text{AWGN}} = \frac{1}{2} \log_2(1 + \text{SNR}) \text{ bit/channel use}, \quad (6.9)$$

where $\text{SNR} = S/N$ is the signal-to-noise ratio.

Thus, capacity grows approximately logarithmically with SNR, or approximately linearly with SNR in dB.

6.1.3 From discrete to continuous time

For the continuous-time bandlimited complex baseband channel that we considered earlier, we have $2W$ uses per second of the discrete-time real AWGN channel (6.7). With the normalization we employed in (6.4), we have that, per real-valued sample, the average signal energy $S = P/2$ and the noise energy

$N = N_0 W/2$, where P is the power constraint on the continuous-time signal. Plugging in, we get

$$C_{\text{cont-time}} = 2W \frac{1}{2} \log_2 \left(1 + \frac{P}{N_0 W} \right) \text{ bit/s},$$

which gives (6.1).

As the invocation of the LLN in the sphere packing based derivation shows, capacity for the discrete-time channel is achieved by using codewords that span a large number of symbols. Suppose, now, that we have designed a capacity-achieving strategy for the discrete-time channel; that is, we have specified a good code, or signal set. A codeword from this set is a discrete-time sequence $\{s[i]\}$. We can now translate this design to continuous time by using the modulation formula (6.4) to send the symbols $\{s[i] = s(i/W)\}$. Of course, as we discussed in Section 2, the sinc pulse used in this formula cannot be used in practice, and should be replaced by a modulating pulse whose bandwidth is larger than the symbol rate employed. A good choice would be a square root Nyquist modulating pulse at the transmitter, and its matched filter at the receiver, which again yields the ISI-free discrete-time model (6.6) with uncorrelated noise samples.

In summary, good codes for the discrete-time AWGN channel (6.6) can be translated into good signal designs for the continuous-time bandlimited AWGN channel using practical linear modulation techniques; this corresponds to using translates of a square root Nyquist pulse as an orthonormal basis for the signal space. It is also possible to use an entirely different basis: for example, orthogonal frequency division multiplexing, which we discuss in Chapter 8, employs complex sinusoids as basis functions. In general, the use of appropriate signal space arguments allows us to restrict attention to discrete-time models, both for code design and for deriving information-theoretic benchmarks.

Real baseband channel The preceding observations also hold for a physical (i.e., real-valued) baseband channel. That is, both the AWGN capacity formula (6.1) and its corollary (6.2) hold, where W for a physical baseband channel refers to the bandwidth occupancy for positive frequencies. Thus, a real baseband signal $s(t)$ occupying a bandwidth W actually spans the interval $[-W, W]$, with the constraint that $S(f) = S^*(-f)$. Using the sampling theorem, such a signal can be represented by $2W$ real-valued samples per second. This is the same result as for a passband signal of bandwidth W , so that the arguments made so far, relating the continuous-time model to the discrete-time real AWGN channel, apply as before. For example, suppose that we wish to find out how far uncoded binary antipodal signaling at BER of 10^{-5} is from Shannon capacity. Since we transmit at 1 bit per sample, the information rate is $2W$ bits per second, corresponding to a spectral efficiency of $r = R/W = 2$. This corresponds to a Shannon limit of 1.8 dB E_b/N_0 , using (6.2). Setting the BER of $Q\left(\sqrt{2E_b/N_0}\right)$ for binary antipodal signaling to

10^{-5} , we find that the required E_b/N_0 is 9.5 dB, which is 7.7 dB away from the Shannon limit. There is good reason for this computation looking familiar: we obtained exactly the same result earlier for uncoded QPSK on a passband channel. This is because QPSK can be interpreted as binary antipodal modulation along the I and Q channels, and is therefore exactly equivalent to binary antipodal modulation for a real baseband channel.

At this point, it is worth mentioning the potential for confusion when dealing with Shannon limits in the literature. Even though PSK is a passband technique, the term BPSK is often used when referring to binary antipodal signaling on a real baseband channel. Thus, when we compare the performance of BPSK with rate $1/2$ coding to the Shannon limit, we should actually be keeping in mind a real baseband channel, so that $r = 1$, corresponding to a Shannon limit of 0 dB E_b/N_0 . (On the other hand, if we had literally interpreted BPSK as using only the I channel in a passband system, we would have gotten $r = 1/2$.) That is, whenever we consider real-valued alphabets, we restrict ourselves to the real baseband channel for the purpose of computing spectral efficiency and comparing Shannon limits. For a passband channel, we can use the same real-valued alphabet over the I and Q channels (corresponding to a rectangular complex-valued alphabet) to get exactly the same dependence of spectral efficiency on E_b/N_0 .

6.1.4 Summarizing the discrete-time AWGN model

In previous chapters, we have used constellations over the AWGN channel with a finite number of signal points. One of the goals of this chapter is to be able to compute Shannon theoretic limits for performance when we constrain ourselves to using such constellations. In Chapters 3 to 5, when sampling signals corrupted by AWGN, we model the discrete-time AWGN samples as having variance $\sigma^2 = N_0/2$ per dimension. On the other hand, the noise variance in the discrete-time model in Section 6.1.3 depends on the system bandwidth W . We would now like to reconcile these two models, and use a notation that is consistent with that in the prior chapters.

Real discrete-time AWGN channel Consider the following model for a real-valued discrete-time channel:

$$Y = X + Z, \quad Z \sim N(0, \sigma^2) \quad (6.10)$$

where X is a power-constrained input, $\mathbb{E}[X^2] \leq E_s$, as well as possibly constrained to take values in a given alphabet (e.g., BPSK or 4-PAM). This notation is consistent with that in Chapter 3, where we use E_s to denote the average energy per symbol. Suppose that we compute the capacity of this discrete-time model as C_d bits per channel use, where C_d is a function of $\text{SNR} = E_s/\sigma^2$. If E_b is the energy per information bit, we must have $E_s = E_b C_d$ joules per channel use. Now, if this discrete-time channel arose from a real

baseband channel of bandwidth W , we would have $2W$ channel uses per second, so that the capacity of the continuous-time channel is $C_c = 2WC_d$ bits per second. This means that the spectral efficiency is given by

$$r = \frac{C_c}{W} = 2C_d \quad \text{Real discrete-time channel.} \quad (6.11)$$

Thus, the SNR for this system is given by

$$\text{SNR} = \frac{E_s}{\sigma^2} = 2C_d \frac{E_b}{N_0} = r \frac{E_b}{N_0} \quad \text{Real discrete-time channel.} \quad (6.12)$$

Thus, we can restrict attention to the real discrete-time model (6.10), which is consistent with our notation in prior chapters. To apply the results to a bandlimited system as in Sections 6.1.1 and 6.1.3, all we need is the relationship (6.11) which specifies the spectral efficiency (bits per Hz) in terms of the capacity of the discrete-time channel (bits per channel use).

Complex discrete-time AWGN model The real-valued model (6.10) can be used to calculate the capacity for rectangular complex-valued constellations such as rectangular 16-QAM, which can be viewed as a product of two real-valued 4-PAM constellations. However, for constellations such as 8-PSK, it is necessary to work directly with a two-dimensional observation. We can think of this as a complex-valued symbol, plus proper complex AWGN (discussed in Chapter 4). The discrete-time model we employ for this purpose is

$$Y = X + Z, \quad Z \sim \mathcal{CN}(0, 2\sigma^2), \quad (6.13)$$

where $\mathbb{E}[|X|^2] \leq E_s$ as before. However, we can also express this model in terms of a two-dimensional real-valued observation (in which case, we do not need to invoke the concepts of proper complex Gaussianity covered in Chapter 4):

$$Y_c = X_c + Z_c, \quad Y_s = X_s + Z_s, \quad (6.14)$$

with Z_c, Z_s i.i.d. $\mathcal{N}(0, \sigma^2)$, and $\mathbb{E}[X_c^2 + X_s^2] \leq E_s$. The capacity C_d bits per channel use for this system is a function of the SNR, which is given by $E_s/2\sigma^2$, as well as any other constraints (e.g., that X is drawn from an 8-PSK constellation). If this discrete-time channel arises from a passband channel of bandwidth W , we have W channel uses per second for the corresponding complex baseband channel, so that the capacity of the continuous-time channel is $C_c = WC_d$ bits per second, so that the spectral efficiency is given by

$$r = \frac{C_c}{W} = C_d \quad \text{Complex discrete-time channel.} \quad (6.15)$$

The SNR is given by

$$\text{SNR} = \frac{E_s}{2\sigma^2} = C_d \frac{E_b}{N_0} = r \frac{E_b}{N_0} \quad \text{Complex discrete-time channel.} \quad (6.16)$$

Comparing (6.12) with (6.16), we note that the relation of SNR with E_b/N_0 and spectral efficiency is the same for both systems. The relations (6.11) and

(6.15) are also consistent: if we get a given capacity for a real-valued model, we should be able to double that in a consistent complex-valued model by using the real-valued model twice.

6.2 Shannon theory basics

From the preceding sphere packing arguments, we take away the intuition that we need to design codewords so as to achieve a good packing of decoding spheres in n dimensions. A direct approach to trying to realize this intuition is not easy (although much progress has been made in recent years in the encoding and decoding of lattice codes that attempt to implement the sphere packing prescription directly). We are interested in determining whether standard constellations (e.g., PSK, QAM), in conjunction with appropriately chosen error-correcting codes, can achieve the same objectives. In this section, we discuss just enough of the basics of Shannon theory to enable us to develop elementary capacity computation techniques. We introduce the general discrete memoryless channel model, for which the model (6.7) is a special case. Key information-theoretic quantities such as entropy, mutual information, and divergence are discussed. We end this section with a statement and partial proof of the channel coding theorem.

While developing this framework, we emphasize the role played by the LLN as the fundamental basis for establishing information-theoretic benchmarks: roughly speaking, the randomness that is inherent in one channel use is averaged out by employing signal designs spanning multiple independent channel uses, thus leading to reliable communication. We have already seen this approach at work in the sphere packing arguments in Section 6.1.2.

Definition 6.2.1 (Discrete memoryless channel) *A discrete memoryless channel is specified by a transition density or probability mass function $p(y|x)$ specifying the conditional distribution of the output y given the input x . For multiple channel uses, the outputs are conditionally independent given the inputs. That is, if $x_1, \dots, x_n$ are the inputs, and $y_1, \dots, y_n$ denote the corresponding outputs, for n channel uses, then*

$$p(y_1, \dots, y_n | x_1, \dots, x_n) = p(y_1 | x_1) \dots p(y_n | x_n).$$

Real AWGN channel For the real Gaussian channel (6.10), the channel transition density is given by

$$p(y|x) = \frac{e^{-\frac{(y-x)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}}, \quad y \text{ real.} \quad (6.17)$$

Here both the input and the output are real numbers, but we typically constrain the input to average symbol energy E_s . In addition, we can constrain the input

x to be drawn from a finite constellation: for example, for BPSK, the input would take values $x = \pm\sqrt{E_s}$.

Complex AWGN channel For the complex Gaussian channel (6.13) or (6.14), the channel transition density is given by

$$p(y|x) = \frac{e^{-\frac{|y-x|^2}{2\sigma^2}}}{2\pi\sigma^2} = \frac{e^{-\frac{(y_c-x_c)^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma} \frac{e^{-\frac{(y_s-x_s)^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma}, \quad (6.18)$$

where the output $y = y_c + jy_s$ and input $x = x_c + jx_s$ can be viewed as complex numbers or two-dimensional real vectors. We typically constrain the input to average symbol energy E_s , and may also constrain it to be drawn from a finite constellation: for example, for M -ary PSK, the input $x \in \{\sqrt{E_s}e^{j2\pi i/M}, i = 0, 1, \dots, M-1\}$. Equation (6.18) makes it transparent that the complex AWGN model is equivalent to two uses of the real model (6.17), where the I component x_c and the Q component x_s of the input may be correlated due to constraints on the input alphabet.

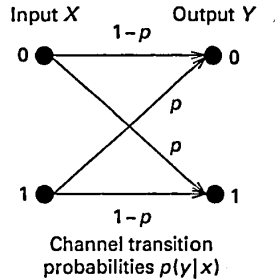
Binary symmetric channel (BSC) In this case, x and y both take values in $\{0, 1\}$, and the transition “density” is now a probability mass function:

$$p(y|x) = \begin{cases} 1-p, & y = x, \\ p, & y = 1-x. \end{cases} \quad (6.19)$$

That is, the BSC is specified by a “crossover” probability p , as shown in Figure 6.3.

Consider BPSK transmission over an AWGN channel. When we make symbol-by-symbol ML decisions, we create a BSC with crossover probability $p = Q(\sqrt{2E_b/N_0})$. Of course, we know that such symbol-by-symbol *hard decisions* are not optimal; for example, ML decoding using the Viterbi algorithm for a convolutional code involves real-valued observations, or *soft decisions*. In Problem 6.10, we quantify the fundamental penalty for hard decisions by comparing the capacity of the BSC induced by hard decisions to the maximum achievable rate on the AWGN channel with BPSK input.

Figure 6.3 Binary symmetric channel with crossover probability p .



6.2.1 Entropy, mutual information, and divergence

We now provide a brief discussion of relevant information-theoretic quantities and discuss their role in the law of large numbers arguments invoked in information theory.

Definition 6.2.2 (Entropy) For a discrete random variable (or vector) X with probability mass function $p(x)$, the entropy $H(X)$ is defined as

$$H(X) = -\mathbb{E}[\log_2 p(X)] = -\sum_i p(x_i) \log_2 p(x_i) \quad \text{Entropy,} \quad (6.20)$$

where $\{x_i\}$ is the set of values taken by X .

Entropy is a measure of the information gained from knowing the value of the random variable X . The more uncertain we are regarding the random variable from just knowing its distribution, the more information we gain when its value is revealed, and the larger its entropy. The information is measured in bits, corresponding to the base 2 used in the logarithms in (6.20).

Example 6.2.1 (Binary entropy) We set aside the special notation $H_B(p)$ for the entropy of a Bernoulli random variable X with $P[X = 1] = p = 1 - P[X = 0]$. From (6.20), we can compute this entropy as

$$H_B(p) = -p \log_2 p - (1-p) \log_2 (1-p) \quad \text{Binary entropy function.} \quad (6.21)$$

Note that $H_B(p) = H_B(1-p)$: as expected, the information content of X does not change if we switch the labels 0 and 1. The binary entropy function is plotted in Figure 6.4. The end points $p = 0$ and $p = 1$ correspond to certainty regarding the value of the random variable, so that no information is gained by revealing its value. On the other hand, $H_B(p)$ attains its maximum value of 1 bit at $p = 1/2$, which corresponds to maximal uncertainty regarding the value of the random variable (which maximizes the information gained by revealing its value).

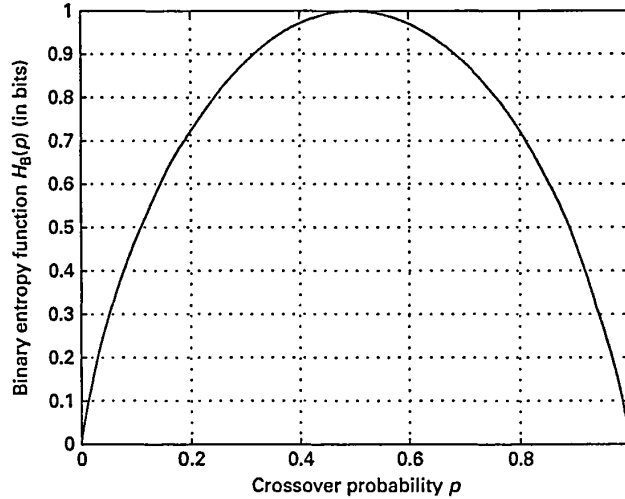
Law of large numbers interpretation of entropy Let $X_1, \dots, X_n$ be i.i.d. random variables, each with pmf $p(x)$. Then their joint pmf satisfies

$$\frac{1}{n} \log_2 p(X_1, \dots, X_n) = \frac{1}{n} \sum_{i=1}^n \log_2 p(X_i) \rightarrow \mathbb{E}[\log_2 p(X_1)] = -H(X), \quad n \rightarrow \infty. \quad (6.22)$$

We can therefore infer that, with high probability, we see the “typical” behavior

$$p(X_1, \dots, X_n) \approx 2^{-nH(X)} \quad \text{typical behavior.} \quad (6.23)$$

Figure 6.4 The binary entropy function.



A sequence that satisfies this behavior is called a *typical* sequence. The set of such sequences is called the *typical set*. The LLN implies that

$$P[(X_1, \dots, X_n) \text{ is typical}] \rightarrow 1, \quad n \rightarrow \infty. \quad (6.24)$$

That is, any sequence of length n that is not typical is extremely unlikely to occur. Using (6.23) and (6.24), we infer that there must be approximately $2^{nH(X)}$ sequences in the typical set. We have thus inferred a very important principle, called the *asymptotic equipartition property (AEP)*, stated informally as follows.

Asymptotic equipartition property (Discrete random variables) For a length n sequence of i.i.d. discrete random variables $X_1, \dots, X_n$, where n is large, the typical set consists of about $2^{nH(X)}$ sequences, each occurring with probability approximately $2^{-nH(X)}$. Sequences outside the typical set occur with negligible probability for large n .

Since $nH(X)$ bits are required to specify the $2^{nH(X)}$ typical sequences, the AEP tells us that describing n i.i.d. copies of the random variable X requires about $nH(X)$ bits, so that the average number of bits per copy of the random variable is $H(X)$. This gives a concrete interpretation for what we mean by entropy measuring information content. The implications for data compression (not considered in detail here) are immediate: by arranging i.i.d. copies of the source in long blocks, we can describe it at rates approaching $H(X)$ per source symbol, by only assigning bits to represent the typical sequences.

We have defined entropy for discrete random variables. We also need an analogous notion for continuous random variables, termed *differential entropy*, defined as follows.

Definition 6.2.3 (Differential entropy) For a continuous random variable (or vector) X with probability density function $p(x)$, the differential entropy $h(X)$ is defined as

$$h(X) = -\mathbb{E}[\log_2 p(X)] = -\int p(x) \log_2 p(x) dx \quad \text{Differential entropy.}$$

Example 6.2.2 (Differential entropy for a Gaussian random variable)
For $X \sim N(m, v^2)$,

$$-\log_2 p(x) = \frac{(x-m)^2}{2v^2} (\log_2 e) + \frac{1}{2} \log_2(2\pi v^2).$$

Thus, we obtain

$$\begin{aligned} h(X) &= -\mathbb{E}[\log_2 p(X)] = \mathbb{E} \left[\frac{(X-m)^2}{2v^2} (\log_2 e) + \frac{1}{2} \log_2(2\pi v^2) \right] \\ &= \frac{1}{2} \log_2 e + \frac{1}{2} \log_2(2\pi v^2). \end{aligned}$$

We summarize as follows:

$$h(X) = \frac{1}{2} \log_2(2\pi e v^2) \quad \text{Differential entropy for Gaussian } N(m, v^2) \text{ random variable.} \quad (6.25)$$

Note that the differential entropy does not depend on the mean, since that is a deterministic parameter that can be subtracted out from X without any loss of information.

Cautionary note There are key differences between entropy and differential entropy. While entropy must be nonnegative, this is not true of differential entropy (e.g., set $v^2 < 1/2\pi e$ in Example 6.2.2). While entropy is scale-invariant, differential entropy is not, even though scaling a random variable by a known constant should not change its information content. These differences can be traced to the differences between probability mass functions and probability density functions. Scaling changes the location of the mass points for a discrete random variable, but does not change their probabilities. On the other hand, scaling changes both the location and size of the infinitesimal intervals used to define a probability density function for a continuous random variable. However, such differences between entropy and differential entropy are irrelevant for our main purpose of computing channel capacities, which, as we shall see, requires computing *differences* between unconditional and conditional entropies or differential entropies. The effect of scale factors “cancels out” when we compute such differences.

Law of large numbers interpretation of differential entropy Let $X_1, \dots, X_n$ be i.i.d. random variables, each with density $f(x)$, then their joint density satisfies

$$\frac{1}{n} \log_2 f(X_1, \dots, X_n) = \frac{1}{n} \sum_{i=1}^n \log_2 f(X_i) \rightarrow \mathbb{E}[\log_2 f(X_1)] = -h(X), \quad n \rightarrow \infty. \quad (6.26)$$

We now define “typical” behavior in terms of the value of the joint density

$$f(X_1, \dots, X_n) \approx 2^{-nh(X)} \quad \text{typical behavior}, \quad (6.27)$$

and invoke the LLN to infer that (6.24) holds. Since the “typical” value of the joint density is a constant, $2^{-nh(X)}$, we infer that the typical set must have volume approximately $2^{nh(X)}$, in order for the joint density to integrate to one. This leads to the AEP for continuous random variables stated below.

Asymptotic equipartition property (Continuous random variables) For a length n sequence of i.i.d. continuous random variables $X_1, \dots, X_n$, where n is large, the joint density takes value approximately $2^{-nh(X)}$ over a typical set of volume $2^{nh(X)}$. The probability mass outside the typical set is negligible for large n .

Joint entropy and mutual information The entropy $H(X, Y)$ of a pair of random variables (X, Y) (e.g., the input and output of a channel) is called the joint entropy of X and Y , and is given by

$$H(X, Y) = -\mathbb{E}[\log_2 p(X, Y)], \quad (6.28)$$

where $p(x, y) = p(x)p(y|x)$ is the joint pmf. The mutual information between X and Y is defined as

$$I(X; Y) = H(X) + H(Y) - H(X, Y). \quad (6.29)$$

Conditional entropy The conditional entropy $H(Y|X)$ is defined as

$$H(Y|X) = -\mathbb{E}[\log_2 p(Y|X)] = -\sum_x \sum_y p(x, y) \log_2 p(y|x). \quad (6.30)$$

Since $p(y|x) = p(x, y)/p(x)$, we have

$$\log_2 p(Y|X) = \log_2 p(X, Y) - \log_2 p(X).$$

Taking expectations and changing sign, we get

$$H(Y|X) = H(X, Y) - H(X).$$

Substituting into (6.29), we get an alternative formula for the mutual information (6.29): $I(X; Y) = H(Y) - H(Y|X)$. By symmetry, we also have

$I(X; Y) = H(X) - H(X|Y)$. For convenience, we state all of these formulas for mutual information together:

$$\begin{aligned} I(X; Y) &= H(Y) - H(Y|X) \\ &= H(X) - H(X|Y) \\ &= H(X) + H(Y) - H(X, Y). \end{aligned} \quad (6.31)$$

It is also useful to define the entropy of Y conditioned on a particular value of $X = x$, as follows:

$$H(Y|X = x) = -\mathbb{E}[\log_2 p(Y|X)|X = x] = -\sum_y p(y|x) \log_2 p(y|x),$$

and note that

$$H(Y|X) = \sum_x p(x) H(Y|X = x). \quad (6.32)$$

The preceding definitions and formulas hold for continuous random variables as well, with entropy replaced by differential entropy.

One final concept that is closely related to entropies is information-theoretic *divergence*, also termed the *Kullback–Leibler (KL) distance*.

Divergence The divergence $D(P||Q)$ between two distributions P and Q (with corresponding densities $p(x)$ and $q(x)$) is defined as

$$D(P||Q) = \mathbb{E}_P \left[\log \left(\frac{p(X)}{q(X)} \right) \right] = \sum_x p(x) \log \left(\frac{p(x)}{q(x)} \right),$$

where $\mathbb{E}_P$ denotes expectation computed using the distribution P (i.e., X is a random variable with distribution P).

Divergence is nonnegative The divergence $D(P||Q) \geq 0$, with equality if and only if $P \equiv Q$.

The proof is as follows:

$$\begin{aligned} -D(P||Q) &= \mathbb{E}_P \left[\log \left(\frac{q(X)}{p(X)} \right) \right] = \sum_{x:p(x)>0} p(x) \log \left(\frac{q(x)}{p(x)} \right) \\ &\leq \sum_{x:p(x)>0} p(x) \left(\frac{q(x)}{p(x)} - 1 \right) = \left(\sum_{x:p(x)>0} q(x) \right) - 1 \leq 0, \end{aligned}$$

where the first inequality is because $\log x \leq x - 1$. Since equality in the latter inequality occurs if and only if $x = 1$, the first inequality is an equality if and only if $q(x)/p(x) = 1$ wherever $p(x) > 0$. The second inequality follows from the fact that q is a pmf, and is an equality if and only if $q(x) = 0$ wherever $p(x) = 0$. Thus, we find that $D(P||Q) = 0$ if and only if $p(x) = q(x)$ for all x (for continuous random variables, the equalities would only need to hold “almost everywhere”).

Mutual information as a divergence The mutual information between two random variables can be expressed as a divergence between their joint distribution, and a distribution corresponding to independent realizations of these random variables, as follows:

$$I(X; Y) = D(P_{X,Y} || P_X P_Y). \quad (6.33)$$

This follows by noting that

$$\begin{aligned} I(X; Y) &= H(X) + H(Y) - H(X, Y) \\ &= -\mathbb{E}[\log p(X)] - \mathbb{E}[\log p(Y)] + \mathbb{E}[\log p(X, Y)] \\ &= \mathbb{E} \left[\log \left(\frac{p(X, Y)}{p(X)p(Y)} \right) \right], \end{aligned}$$

where the expectation is computed using the joint distribution of X and Y .

6.2.2 The channel coding theorem

We first introduce joint typicality, which is the central component of a random coding argument for characterizing the maximum achievable rate on a DMC.

Joint typicality Let X and Y have joint density $p(x, y)$. Then the law of large numbers can be applied to n channel uses with i.i.d. inputs $X_1, \dots, X_n$, leading to outputs $Y_1, \dots, Y_n$, respectively. Note that the pairs (X_i, Y_i) are i.i.d., as are the outputs $\{Y_i\}$. We thus get three LLN-based results:

$$\begin{aligned} \frac{1}{n} \log_2 p(X_1, \dots, X_n) &\rightarrow -H(X) \\ \frac{1}{n} \log_2 p(Y_1, \dots, Y_n) &\rightarrow -H(Y) \\ \frac{1}{n} \log_2 p(X_1, Y_1, \dots, X_n, Y_n) &\rightarrow -H(X, Y). \end{aligned} \quad (6.34)$$

For an input sequence $\mathbf{x} = (x_1, \dots, x_n)^T$ and an output sequence $\mathbf{y} = (y_1, \dots, y_n)^T$, the pair $(\mathbf{x}, \mathbf{y})$ is said to be *jointly typical* if its empirical characteristics conform to the statistical averages in (6.34); that is, if

$$\begin{aligned} p(\mathbf{x}) &\approx 2^{-nH(X)} \\ p(\mathbf{y}) &\approx 2^{-nH(Y)} \\ p(\mathbf{x}, \mathbf{y}) &\approx 2^{-nH(X, Y)}. \end{aligned} \quad (6.35)$$

We also infer that there are about $2^{nH(X, Y)}$ jointly typical sequences, since

$$\sum_{\mathbf{x}, \mathbf{y} \text{ jointly typical}} p(\mathbf{x}, \mathbf{y}) \approx 1.$$

In the following, we apply the concept of joint typicality to a situation in which X is the input to a DMC, and Y its output. In this case, $p(x, y) = p(x)p(y|x)$, where $p(x)$ is the marginal pmf of X , and $p(y|x)$ is the channel transition pmf.

Random coding For communicating at rate R bit/channel use over a DMC $p(y|x)$, we use 2^{nR} codewords, where a codeword of the form $\mathbf{X} = (X_1, \dots, X_n)^T$ is sent using n channel uses (input X_i sent for i th channel use). The elements $\{X_i\}$ are chosen to be i.i.d., drawn from a pmf $p(x)$. Thus, all elements in all codewords are i.i.d., hence the term random coding (of course, the encoder and decoder both know the set of codewords once the random codebook choice has been made). All codewords are equally likely to be sent.

Joint typicality decoder While ML decoding is optimal for equiprobable transmission, it suffices to consider the following *joint typicality* decoder for our purpose. This decoder checks whether the received vector $\mathbf{Y} = (Y_1, \dots, Y_n)$ is jointly typical with any codeword $\hat{\mathbf{X}} = (\hat{X}_1, \dots, \hat{X}_n)^T$. If so, and if there is exactly one such codeword, then the decoder outputs $\hat{\mathbf{X}}$. If not, it declares decoding failure. Decoding error occurs if $\hat{\mathbf{X}} \neq \mathbf{X}$, where $\mathbf{X}$ is the transmitted codeword. Let us now estimate the probability of decoding error or failure.

If $\mathbf{X}$ is the transmitted codeword, and $\hat{\mathbf{X}}$ is any other codeword, then $\hat{\mathbf{X}}$ and the output $\mathbf{Y}$ are independent by our random coding construction, so that $p(\hat{\mathbf{X}}, \mathbf{Y}) = p(\hat{\mathbf{X}})p(\mathbf{Y}) \approx 2^{-n(H(\hat{\mathbf{X}}) + H(\mathbf{Y}))}$ if $\hat{\mathbf{X}}$ and $\mathbf{Y}$ are typical. Now, the probability that they are jointly typical is

$$\begin{aligned} P[\hat{\mathbf{X}}, \mathbf{Y} \text{ jointly typical}] &= \sum_{\hat{\mathbf{X}}, \mathbf{Y} \text{ jointly typical}} p(\hat{\mathbf{X}})p(\mathbf{Y}) \\ &\approx 2^{nH(\hat{\mathbf{X}}, \mathbf{Y})} 2^{-n(H(\hat{\mathbf{X}}) + H(\mathbf{Y}))} = 2^{-nI(\hat{\mathbf{X}}; \mathbf{Y})}. \end{aligned}$$

Since there are $2^{nR} - 1$ possible incorrect codewords, the probability of at least one of them being jointly typical with the received vector can be estimated using the union bound

$$(2^{nR} - 1)2^{-nI(\hat{\mathbf{X}}; \mathbf{Y})} \leq 2^{-n(I(\hat{\mathbf{X}}; \mathbf{Y}) - R)}, \quad (6.36)$$

which tends to zero as $n \rightarrow \infty$, as long as $R < I(\hat{\mathbf{X}}; \mathbf{Y})$.

There are some other possible events that lead to decoding error that we also need to estimate (but that we omit here). However, the estimate (6.36) is the crux of the random coding argument for the “forward” part of the noisy channel coding theorem, which we now state below.

Theorem 6.2.1 (Channel coding theorem: achievability)

(a) For a DMC with channel transition pmf $p(y|x)$, we can use i.i.d. inputs with pmf $p(x)$ to communicate reliably, as long as the code rate satisfies

$$R < I(X; Y).$$

(b) The preceding achievable rate can be maximized over the input density $p(x)$ to obtain the channel capacity

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

We omit detailed discussion and proof of the “converse” part of the channel coding theorem, which states that it is not possible to do better than the achievable rates promised by the preceding theorem.

Note that, while we considered discrete random variables for concreteness, the preceding discussion goes through unchanged for continuous random variables (as well as for mixed settings, such as when X is discrete and Y is continuous), by appropriately replacing entropy by differential entropy.

6.3 Some capacity computations

We are now ready to undertake some example capacity computations. In Section 6.3.1, we discuss capacity computations for guiding the choice of signal constellations and code rates on the AWGN channel. Specifically, for a given constellation, we wish to establish a benchmark on the best rate that it can achieve on the AWGN channel as a function of SNR. Such a result is nonconstructive, saying only that there is *some* error-correcting code which, when used with the constellation, achieves the promised rate (and that no code can achieve reliable communication at a higher rate). However, as mentioned earlier, it is usually possible with a moderate degree of ingenuity to obtain a turbo-like coded modulation scheme that approaches these benchmarks quite closely. Thus, the information-theoretic benchmarks provide valuable guidance on choice of constellation and code rate. We then discuss the parallel Gaussian channel model, and its application to modeling dispersive channels, in Section 6.3.2. The optimal “waterfilling” power allocation for this model is an important technique that appears in many different settings.

6.3.1 Capacity for standard constellations

We now compute mutual information for some examples. We term the maximum mutual information attained under specific input constraints as the channel capacity under those constraints. For example, we compute the capacity of the AWGN channel with BPSK signaling and a power constraint. This is, of course, smaller than the capacity of power-constrained AWGN signaling when there are no constraints on the input alphabet, which is what we typically refer to as *the* capacity of the AWGN channel.

Binary symmetric channel capacity Consider the BSC with crossover probability p as in Figure 6.3. Given the symmetry of the channel, it is plausible that the optimal input distribution is to send 0 and 1 with equal probability (see Section 6.4 for techniques for validating such guesses, as well as for computing optimal input distributions when the answer is not

6.3 Some capacity computations

“obvious”). We now calculate $C = I(X; Y) = H(Y) - H(Y|X)$. By symmetry, the resulting output distribution is also uniform over $\{0, 1\}$, so that

$$H(Y) = -\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} = 1.$$

Now,

$$\begin{aligned} H(Y|X=0) &= -p(Y=1|X=0) \log_2 p(Y=1|X=0) - p(Y=0|X=0) \times \\ &\quad \log_2 p(Y=0|X=0) \\ &= -p \log_2 p - (1-p) \log_2 (1-p) = H_B(p), \end{aligned}$$

where $H_B(p)$ is the entropy of a Bernoulli random variable with probability p of taking the value one. By symmetry, we also have $H(Y|X=1) = H_B(p)$, so that, from (6.32), we get

$$H(Y|X) = H_B(p).$$

We therefore obtain the capacity of the BSC with crossover probability p as

$$C_{\text{BSC}}(p) = 1 - H_B(p). \quad (6.37)$$

AWGN channel capacity Consider the channel model (6.10), with the observation

$$Y = X + Z$$

with input $E[X^2] \leq E_s$ and $Z \sim N(0, \sigma^2)$. We wish to compute the capacity

$$C = \max_{p(x): E[X^2] \leq E_s} I(X; Y).$$

Given $X = x$, $h(Y|X=x) = h(Z)$, so that $h(Y|X) = h(Z)$. We therefore have

$$I(X; Y) = h(Y) - h(Z), \quad (6.38)$$

so that maximizing mutual information is equivalent to maximizing $h(Y)$. Since X and Z are independent (the transmitter does not know the noise realization Z), we have $E[Y^2] = E[X^2] + E[Z^2] \leq E_s + \sigma^2$. Subject to this constraint, it follows from Problem 6.3 that $h(Y)$ is maximized if Y is zero mean Gaussian. This is achieved if the input distribution is $X \sim N(0, E_s)$, independent of the noise Z , which yields $Y \sim N(0, E_s + \sigma^2)$. Substituting the expression (6.25) for the entropy of a Gaussian random variable into (6.38), we obtain the capacity:

$$\begin{aligned} I(X; Y) &= \frac{1}{2} \log_2(2\pi e(E_s + \sigma^2)) - \frac{1}{2} \log_2(2\pi e\sigma^2) \\ &= \frac{1}{2} \log_2\left(1 + \frac{E_s}{\sigma^2}\right) = \frac{1}{2} \log_2(1 + \text{SNR}); \end{aligned}$$

the same formula that we got from the sphere packing arguments. We have now in addition proved that this capacity is attained by Gaussian input $X \sim N(0, E_s)$.

We now consider the capacity of the AWGN channel when the signal constellation is constrained.

Example 6.3.1 (AWGN capacity with BPSK signaling) Let us first consider BPSK signaling, for which we have the channel model

$$Y = \sqrt{E_s}X + Z, \quad X \in \{-1, +1\}, \quad Z \sim \mathcal{N}(0, \sigma^2).$$

It can be shown (e.g., using the techniques to be developed in Section 6.4.1) that the mutual information $I(X; Y)$, subject to the constraint of BPSK signaling, is maximized for equiprobable signaling. Let us now compute the mutual information $I(X; Y)$ as a function of the signal power E_s and the noise power σ^2 . We first show that, as with the capacity without an input alphabet constraint, the capacity for BPSK also depends on these parameters only through their ratio, the SNR E_s/σ^2 . To show this, replace Y by Y/σ to get the model

$$Y = \sqrt{\text{SNR}} X + Z, \quad X \in \{-1, +1\}, \quad Z \sim \mathcal{N}(0, 1). \quad (6.39)$$

For notational simplicity, set $A = \sqrt{\text{SNR}}$. We have

$$p(y|+1) = \frac{1}{\sqrt{2\pi}} e^{-(y-A)^2/2},$$

$$p(y|-1) = \frac{1}{\sqrt{2\pi}} e^{-(y+A)^2/2},$$

and

$$p(y) = \frac{1}{2}p(y|+1) + \frac{1}{2}p(y|-1). \quad (6.40)$$

We can now compute

$$I(X; Y) = h(Y) - h(Y|X).$$

As before, we can show that $h(Y|X) = h(Z) = 1/2 \log_2(2\pi e)$. We can now compute

$$h(Y) = - \int \log_2(p_Y(y)) p_Y(y) dy$$

by numerical integration, plugging in (6.40). An alternative approach, which is particularly useful for more complicated constellations and channel models, is to use Monte Carlo integration (i.e., simulation-based empirical averaging) for computing the expectation $h(Y) = -\mathbb{E}[\log_2 p(Y)]$. For this method, we generate i.i.d. samples $Y_1, \dots, Y_n$ using the model (6.39), and then use the estimate

$$\hat{h}(Y) = -\frac{1}{n} \sum_{i=1}^n \log_2 p(Y_i).$$

We can also use the alternative formula

$$I(X; Y) = H(X) - H(X|Y)$$

to compute the capacity. For equiprobable binary input, $H(X) = H_B(\frac{1}{2}) = 1$ bit/symbol. It remains to compute

$$H(X|Y) = \int H(X|Y=y)p_Y(y) dy. \quad (6.41)$$

By Bayes' rule, we have

$$\begin{aligned} P[X = +1|Y = y] &= \frac{P[X = +1]p(y|+1)}{p(y)} \\ &= \frac{P[X = +1]p(y|+1)}{P[X = +1]p(y|+1) + P[X = -1]p(y|-1)} \\ &= \frac{e^{Ay}}{e^{Ay} + e^{-Ay}} \quad (\text{equal priors}). \end{aligned}$$

We also have

$$P[X = -1|Y = y] = 1 - P[X = +1|Y = y] = \frac{e^{-Ay}}{e^{Ay} + e^{-Ay}}.$$

Such a posteriori probability computations can be thought of as *soft decisions* on the transmitted bits, and are employed extensively when we discuss iterative decoding. We can now use the binary entropy function to compute

$$H(X|Y=y) = H_B(P[X = +1|Y = y]).$$

The average in (6.41) can now be computed by direct numerical integration or by Monte Carlo integration as before. The latter, which generalizes better to more complex models, gives the estimate

$$\hat{H}(X|Y) = \frac{1}{n} \sum_{i=1}^n H_B(P[X = +1|Y_i = y_i]).$$

The preceding methodology generalizes in a straightforward manner to PAM constellations. For complex-valued constellations, we need to consider the complex discrete-time AWGN channel model (6.13). For rectangular QAM constellations, one use of the complex channel with QAM input is equivalent to two uses of a real channel using PAM input, so that the same methodology applies again. However, for complex-valued constellations that cannot be decomposed in this fashion (e.g., 8-PSK and other higher order PSK alphabets), we must work directly with the complex AWGN channel model (6.18).

Example 6.3.2 (Capacity with PSK signaling) For PSK signaling over the complex AWGN channel (6.13), we have the model:

$$Y = X + Z,$$

where $X \in A = \{\sqrt{\text{SNR}} e^{j2\pi i/M}, i = 0, 1, \dots, M-1\}$ and $Z \sim \text{CN}(0, 1)$ with density

$$p_Z(z) = \frac{e^{-|z|^2}}{\pi}, \quad z \text{ complex-valued},$$

where we have normalized the noise to obtain a scale-invariant model. As before, for an additive noise model in which the noise is independent of the input, we have $h(Y|X) = h(Z)$. The differential entropy of the proper complex Gaussian random variable Z can be inferred from that for a real Gaussian random variable given by (6.25), or by specializing the formula in Problem 6.4(b). This yields that

$$h(Z) = \log_2 \pi e.$$

Furthermore, assuming that a uniform distribution achieves capacity (this can be proved using the techniques in Section 6.4.2), we have

$$p_Y(y) = \frac{1}{M} \sum_{x \in A} p_Z(y-x) = \frac{1}{M} \sum_{i=0}^{M-1} \frac{1}{\pi} \exp\left(-|y - \sqrt{\text{SNR}} e^{j2\pi i/M}|^2\right).$$

We can now use Monte Carlo integration to compute $h(Y)$, and then compute the mutual information $I(X; Y) = h(Y) - h(Z)$.

Figure 6.5 plots the capacity (in bits per channel use) for QPSK, 16PSK and 16-QAM versus SNR (dB).

Power-bandwidth tradeoffs Now that we know how to compute capacity as a function of SNR for specific constellations using the discrete-time AWGN channel model, we can relate it, as discussed in Section 6.1.4, back to the

Figure 6.5 The capacity of the AWGN channel with different constellations as a function of SNR.

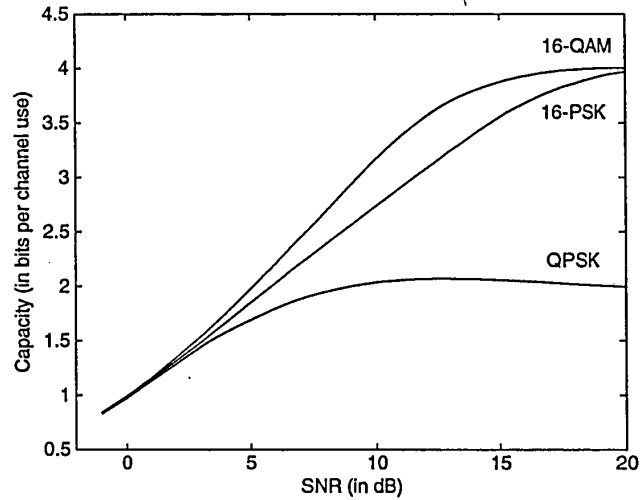
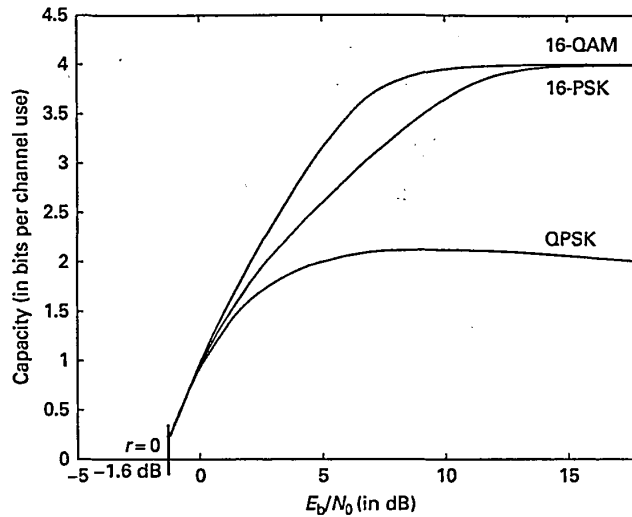


Figure 6.6 The capacity of the AWGN channel with different constellations as a function of E_b/N_0 .



continuous-time AWGN channel to understand the tradeoff between power efficiency and spectral efficiency. As shown in Section 6.1.4, we have, for both the real and complex models,

$$\text{SNR} = r \frac{E_b}{N_0},$$

where r is the spectral efficiency in bit/s per Hz (not accounting for excess bandwidth requirements imposed by implementation considerations).

For a given complex-valued constellation A , suppose that the capacity in discrete time is $C_A(\text{SNR})$ bit/channel use. Then, using (6.15), we find that the feasible region for communication using this constellation is given, as a function of E_b/N_0 , by

$$r < C_A \left(r \frac{E_b}{N_0} \right), \quad \text{complex-valued constellation } A. \quad (6.42)$$

Figure 6.6 expresses the plots in Figure 6.5 in terms of E_b/N_0 (dB), obtained by numerically solving for equality in (6.42). (Actually, for each value of SNR, we compute the capacity, and then the corresponding E_b/N_0 value, and then plot the latter two quantities against each other.)

For real-valued constellations such as 4-PAM, we would use (6.11), and obtain

$$r < 2C_A \left(r \frac{E_b}{N_0} \right), \quad \text{real-valued constellation } A. \quad (6.43)$$

6.3.2 Parallel Gaussian channels and waterfilling

A useful generalization of the AWGN channel model is the parallel Gaussian channel model (we can use this to model both dispersive channels and colored

noise, as we shall see shortly), stated as follows. We have access to K parallel complex Gaussian channels, with the output of the k th channel, $1 \leq k \leq K$, modeled as

$$Y_k = h_k X_k + Z_k,$$

where h_k is the channel gain, $Z_k \sim CN(0, N_k)$ is the noise on the k th channel, and $E[|X_k|^2] = P_k$, with a constraint on the total power:

$$\sum_{k=1}^K P_k \leq P. \quad (6.44)$$

The noises $\{Z_k\}$ are independent across channels, as well as across time. The channel is characterized by the gains $\{h_k\}$ and the noise variances $\{N_k\}$. The goal is to derive the capacity of this channel model for fixed $\{P_k\}$, which is appropriate when the transmitter does not know the channel characteristics, as well as to optimize the capacity over the power allocation $\{P_k\}$ when the channel characteristics are known at the transmitter.

The mutual information between the input vector $\mathbf{X} = (X_1, \dots, X_K)$ and the output vector $\mathbf{Y} = (Y_1, \dots, Y_K)$ is given by

$$I(\mathbf{X}; \mathbf{Y}) = h(\mathbf{Y}) - h(\mathbf{Y}|\mathbf{X}) = h(\mathbf{Y}) - h(\mathbf{Z}).$$

Owing to the independence of the noises, $h(\mathbf{Z}) = \sum_{k=1}^K h(Z_k)$. Furthermore, we can bound the joint differential entropy of the output as the sum of the individual differential entropies:

$$h(\mathbf{Y}) = h(Y_1, \dots, Y_K) \leq \sum_{k=1}^K h(Y_k)$$

with equality if $Y_1, \dots, Y_K$ are independent. Thus, we obtain

$$I(\mathbf{X}; \mathbf{Y}) \leq \sum_{k=1}^K (h(Y_k) - h(Z_k)).$$

Each of the K terms on the right-hand side can be maximized as for a standard Gaussian channel, by choosing $X_k \sim CN(0, P_k)$. We therefore find that, for a given power allocation $\mathbf{P} = (P_1, \dots, P_K)$, the capacity is given by

$$C(\mathbf{P}) = \sum_{k=1}^K \log_2 \left(1 + \frac{|h_k|^2 P_k}{N_k} \right) \quad \text{Fixed power allocation} \quad (6.45)$$

(the received signal power on the k th channel is $S_k = |h_k|^2 P_k$). The preceding development also holds for real-valued parallel Gaussian channels, except that a factor of $1/2$ must be inserted in (6.45).

Optimizing the power allocation We can now optimize $C(\mathbf{P})$ subject to the constraint (6.44) by maximizing the Lagrangian

$$J(\mathbf{P}) = C(\mathbf{P}) + \lambda \sum_{k=1}^K P_k = \sum_{k=1}^K \log_2 \left(1 + \frac{|h_k|^2 P_k}{N_k} \right) - \lambda \sum_{k=1}^K P_k. \quad (6.46)$$

We discuss the theory of such convex optimization problems very briefly in Section 6.4.1, but for now, it suffices to note that we can optimize $J(\mathbf{P})$ by setting the partial derivative with respect to P_k to zero. This yields

$$0 = \frac{\partial J(\mathbf{P})}{\partial P_k} = \frac{|h_k|^2/N_k}{1 + \frac{|h_k|^2 P_k}{N_k}} - \lambda,$$

so that

$$P_k = a - \frac{N_k}{|h_k|^2}$$

for some constant a . However, we must also satisfy the constraint that $P_k \geq 0$. We therefore get the following solution.

Waterfilling power allocation

$$P_k = \begin{cases} a - \frac{N_k}{|h_k|^2}, & \frac{N_k}{|h_k|^2} \leq a, \\ 0, & \frac{N_k}{|h_k|^2} > a, \end{cases} \quad (6.47)$$

where a is chosen so as to satisfy the power constraint with equality:

$$\sum_{k=1}^K P_k = P.$$

This has the waterfilling interpretation depicted in Figure 6.7. The water level a is determined by pouring water until the net amount equals the power budget P . Thus, if the normalized noise level $N_k/|h_k|^2$ is too large for a channel, then it does not get used (the corresponding $P_k = 0$ in the optimal allocation).

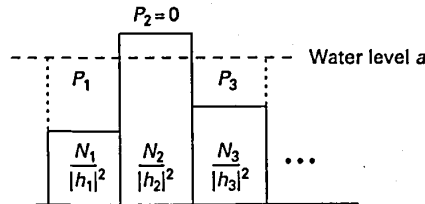
Application to dispersive channels The parallel Gaussian model provides a means of characterizing the capacity of a dispersive channel with impulse response $h(t)$ (we are working in complex baseband now), by signaling across parallel frequency bins. Let us assume colored proper complex Gaussian noise with PSD $S_n(f)$. Then a frequency bin of width Δf around f_k follows the model

$$Y_k = H(f_k)X_k + Z_k,$$

where $N_k \sim CN(0, S_n(f_k)\Delta f)$ and $E[|X_k|^2] = S_s(f_k)\Delta f$, where $S_s(f)$ is the PSD of the input (which must be proper complex Gaussian to achieve capacity). Note that the SNR on the channel around f_k is

$$\text{SNR}(f_k) = \frac{|H(f_k)|^2 S_s(f_k)\Delta f}{S_n(f_k)\Delta f} = \frac{|H(f_k)|^2 S_s(f_k)}{S_n(f_k)}.$$

Figure 6.7 Waterfilling power allocation for the parallel Gaussian channel.



The net capacity is now given by

$$\sum_k \Delta f \log_2 (1 + \text{SNR}(f_k)).$$

By letting $\Delta f \rightarrow 0$, the sum above tends to an integral, and we obtain the capacity as a function of the input PSD:

$$C(S_s) = \int_{-W/2}^{W/2} \log_2 \left(1 + \frac{|H(f)|^2 S_s(f)}{S_n(f)} \right) df, \quad (6.48)$$

where W is the channel bandwidth, and the input power is given by

$$\int_{-W/2}^{W/2} S_s(f) df = P. \quad (6.49)$$

This reduces to the formula (6.1) for the complex baseband AWGN channel by setting $H(f) \equiv 1$, $S_s(f) \equiv P/W$ and $S_n(f) \equiv N_0$ for $-W/2 \leq f \leq W/2$.

Waterfilling can now be used to determine the optimum input PSD as follows:

$$S_s(f) = \begin{cases} a - \frac{S_n(f)}{|H(f)|^2}, & \frac{S_n(f)}{|H(f)|^2} \leq a, \\ 0, & \frac{S_n(f)}{|H(f)|^2} > a, \end{cases} \quad (6.50)$$

with a chosen to satisfy the power constraint (6.49).

An important application of the parallel Gaussian model is orthogonal frequency division multiplexing (OFDM), also called discrete multitone, in which data are modulated onto a discrete set of subcarriers in parallel. Orthogonal frequency division multiplexing is treated in detail in Chapter 8, where we focus on its wireless applications. However, OFDM has also been successfully applied to dispersive wireline channels such as digital subscriber loop (DSL). In such settings, the channel can be modeled as time-invariant, and can be learnt by the transmitter using channel sounding and receiver feedback. Waterfilling, appropriately modified to reflect practical constraints such as available constellation choices and the gap to capacity for the error correction scheme used, then plays an important role in optimizing the constellations to be used on the different subcarriers, with larger constellations being used on subcarriers seeing a better channel gain.

6.4 Optimizing the input distribution

We have shown how to compute mutual information between the input and output of a discrete memoryless channel (DMC) for a given input distribution. For finite constellations over the AWGN channel, we have, for example, considered input distributions that are uniform over the alphabet. This is an intuitively pleasing and practical choice, and indeed, it is optimal in certain situations, as we shall show. However, the optimal input distribution

6.4 Optimizing the input distribution

is by no means obvious in all cases, hence it is important to develop a set of tools for characterizing and computing it in general. The key ideas behind developing such tools are as follows:

- (a) Mutual information is a concave function of the input distribution, hence a unique maximizing input distribution exists.
- (b) There are necessary and sufficient conditions for optimality that can easily be used to check guesses regarding the optimal input distribution. However, directly solving for the optimal input distribution based on these conditions is difficult.
- (c) An iterative algorithm to find the optimal input distribution can be obtained by writing the maximum mutual information as the solution to a two-stage maximization problem, such that it is easy to solve each stage. Convergence to the optimal input distribution is obtained by alternating between the two stages. This algorithm is referred to as the Blahut–Arimoto algorithm.

We begin with a brief discussion of concave functions and their maximization. We apply this to obtain necessary and sufficient conditions that must be satisfied by the optimal input distribution. We end with a discussion of the Blahut–Arimoto algorithm.

6.4.1 Convex optimization

A set C is *convex* if, given $x_1, x_2 \in C$, $\lambda x_1 + (1 - \lambda)x_2 \in C$ for any $\lambda \in [0, 1]$. We are interested in optimizing mutual information over a set of probability distributions, which is a convex set. Thus, we consider functions whose arguments lie in a convex set.

A function $f(x)$ (whose argument may be a real or complex vector x in a convex set C) is *convex* (also termed *convex up*) if

$$f(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f(x_1) + (1 - \lambda)f(x_2) \quad (6.51)$$

for any x_1, x_2 , and any $\lambda \in [0, 1]$. That is, the line joining any two points on the graph of the function lies above the function.

Similarly, $f(x)$ is *concave* (also termed *convex down*) if

$$f(\lambda x_1 + (1 - \lambda)x_2) \geq \lambda f(x_1) + (1 - \lambda)f(x_2). \quad (6.52)$$

From the preceding definitions, it is easy to show that linear combinations of convex (concave) functions are convex (concave). Also, the negative of a convex function is concave, and vice versa. Affine functions (i.e., linear functions plus constants) are both convex and concave, since they satisfy (6.51) and (6.52) with equality.

Example 6.4.1 A twice differentiable function $f(x)$ with a one-dimensional argument x is convex if $f''(x) \geq 0$, and concave if $f''(x) \leq 0$. Thus, $f(x) = x^2$ is convex, $f(x) = \log x$ is concave, and a line has second derivative zero, and is therefore both convex and concave.

Entropy is a concave function of the probability density/mass function
 The function $f(x) = -x \log x$ is concave (verify by differentiating twice). Use this to show that

$$-\sum_x p(x) \log p(x)$$

is concave in the probability mass function $\{p(x)\}$, where the latter is viewed as a real-valued vector.

Mutual information between input and output of a DMC is a concave function of the input probability distribution The mutual information is given by $I(X; Y) = H(Y) - H(Y|X)$. The output entropy $H(Y)$ is a concave function of p_Y , and p_Y is a linear function of p_X . It is easy to show, proceeding from the definition (6.52) that $H(Y)$ is a concave function of p_X . The conditional entropy $H(Y|X)$ is easily seen to be a linear function of p_X .

Kuhn–Tucker conditions for constrained maximization of a concave function We state without proof necessary and sufficient conditions for optimality for a special case of constrained optimization, which are specializations of the so-called Kuhn–Tucker conditions for constrained convex optimization. Suppose that $f(\mathbf{x})$ is a concave function to be maximized over $\mathbf{x} = (x_1, \dots, x_m)^T$, subject to the constraints $x_k \geq 0$, $1 \leq k \leq m$, and $\sum_{k=1}^m x_k = c$, where c is a constant. Then the following conditions are necessary and sufficient for optimality: for $1 \leq k \leq m$, we have

$$\begin{aligned} \frac{\partial f}{\partial x_k} &= \lambda, & x_k &> 0, \\ \frac{\partial f}{\partial x_k} &\leq \lambda, & x_k &= 0, \end{aligned} \tag{6.53}$$

for a value of λ such that $\sum_k x_k = c$.

We can interpret the Kuhn–Tucker conditions in terms of the Lagrangian for the constrained optimization problem at hand:

$$J(\mathbf{x}) = f(\mathbf{x}) - \lambda \sum_k x_k.$$

For $x_k > 0$, we set $\partial/\partial x_k J(\mathbf{x}) = 0$. For a point on the boundary with $x_k = 0$, the performance must get worse when we move in from the boundary by increasing x_k , so that $\partial/\partial x_k J(\mathbf{x}) \leq 0$.

We apply these results in the next section to characterize optimal input distributions for a DMC.

6.4.2 Characterizing optimal input distributions

A capacity-achieving input distribution must satisfy the following conditions.

Necessary and sufficient conditions for optimal input distribution For a DMC with transition probabilities $p(y|x)$, an input distribution $p(x)$ is optimal, achieving a capacity C , if and only if, for each input x_k ,

6.4 Optimizing the input distribution

$$\begin{aligned} D(P_{Y|X=x_k}||P_Y) &= C, & p(x_k) > 0, \\ D(P_{Y|X=x_k}||P_Y) &\leq C, & p(x_k) = 0. \end{aligned} \quad (6.54)$$

Interpretation of optimality condition We show that the mutual information is the average of the terms $D(P_{Y|X=x}||P_Y)$ as follows.

$$\begin{aligned} I(X; Y) &= D(P_{X,Y}||P_X P_Y) = \sum_{x,y} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right) \\ &= \sum_x p(x) \sum_y p(y|x) \log \left(\frac{p(y|x)}{p(y)} \right) \\ &= \sum_x p(x) D(P_{Y|X=x}||P_Y). \end{aligned} \quad (6.55)$$

The optimality conditions state that each term making a nontrivial contribution to the average mutual information must be equal. That is, each term equals the average, which for the optimal input distribution equals the capacity C . Terms corresponding to $p(x) = 0$ do not contribute to the average, and are smaller (otherwise we could get a bigger average by allocating probability mass to them). We now provide a proof of these conditions.

Proof The Kuhn–Tucker conditions for capacity maximization are as follows:

$$\begin{aligned} \frac{\partial}{\partial p(x)} I(X; Y) - \lambda &= 0, & p(x) > 0, \\ \frac{\partial}{\partial p(x)} I(X; Y) - \lambda &\leq 0, & p(x) = 0. \end{aligned} \quad (6.56)$$

Now to evaluate the partial derivatives of $I(X; Y) = H(Y) - H(Y|X)$. Since

$$H(Y) = - \sum_y p(y) \log p(y),$$

we have, using the chain rule,

$$\begin{aligned} \frac{\partial}{\partial p(x_k)} H(Y) &= \sum_y \left[\frac{\partial H(Y)}{\partial p(y)} \frac{\partial p(y)}{\partial p(x_k)} \right] \\ &= \sum_y [-1 - \log p(y)] p(y|x_k) \\ &= -1 - \sum_y p(y|x_k) \log p(y). \end{aligned} \quad (6.57)$$

Also,

$$H(Y|X) = - \sum_{x,y} p(x)p(y|x) \log p(y|x),$$

so that

$$\frac{\partial}{\partial p(x_k)} H(Y|X) = - \sum_y p(y|x_k) \log p(y|x_k) = H(Y|X = x_k). \quad (6.58)$$

Using (6.57) and (6.58), we obtain

$$\frac{\partial}{\partial p(x_k)} I(X; Y) = -1 + \sum_y p(y|x_k) \log \frac{p(y|x_k)}{p(y)} = -1 + D(P_{Y|X=x_k} \| P_Y). \quad (6.59)$$

Plugging into (6.56), we get $D(P_{Y|X=x_k} \| P_Y) \leq \lambda + 1$, with equality for $p(x_k) > 0$. Averaging over the input distribution, we realize from (6.55) that we must have $\lambda + 1 = C$, completing the proof. $\square$

Remark While we prove all results for discrete random variables, their natural extensions to continuous random variables hold, with probability mass functions replaced by probability density functions, and summations replaced by integrals. In what follows, we use the term *density* to refer to either probability mass function or probability density function.

Symmetric channels The optimality conditions (6.54) impose a symmetry in the input–output relation. When the channel transition probabilities exhibit a natural symmetry, it often suffices to pick an input distribution that is uniform over the alphabet to achieve capacity. Rather than formally characterizing the class of symmetric channels for which this holds, we leave it to the reader to check, for example, that uniform inputs work for the BSC, and for PSK constellations over the AWGN channel.

While the conditions (6.54) are useful for checking guesses as to the optimal input distribution, they do not provide an efficient computational procedure for obtaining the optimal input distribution. For channels for which guessing the optimal distribution is difficult, a general procedure for computing it is provided by the Blahut–Arimoto algorithm, which we describe in the next section.

6.4.3 Computing optimal input distributions

A key step in the Blahut–Arimoto algorithm is the following lemma, which expresses mutual information as the solution to a maximization problem with an explicit solution. This enables us to write the maximum mutual information, or capacity, as the solution to a double maximization that can be obtained by an alternating maximization algorithm.

Lemma 6.4.1 The mutual information between X and Y can be written as

$$I(X; Y) = \max_{q(x|y)} \sum_{x,y} p(x)p(y|x) \log \frac{q(x|y)}{p(x)},$$

where $\{q(x|y)\}$ is a set of conditional densities for X (that is, $\sum_x q(x|y) = 1$ for each y). The maximum is achieved by the conditional distribution $p(x|y)$ that is consistent with $p(x)$ and $p(y|x)$. That is, the optimizing q is given by

$$q^*(x|y) = \frac{p(x)p(y|x)}{\sum_{x'} p(x')p(y|x')}.$$

Proof We show that the difference in values attained by q^* and any other q is nonnegative as follows:

$$\begin{aligned}
 \sum_{x,y} p(x)p(y|x) \log \frac{q^*(x|y)}{p(x)} - \sum_{x,y} p(x)p(y|x) \log \frac{q(x|y)}{p(x)} \\
 &= \sum_{x,y} p(x)p(y|x) \log \frac{q^*(x|y)}{q(x|y)} \\
 &= \sum_y p(y) \sum_x q^*(x|y) \log \frac{q^*(x|y)}{q(x|y)} \\
 &= \sum_y p(y) D(Q^*(\cdot|y) || Q(\cdot|y)) \geq 0,
 \end{aligned}$$

where we have used $p(x, y) = p(x)p(y|x) = p(y)p(x|y) = p(y)q^*(x|y)$, and where $Q^*(\cdot|y)$, $Q(\cdot|y)$ denote the conditional distributions corresponding to the conditional densities $q^*(x|y)$ and $q(x|y)$, respectively. $\square$

The capacity of a DMC characterized by transition densities $\{p(y|x)\}$ can now be written as

$$C = \max_{p(x)} I(X; Y) = \max_{p(x)} \max_{q(x|y)} \sum_{x,y} p(x)p(y|x) \log \frac{q(x|y)}{p(x)}.$$

We state without proof that an alternating maximization algorithm, which maximizes over $q(x|y)$ keeping $p(x)$ fixed, and then maximizes over $p(x)$ keeping $q(x|y)$ fixed, converges to the global optimum. The utility of this procedure is that each maximization can be carried out explicitly. The lemma provides an explicit form for the optimal $q(x|y)$ for fixed $p(x)$. It remains to provide an explicit form for the optimal $p(x)$ for fixed $q(x|y)$. To this end, consider the Lagrangian

$$\begin{aligned}
 J(p) &= \sum_{x,y} p(x)p(y|x) \log \frac{q(x|y)}{p(x)} - \lambda \sum_x p(x) \\
 &= \sum_{x,y} [p(x)p(y|x) \log q(x|y) - p(x)p(y|x) \log p(x)] - \lambda \sum_x p(x), \quad (6.60)
 \end{aligned}$$

corresponding to the usual sum constraint $\sum_x p(x) = 1$. Setting partial derivatives to zero, we obtain

$$\frac{\partial}{\partial p(x_k)} J(p) = \sum_y [p(y|x_k) \log q(x_k|y) - p(y|x_k) - p(y|x_k) \log p(x_k)] - \lambda = 0.$$

Noting that $\sum_y p(y|x_k) = 1$, we get

$$\log p(x_k) = -\lambda - 1 + \sum_y p(y|x_k) \log q(x_k|y),$$

from which we conclude that

$$p^*(x_k) = K \exp\left(\sum_y p(y|x_k) \log q(x_k|y)\right) = K \Pi_y (q(x_k|y))^{p(y|x_k)},$$

where the constant K is chosen so that $\sum_x p^*(x) = 1$.

We can now state the Blahut–Arimoto algorithm for computing optimal input distributions.

Blahut–Arimoto algorithm

Step 0 Choose an initial guess $p(x)$ for input distribution, ensuring that there is nonzero probability mass everywhere that the optimal input distribution is expected to have nonzero probability mass (e.g., for finite alphabets, a uniform distribution is a safe choice).

Step 1 For the current $p(x)$, compute the optimal $q(x|y)$ using

$$q^*(x|y) = \frac{p(x)p(y|x)}{\sum_x p(x)p(y|x)}.$$

Set this to be the current $q(x|y)$.

Step 2 For the current $q(x|y)$, compute the optimal $p(x)$ using

$$p^*(x) = \frac{\prod_y (q(x|y))^{p(y|x)}}{\sum_x \prod_y (q(x'|y))^{p(y|x)}}.$$

Set this to be the current $p(x)$. Go back to Step 1.

Alternate Steps 1 and 2 until convergence (using any sensible stopping criterion to determine when the changes in $p(x)$ are sufficiently small).

Example 6.4.2 (Blahut–Arimoto algorithm applied to BSC) For a BSC with crossover probability α , we know that the optimal input distribution is uniform. However, let us apply the Blahut–Arimoto algorithm, starting with an arbitrary input distribution $P[X=1] = p = 1 - P[X=0]$, where $0 < p < 1$. We can now check that Step 1 yields

$$q(1|0) = \frac{p\alpha}{p\alpha + (1-p)(1-\alpha)} = 1 - q(0|0)$$

$$q(0|1) = \frac{(1-p)\alpha}{(1-p)\alpha + p(1-\alpha)} = 1 - q(1|1)$$

and Step 2 yields

$$p = p(1) = \frac{q(1|0)^\alpha q(1|1)^{1-\alpha}}{q(1|0)^\alpha q(1|1)^{1-\alpha} + q(0|1)^\alpha q(0|0)^{1-\alpha}}.$$

Iterating these steps should yield $p \rightarrow 1/2$.

Extensions of the basic Blahut–Arimoto algorithm Natural extensions of the Blahut–Arimoto algorithm provide methods for computing optimal input distributions that apply in great generality. As an example of a simple extension, Problem 6.16 considers optimization of the input probabilities for a 4-PAM alphabet $\{\pm d, \pm 3d\}$ over the AWGN channel with a power constraint. The Blahut–Arimoto iterations must now account for the fact that the signal power depends both on d and the input probabilities.

6.5 Further reading

The information theory textbook by Cover and Thomas [14] provides a lucid exposition of the fundamental concepts of information theory, and is perhaps the best starting point for delving further into this field. The classic text by Gallager [42] is an important reference for many topics. The text by Csiszar and Korner [43] is the definitive work on the use of combinatorial techniques and the method of types in proving fundamental theorems of information theory. Other notable texts providing in-depth treatments of information theory include Blahut [44], McEliece [45], Viterbi and Omura [12], and Wolfowitz [46]. Shannon's original work [47, 48] is a highly recommended read, because of its beautiful blend of intuition and rigor in establishing the foundations of the field.

For most applications, information-theoretic quantities such as capacity must be computed numerically as solutions to optimization problems. The Blahut–Arimoto algorithm discussed here [49, 50] is the classical technique for optimizing input distributions. More recently, however, methods based on convex optimization and duality [51, 52] and on linear programming [53] have been developed for deeper insight into, and efficient solution of, optimization problems related to the computation of information-theoretic quantities. Much attention has been focused in recent years on information-theoretic limits for the wireless channel, as discussed in Chapter 8.

Good sources for recent results in information theory are the *Proceedings of the International Symposium on Information Theory (ISIT)*, and the journal *IEEE Transactions on Information Theory*. The October 1998 issue of the latter commemorates the fiftieth anniversary of Shannon's seminal work, and provides a perspective on the state of the field at that time.

6.6 Problems

Problem 6.1 (Estimating the capacity of a physical channel) Consider a line of sight radio link with free space propagation. Assume transmit and receive antenna gains of 10 dB each, a receiver noise figure of 6 dB, and a range of 1 km. Using the Shannon capacity formula for AWGN channels, what is the transmit power required to attain a link speed of 1 gigabit/s using a bandwidth of 1.5 GHz (assuming 50% excess bandwidth)?

Problem 6.2 (Entropy for an M -ary random variable) Suppose that X is a random variable taking one of M possible values (e.g., X may be the index of the transmitted signal in an M -ary signaling scheme).

- (a) What is the entropy of X , assuming all M values are equally likely?
- (b) Denoting the pmf for the uniform distribution in (a) by $q(x)$, suppose now that X is distributed according to pmf $p(x)$. Denote the entropy of X under pmf p by $H_p(X)$. Show that the divergence between p and q equals

$$D(p||q) = \mathbb{E}_p[\log_2 \frac{p(X)}{q(X)}] = \log_2 M - H_p(X).$$

- (c) Infer from (b) that the maximum possible entropy for X is $\log_2 M$, which is achieved by the uniform distribution.

Problem 6.3 (Differential entropy is maximum for Gaussian random variables) Consider a zero mean random variable X with density $p(x)$ and variance v^2 . Let $q(x)$ denote the density of an $N(0, v^2)$ random variable with the same mean and variance.

- (a) Compute the divergence $D(p||q)$ in terms of $h(X)$ and v^2 .
 (b) Use the nonnegativity of divergence to show that

$$h(X) \leq \frac{1}{2} \log_2(2\pi e v^2) = h(N(0, v^2)).$$

That is, the Gaussian density maximizes the differential entropy over all densities with the same variance.

Remark This result, and the technique used for proving it, generalizes to random vectors, with Gaussian random vectors maximizing differential entropy over all densities with the same covariance.

Problem 6.4 (Differential entropy for Gaussian random vectors) Derive the following results.

- (a) If $\mathbf{X} \sim N(\mathbf{m}, \mathbf{C})$ is an n -dimensional Gaussian random vector with mean vector $\mathbf{m}$ and covariance matrix $\mathbf{C}$, then its differential entropy is given by

$$h(\mathbf{X}) = \frac{1}{2} \log_2((2\pi e)^n |\mathbf{C}|) \quad \text{Differential entropy for real Gaussian.}$$

- (b) If $\mathbf{X} \sim CN(\mathbf{m}, \mathbf{C})$ is an n -dimensional proper complex Gaussian random vector with mean vector $\mathbf{m}$ and covariance matrix $\mathbf{C}$, then its differential entropy is given by

$$h(\mathbf{X}) = \log_2((\pi e)^n |\mathbf{C}|) \quad \text{Differential entropy for proper complex Gaussian.}$$

Problem 6.5 (Entropy under simple transformations) Let X denote a random variable, and a, b denote arbitrary constants.

- (a) If X is discrete, how are the entropies $H(aX)$ and $H(X+b)$ related to $H(X)$?
 (b) If X is continuous, how are the differential entropies $h(aX)$ and $h(X+b)$ related to $h(X)$?

Figure 6.8 The binary (symmetric) erasures channel.

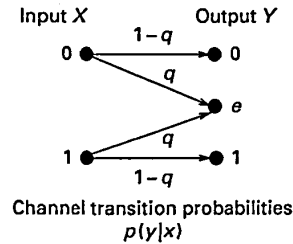
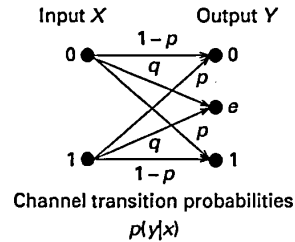


Figure 6.9 The binary (symmetric) errors and erasures channel.



Problem 6.6 (Binary erasures channel) Show that the channel capacity of the binary erasures channel with erasure probability q , as shown in Figure 6.8, is given by $1 - q$.

Problem 6.7 (Binary errors and erasures channel) Find the channel capacity of the binary errors and erasures channel with error probability p and erasures probability q , as shown in Figure 6.9.

Problem 6.8 (AWGN capacity plots for complex constellations) Write computer programs for reproducing the capacity plots in Figures 6.5 and 6.6.

Problem 6.9 (Shannon theory for due diligence) A binary noncoherent FSK system is operating at an E_b/N_0 of 5 dB, and passes hard decisions (i.e., decides whether 0 or 1 was sent) up to the decoder. The designer claims that her system achieves a BER of 10^{-5} using a powerful rate 1/2 code. Do you believe her claim?

Problem 6.10 (BPSK with errors and erasures) Consider BPSK signaling with the following scale-invariant model for the received samples:

$$Y = A(-1)^X + Z,$$

where $X \in \{0, 1\}$ is equiprobable, and $Z \sim N(0, 1)$, with $A^2 = \text{SNR}$.

(a) Find the capacity in bits per channel use as outlined in the text, and plot it as a function of SNR (dB).

- (b) Specify the BSC induced by hard decisions. Find the capacity in bits per channel use and plot it as a function of SNR (dB).
- (c) What is the degradation in dB due to hard decisions at a rate of 1/4 bits per channel use?
- (d) What is the E_b/N_0 (dB) corresponding to (c), for both soft and hard decisions?
- (e) Now suppose that the receiver supplements hard decisions with erasures. That is, the receiver declares an erasure when $|Y| < \alpha$, with $\alpha \geq 0$. Find the error and erasure probabilities as a function of α and SNR.
- (f) Apply the result of Problem 6.7 to compute the capacity as a function of α and SNR. Set SNR at 3 dB, and plot capacity as a function of α . Compare with the capacity for hard decisions.
- (g) Find the best value of α for SNR of 0 dB, 3 dB and 6 dB. Is there a value of α that works well over the range 0–6 dB?

Problem 6.11 (Gray coded two-dimensional modulation with hard decisions) A communication system employs Gray coded 16-QAM, with the demodulator feeding hard decisions to an outer binary code.

- (a) What is a good channel model for determining information-theoretic limits on the rate of the binary code as a function of E_b/N_0 ?
- (b) We would like to use the system to communicate at an information rate of 100 Mbps using a bandwidth of 150 MHz, where the modulating pulse uses an excess bandwidth of 50%. Use the model in (a) to determine the minimum required value of E_b/N_0 for reliable communication.
- (c) Now suppose that we use QPSK instead of 16-QAM in the setting of (b). What is the minimum required value of E_b/N_0 for reliable communication?

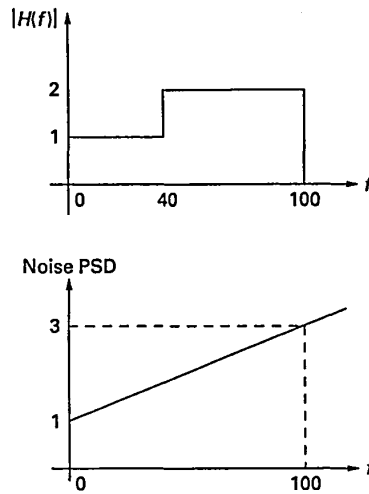
Problem 6.12 (Parallel Gaussian channels) Consider two parallel complex Gaussian channels with channel gains $h_1 = 1 + j$, $h_2 = -3j$ and noise covariances $N_1 = 1$, $N_2 = 2$. Assume that the transmitter knows the channel characteristics.

- (a) At “low” SNR, which of the two channels would you use?
- (b) For what values of net input power P would you start using both channels?
- (c) Plot the capacity as a function of net input power P using the waterfilling power allocation.

Also plot for comparison the capacity attained if the transmitter does not know the channel characteristics, and splits power evenly across the two channels.

Problem 6.13 (Waterfilling for a dispersive channel) A real baseband dispersive channel with colored Gaussian noise is modeled as in Figure 6.10.

Figure 6.10 Channel characteristics for Problem 6.13.



We plan to use the channel over the band $[0, 100]$. Let $N = \int_0^{100} S_n(f) df$ denote the net noise power over the band. If the net signal power is P , then we define SNR as P/N .

- Assuming an SNR of 10 dB, find the optimal signal PSD using waterfilling. Find the corresponding capacity.
- Repeat (a) for an SNR of 0 dB.
- Repeat (a) and (b) assuming that signal power is allocated uniformly over the band $[0, 100]$.

Problem 6.14 (Multipath channel) Consider a complex baseband multipath channel with impulse response

$$h(t) = 2\delta(t-1) - \frac{j}{2}\delta(t-2) + (1+j)\delta(t-3.5).$$

The channel is used over the band $[-W/2, W/2]$. Let $C_W(\text{SNR})$ denote the capacity as a function of bandwidth W and SNR, assuming that the input power is spread evenly over the bandwidth used and that the noise is AWGN.

- Plot $C_W(\text{SNR})/W$ versus W over the range $1 < W < 20$, fixing the SNR at 10 dB. Do you notice any trends?
- For $W = 10$, find the improvement in capacity due to waterfilling at an SNR of 10 dB.

Problem 6.15 (Blahut–Arimoto iterations for BSC) Consider the binary symmetric channel with crossover probability 0.1. Starting from an initial input distribution with $P[X = 1] = p = 0.3$, specify the values of p obtained

in the first five iterations of the Blahut–Arimoto algorithm. Comment on whether the iterations are converging to the result you would expect.

Problem 6.16 (Extension of Blahut–Arimoto algorithm for constellation optimization) Consider a 4-PAM alphabet $\{\pm d, \pm 3d\}$ to be used on the real, discrete-time AWGN channel. Without loss of generality, normalize the noise variance to one. Assuming that the input distribution satisfies a natural symmetry condition:

$$P[X = \pm d] = p, \quad P[X = \pm 3d] = \frac{1}{2} - p.$$

- (a) What is the relation between p and d at SNR of 3 dB?
- (b) Starting from an initial guess of $p = 1/4$, iterate the Blahut–Arimoto algorithm to find the optimal input distribution at SNR of 3 dB, modifying as necessary to satisfy the SNR constraint.
- (c) Comment on how the optimal value of p varies with SNR by running the Blahut–Arimoto algorithm for a few other values.